

```

;      TERMINAL.ASM   APRIL 10, 1982
;
;      THIS PROGRAM WAS ORIGINALLY DEVELOPED BY THE SMALL SYSTEMS
;      GROUP OF THE INSTITUTE OF COMPUTER SCIENCE, UNIVERSITY OF GUELPH,
;      COPYRIGHT 1979 BY DON GENNER.
;      ADAPTED TO RUN IN A MACHINE INDEPENDENT CP/M ENVIRONMENT
;      BY JOHN WILSON.
;      MODIFIED BY J. LAW, DEPT. OF PHYSICS, UNIVERSITY OF GUELPH,
;      WITH DON GENNER'S HELP.
;      EXTENSIVELY MODIFIED TO BE MORE NEARLY HOST INDEPENDENT,
;      TO RECEIVE INTO A BUFFER, TO SEND BY LOGICAL DISK SECTOR,
;      EXPANDED LOCAL MODE, TO WORK WITH STANDARD BDOS, AND MANY OTHERS
;      BY GEORGE CORLISS, DEPT. OF MATH., MARQUETTE UNIVERSITY.
;      MODIFIED FOR SLOW PRINTER BY J. GLENN BROOKSHEAR, MARQUETTE
;      UNIVERSITY.
;      MODIFIED FOR RECEIVER BUFFER OVERFLOW BY J. GLENN BROOKSHEAR
;      MARQUETTE UNIVERSITY.

```

```

;
;      DOCUMENTATION AND INSTRUCTIONS ARE IN FILE TERMINAL.DOC
;
;-----
;

```

```

0000 = BASE      EQU      0000H
0100 = USER     EQU      BASE+100H
005C = FCB       EQU      BASE+005CH
0E00 = STACK    EQU      USER+0D00H
0F00 = BUFF     EQU      USER+0E00H
0D00 = PRTBUF   EQU      USER+0C00H
0D00 = OVFLBF   EQU      USER+0C00H
0000 = CPM      EQU      BASE
0005 = BDOS     EQU      BASE+0005H
;
0001 = CTLA     EQU      01H
0002 = CTLB     EQU      02H
0003 = CTLC     EQU      03H
0005 = CTLE     EQU      05H
0007 = BEL      EQU      07H      ;CTRL G
0008 = BS       EQU      08H      ;BACKSPACE = CTRL H
0009 = TAB      EQU      09H      ;CTRL I
000A = LF       EQU      0AH      ;LINE FEED = CTRL J
000C = FF       EQU      0CH      ;FORM FEED = CTRL L
000D = CR       EQU      0DH      ;CTRL M
0010 = CTLP     EQU      10H
0011 = XON      EQU      11H      ;CTRL Q
0013 = XOFF     EQU      13H      ; CTRL S
0019 = CTLY     EQU      19H      ;CTRL Y
001A = EOD      EQU      1AH      ;CP/M END OF FILE = CTRL Z
001B = ESC      EQU      1BH      ;ESCAPE
007F = DEL      EQU      7FH

```

```

;----- BEGIN
;

```

```

0100          ORG      USER
0100 C3B501   JMP      ENTER
0103 01      HOSTON  DB      01H
0104 000D    HEAD   DW      PRTBUF
0106 000D    TAIL   DW      PRTBUF
0108 00      BAKLOG DB      00
00C0 =       STOPLMT EQU     0C0H
0020 =       STRTLMT EQU     20H

```

```

;
; HOST DEPENDENCIES:
0109 0D24      CRMSG  DB      CR,'$'      ;CARRIAGE RETURN MESSAGE
0008 =        BADBS  EQU      BS          ;WRONG BS/DEL CHAR FOR HOST
007F =        GOODBS EQU      DEL        ;CORRECT BS/DEL CHAR FOR HOST
000D =        HSTEOL EQU      CR         ;HOST END OF LINE
001A =        HSTEOD EQU      EOD        ;HOST END OF FILE
;
010B =        HOSTBF EQU      $          ;INITIALIZED FOR SIGMA 9
010B 0D24333435HGO  DB      CR,'$','345' ; START HOST
0110 1124333435HREC DB      XON,'$','345' ; SET UP HOST TO BEGIN RECEIVING
0115 1B46132435HCLOSE DB     ESC,'F',XOFF,'$','5' ;CLOSE HOST FILE AFTER REC.
011A 1B48243435HSTOP DB     ESC,'H', '$', '45' ;HOST, STOP SENDING ME STUFF
011F 363237342CLONMSG DB     '6274,BROOKSHEARG',CR
0130 4548420D54    DB     'EHB',CR,'T 37,0',CR,'$'
013C 3031323334    DB     '01234567890'
0147 4F46460D24OFFMSG DB     'OFF',CR,'$','6789012345'
0156 1B480D1951HNOSND DB     ESC,'H',CR,CTLY,'Q',CR,'$','890'
;
0160 =        VAXMSG EQU      $          ; VAX-11 MESSAGES START HERE.
0160 1124333435VHGO  DB      XON,'$','345' ; START HOST
0165 2432333435VHREC DB      '$', '2345' ; SET UP HOST TO BEGIN RECEIVING
016A 1A24333435VHCLOSE DB     EOD,'$','345' ;CLOSE HOST FILE AFTER REC.
016F 1324333435VHSTOP DB     XOFF,'$','345' ;HOST, STOP SENDING ME STUFF
0174 363237340DV LONMSG DB     '6274',CR,'EHB',CR
017D 5345542054    DB     'SET TERM/HOSTSYNC',CR,'$'
0190 3930313233    DB     '901234567890'
019C 4C4F474F55VOFFMSG DB     'LOGOUT',CR,'$','9012345'
01AB 1303243435VHNOSND DB     XOFF,CTLC,'$','4567890'
;

```

```

-----
;
; PROGRAM LAYOUT
; MAIN PROGRAM
; I/O ROUTINES CONIO, PRNT, SOUT
; SPECIAL CONTROL CODE HANDLERS
; SWCHPT, CRTOUT, TRADE, LOCAL
; SUBROUTINES FOR LOCAL
; DSKDIR, HELP, LOGON, OFF, PRINTR, RECEIVE, SEND
; CP/M BDOS CALLS
;

```

```

01B5 210000      ENTER  LXI  H,0
01B8 39          DAD   SP
01B9 22FE0E      SHLD  STACK+254
01BC 31FE0E      LXI  SP,STACK+254
;

```

```

-----
;
; MVI  A,83H      ; THIS BLOCK USED TO INITIALIZE MODEM
; OUT  330Q+3     ;PORT ON HEATH H8/H89 MACHINES TO
; MVI  A,3        ;TO 300 BAUD, 8 DATA DITS, NO PARITY,
; OUT  330Q+4     ;AND 2 STOP BITS.
; MVI  A,80H
; OUT  330Q
; MVI  A,1
; OUT  330Q+1
; MVI  A,3
; OUT  330Q+3
; XRA  A
; OUT  330Q+1
;

```

```

01BF C3C901      JMP  TERMINAL
;

```

01C2 2AFE0E EXIT LHLD STACK+254  
01C5 F9 SPHL  
01C6 C30000 JMP CPM

;  
;  
-----  
; BEGIN MAIN PROGRAM  
;

TERMINAL:  
01C9 11D801 LXI D,VERMSG  
01CC CD820C CALL STRINGF  
01CF 11F701 LXI D,TMESS  
01D2 CD820C CALL STRINGF  
01D5 C30D02 JMP TERM2

01D8 0D0A544552VERMSG DB CR,LF,'TERMINAL PROGRAM VER. 2',CR,LF,'\$'  
01F7 0D0A544552TMESS DB CR,LF,'TERMINAL MODE',CR,LF,LF,' \$'

020D CD8F02 TERM2 CALL CONIO ;ANYTHING FROM TERMINAL?  
0210 CA4802 JZ SGET ;NO - MOVE ON  
0213 5F MOV E,A ;YES - LOAD E

;  
; PROCESS SPECIAL CONTROL CHARACTERS  
;  
; P TOGGLE PRINTER ON/OFF  
; A ENTER LOCAL MODE  
; B SET BELL TO RING ON NEXT HOST CHAR  
; C RETURN TO CP/M  
; E TOGGLE CRT ECHO ON/OFF  
; ESC FORCE SEND NEXT CHAR TO HOST  
;

0214 FE10 CPI CTLP  
0216 CA1203 JZ SWCHPT ; IF P  
0219 FE01 CPI CTLA  
021B CA3C03 JZ LOCAL ; IF A  
021E FE02 CPI CTLB  
0220 CA2003 JZ SWBEL ; IS B  
0223 FE03 CPI CTLC  
0225 CAC201 JZ EXIT ; IF C  
0228 FE05 CPI CTLE  
022A CA2903 JZ SWECHO ; IF E  
022D FE1B CPI ESC  
022F CA3503 JZ FORSND ; IF ESCAPE  
0232 FE08 CPI BADBS  
0234 CAE703 JZ TRADE ; IF WRONG BS/DEL FOR HOST

0237 D5 SPUT PUSH D  
0238 CD660C CALL PUNOUTF ;SEND TO HOST  
023B D1 POP D  
023C 3A3403 LDA ECHO ;SHOULD CHARACTERS BE SENT TO CRT?  
023F FE00 CPI 0 ;0 - YES, 1 - NO  
0241 C24802 JNZ SGET  
0244 7B MOV A,E  
0245 CD9602 DPUTB CALL PRNT ;SEND TO CONSOLE  
0248 CD4F0C SGET CALL RDRINPF ;ANYTHING FROM HOST?  
024B CA6202 JZ CKPRT ;NO. CHECK PRINT BUFFER  
024E CD9602 CALL PRNT ;ECHO IT  
0251 3A2803 LDA BELWAK ;IS BELL SWITCH SET?  
0254 FE00 CPI 0  
0256 CA6202 JZ CKPRT ;IF NOT SET  
0259 1E07 MVI E,BEL ;RING BELL  
025B CD4A0C CALL CONOUTF  
025E AF XRA A ;RESET BELL SWITCH  
025F 322803 STA BELWAK  
0262 3A0801 CKPRT LDA BAKLOG ;ANYTHING IN PRINT BUFFER?  
0265 FE00 CPI 00

```

0267 CA0D02      JZ      TERM2      ;NO. TRY TERMINAL
026A CD750C      CALL     PRTSTAT      ;PRINTER BUSY?
026D FE00        CPI      00
026F CA0D02      JZ      TERM2      ;YES. TRY TERMINAL
0272 2A0401      LHL D   HEAD      ;SEND CHARACTER TO PRINTER
0275 5E          MOV     E,M      ;AND SHIFT HEAD POINTER.
0276 2C          INR     L
0277 220401      SHLD   HEAD
027A CD700C      CALL     LSTOUTF
027D 3A0801      LDA     BAKLOG      ;IF BACKLOG HAS DROPPED
0280 3D          DCR     A      ;TO START LIMIT THEN TELL
0281 320801      STA     BAKLOG      ;HOST TO START SENDING AGAIN.
0284 FE20        CPI      STRTLMT
0286 C20D02      JNZ     TERM2
0289 CD0003      CALL     STRTHST
028C C30D02      JMP     TERM2      ;CHECK ON TERMINAL

```

```

;
;----- END OF MAIN PROGRAM
;
;

```

```

028F 1EFF        CONIO   MVI     E,0FFH ;DIRECT CONSOLE INPUT
0291 CD7A0C      CALL     DIRCONF ;(A) = CHAR OR 00 IF NO CHAR
0294 B7          ORA     A ;FORCE FLAG
0295 C9          RET

```

```

;
0296 E5          PRNT   PUSH    H      ;PRINT CHAR (A) AT CONSOLE
0297 D5          PUSH    D
0298 5F          MOV     E,A
0299 FE0D        CPI     HSTEOL ;END OF LINE?
029B CAB102      JZ      PRNT2
029E D5          PUSH    D
029F CD4A0C      CALL     CONOUTF
02A2 D1          POP     D
02A3 3A1F03      LDA     AUXON
02A6 FE00        CPI     0
02A8 C2AE02      JNZ     PRNT1 ;AND PERHAPS AT THE PRINTER
02AB CDD802      CALL     LIST
02AE D1          PRNT1  POP     D
02AF E1          POP     H
02B0 C9          RET

```

```

;
02B1 3A1F03      PRNT2  LDA     AUXON
02B4 FE00        CPI     0
02B6 C2CC02      JNZ     PRNT3
02B9 11D502      LXI     D,CRLF
02BC CD820C      CALL     STRINGF
02BF 1E0D        MVI     E,CR
02C1 CDD802      CALL     LIST
02C4 1E0A        MVI     E,LF
02C6 CDD802      CALL     LIST
02C9 C3AE02      JMP     PRNT1
02CC 11D502      PRNT3  LXI     D,CRLF
02CF CD820C      CALL     STRINGF
02D2 C3AE02      JMP     PRNT1
02D5 0D0A24      CRLF   DB     CR,LF,'$'

```

```

;
; PLACE CONTENTS OF E REGISTER IN PRINT BUFFER.
; IF BACKLOG IS THEN COH, TELL HOST TO WAIT.
;

```

```

; CAUTION*** HOST WAIT ROUTINE USES E REGISTER.
;

```

```

02D8 2A0601      LIST   LHL D   TAIL
02DB 73          MOV     M,E
02DC 2C          INR     L
02DD 220601      SHLD   TAIL
02E0 3A0801      LDA     BAKLOG

```

```

02E3 3C      INR      A
02E4 320801  STA      BAKLOG
02E7 FEC0    CPI      STOPLMT
02E9 C0      RNZ
02EA CDEE02  CALL     STOPHST
02ED C9      RET

;
02EE 3A0301  STOPHST LDA     HOSTON ;IS HOST ALREADY STOPPED?
02F1 FE00    CPI      00
02F3 C8      RZ          ;YES. RETURN
02F4 111A01  LXI      D,HSTOP ;NO. THEN STOP IT.
02F7 CD870A  CALL     MSGHOST
02FA 3E00    MVI      A,00
02FC 320301  STA      HOSTON
02FF C9      RET

;
0300 3A0301  STRTHST LDA     HOSTON ;IS HOST ALREADY STARTED?
0303 FE00    CPI      00
0305 C0      RNZ          ;YES. RETURN
0306 110B01  LXI      D,HGO   ;NO. THEN START IT.
0309 CD870A  CALL     MSGHOST
030C 3E01    MVI      A,01
030E 320301  STA      HOSTON
0311 C9      RET

;
;----- SPECIAL CONTROL CODE HANDLERS
;
0312 3A1F03  SWCHPT  LDA     AUXON
0315 47      MOV     B,A
0316 3E01    MVI     A,1
0318 90      SUB     B
0319 321F03  STA     AUXON
031C C34802  JMP     SGET

;
031F 01      AUXON  DB     1

;
0320 3E01    SWBEL  MVI     A,1 ;SET BELL SWITCH
0322 322803  STA     BELWAK
0325 C34802  JMP     SGET

;
0328 00      BELWAK DB     0

;
0329 3A3403  SWECHO  LDA     ECHO
032C EE01    XRI     01H
032E 323403  STA     ECHO
0331 C34802  JMP     SGET

;
0334 01      ECHO  DB     1 ;1 - NO ECHO, 0 - ECHO TO CRT

;
0335 CD400C  FORSND  CALL    CONINPF ;FORCE SEND NEXT CHAR TO HOST
0338 5F      MOV     E,A
0339 C33702  JMP     SPUT

;
033C 11AB03  LOCAL  LXI     D,LMESS ;DISPLAY 'LOCAL MODE'
033F CD820C  CALL    STRINGF
0342 CD400C  CALL    CONINPF
0345 FE0D    CPI     CR
0347 CAEF03  JZ      HELP
034A FE03    CPI     CTLC
034C CAC201  JZ      EXIT
034F E65F    ANI     5FH ;CONVERT TO UPPER CASE
0351 D641    SUI     'A' ;SET UP FOR JUMP TABLE
0353 FAA203  JM      NOTYET
0356 FE1A    CPI     1AH
0358 F2A203  JP      NOTYET ;IF ASCII 'Z'

```

5

```

035B 216E03      LXI      H,IFA      ;LB OF INDEX
035E 5F          MOV      E,A        ;HB OF INDEX
035F 1600        MVI      D,0        ;ADD INDEX TO BASE OF JUMP TABLE
0361 19          DAD      D          ; TWICE BECAUSE 2 BYTES/WORD
0362 19          DAD      D          ;ADDRESS HIGH BYTE FIRST
0363 23          INX      H          ;HB OF JUMP ADDRESS
0364 56          MOV      D,M
0365 2B          DCX      H
0366 5E          MOV      E,M      ;LB OF JUMP ADDRESS
0367 EB          XCHG     ;JUMP ADDRESS TO HL
0368 226C03      SHLD    DUMMY+1    ;FILL IN JUMP ADDRESS
036B C36B03      DUMMY   JMP      DUMMY ;JUMP ADDRESS IS SUPPLIED ABOVE

```

```

;
; TABLE OF JUMP ADDRESSES:
;

```

```

;
; A B C D E F G
036E A203D605C2IFA DW NOTYET,CHGBUF,EXIT,DSKDIR,TERMINAL,NOTYET,NOTYET
; H I J K L M N O
037C EF03A203A2 DW HELP,NOTYET,NOTYET,NOTYET,LOGON,NOTYET,NOTYET,OFF
; P Q R S T U V
038C EE07A203F9 DW PRINTR,NOTYET,RECEIVE,SEND,TERMINAL,NOTYET,VAXHST
; W X Y Z
039A D009A203A2 DW WAIT,NOTYET,NOTYET,NOTYET
;
;

```

```

03A2 11BD03      NOTYET LXI      D,YETMSG
03A5 CD820C      CALL    STRINGF
03A8 C33C03      JMP     LOCAL
;

```

```

03AB 0D0A4C4F43LMESS DB CR,LF,'LOCAL MODE $'
03BD 070D0A4E4FYETMSG DB BEL,CR,LF,'NOT IMPLEMENTED YET.$'
03D5 0D0A46494CFMESS DB CR,LF,'FILE NAME? $'
;
;

```

```

03E7 1E7F        TRADE  MVI      E,GOODBS ;SUBSTITUTE GOOD BS/DEL CHAR
03E9 C33702      JMP     SPUT      ;FOR A BAD ONE
;
;

```

```

;----- SUBROUTINES FOR LOCAL MODE
;

```

```

03EC C3A203      DSKDIR JMP      NOTYET
;

```

```

03EF 11F803      HELP   LXI      D,HLPMSG
03F2 CD820C      CALL    STRINGF
03F5 C33C03      JMP     LOCAL
;

```

```

03F8 0C0A0A2020HLPMSG DB FF,LF,LF,' OPTIONS AVAILABLE:',CR,LF,LF
0413 0942094348 DB TAB,'B',TAB,'CHANGE RECEIVING BUFFER SIZE (E)',CR,LF
0438 0943094558 DB TAB,'C',TAB,'EXIT TO CP/M (E)',CR,LF
044D 0944094449 DB TAB,'D',TAB,'DISK DIRECTORY (L)',CR,LF
0464 0945094558 DB TAB,'E',TAB,'EXIT TO TERMINAL MODE (E)',CR,LF
0482 0948094845 DB TAB,'H',TAB,'HELP (L)',CR,LF
048F 094C095345 DB TAB,'L',TAB,'SEND LOGON TO HOST (E)',CR,LF
04AA 094F095345 DB TAB,'O',TAB,'SEND LOGOFF TO HOST (E)',CR,LF
04C6 095009544F DB TAB,'P',TAB,'TOGGLE PRINTER ON/OFF (L)',CR,LF
04E4 0952095245 DB TAB,'R',TAB,'RECEIVE FILE FROM HOST (L)',CR,LF
0503 0953095345 DB TAB,'S',TAB,'SEND FILE TO HOST (L)',CR,LF
051D 0954095445 DB TAB,'T',TAB,'TERMINAL MODE (E)',CR,LF
0533 0956094348 DB TAB,'V',TAB,'CHANGE TO VAX-11 MODE (E)',CR,LF
0551 0957095741 DB TAB,'W',TAB,'WAIT - SET SENDING DELAY (L)',CR,LF
0572 0A09284C29 DB LF,TAB,'(L) COMMANDS RETURN TO LOCAL MODE',CR,LF
0597 0928452920 DB TAB,'(E) COMMANDS EXIT LOCAL MODE',CR,LF,LF,'$'
;

```

```

05B8 11D502      LOGON  LXI      D,CRLF
05BB CD820C      CALL    STRINGF
05BE 111F01      LXI      D,LONMSG
05C1 CD870A      CALL    MSGHOST

```

```

05C4 C30D02      ;
05C7 11D502     OFF    LXI    D,CRLF
05CA CD820C      CALL   STRINGF
05CD 114701     LXI    D,OFFMSG
05D0 CD870A     CALL   MSGHOST
05D3 C30D02     JMP    TERM2

;
05D6 111606     CHGBUF LXI    D,CHG1
05D9 CD820C     CALL   STRINGF
05DC CD8F02     GETOPT CALL   CONIO
05DF CADC05     JZ     GETOPT
05E2 FE41       CPI    41H
05E4 CAFA05     JZ     GOTA
05E7 FE42       CPI    42H
05E9 CA0806     JZ     GOTB
05EC 3E00       MVI    A,00
05EE 323D0C     STA    BFSIZE
05F1 11A707     LXI    D,CMSG
05F4 CD820C     CALL   STRINGF
05F7 C30D02     JMP    TERM2
05FA 3E01       GOTA   MVI    A,01
05FC 323D0C     STA    BFSIZE
05FF 117907     LXI    D,AMSG
0602 CD820C     CALL   STRINGF
0605 C30D02     JMP    TERM2
0608 3E80       GOTB   MVI    A,80H
060A 323D0C     STA    BFSIZE
060D 119007     LXI    D,BMSG
0610 CD820C     CALL   STRINGF
0613 C30D02     JMP    TERM2
0616 0D0A0A5345CHG1 DB    CR,LF,LF,'SELECT THE SIZE OF THE RECEIVING'
0639 2042554646 DB    ' BUFFER BY TYPING THE LETTER REPRESENTING',CR,LF
0664 594F555220 DB    'YOUR CHOICE IN THE FOLLOWING TABLE.',CR,LF,LF,LF
068B 53454C4543 DB    'SELECTION          BUFFER SIZE          FOR SYSTEMS',CR,LF
06BC 2020202020 DB    '                (IN CP/M SECTORS)        WITH',CR,LF,LF
06EA 2020204120 DB    '   A                1                32K RAM',CR,LF
0719 2020204220 DB    '   B                128             48K RAM',CR,LF
0748 2020204320 DB    '   C (DEFAULT)        256             64K RAM',CR,LF
0777 0A24       DB    LF,'$'
0779 4F5054494FAMSG DB    'OPTION A  SELECTED.',CR,LF,'$'
0790 4F5054494FBMSG DB    'OPTION B  SELECTED.',CR,LF,'$'
07A7 4F5054494FCMSG DB    'OPTION C  SELECTED.',CR,LF,'$'

;
07BE 0655       VAXHST MVI    B,55H          ;MOVE VAXMSG BLOCK TO
07C0 216001     LXI    H,VAXMSG      ;HOSTBF BLOCK AND THEN
07C3 110B01     LXI    D,HOSTBF      ;GO TO TERM2.
07C6 7E         MOVBYT MOV    A,M
07C7 EB         XCHG
07C8 77         MOV    M,A
07C9 05         DCR    B
07CA FE00       CPI    00
07CC CAD507     JZ     VAXH1
07CF 13         INX    D
07D0 23         INX    H
07D1 EB         XCHG
07D2 C3C607     JMP    MOVBYT
07D5 11DE07     VAXH1 LXI    D,VAXMOD      ;INDICATE MODE IS VAX
07D8 CD820C     CALL   STRINGF
07DB C30D02     JMP    TERM2
07DE 5641582D31VAXMOD DB    'VAX-11 MODE.',CR,LF,LF,'$'

;
07EE 3A1F03     PRINTR LDA    AUXON        ;TOGGLE PRINTER OFF/ON
07F1 EE01       XRI    01H
07F3 321F03     STA    AUXON
07F6 C33C03     JMP    LOCAL

```

```

;
; RECEIVE A FILE FROM HOST
; ESTABLISH STATE
; LOOP FOR EACH CHAR
; CHECK CONSOLE
; ESC -----
; GET CHAR FROM HOST
; NONE -----
; END OF FILE -----
; WRITE CHAR TO MEMORY
; MANAGE BUFFER
; COUNT CHARS IN SECTOR
; WHEN NECESSARY, WRITE SECTOR
; HOST WAIT
; DISK WRITE
; HOST CONTINUE
;
; CLOSE FILE
;

```

```

07F9 3E01 RECEIVE MVI A,01 ;TURN PRINTER OFF
07FB 321F03 STA AUXON
07FE CDFC09 CALL GETFN ;GET FILE NAME
0801 115C00 LXI D,FCB
0804 CD8C0C CALL OPENFIF
0807 FEFF CPI OFFH
0809 CA1808 JZ REC2 ;IF FILE IS NEW
080C 112C08 LXI D,EXISTM ;FILE EXISTS.
080F CD820C CALL STRINGF ;WARN
0812 115C00 LXI D,FCB ;AND DELETE OLD ONE
0815 CD960C CALL DELETF
0818 115C00 REC2 LXI D,FCB
081B CDA50C CALL MAKEFIF
081E FEFF CPI OFFH
0820 C26D08 JNZ REC7 ;IF MAKE WAS SUCCESSFUL
0823 114708 LXI D,FULLMSG
0826 CD820C CALL STRINGF
0829 C33C03 JMP LOCAL

;
082C 070D0A4558EXISTM DB BEL,CR,LF,'EXISTING FILE DELETED',CR,LF,'$'
0847 070D0A4449FULLMSG DB BEL,CR,LF,'DIRECTORY FULL. ACTION CANCELLED.$'

;
086D 110901 REC7 LXI D,CRMSG
0870 CD870A CALL MSGHOST
0873 CD4F0C REC4 CALL RDRINPF ;WAIT FOR ECHO OF CR SENT TO START TRANSMISSION
0876 FE0D CPI CR
0878 C27308 JNZ REC4
087B CD8F02 REC3 CALL CONIO ;LOOP FOR EACH CHAR
087E FE1A CPI EOD ;TO TERMINATE TRANSMISSION
0880 CAAB08 JZ RECEOD ;IF ESCAPE, ELSE IGNOR
0883 CD4F0C CALL RDRINPF ;WAIT FOR CHAR
0886 CA7B08 JZ REC3 ;IF NOTHING WAS RECEIVED
0889 FE1A CPI HSTEOD
088B CAAB08 JZ RECEOD ;IF END OF FILE
088E F5 PUSH PSW
088F CD9602 CALL PRNT ;ECHO TO CONSOL
0892 F1 POP PSW
0893 F5 PUSH PSW
0894 CDBC0A CALL WRTMEMF
0897 C23C03 JNZ LOCAL ;IF WRITE WAS NOT SUCCESSFUL
089A F1 POP PSW
089B FE0D CPI CR
089D C27B08 JNZ REC3
08A0 3E0A MVI A,LF
08A2 CDBC0A CALL WRTMEMF
08A5 C23C03 JNZ LOCAL
08A8 C37B08 JMP REC3

```



```

;
08AB 115601 RECEOD LXI D,HNOSND ;END OF TRANSMISSION. CLOSE FILES
08AE CD870A CALL MSGHOST
08B1 3E1A MVI A,EOD ;CP/M END OF FILE
08B3 CDBC0A CALL WRTMEMF
08B6 C33C03 JMP LOCAL

```

```

;
; SEND A FILE FROM TERMINAL TO HOST
; ESTABLISH STATE
; OPEN FILE
; SEND START SIGNAL
; LOOP FOR EACH LOGICAL SECTOR (128 BYTES)
; READ SECTOR FROM DISK
; LOOP FOR EACH OF 128 CHARS
; GET CHAR FROM MEMORY
; LF? -----
; CR? -----
; TAB? -----
; EOD? -----
; INVALID? ----
; SEND CHAR TO HOST
; IF RESPONSE, THEN XOFF? ----
; ELSE ECHO RESPONSE
; CHECK CONSOLE
; ESC -----
; EOD OR ESC: CLOSE FILES

```

```

08B9 3E01 SEND MVI A,01 ;TURN PRINTER OFF
08BB 321F03 STA AUXON
08BE CDFC09 CALL GETFN
08C1 115C00 LXI D,FCB
08C4 CD8C0C CALL OPENFIF ;OPEN SOURCE FILE
08C7 FEFF CPI OFFH
08C9 CAB309 JZ SENDNOT ;IF FILE NOT FOUND
08CC 111001 LXI D,HREC ;SET HOST TO RECEIVE A FILE
08CF CD870A CALL MSGHOST

```

```

; SENDL1 LXI D,BUFF ;LOOP FOR EACH SECTOR
08D2 11000F SENDL1 PUSH D ;BUFF REAPPEARS IN HL
08D5 D5 CALL SETDMAF
08D6 CDAA0C LXI D,FCB
08D9 115C00 CALL READSQF
08DC CD9B0C ORA A
08DF B7 JZ SENDL7 ;IF NOT END OF FILE
08E0 CAE808 PUSH B ;FILL STACK TO BE CLEARED
08E3 C5 PUSH B ;IN SENDOED
08E4 C5 JMP SENDEOD
08E5 C3A809

```

```

; SENDL7 MVI B,129 ;INITIALIZE LOOP COUNTER
08E8 0681 SENDL7 PUSH B
08EA C5

```

```

; SENDL2 POP B ;LOOP FOR EACH CHAR
08EB C1 SENDL2 POP H
08EC E1 DCR B
08ED 05 XRA A
08EE AF ADD B ;DONE WITH THIS SECTOR?
08EF 80 JZ SENDL1 ;GET NEXT SECTOR
08F0 CAD208 MOV A,M ;GET CHAR FROM MEMORY
08F3 7E INX H
08F4 23 PUSH H
08F5 E5 PUSH B
08F6 C5 ANI 7FH ;GUARANTEE VALID ASCII CHAR
08F7 E67F MOV E,A
08F9 5F CPI LF
08FA FE0A JZ SENDLF
08FC CAA509

```

```

08FF FE0D      CPI      CR
0901 CA1309    JZ      SENDL3
0904 FE09      CPI      TAB
0906 CALD09    JZ      SENDTAB
0909 FE1A      CPI      EOD
090B CAA809    JZ      SENDEOD
090E FE20      CPI      20H      ; 20H ARE CONTROL CODES
0910 FA2009    JM      SENDINV
0913 CD4809    SENDL3  CALL    SENDCHR      ;RETURNS (A) = 0 IF EOD.
0916 B7        ORA      A
0917 CAA809    JZ      SENDEOD
091A C3EB08    JMP      SENDL2      ;GET NEXT CHAR

;
091D C31309    SENDTAB  JMP      SENDL3      ;EXPAND TABS LATER
;
0920 112909    SENDINV  LXI      D,INVMMSG      ;INVALID CHAR IN FILE
0923 CD820C    CALL    STRINGF      ;WARN AND IGNOR
0926 C3EB08    JMP      SENDL2

;
0929 070D0A494EINVMMSG  DB      BEL,CR,LF,'INVALID CHARACTER IN FILE',CR,LF,'$'
;
0948 3A3403    SENDCHR  LDA      ECHO
094B FE00      CPI      0
094D C25409    JNZ      SENDL8      ;IF NO ECHO TO CONSOLE BEFORE SEND
0950 7B        MOV      A,E
0951 CD9602    CALL    PRNT
0954 CD660C    SENDL8  CALL    PUNOUTF      ;SEND CHAR (E) TO HOST
0957 3AFA09    LDA      NDELAY      ;NUMBER OF DELAY LOOPS
095A B7        ORA      A
095B CA6109    JZ      SENDL4      ;IF NO DELAY
095E CDA60A    CALL    DELAYF
0961 CD4F0C    SENDL4  CALL    RDRINPF      ;RESPONSE?
0964 B7        ORA      A
0965 CA7309    JZ      CKPUN      ;IF NOTHING RECEIVED
0968 FE13      CPI      XOFF      ;XOFF MEANS WE MUST WAIT
096A CA8709    JZ      SENDOFF
096D CD9602    CALL    PRNT      ;ECHO IT TO CONSOL
0970 C36109    JMP      SENDL4      ;MORE RESPONSE?
0973 CD6B0C    CKPUN   CALL    PUNSTAT      ;KEEP WATCHING INPUT FROM
0976 FE00      CPI      00      ;HOST UNTIL PUN: PORT
0978 CA6109    JZ      SENDL4      ;IS CLEAR.
097B CD8F02    SENDL5  CALL    CONIO      ;CHECK FOR CONSOL INPUT
097E D61A      SUI      EOD      ;RETURN (A) = 0 IF Z
0980 C9        RET

;
0981 119909    SENDL6  LXI      D,WAITMG      ;TELL CONSOL WE ARE WAITING
0984 CD820C    CALL    STRINGF
0987 CD4F0C    SENDOFF CALL    RDRINPF      ;HOST IS SLOW - WAIT FOR XON
098A FE11      CPI      XON
098C CA7309    JZ CKPUN
098F B7        ORA      A
0990 CA8709    JZ      SENDOFF      ;NOTHING RECEIVED, WAIT
0993 CD9602    CALL    PRNT      ;ELSE ECHO RECEIVED CHAR
0996 C38709    JMP      SENDOFF      ;AND CONTINUE WAITING

;
0999 0D0A574149WAITMG  DB      CR,LF,'WAITING',CR,LF,'$'
;
09A5 C31309    SENDLF  JMP      SENDL3      ;LF CAN BE HANDLED LATER
;
09A8 111501    SENDEOD  LXI      D,HCLOSE      ;CLOSE HOST FILE
09AB CD870A    CALL    MSGHOST
09AE C1        POP      B
09AF E1        POP      H      ;CLEAR STACK
09B0 C33C03    JMP      LOCAL

;
09B3 11BC09    SENDNOT  LXI      D,ENEMSG      ;FILE NOT FOUND TO SEND

```

```

09B6 CD820C          CALL STRINGF
09B9 C33C03          JMP LOCAL

;
09BC 070D0A4649FNMSG DB BEL,CR,LF,'FILE NOT FOUND',CR,LF,'$'

;
09D0 11DF09          WAIT LXI D,DELAYM ;SET COUNTER FOR NUMBER
09D3 CD820C          CALL STRINGF ;DELAY LOOPS
09D6 CD400C          CALL CONINPF ;USE THE ASCII CODE AS INDEX
09D9 32FA09          STA NDELAY
09DC C33C03          JMP LOCAL

;
09DF 0D0A454E54DELAYM DB CR,LF,'ENTER DELAY COUNTER $'
09FA 0000            NDELAY DB 0,0

;
; GET A FILE NAME FROM THE OPERATOR
;
; STORE FILE NAME IN FCB
;
09FC 215C00          GETFN LXI H,FCB
09FF AF              XRA A
0A00 326800          STA FCB+12 ;ZERO CURRENT EXTENT
0A03 327C00          STA FCB+32 ;ZERO CURRENT RECORD
0A06 3600            MVI M,0
0A08 23              INX H
0A09 0E0B            MVI C,11
0A0B 3E20            MVI A,' '
0A0D CD9F0A          CALL FILLF ;BLANK NAME AREA
0A10 11D503          LXI D,FMESS
0A13 CD820C          CALL STRINGF
0A16 21770A          LXI H,GETFN3
0A19 0E0E            MVI C,14
0A1B 3E20            MVI A,' '
0A1D CD9F0A          CALL FILLF ;BLANK BUFFER AREA
0A20 11750A          LXI D,GETFN3
0A23 CD870C          CALL READCNF
0A26 11770A          LXI D,GETFN3
0A29 215C00          LXI H,FCB
0A2C 13              INX D
0A2D 1A              LDAX D
0A2E 1B              DCX D
0A2F FE3A            CPI ':'
0A31 C23F0A          JNZ GETNAM
0A34 1A              LDAX D
0A35 E60F            ANI 0FH
0A37 FE03            CPI 3 ;VALID DISK NUMBER?
0A39 F2FC09          JP GETFN
0A3C 77              MOV M,A
0A3D 13              INX D
0A3E 13              INX D
0A3F 23              GETNAM INX H ;FILE NAME
0A40 E5              PUSH H
0A41 0E09            MVI C,9
0A43 1A              GETFNLL LDAX D
0A44 FE2E            CPI '.'
0A46 CA5A0A          JZ GETFNLLF
0A49 FE20            CPI ' '
0A4B C8              RZ
0A4C FE41            CPI 'A'
0A4E FA530A          JM GETFN3 ;IF NOT LOWER CASE
0A51 E65F            GETFN3 ANI 5FH ;CONVERT TO UPPER CASE
0A53 77              MOV M,A
0A54 13              INX D
0A55 23              INX H
0A56 0D              DCR C
0A57 C2420A          INZ GETFN1

```

```

0A5A E1 GETFNL1F POP H
0A5B 010800 LXI B,8
0A5E 09 DAD B
0A5F 13 INX D
0A60 0E03 MVI C,3
0A62 1A GETFNL2 LDAX D ;EXTENSION
0A63 FE20 CPI ' '
0A65 C8 RZ
0A66 FE41 CPI 'A'
0A68 FA6D0A JM GETFN4 ;IF NOT LOWER CASE
0A6B E65F ANI 5FH
0A6D 77 GETFN4 MOV M,A
0A6E 13 INX D
0A6F 23 INX H
0A70 0D DCR C
0A71 C2620A JNZ GETFNL2
0A74 C9 RET

;
0A75 0E00313233GETFNRB DB 14,0,'1234567890123456'
0A77 = GETFN EQU GETFNRB+2
;
; MSGHOST: SEND MESSAGE TO THE HOST
; DE POINTS TO THE MESSAGE (MUST END IN A $)
;
0A87 D5 MSGHOST PUSH D
0A88 D1 MSGL1 POP D
0A89 1A LDAX D
0A8A 13 INX D
0A8B FE24 CPI '$'
0A8D C8 RZ ;END OF MESSAGE
0A8E D5 PUSH D
0A8F 5F MOV E,A
0A90 CD660C CALL PUNOUTF ;SEND CHAR TO HOST
0A93 CD4F0C CALL RDRINPF ;HAS HOST SENT ANYTHING BACK?
0A96 CA880A JZ MSGL1 ;IF NOTHING RECEIVED
0A99 CD9602 CALL PRNT ;ECHO RECEIVED CHAR
0A9C C3880A JMP MSGL1

;
;
; FILLF FILL MEMORY WITH CHAR
;
; ENTRY: (A) CHAR TO BE FILLED
; (C) NUMBER OF LOCATIONS TO BE FILLED
; (HL) BEGINNING ADDRESS
;
0A9F 77 FILLF MOV M,A
0AA0 23 INX H
0AA1 0D DCR C
0AA2 C29F0A JNZ FILLF
0AA5 C9 RET

;
;
; DELAYF: DELAY FOR A WHILE. (A) CONTAINS COUNTER
;
0AA6 3D DELAYF DCR A
0AA7 C8 RZ
0AA8 C5 PUSH B
0AA9 0620 MVI B,20H
0AAB 05 DEL1 DCR B
0AAC CAB80A JZ DEL2
0AAF 2AFA09 LHLD NDELAY
0AB2 22FA09 SHLD NDELAY
0AB5 C3AB0A JMP DEL1
0AB8 C1 DEL2 POP B
0AB9 C21601 JMP DELAYF

```

```

;
;
;   WRTMEMF:  HANDLE WRITING A CHAR IN REGISTER A TO MEMORY
;             AND TO DISK.
;             ASSUMES DISK FILE HAS ALREADY BEEN OPENED.
;             IF (A) = EOD ( Z), DISK FILE IS CLOSED
;
;           RETURNS (A) = 0 IF SUCCESSFUL,      0 OTHERWISE
;
0ABC 2A330C  WRTMEMF  LHLD  MEMADD
0ABF 77      MOV      M,A
0AC0 23      INX      H
0AC1 22330C  SHLD     MEMADD
0AC4 FE1A    CPI      EOD
0AC6 CA7A0B  JZ       MEMCLOS      ;IF END OF FILE
0AC9 3A390C  LDA      MEMCNT      ;MEMORY COUNT 0 .. 128
0ACC C601    ADI      1
0ACE FE80    CPI      80H
0AD0 CAD80A  JZ       SECFUL      ;IF LOGICAL SECTOR IS FULL
0AD3 32390C  STA      MEMCNT
0AD6 AF      XRA      A      ;SUCCESS
0AD7 C9      RET

;
0AD8 3A3D0C  SECFUL  LDA      BFSIZE
0ADB 47      MOV      B,A
0ADC 3A3A0C  LDA      SECNT      ;LOGICAL SECTOR FULL
0ADF C601    ADI      1      ;INCREMENT SECTOR COUNTER
0AE1 B8      CMP      B
0AE2 CA260B  JZ       BUFFUL      ;IF BUFFER FULL
0AE5 323A0C  STA      SECNT
0AE8 AF      XRA      A      ;RESET CHAR COUNTER
0AE9 32390C  STA      MEMCNT
0AEC C9      RET

;
0AED 070D0A534FBFLMSG  DB      BEL,CR,LF,'SOURCE FILE IS TOO LARGE.'
0B09 0D0A444553        DB      CR,LF,'DESTINATION FILE CLOSED.','CR,LF','$'

;
0B26 CDEE02  BUFFUL  CALL     STOPHST      ;HOST MUST STOP SENDING
0B29 3A3B0C  LDA      BFFL      ;SET BUFFER FULL FLAG
0B2C 3C      INR      A
0B2D 323B0C  STA      BFFL
0B30 21000D  LXI      H,OVLBF      ;PREPARE FOR OVERFLOW
0B33 22370C  SHLD     OVBFAD
0B36 3E00    GETSTRG  MVI      A,0      ;GET STRAGGLERS UNTIL TIMED OUT
0B38 323E0C  STA      TIME1
0B3B 3E1F    MVI      A,1FH
0B3D 323F0C  STA      TIME2
0B40 CD4F0C  GET1     CALL     RDRINPF
0B43 FE00    CPI      0
0B45 C25F0B  JNZ     GOTIT
0B48 3A3E0C  LDA      TIME1
0B4B 3D      DCR      A
0B4C 323E0C  STA      TIME1
0B4F C2400B  JNZ     GET1
0B52 3A3F0C  LDA      TIME2
0B55 3D      DCR      A
0B56 323F0C  STA      TIME2
0B59 C2400B  JNZ     GET1
0B5C C37A0B  JMP     MEMCLOS      ;ASSUME NO MORE COMING

;
0B5F 2A370C  GOTIT   LHLD     OVBFAD      ;PUT STRAGGLER IN OVERFLOW BUFFER,
0B62 47      MOV      B,A
0B63 77      MOV      M,A
0B64 23      INX      H      ;.....INCREMENT COUNTERS,
0B65 22370C  SHLD     OVBFAD

```

```

0B6B 3C          INR      A
0B6C 323C0C     STA      OVCNT
0B6F 78         MOV      A,B
0B70 FE0D       CPI      CR
0B72 C2360B     JNZ      GETSTRG
0B75 3E0A       MVI      A,LF
0B77 C35F0B     JMP      GOTIT

;
0B7A 3A3A0C     MEMCLOS LDA      SECNT
0B7D 47         MOV      B,A
0B7E 04         INR      B
0B7F 2A350C     LHL D   BUFADD
0B82 22330C     SHLD    MEMADD
0B85 3E80       MEML1  MVI      A,80H
0B87 2A330C     LHL D   MEMADD
0B8A 54         MOV      D,H
0B8B 5D         MOV      E,L
0B8C 83         ADD      E
0B8D 6F         MOV      L,A
0B8E D2950B     JNC      MEML4
0B91 3E01       MVI      A,1
0B93 82         ADD      D
0B94 67         MOV      H,A
0B95 22330C     MEML4  SHLD    MEMADD
0B98 C5         PUSH    B
0B99 CDAA0C     CALL    SETDMAF
0B9C 115C00     LXI     D,FCB
0B9F CDA00C     CALL    WRITSQF
0BA2 C1         POP     B
0BA3 B7         ORA     A
0BA4 C20D0C     JNZ     MEML3
0BA7 05         DCR     B
0BA8 AF         XRA     A
0BA9 80         ADD     B
0BAA C2850B     JNZ     MEML1

;
0BAD 3A3B0C     LDA     BFFL
0BB0 FE00       CPI     00H
0BB2 CAF70B     JZ      MEML2
0BB5 AF         XRA     A
0BB6 323B0C     STA     BFFL
0BB9 323A0C     STA     SECNT
0BBC 32390C     STA     MEMCNT
0BBF 2A350C     LHL D   BUFADD
0BC2 22330C     SHLD    MEMADD
0BC5 3A3C0C     LDA     OVCNT
0BC8 FE00       CPI     0
0BCA CAEF0B     JZ      HOSTGO
0BCD 3A3C0C     LDA     OVCNT
0BD0 47         MOV     B,A
0BD1 21000D     LXI     H,OVFLBF
0BD4 EB         XCHG
0BD5 2A330C     LHL D   MEMADD
0BD8 1A         MOVE    LDAX   D
0BD9 FE1A       CPI     EOD
0BDB CA7A0B     JZ      MEMCLOS
0BDE 77         MOV     M,A
0BDF 23         INX     H
0BE0 22330C     SHLD    MEMADD
0BE3 13         INX     D
0BE4 3A390C     LDA     MEMCNT
0BE7 3C         INR     A
0BE8 32390C     STA     MEMCNT
0BEB 05         DCR     B
0BEC C2D80B     JNZ     MOVE

;.....AND INSERT LINE FEEDS
; AFTER CARRIAGE RETURNS.

;WRITE MEMORY BUFFER TO DISK
;AND CLOSE DESTINATION FILE
;(WE COUNTED FROM 0)
;(B) COUNTS DOWN LOGICAL SECTORS
;RESET ADDRESS OF MEMORY BUFFER
;INCREMENT MEMADD BY 80H
;BYTES WRITTEN PER SECTOR

;IF DISK FULL

;IF NOT ALL SECTORS ARE WRITTEN

;HOW DID WE GET HERE?

;.....BECAUSE OF EOF
;.....BECAUSE OF FULL BUFFER
;INDICATE BUFFER NOT FULL
;ZERO SECTOR COUNT
; AND CHAR COUNT
;RESET ADDRESS OF MEMORY BUFFER

;ARE THERE ANY
OVERFLOW CHARACTERS?
;NO
;YES...SO MOVE THEM INTO
; THE RECEIVER BUFFER.

;OOPS..END OF DATA IN OVERFLOW

```

```

0BE1 CD0003 ;ROSIGO CALL SIRINSI ;TELL HOST TO START SENDING AGAIN
0BF2 AF XRA A
0BF3 323C0C STA OVCNT ;ZERO OVERFLOW COUNT
0BF6 C9 RET

;
0BF7 115C00 MEML2 LXI D,FCB ;DONE WRITING. CLOSE FILE
0BFA CD910C CALL CLOSEFF
0BFD AF XRA A ;RESET ALL COUNTERS AND ADDRESSES
0BFE 32390C STA MEMCNT
0C01 323A0C STA SECNT
0C04 2A350C LHLD BUFADD
0C07 22330C SHLD MEMADD
0C0A C601 ADI 1 ;FORCE FLAG
0C0C C9 RET

;
0C0D 11160C MEML3 LXI D,LOSTMG ;DISK FULL
0C10 CD820C CALL STRINGF
0C13 C3F70B JMP MEML2

;
0C16 070D0A4449LOSTMG DB BEL,CR,LF,'DISK FULL. FILE CLOSED',CR,LF,'$'

;
0C33 000F MEMADD DW BUFF
0C35 000F BUFADD DW BUFF
0C37 000D OVBFAD DW OVFLBF
0C39 00 MEMCNT DB 0
0C3A 00 SECNT DB 0
0C3B 00 BFFL DB 0
0C3C 00 OVCNT DB 0
0C3D 00 BFSIZE DB 00H ;INITIALIZED FOR 256 SECTORS
0C3E 00 TIME1 DB 0
0C3F 00 TIME2 DB 0

;
;
; FUNCTION 1: CONSOLE INPUT
;
; RETURN: (A) = ASCII CHARACTER
;
0C40 DBED CONINPF IN 350Q+5
0C42 E601 ANI 1
0C44 CA400C JZ CONINPF
0C47 DBE8 IN 350Q
0C49 C9 RET

;
;CONINPF MVI C,1
; JMP BDOS

;
; FUNCTION 2: CONSOLE OUTPUT
;
; ENTRY: (E) = ASCII CHARACTER
;
0C4A 0E02 CONOUTF MVI C,2
0C4C C30500 JMP BDOS

;
; FUNCTION 3: READER INPUT
;
; RETURN: (A) = ASCII CHARACTER
;
;RDRINPF IN 330Q+5
; ANI 1
; RZ
; IN 330Q
; ANI 7FH
0C4F 0E03 RDRINPF MVI C,3
0C51 CD0500 CALL BDOS

```

```

0C54 E67F ANI 7FH
0C56 C8 RZ
0C57 FE7F CPI DEL
0C59 CA640C JZ IGNOR
0C5C FE07 CPI BEL
0C5E CA640C JZ IGNOR
0C61 FE0A CPI LF
0C63 C0 RNZ
0C64 AF IGNOR XRA A
0C65 C9 RET

```

```

;
;
;
; FUNCTION 4: PUNCH OUTPUT
;
; ENTRY: (E) = ASCII CHARACTER
;
;PUNOUTF IN 330Q+5
; ANI 20H
; JZ PUNOUTF
; MOV A,E
; OUT 330Q
; RET
0C66 0E04 PUNOUTF MVI C,4
0C68 C30500 JMP BDOS

```

```

;
; PUNCH STATUS
;
0C6B DBDD PUNSTAT IN 330Q+5
0C6D E620 ANI 20H
0C6F C9 RET

```

```

;
; FUNCTION 5: LIST OUTPUT
;
; ENTRY: (E) = ASCII CHARACTER
;
0C70 0E05 LSTOUTF MVI C,5
0C72 C30500 JMP BDOS

```

```

;
; CHECK PRINTER STATUS
;
0C75 DB41 PRTSTAT IN 1010 340Q+5
0C77 E601 ANI 01H 20H
0C79 C9 RET RZ

```

*Change back to serial printer part*

*IN 340Q+6  
XRI FFH  
ANI 10H  
RET*

```

;
; FUNCTION 6: DIRECT CONSOLE I/O
;
; ENTRY: (E) = 0FFH FOR INPUT FROM CONSOLE
; = ASCII CHAR FOR OUTPUT TO CONSOLE
; RETURN: (A) = ASCII CHAR, IF ANY
; = 00 IF NO CHAR WAS ENTERED
;
; IF BDOS VERSION DOES NOT SUPPORT FUNCTION 6
;DIRCONF MVI C,11 ;GET CONSOLE STATUS
; CALL BDOS
; RZ ;NOTHING WAITING
; MVI C,1 ;CONSOLE READ (BETTER IF NO ECHO)
; CALL BDOS
; ENDIF

```

```

0C7A DBED DIRCONF IN 350Q+5
0C7C E601 ANI 1
0C7E C8 RZ
0C7F DBE8 IN 350Q

```



```

0C81 C9      RET
;
;DIRCONF   MVI    C,6
;          JMP    BDOS
;
;
; FUNCTION 9: PRINT STRING
;
;          ENTRY: (DE) = ADDRESS OF MESSAGE
;                  MESSAGE MUST END WITH '$'
;
0C82 0E09   STRINGF MVI    C,9
0C84 C30500 JMP    BDOS
;
;
; FUNCTION 10: READ CONSOLE BUFFER
;
;          ENTRY:  (DE) = BUFFER ADDRESS
;          RETURN: ((DE)) = CONSOLE CHARACTERS
;
0C87 0E0A   READCNF MVI    C,10
0C89 C30500 JMP    BDOS
;
;
; FUNCTION 15: OPEN FILE
;
;          ENTRY:  (DE) = FCB ADDRESS
;          RETURN: (A) = 0 - 3 IF SUCCESS, OFFH OTHERWISE
;
0C8C 0E0F   OPENFIF MVI    C,15
0C8E C30500 JMP    BDOS
;
;
; FUNCTION 16: CLOSE FILE
;
;          ENTRY:  (DE) = FCB ADDRESS
;          RETURN: (A) = 0 - 3 IF SUCCESS, OFFH OTHERWISE
;
0C91 0E10   CLOSEFF MVI    C,16
0C93 C30500 JMP    BDOS
;
;
; FUNCTION 19: DELETE FILE
;
;          ENTRY:  (DE) = FCB ADDRESS
;          RETURN: (A) = 0 - 3 IF SUCCESS, OFFH OTHERWISE
;
0C96 0E13   DELETEF MVI    C,19
0C98 C30500 JMP    BDOS
;
;
; FUNCTION 20: READ SEQUENTIAL
;
;          ENTRY:  (DE) = FCB ADDRESS
;          RETURN: (A) = 0 IF SUCCESS, ELSE NONZERO
;
0C9B 0E14   READSQF MVI    C,20
0C9D C30500 JMP    BDOS
;
;
; FUNCTION 21: WRITE SEQUENTIAL
;
;          ENTRY:  (DE) = FCB ADDRESS
;          RETURN: (A) = 0 IF SUCCESS, ELSE NONZERO
;
0CA0 0E15   WRITSQF MVI    C,21

```

```
0CA2 C30500      JMP      BDOS
;
;
; FUNCTION 22:  MAKE FILE - OPEN FOR OUTPUT
;
; ENTRY:      (DE) = FCB ADDRESS
; RETURN:     (A) = 0 - 3 IF SUCCESS, ELSE 0FFH
;
0CA5 0E16      MAKEFIF MVI      C,22
0CA7 C30500      JMP      BDOS
;
;
; FUNCTION 26:  SET DMA ADDRESS
;
; ENTRY:      (DE) = DMA ADDRESS
;
0CAA 0E1A      SETDMAF MVI      C,26
0CAC C30500      JMP      BDOS
;
;
;
0CAF                      END
```