

II DIGITAL RESEARCH

Post Office Box 579, Pacific Grove, California 93950, (408) 373-3403

CONSOLE COMMAND PROCESSOR (CCP)

CP/M VERSION _____

COPYRIGHT © 1976

DIGITAL RESEARCH

P. O. BOX 579

PACIFIC GROVE, CA. 93950

SER. # _____

CCP

```

0000 = BIAS EQU 800H
; ASSEMBLY LANGUAGE VERSION. OF THE CP/M CONSOLE COMMAND PROCESSOR
; VERSION. 1.1 OCTOBER, 1976
;
; COPYRIGHT (C) 1976
; DIGITAL RESEARCH
;
0000 = FALSE EQU 0000H
FFFF = TRUE EQU NOT FALSE
0000 = TESTING EQU FALSE ; TRUE IF DEBUGGING
0000 = CCPLOC EQU BIAS ; BASE ADDRESS OF CCP
0900 = BDOSL EQU 900H+BIAS ; BASE OF DOS
;
; IF TESTING
; ORG 100H
; MVI C,0
TRAN EQU 01000H ; TRANSIENT PROGRAM AREA BASE (TPA)
TRANM EQU 1400H ; LAST TRANSIENT LOCATION. + 1
; ENDDIF
;
; IF NOT TESTING
; ORG CCPLOC
; MVI C,1
; ENDDIF
;
0000 C3E702 JMP CCPSTART
0003 434F505952 DB 'COPYRIGHT (C) 1976, DIGITAL RESEARCH'
;
0004 = DISKA EQU 0004H ; DISK ADDRESS FOR CURRENT DISK
0005 = BDOS EQU 0005H ; PRIMARY BDOS ENTRY POINT
0006 = BUFF EQU 0006H ; DEFAULT BUFFER
005C = FCB EQU 005CH ; DEFAULT FILE CONTROL BLOCK
;
0001 = RCHARF EQU 1 ; READ CHARACTER FUNCTION
0002 = PCHARF EQU 2 ; PRINT CHARACTER FUNCTION
0009 = PBUF EQU 9 ; PRINT BUFFER FUNCTION
000A = RBUF EQU 10 ; READ BUFFER FUNCTION
000B = BREAKF EQU 11 ; BREAK KEY FUNCTION
000C = LIFTF EQU 12 ; LIFT HEAD FUNCTION. (SHUGART SA3900 ONLY)
000D = INITF EQU 13 ; INITIALIZE BDOS FUNCTION
000E = SELF EQU 14 ; SELECT DISK FUNCTION
000F = OPENF EQU 15 ; OPEN FILE FUNCTION
0010 = CLOSEF EQU 16 ; CLOSE FILE FUNCTION
0011 = SEARF EQU 17 ; SEARCH FOR FILE FUNCTION
0012 = SEARNF EQU 18 ; SEARCH FOR NEXT FILE FUNCTION
0013 = DELF EQU 19 ; DELETE FILE FUNCTION
0014 = DREADF EQU 20 ; DISK READ FUNCTION
0015 = DWRITE EQU 21 ; DISK WRITE FUNCTION
0016 = MAKEF EQU 22 ; FILE MAKE FUNCTION
0017 = RENF EQU 23 ; RENAME FILE FUNCTION

```

CP/M VERSION _____
 COPYRIGHT © 1976
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA. 93950
 SER. # _____

CP/M VERSION _____
 COPYRIGHT © 1976
 DIGITAL RESEARCH
 P. O. BOX 579
 PACIFIC GROVE, CA. 93950
 SER. # _____

```

0018 = LOGF EQU 24 ; RETURN. LOGIN VECTOR
0019 = CSELF EQU 25 ; RETURN. CURRENTLY SELECTED DRIVE NUMBER
001A = DMAF EQU 26 ; SET DMA ADDRESS
;
000D = CR EQU 13 ; CARRIAGE RETURN
000A = LF EQU 10 ; LINE FEED
005F = LA EQU 5FH ; LEFT ARROW
001A = EOFILE EQU 1AH ; END OF FILE
0002 = NDISKS EQU 2 ; NUMBER OF DISKS
;
; UTILITY PROCEDURES
; PRINTCHAR,
0027 5F0E02C305 MOV E,A! MVI C,PCHARF! JMP BDOS
; PRINTBC,
; PRINT CHARACTER, BUT SAVE B,C REGISTERS
002D C5CD2700C1 PUSH B! CALL PRINTCHAR! POP B! RET
; CRLF,
0033 3E0DCD2700 MVI A,C! CALL PRINTCHAR
0038 3E0AC32700 MVI A,L! JMP PRINTCHAR
; PRINT,
; PRINT STRING STARTING AT B,C UNTIL NEXT DOLLAR SYMBOL
003D C5CD3300D1 PUSH B! CALL CRLF! POP B! MVI C,PBUFF! JMP BDOS
; INITIALIZE,
0047 0E0DC30500 MVI C,INITF! JMP BDOS
; SELECT,
004C 5F0E0EC305 MOV E,A! MVI C,SELF! JMP BDOS
; OPEN,
; OPEN THE FILE GIVEN. BY D,E
0052 0E0FCD0500 MVI C,OPENF! CALL BDOS! STA DCNT! INR A! RET
; OPENC,
; OPEN. COMFCB
005C AF324007 XRA A! STA COMREC ; CLEAR NEXT RECORD TO READ
0060 112007C352 LXI D,COMFCB! JMP OPEN
; CLOSE,
; CLOSE THE FILE GIVEN. BY D,E
0066 0E10CD0500 MVI C,CLOSEF! CALL BDOS! STA DCNT! INR A! RET
; SEARCH,
; SEARCH FOR THE FILE GIVEN BY D,E
0070 0E11CD0500 MVI C,SEARF! CALL BDOS! STA DCNT! INR A! RET
; SEARCHN,
; SEARCH FOR THE NEXT OCCURRENCE OF THE FILE GIVEN. BY D,E
007A 0E12CD0500 MVI C,SEARNF! CALL BDOS! STA DCNT! INR A! RET
; SEARCHCOM,
; SEARCH FOR COMFCB FILE
0084 112007C370 LXI D,COMFCB! JMP SEARCH
; DELETE,
; DELETE THE FILE GIVEN. BY D,E
008A 0E13CD0500 MVI C,DELF! JMP BDOS
; DISKREAD,
; READ THE NEXT RECORD FROM THE FILE GIVEN. BY D,E
000F 0E14CD0500 MVI C,DREADF! CALL BDOS! ORA A! RET
; DISKREADC,

```

```

>
> 0096 112007C38F ;READ THE COMFCB FILE
;LXI D,COMFCB; JMP DISKREAD
;
; DISKWRITE.
;WRITE THE NEXT RECORD TO THE FILE GIVEN. BY D,E
009C 0E15CD0500 MVI C,DURITF; CALL BDOS; ORA A; RET
;
; MAKE. ;CREATE THE FILE GIVEN. BY D,E
00A3 0E16CD0500 MVI C,MAKEF; CALL BDOS; STA DCHT; RET
;
; RENAM. ;RENAME THE FILE GIVEN BY D,E
00AC 0E17C30500 MVI C,RENFI; JMP BDOS
;
; TRANSLATE.
;TRANSLATE CHARACTER INH REGISTER A TO UPPER CASE
0091 FE61D8 CPI 61H; RC ;RETURN. IF BELOW LOWER CASE A
0094 FE70D0 CPI 70H; RNC ;RETURN IF ABOVE LOWER CASE Z
0087 E65FC9 ANI 5FH; RET ;TRANSLATED TO UPPER CASE
;
; READCOM.
;READ THE NEXT COMMAND INTO THE COMMAND BUFFER
;CHECK FOR SUBMIT FILE
008A 3AFE06B7CA LDA SUBMIT; ORA A; JZ NOSUB
;SCANNING A SUBMIT FILE
;CHANGE DRIVES TO OPENH AND READ THE FILE
00C1 3AC087873E LDA CDISK; ORA A; MVI A,0; CNZ SELECT
00CA 3A8E073D LDA SUBRC; DCR A ;READ LAST RECORD(S) FIRST
00CE 321F07 STA SUBCR ;CURRENT RECORD TO READ
00D1 11FF06CD0F LXI D,SUBFCB; CALL DISKREAD ;END OF FILE IF LAST RECORD
00D7 C20D01 JNZ NOSUB
;DISK READ IS OK, TRANSFER TO COMBUF
00DA 1142072180 LXI D,COMLEN; LXI H,BUFF; MVI B,120; CALL MOVE0
;LINE IS TRANSFERRED, CLOSE THE FILE WITH A
;DELETED RECORD
00E5 210E0735 LXI H,SUBRC; DCR H ;ONE LESS RECORD
00E9 11FF06CD66 LXI D,SUBFCB; CALL CLOSE; JZ NOSUB
;CLOSE WENT OK, RETURN. TO ORIGINAL DRIVE
00F2 3AC087B7C4 LDA CDISK; ORA A; CNZ SELECT
00F9 0E09114307 MVI C,PBUFF; LXI D,COMBUF; CALL BDOS
0101 CD3301CA18 CALL BREAK*KEY; JZ WOREAD
0107 CD5501C3FC CALL DEL$SUB; JMP CCP ;BREAK KEY DEPRESSED
;
;NOSUB. ;NO SUBMIT FILE; CALL DEL$SUB
;TRANSLATE TO UPPER CASE, STORE ZERO AT END
010D CD5901 MVI C,RBUFF; LXI D,MAXLEN; CALL BDOS
;ENTER HERE FROM SUBMIT FILE
;SET THE LAST CHARACTER TO ZERO FOR LATER SCANS
0110 0E0A114107 LXI H,COMLEN; MOV B,M ;LENGTH IS IN. B
;READCOM0. INX H; MOV A,B; ORA A ;END OF SCAN?
0116 21420746 JZ READCOM1; MOV A,M ;GET CHARACTER AND TRANSLATE
011C 237007 CPI ' ' ;CALL PRINTCHAR AND TRANSLATE
011F CA20017E CALL TRANSLATE; MOV M,A; DCR B; JMP READCOM0
0123 CDB1007705
;
;READCOM1. ;END OF SCAN, H,L ADDRESS END OF COMMAND
;MOV M,A ;STORE A ZERO
012B 77 LXI H,COMBUF; SHLD COMADDR ;READY TO SCANN TO ZERO
012C 21430722C3
0132 C9 RET
;
;BREAK*KEY.
;CHECK FOR A CHARACTER READY AT THE CONSOLE
0133 0E06CD0500 MVI C,BREAKF; CALL BDOS

```

```

0130 B7C0 ORA A; RZ
013A 0E01CD0500 MVI C,RCHARF; CALL BDOS ;CHARACTER CLEARED
013F B7C9 ORA A; RET
;
;LIFTHEAD.
;LIFT THE HEAD ON THE SA3900
0141 0E0CC30500 MVI C,LIFTF; JMP BDOS
;
;CSELECT.
;GET THE CURRENTLY SELECTED DRIVE NUMBER TO REG-A
0146 0E19C30500 MVI C,CSELF; JMP BDOS
;
;MOVE. ;MOVE DATA FROM D,E TO H,L FOR C BYTES
014B 1A7713230D LDAX DI ;MOV M,A; INX DI ;INX HI ;DCR CI ;JNZ MOVE; RET
;
;SETDMA.
;SET DMA ADDRESS TO D,E
0154 0E1AC30500 MVI C,DMAF; JMP BDOS
;
;DEL$SUB.
;DELETE THE SUBMIT FILE, AND SET SUBMIT FLAG TO FALSE
0159 21FE067EB7 LXI H,SUBMIT; MOV A,M; ORA A; RZ ;RETURN. IF NO SUB FILE
015F 3600 MVI M,0 ;SUBMIT FLAG IS SET TO FALSE
0161 AFC04C00 XRA A; CALL SELECT ;ON. DRIVE A TO ERASE FILE
0165 11FF06CD0A LXI D,SUBFCB; CALL DELETE
016B 3AC007C34C LDA CDISK; JMP SELECT ;BACK TO ORIGINAL DRIVE
;
;SERIALIZE.
;CHECK SERIALIZATION
0171 11A0022100 LXI D,SERIAL; LXI H,BDOSL; MVI B,6 ;CHECK SIX BYTES
0179 1ABEC24403 SER0. LDAX DI ;CMP M; JNZ BADSERIAL
017E 132305C279 INX DI ;INX HI ;DCR B; JNZ SER0
0184 C9 RET ;SERIAL NUMBER IS OK
;
;COMERR.
;ERROR IN. COMMAND STRING STARTING AT POSITION
;'STADDR' AND ENDING WITH FIRST DELIMITER
0185 CD3300 CALL CRLF ;SPACE TO NEXT LINE
0188 2AC507 LHLD STADDR ;H,L ADDRESS FIRST TO PRINT
COMERR0. ;PRINT CHARACTERS UNTIL BLANK OR ZERO
018B 7EFE20C9E MOV A,M; CPI ' ' ;JZ COMERR1; NOT BLANK
0191 B7C9E01 ORA A; JZ COMERR1; NOT ZERO, SO PRINT IT
0195 E5CD2700E1 PUSH H; CALL PRINTCHAR; POP H; INX H
019B C38001 JMP COMERR0 ;FOR ANOTHER CHARACTER
COMERR1. ;PRINT QUESTION. MARK,AND DELETE SUB FILE
019E 3E3FCD2700 MVI A,'?' ;CALL PRINTCHAR
01A3 CD3300CD59 CALL CRLF; CALL DEL$SUB
01A9 C3FC02 JMP CCP ;RESTART WITH NEXT COMMAND
;
;FCB SCAN. AND FILL SUBROUTINE (ENTRY IS AT FILLFCB BELOW)
;FILL THE COMFCB, INDEXED BY A (0 OR 16)
;SUBROUTINES
;DELIM. ;LOOK FOR A DELIMITER
01AC 1AB7C0 LDAX DI ;ORA A; RZ ;NOT THE LAST ELEMENT
01AF FE20DA9501 CPI ' ' ;JC COMERR ;NON. GRAPHIC
01B4 C9 RZ ;TREAT BLANK AS DELIMITER
01B5 FE3DC0 CPI '=' ;RZ
01B8 FE5FC0 CPI LAI ;RZ ;LEFT ARROW
01BB FE2EC0 CPI '.' ;RZ
01BE FE3AC0 CPI '!' ;RZ

```

```

01C1 FE3BC8      CPI ' ' RZ
01C4 FE3CC8      CPI '<' RZ
01C7 FE3EC8      CPI '>' RZ
01CA C9          RET      DELIMITER NOT FOUND

```

```

DEBLANK: DEBLANK THE INPUT LINE
LDAX D1 ORA A1 RZ ;TREAT END OF LINE AS BLANK
CPI ' ' RZ ;DELIMITER NOT FOUND
JMP DEBLANK

```

```

ADDH: ADD A TO H.L
ADD L1 MOV L,A1 RNC
INR H1 RET

```

```

FILLFCB:
EQUIVALENT TO FILLFCB(8)
MVI A,8

```

```

FILLFCB:
LXI H,COMFCB1 CALL ADDH1 PUSH H1 PUSH H ;FCB RESEARCHED AT END
XRA A1 STA SDISK ;CLEAR SELECTED DISK (IN. CASE A1...)
LMDL COMADDR1 XCHG ;COMMAND ADDRESS IN. D,E
CALL DEBLANK ;TO FIRST NON-BLANK CHARACTER
XCHG1 SHLD STADDR ;IN. CASE OF ERRORS
XCHG1 POP H ;D,E HAS COMMAND, H,L HAS FCB ADDRESS
;LOOK FOR PRECEDING FILE NAME A, B, ...
LDAX D1 ORA A1 JZ SETCUR0 ;USE CURRENT DISK IF EMPTY COMMAND
SBI 'A'-11 MOV B,A ;DISK NAME HELD IN. B IF ;FOLLOWS
INX D1 LDAX D1 CPI ' ' JZ SETDSK ;SET DISK NAME IF ;

```

```

SETCUR: ;SET CURRENT DISK
DCX D ;BACK TO FIRST CHARACTER OF COMMAND

```

```

SETCUR0: LDA CDISK1 MOV M,A1 JMP SETNAME

```

```

SETDSK: ;SET DISK TO NAME IN. REGISTER B
MOV A,B1 STA SDISK ;MARK AS DISK SELECTED
MOV M,B1 INX D ;PAST THE ;

```

```

SETNAME: ;SET THE FILE NAME FIELD
MVI B,8 ;FILE NAME LENGTH (MAX)
SETNAM0: CALL DELIM1 JZ PADNAME ;NOT A DELIMITER
INX H1 CPI ' ' JNZ SETNAM1 ;MUST BE '?'S
MVI M,'?'1 JMP SETNAM2 ;TO DEC COUNT

```

```

SETNAM1: MOV M,A ;STORE CHARACTER TO FCB1 INX D
SETNAM2: DCR B ;COUNT DOWN. LENGTH1 JNZ SETNAM0

```

```

;END OF NAME, TRUNCATE REMAINDER
TRNAME: CALL DELIM1 JZ SETTY ;SET TYPE FIELD IF DELIMITER
INX D1 JMP TRNAME

```

```

PADNAME: INX H1 MVI M,' '1 DCR B1 JNZ PADNAME

```

```

SETTY: ;SET THE TYPE FIELD
MVI B,31 CPI ' '1 JNZ PADTY ;SKIP THE TYPE FIELD IF NO ;
INX D ;PAST THE ; TO THE FILE TYPE FIELD
SETTY0: ;SET THE FIELD FROM THE COMMAND BUFFER
CALL DELIM1 JZ PADTY1 INX H1 CPI ' '1 JNZ SETTY1
MVI M,'?' ;SINCE * SPECIFIED1 JMP SETTY2

```

```
0255 7713
```

```
0257 05C24402
```

```
025B CDAC01CA6C
```

```
0265 23362005C2
```

```
026C 0603
026E 23360005C2
0275 EB22C77
```

```
0279 E1010B00
027D 237EFE3FC2
0284 04
0285 0DC27D02
```

```
0289 78B7C9
```

INTVEC:

```
028C 44495220
0290 45524120
0294 54595045
0298 53415645
029C 52454E20
0005 =
02A0 0000000000
```

INTRINSIC:

```
02A6 218C020E00
02AB 79FE05D0
```

```
02AF 112107
02B2 0604
02B4 1A8E
02B6 C2C702
02B9 132305
02BC C2B402
```

```
02BF 1AFE20C2CC
02C5 79C9
```

```
02C7 2305C2C702
```

```
02CC 0C
02CD C3AB02
```

```
SETTY1: ;NOT A *, SO COPY TO TYPE FIELD
MOV M,A1 INX D

```

```
SETTY2: ;DECREMENT COUNT AND GO AGAIN
DCR B1 JNZ SETTY0

```

```
TRTYP: ;END OF TYPE FIELD, TRUNCATE
TRUNCATE TYPE FIELD
CALL DELIM1 JZ EFILL1 INX D1 JMP TRTYP

```

```
PADTY: ;PAD THE TYPE FIELD WITH BLANKS
INX H1 MVI M,' '1 DCR B1 JNZ PADTY

```

```
EFILL: ;END OF THE FILENAME/FILETYPE FILL, SAVE COMMAND ADDRESS;
;FILL THE REMAINING FIELDS FOR THE FCB
MVI B,3

```

```
EFILL0: INX H1 MVI M,01 DCR B1 JNZ EFILL0
KCHG1 SHLD COMADDR ;SET NEW STARTING POINT

```

```
;RECOVER THE START ADDRESS OF THE FCB AND COUNT ?'S
POP H1 LXI B,11 ;B=0, C=8+3
SCHQ: INX H1 MOV A,M1 CPI '?'1 JNZ SCHQ0
;? FOUND, COUNT IT IN. B1 INR B
SCHQ0: DCR C1 JNZ SCHQ

```

```
;NUMBER OF ?'S INH C, MOVE TO A AND RETURN. WITH FLAGS
MOV A,B1 ORA A1 RET

```

```
INTRINSIC FUNCTION. NAMES (ALL ARE FOUR CHARACTERS)
```

```
DB 'DIR '
DB 'ERA '
DB 'TYPE'
DB 'SAVE'
DB 'REN '
INTLEN. EQU (I-INTVEC)/4 ;INTRINSIC FUNCTION. LENGTH
SERIAL: DB 0,0,0,0,0,0

```

```
LOOK FOR INTRINSIC FUNCTIONS (COMFCB HAS BEEN. FILLED)
```

```
LXI H,INTVEC1 MVI C,0 ;C COUNTS INTRINSICS AS SCANNED
INTRIN0: MOV A,C1 CPI INTLEN. ;DONE WITH SCAN?!! RNC
;NO. MORE TO SCAN
LXI D,COMFCB+1 ;BEGINNING OF NAME
MVI B,4 ;LENGTH OF MATCH IS IN. B
INTRIN1: LDAX D1 CMP M ;MATCH?
JNZ INTRIN2 ;SKIP IF NO MATCH
INX D1 INX H1 DCR B
JNZ INTRIN1 ;LOOP WHILE MATCHING

```

```
;COMPLETE MATCH ON. NAME. CHECK FOR BLANK IN. FCB
LDAX D1 CPI ' '1 JNZ INTRIN3 ;OTHERWISE MATCHED
MOV A,C1 RET ;WITH INTRINSIC NUMBER IN. A

```

```
INTRIN2: ;MISMATCH, MOVE TO END OF INTRINSIC
INX H1 DCR B1 JNZ INTRIN2

```

```
INTRIN3: ;TRY NEXT INTRINSIC
INR C ;TO NEXT INTRINSIC NUMBER
JMP INTRIN0 ;FOR ANOTHER ROUND

```

```

CHECKSUB.
02D8 B7C2E282 ;CHECK FOR A SUBMIT FILE (REGISTER A HAS DISK NAME)
ORA A! JNZ SETSUB00 ;CLEAR SUBMIT IF NOT DISK A
;OPENN SUBMIT FILE - IF POSSIBLE
02D4 11FF06C052 LXI D,SUBFCB01 CALL OPENI JZ SETSUB00
;SUBMIT FILE OPENED, SET SUBMIT TO TRUE
MVI A,0FFH! JMP SETSUB

02DD 3EFFC3E302
;
SETSUB00:
02E2 AF XRA A ;CLEAR SUBMIT VARIABLE
02E3 32FE06 SETSUB. STA SUBMIT
02E6 C9 RET

```

```

CCPSTART.
02E7 31FE06C5C5 ;ENTER HERE FROM BOOT LOADER
02EC CD4700C1 LXI SP,STACK! PUSH B! PUSH 0 ;SAVE INITIAL DISK NUMBER
02F0 F5 CALL INITIALIZE! POP B ;RECOVERED DISK NUMBER TO C
02F1 79 PUSH PSU ;SAVE * FLAG FROM INITIALIZATION. (00 IF NO * FILE)
02F2 CD4C00 MOV A,C ;DISK NUMBER IN. ACCUMULATOR
02F5 F1 CALL SELECT ;PROPER DISK IS SELECTED, NOW CHECK SUB FILES
02F6 C1 POP PSU ;RECOVER * FILE FLAG FROM INITIALIZATION
02F7 3CB1 POP B ;ORIGINAL DISK NUMBER IS IN. C
;INR A ;00 BECOMES 01, FF BECOMES 00! ORA C ;OR DISK NUMBER
;ACCUMULATOR CONTAINS 00 IF ONN DISK 0, AND * FILE PRESENT
;OTHERWISE, THE ACCUMULATOR CONTAINS A NON-ZERO VALUE
02F9 CDD002 CALL CHECKSUB

```

```

CCP.
02FC 31FE06 ;ENTER HERE ON. EACH COMMAND OR ERROR CONDITION
02FF CD3300 LXI SP,STACK
0302 CD450132C0 CALL CRLF ;PRINT D> PROMPT, WHERE D IS DISK NAME
0308 C641CD2700 CALL CSELECT! STA CDISK!
030D 3E3E0D2700 ADI 'A'! CALL PRINTCHAR
0312 110000CD54 MVI A,'>'! CALL PRINTCHAR
0318 CDE400CDDA LXI D,BUFF! CALL SETDMA ;DEFAULT DMA ADDRESS IS AT BUFF
031E C4A501 CALL READCOM! CALL FILLFCB0
0321 3AC907B7C2 CNZ COMERR ;THE NAME CANNOT BE ANH AMBIGUOUS REFERENCE
LDA SDISK! ORA A! JNZ USERFUNC

```

```

;CHECK FOR AN. INTRINSIC FUNCTION
CALL INTRINSIC
0329 CDA602 LXI H,JMPTAB ;INDEX IS IN. THE ACCUMULATOR
032B 213803 MOV E,A! MVI D,0! DAD D! DAD D ;INDEX INN D,E
032E 5F16001919 MOV A,M! INX H! MOV H,M! MOV L,A! PCHL
0333 7E23666FE9 ;JPC CHANGES TO THE PROPER INTRINSIC OR USER FUNCTION
JMPTAB.

```

```

0338 E403 DW DIRECT ;DIRECTORY SEARCH
033A 5F04 DW ERASE ;FILE ERASE
033C A404 DW TYPE ;TYPE FILE
033E F404 DW SAVE ;SAVE MEMORY IMAGE
0340 6F05 DW RENAME ;FILE RENAME
0342 ED05 DW USERFUNC;USER-DEFINED FUNCTION

```

```

BADSERIAL.
0344 21F37E LXI H,D1 OR (HLT SHL 0)
0347 2200002100 SHLD CCPLOC1 LXI H,CCPLOC1 PCHL

```

```

;UTILITY SUBROUTINES FOR INTRINSIC HANDLERS
READERR.
;PRINT THE READ ERROR MESSAGE

```

```

034E 015403C33D
0354 5245414420

```

```

035F 016503C33D
0365 4E4F542046

```

```

036F 2121070100

```

```

0375 7EFE20CAA0

```

```

037B 23D630FE0A
0383 5778
0385 E6E0C285J1
038A 78
038B 070707
038E 80DA0501
0392 80DA0501
0396 82DA0501
039A 470DC27503
039F C9

```

```

03A0 7EFE20C205
03A6 230DC2A003
03AB 78C9

```

```

03AD 0603
03AF 7E122313
0303 05C2AF03
0307 C9

```

```

0388 218000B1CD

```

```

03C1 AF322007
03C5 3AC907B7C8
03CA 3D21C007BE
03D0 C34C00

```

```

03D3 3AC907B7C8
03D8 3D21C007BE
03DE 3AC007C34C

```

```

03E4 CDDA01
03E7 CDC103
03EA 2121077E
03EE FE20C2FC03

```

```

LXI B,RDMSG! JMP PRINT
RDMSG: DB 'READ ERROR*'

```

```

;NOFILE.
;PRINT NO FILE MESSAGE
LXI B,NOFMSG! JMP PRINT
NOFMSG: DB 'NOT FOUND*'

```

```

;CONVERT.
;CONVERT THE BYTE VALUE IN. COMFCB TO BINARY
LXI H,COMFCB+1! LXI B,11 ;(B=0, C=11)
;VALUE ACCUMULATED INN B, C COUNTS NAME LENGTH TO ZERO
CONV0. MOV A,M! CPI ' ' ! JZ CONVI
;MORE TO SCAN, CONVERT CHAR TO BINARY AND ADD
INX H! SUI '0'! CPI 10! JNC COMERR ;VALID?
MOVV D,A ;SAVE VALUE! MOV A,B ;MULT BY 10
ANI 11000000B! JNZ COMERR
MOV A,B ;RECOVER VALUE
RLC! RLC! RLC ;*8
ADD B! JC COMERR
ADD B! JC COMERR ;*8+*2 = *10
ADD B! JC COMERR ;+DIGIT
MOV B,A! DCR C! JNZ CONV0 ;FOR ANOTHER DIGIT
RET
CONVI. ;END OF DIGITS, CHECK FOR ALL BLANKS
MOV A,M! CPI ' ' ! JNZ COMERR ;BLANKS?
INX H! DCR C! JNZ CONVI
MOV A,B ;RECOVER VALUE! RET

```

```

;MOVENAME.
;MOVE 3 CHARACTERS FROM H,L TO D,E ADDRESSES
MVI B,3
MOVE0. MOV A,M! STAX D! INX H! INX D
DCR B! JNZ MOVE0
RET

```

```

;ADDDCF. ;BUFF + A + C TO H,L FOLLOWED BY FETCH
LXI H,BUFF! ADD C! CALL ADDH! MOV A,M! RET

```

```

;SETDISK.
;CHANGE DISKS FOR THIS COMMAND, IF REQUESTED
XRA A! STA COMFCB ;CLEAR DISK NAME FROM FCB
LDA SDISK! ORA A! RZ ;NO ACTION. IF NOT SPECIFIED
DCR A! LXI H,CDISK! CMP M! RZ ;ALREADY SELECTED
JMP SELECT

```

```

;RESETDISK.
;RETURN. TO ORIGINAL DISK AFTER COMMAND
LDA SDISK! ORA A! RZ ;NO ACTION. IF NOT SELECTED
DCR A! LXI H,CDISK! CMP M! RZ ;SAME DISK
LDA CDISK! JMP SELECT

```

```

;INDIVIDUAL INTRINSICS FOLLOW

```

```

DIRECT.
;DIRECTORY SEARCH
CALL FILLFCB0 ;COMFCB GETS FILE NAME
CALL SETDISK ;CHANGE DISK DRIVES IF REQUESTED
LXI H,COMFCB+1! MOV A,M ;MAY BE EMPTY REQUEST
CPI ' ' ! JNZ DIR1 ;SKIP FILL OF ??? IF NOT BLANK
;SET COMFCB TO ALL ??? FOR CURRENT DISK

```

03F3 0608
03F5 363F2385C2

03FC CD9400
03FF C05F03
0402 CA5C04
0405 CD3300

0400 CD4501C641
0410 3E3ACD2700
0415 3E20CD2700

041A 3AC707
041D 0F0F0FE660

0423 0601
0425 78C0B003

0429 FE20C23904
042E 3E03C0B003
0433 3E20BECA50

0439 CD2D00
043C 0478FE0CD2

0443 FE09C22504

0448 3E20CD2D00

0450 CD3301
0453 C25C04
0456 CD7A00C302

045C C3D906

045F CDDA01
0462 FE00
0464 C28784

0467 019304CD3D
046D C0BA00
0470 21420735C2
0477 237EFE59C2

047E 2322C307
0482 212007363F

0487 CDC103
048A 112007CD8A
0490 C3D906

0493 414C4C2046

04A4 CDDA01C285
04AA CDC103CD5C
04B0 CAEE04

MVI B,11 ;LENGTH OF FILL ????????.???
DIP0: MVI M,'?' INX H! DCR B! JNZ DIR0
;NOT A BLANK REQUEST, MUST BE IN. COMFCB
DIR1. CALL SEARCHCOM ;FIRST ONE HAS BEEN FOUND
CZ NOFILE ;NOT FOUND MESSAGE
DIR2. JZ ENDIR
CALL CRLF
;ANOTHER FCB FOUND, PRINT PROMPT
CALL CSELECT! ADI 'A!' CALL PRINTCHAR
MVI A,'!' CALL PRINTCHAR
MVI A,'!' CALL PRINTCHAR
;COMPUTE POSITION. OF NAME IN. BUFFER
LDA DCNT ;GET THE LOCATION. OF THE ELEMENT
RRC! RRC! RRC! ANI 110*0000B! MOV C,A
;C CONTAINS BASE INDEX INTO BUFF FOR DIR ENTRY
MVI B,1 ;START WITH FIRST CHARACTER OF NAME
DIR3. MOV A,B! CALL ADDHCF ;BUFF+A+C FETCHED
;DELETE TRAILING BLANKS
CPI ' ' ! JNZ DIR4 ;CHECK FOR BLANK TYPE
MVI A,9! CALL ADDHCF ;FIRST CHAR OF TYPE
MVI A,' ' ! CMP M! JZ DIR5 ;TERMINATE

DIR4. CALL PRINTBC ;CHAR PRINTED
INR B! MOV A,B! CPI 12! JNC DIR5
;CHECK FOR BREAK BETWEEN NAMES
CPI 9! JNZ DIR3 ;FOR ANOTHER CHAR
;PRINT A BLANK BETWEEN. NAMES
MVI A,' ' ! CALL PRINTBC! JMP DIR3

DIR5. ;END OF CURRENT ENTRY
CALL BREAK\$KEY ;CHECK FOR INTERRUPT AT KEYBOARD
JNZ ENDIR ;ABORT DIRECTORY SEARCH
CALL SEARCHM! JMP DIR2 ;FOR ANOTHER ENTRY
ENDIR. ;END OF DIRECTORY SCAN
JMP RETCOM

ERASE. CALL FILLFC00 ;CANNOT BE ALL ???'S
CPI 11
JNZ ERASEFILE
;ERASING ALL OF THE DISK
LXI B,ERMSG! CALL PRINT!
CALL READCOM
LXI H,COMLEN! DCR M! JNZ CCP ;BAD INPUT
INX H! MOV A,M! CPI 'Y!' JNZ CCP
;OK, ERASE THE ENTIRE DISKETTE
INX H! SHLD COMADDR ;OTHERWISE ERROR AT RETCOM
LXI H,COMFCB! MVI M,'?'

ERASEFILE. CALL SETDISK
LXI D,COMFCB! CALL DELETE
JMP RETCOM

ERMSG. DB 'ALL FILES (Y/N)?'

CALL FILLFC00! JNZ COMERR ;DON'T ALLOW ?'S IN. FILE NAME
CALL SETDISK! CALL OPENC ;OPEN. THE FILE
JZ TYPERR ;ZERO FLAG INDICATES NOT FOUND

04B3 CD330021CA

04B8 21CA077EFE
04C1 DACE04E5

04C5 CD9600E1
04C9 C2E704
04CC AF77

04CE 34
04CF 210000CDD5
04D5 7EFE1ACAD9
04DB CD2700
04DE CD3301C2D9
04E4 C3B004

04E7 3DCAD906
04EB CD4E03
04EE CDD303C305

04F4 CDDA01
04F7 3AC907B7C2
04FE CD6F03
0501 F5

0502 CDDA01
0505 C20501
0508 CDC103
050B 112007D5CD
0512 D1CDA300
0516 CD5C00
0519 CA4105
051C F1
051D 6F260029
0521 110001

0524 7CB5CA4705
0529 28
052A E5
052B 21000019E5
0530 CD5401
0533 112007CD9C
0539 D1E1
053B C24105
053E C32405

0541 015905CD3D

0547 112007CD66
054D C25605
0550 016205CD3D
0556 C3D906
0559 4E4F203350
0562 43414E4E4F

;FILE OPENED, READ 'TIL EOF
CALL CRLF! LXI H,BPTR! MVI M,255 ;READ FIRST BUFFER
TYPE0. ;LOOP ON. BPTR
LXI H,BPTR! MOV A,M! CPI 128 ;END BUFFER
JC TYPE1! PUSH M ;CARRY IF B.1,...,127
;READ ANOTHER BUFFER FULL
CALL DISKREAD! POP H ;RECOVER ADDRESS OF BPTR
JNZ TYPE0 ;HARD END OF FILE
XRA A! MOV M,A ;BPTR = 0
TYPE1. ;READ CHARACTER AT BPTR AND PRINT
INR M ;BPTR = SPTR + 1
LXI H,BUFF! CALL ADDH ;H,L ADDRESSES CHAR
MOV A,M! CPI EOF! JZ RETCOM
CALL PRINTCHAR
CALL BREAK\$KEY! JNZ RETCOM ;ABORT IF BREAK
JMP TYPE0 ;FOR ANOTHER CHARACTER
;TYPEOF. ;END OF FILE, CHECK FOR ERRORS
DCR A! JZ RETCOM
CALL READERR
TYPERR. CALL RESETDISK! JMP COMERR

SAVE. CALL FILLFC00 ;SHOULD BE NUMBER
LDA SDISK! DRA A! JNZ COMERR ;CANNOT BE PREFIXED
CALL CONVERT ;RESULTING VALUE IN. REGISTER A
PUSH PSW ;SAVE IT FOR LATER

;SHOULD BE FOLLOWED BY A FILE TO SAVE THE MEMORY IMAGE
CALL FILLFC00
JNZ COMERR ;CANNOT BE AMBIGUOUS
CALL SETDISK ;MAY BE A DISK CHANGE
LXI D,COMFCB! PUSH D! CALL DELETE ;EXISTING FILE REMOV
POP D! CALL MAKE ;CREATE A NEW FILE ON. DISK
CALL OPENC ;OPEN THE FILE
JZ SAVERR ;NO DIRECTORY SPACE

POP PSW ;PAGES TO WRITE IS IN. A, CHANGE TO #SECTORS
MOV L,A! MVI H,B! DAD H!
LXI D,TRAN. ;H,L IS SECTOR COUNT, D,E IS LOAD ADDRESS
;CHECK FOR SECTOR COUNT ZERO
SAVE0. MOV A,H! ORA L! JZ SAVE1 ;MAY BE COMPLETED
DCX H ;SECTOR COUNT = SECTOR COUNT - 1
PUSH H ;SAVE IT FOR NEXT TIME AROUND
LXI H,128! DAD D! PUSH H ;NEXT DMA ADDRESS SAVED
CALL SETDMA ;CURRENT DMA ADDRESS SET
LXI D,COMFCB! CALL DISKWRITE
POP D! POP H ;DMA ADDRESS, SECTOR COUNT
JNZ SAVERR ;MAY BE DISK FULL CASE
JMP SAVE0 ;FOR ANOTHER SECTOR

SAVERR. ;MUST BE FULL DISK
LXI B,FULLMSG! CALL PRINT

SAVE1. ;END OF DUMP, CLOSE THE FILE
LXI D,COMFCB! CALL CLOSE
JNZ SAVE2 ;OTHERWISE CANNOT CLOSE FILE

SAVE2. LXI B,CLOSMSC! CALL PRINT
JMP RETCOM
FULLMSG. DB 'NO SPACES'
CLOSMSC. DB 'CANNOT CLOSE'


```

RENAME,
055F CDDA01C205 ;RENAME A FILE ON A SPECIFIC DISK
0575 3AC907F5 CALL FILLFCB01 JNZ COMERR ;MUST BE UNAMBIGUOUS
0579 CDC103 LDA SDISK! PUSH PSW ;SAVE FOR LATER COMPARE
057C C0E400 CALL SETDISK ;DISK SELECTED
057F C2E005 CALL SEARCHCOM ;IS NEW NAME ALREADY THERE?
JNZ RENERR3

;FILE DOESN'T EXIST, MOVE TO SECOND HALF OF FCB
0592 2120071130 LXI H,COMFCB! LXI D,COMFCB+16! MVI B,16! CALL MOVED
;CHECK FOR = OR LEFT ARROW
059D 2AC307EBCD LHLD COMADDR! XCHG! CALL DEBLANK
0594 FE3DCA9E05 CPI '='! JZ REN1 ;OK IF =
0599 FE5FC2D205 CPI LA! JNZ RENERR2
059E E02322C307 REN1, XCHG! INX H! SHLD COMADDR ;PAST DELIMITER
;PROPER DELIMITER FOUND
05A3 CDDA01C2D2 CALL FILLFCB01 JNZ RENERR2
;CHECK FOR DRIVE CONFLICT
05A5 F147 POP PSW! MOV B,A ;PREVIOUS DRIVE NUMBER
05AB 21C5077EB7 LXI H,SDISK! MOV A,M! ORA A! JZ REN2
;DRIVE NAME WAS SPECIFIED, SAME ONE?
05B3 B070C2D205 CMP B! MOV M,B! JNZ RENERR2
05B8 78 REN2, MOV M,B ;STORE THE NAME IN CASE DRIVES SWITCHED
05B9 AF3220B7CD XRA A! STA COMFCB! CALL SEARCHCOM ;IS OLD FILE THERE?
05C0 CACC05 JZ PENERR1

;EVERYTHING IS OK, RENAME THE FILE
05C3 112007CDAC LXI D,COMFCB! CALL RENAM
05C9 C3D906 JMP RETCOM

RENERR1, ; NO FILE ON DISK
05CC CD5F03C3D9 CALL NOFILE! JMP RETCOM
RENERR2, ; AMBIGUOUS REFERENCE/NAME CONFLICT
05D2 CDD303C305 CALL RESETDISK! JMP COMERR
RENERR3, ; FILE ALREADY EXISTS
05D8 01E105CD3D LXI B,RENMSG! CALL PRINT! JMP RETCOM
05E1 46494C4520 RENMSG, DB 'FILE EXISTS*'

USERFUNC,
05ED CD7101 CALL SERIALIZE ;CHECK SERIALIZATION
05F0 3A2107FE20 ;LOAD USER FUNCTION, AND SET UP FOR EXECUTION
LDA COMFCB+1! CPI '='! JNZ USER0
;NO FILE NAME, BUT MAY BE DISK SWITCH
05F8 3AC907B7CA LDA SDISK! ORA A! JZ ENDCOM ;NO DISK NAME IF B
05FF 3D32C00732 DCR A! STA CDISK! STA DISKA
0606 CD4C00C3DC CALL SELECT! JMP ENDCOM

USER0,
060C 1120071AFE ;FILE NAME IS PRESENT
0615 25C3C103D1 LXI D,COMFCB+9! LDAX D! CPI '='! JNZ COMERR ;TYPE '='
061D C0A103 PUSH D! CALL SETDISK! POP D! LXI H,COMTYPE ;.COM
0620 C05C00CABC CALL MOVENAME ;FILE TYPE IS SET TO .COM
CALL OPENC! JZ USERER
;FILE OPENED PROPERLY, READ IT INTO MEMORY
0626 210001 LXI H,TRAN. ;TRANSIENT PROGRAM BASE
0629 E5 LOAD0, PUSH H ;SAVE DMA ADDRESS
062A EBC15401 XCHG! CALL SETDMA
062E 112007CD6F LXI D,COMFCB! CALL DISKREAD! JNZ LOAD1
;SECTOR LOADED, SET NEW DMA ADDRESS AND COMPARE
0637 E111000019 POP HI LXI D,120! DAD D
063C 110000 LXI B,TRANM ;HAS THE LOAD OVERFLOWED?
063F 7D937C9AD2 MOV A,LI SUB EI MOV A,HI SBB D! JNC LOADERR
0646 C32906 JMP LOAD0 ;FOR ANOTHER SECTOR

```

```

0649 E13DC2C206
064E CDD303
0651 CDDA0121C9
0658 7E322007
065C 3E10CDDC01
0661 E17E323007
0666 AF324007
066A 115C002120

```

```

0675 214307
0678 7EB7CA8606
0682 23C37006

```

```

0686 0600118100
068B 7E1207CA97

```

```

0691 042313C308

```

```

0697 78320000
069B CD3300

```

```

069E 3AC0073204
06A4 CD41011180
06AD CD0001
06B0 31FE06
06B3 3AC007CD4C
06B9 C3FC02

```

```

06BC CDD303C385

```

```

06C2 01C006CD3D
06C8 C3D906
06CB 4C4F414420
06D6 434F4D

```

```

06D9 CDD303

```

```

06DC CDDA01
06DF 3A2107D620

```

```

06E8 C28501
06EB C3FC02

```

```

06EE

```

```

06FE 00 SUBMIT, DB 0 ;DB IF NO SUBMIT FILE, FF IF SUBMITTING
06FF 0024242420SUBFCB, DB 0, '$$$' ;FILE NAME IS $$$
0700 5355420000 DB 'SUB',0,0,0 ;FILE TYPE IS SUB
070E SUBRC, DS 1 ;RECORD COUNT FILED

```

```

LOAD1, POP HI DCR A! JNZ LOADERR ;END FILE IS 1
CALL RESETDISK ;BACK TO ORIGINAL DISK
CALL FILLFCB01 LXI H,SDISK! PUSH H
MOV A,M! STA COMFCB ;DRIVE NUMBER SET
MVI A,16! CALL FILLFCB ;MOVE ENTIRE FCB TO MEMO
POP HI MOV A,M! STA COMFCB+16
XRA A! STA COMREC ;RECORD NUMBER SET TO ZERO
LXI D,FCB! LXI H,COMFCB! MVI B,33! CALL MOVED
;MOVE COMMAND LINE TO BUFF
LXI H,COMBUF

```

```

BMOVEB, MOV A,M! ORA A! JZ BMOVE1! CPI '='! JZ BMOVE1
INX H! JMP BMOVEB ;FOR ANOTHER SCAN
;FIRST BLANK POSITION, FOUND

```

```

BMOVE1, MVI B,0! LXI D,BUFF+1! ;READY FOR THE MOVE
BMOVE2, MOV A,M! STAX D! ORA A! JZ BMOVE3
;MORE TO MOVE
INR B! INX H! INX D! JMP BMOVE2

```

```

BMOVE3, ;B HAS CHARACTER COUNT
MOV A,B! STA BUFF
CALL CRLF
;NOW GO TO THE LOADED PROGRAM
LDA CDISK! STA DISKA ;CURRENT DISK AT DISKA
CALL LIFTHD! LXI D,BUFF! CALL SETDMA
CALL TRAN. ;GONE TO THE LOADED PROGRAM
LXI SP,STACK ;MAY COME BACK HERE
LDA CDISK! CALL SELECT
JMP CCP

```

```

USERER, ;ARRIVE HERE ON COMMAND ERROR
CALL RESETDISK! JMP COMERR

```

```

LOADERR, ;CANNOT LOAD THE PROGRAM
LXI B,LOADMSG! CALL PRINT
JMP RETCOM
LOADMSG, DB 'LOAD ERROR*'

```

```

COMTYPE, DB 'COM' ;FOR COM FILES

```

```

RETCOM, ;RESET DISK BEFORE END OF COMMAND CHECK
CALL RESETDISK

```

```

ENDCOM, ;END OF INTRINSIC COMMAND
CALL FILLFCB0 ;TO CHECK FOR GARBAGE AT END OF LINE
LDA COMFCB+1! SUI '='! LXI H,SDISK! ORA H
;TO IN ACCUMULATOR IF NO DISK SELECTED, AND BLANK FCB
JNZ COMERR
JMP CCP

```

```

DATA AREAS
DS 16 ;8 LEVEL STACK

```

```

STACK,

```

```

'SUBMIT' FILE CONTROL BLOCK
SUBMIT, DB 0 ;DB IF NO SUBMIT FILE, FF IF SUBMITTING
DB 0, '$$$' ;FILE NAME IS $$$
DB 'SUB',0,0,0 ;FILE TYPE IS SUB
SUBRC, DS 1 ;RECORD COUNT FILED

```

```

070F          DS      16      ;DISK MAP
071F SUBCR:   DS      1      ;CURRENT RECORD TO READ
;
; COMMAND FILE CONTROL BLOCK
0720 COMFCB:  DS      32      ;FIELDS FILLED IN. LATER
0740 COMREC:  DS      1      ;CURRENT RECORD TO READ/WRITE
;
; COMMAND BUFFER
0741 7F MAXLEN:  DB      1 7      ;MAX BUFFER LENGTH
0742 00 COMLEN:  DB      0      ;COMMAND LENGTH (FILLED IN. BY DOS)
0743 COMBUF:  DS     128      ;COMMAND BUFFER
07C3 COMADDR: DS      2      ;ADDRESS OF NEXT TO SCAN
07C5 STADDR:  DS      2      ;STARTING ADDRESS OF CURRENT FILLFCB REQUEST
;
DCNT:   DS      1      ;DISK DIRECTORY COUNT (USED FOR ERROR CODES)
CDISK:  DS      1      ;CURRENT DISK
SDISK:  DS      1      ;SELECTED DISK FOR CURRENT OPERATION
;NONE=0, A=1, B=2 ...
07CA BPTR:   DS      1      ;BUFFER POINTER
07CB          END      CCPLOC

```