

III 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

```
; ASSEMBLY LANGUAGE VERSION. OF THE CP/M CONSOLE COMMAND PROCESSOR
0800 -    BIAS      EQU     808H
;           VERSION. 1.1 OCTOBER, 1976
;           COPYRIGHT (C) 1976
;           DIGITAL RESEARCH
;
0800 -    FALSE     EQU     8000H
FFFF -    TRUE      EQU     NOT FALSE
0800 -    TESTING   EQU     FALSE ; TRUE IF DEBUGGING
0800 -    CCPLOC   EQU     BIAS  ; BASE ADDRESS OF CCP
0900 -    BDOSL    EQU     906H+BIAS ; BASE OF DOS
;
;           IF      TESTING
;           ORG     100H
;           MVI     C,0
TRAN -    EQU     81000H ; TRANSIENT PROGRAM AREA BASE (TPA)
TRANM -   EQU     1400H ; LAST TRANSIENT LOCATION. + 1
ENDIF
;
;           IF      NOT TESTING
;           ORG     CCPLOC
0100 -    TRAN.   EQU     100H
0300 -    TRANM   EQU     $
ENDIF
;
0000 C3E702    JMP     CCPSTART
0003 434F505952 DB      'COPYRIGHT (C) 1976, DIGITAL RESEARCH'
;
0204 -    DISKA   EQU     8004H ; DISK ADDRESS FOR CURRENT DISK
0205 -    BDOS   EQU     8005H ; PRIMARY BDOS ENTRY POINT
0200 -    BUFF   EQU     8008H ; DEFAULT BUFFER
0205C -   FCB    EQU     805CH ; DEFAULT FILE CONTROL BLOCK
;
0201 -    RCHARF  EQU     1     ; READ CHARACTER FUNCTION
0202 -    PCHARF  EQU     2     ; PRINT CHARACTER FUNCTION
0209 -    PBUFF   EQU     9     ; PRINT BUFFER FUNCTION
020A -    RBUFF   EQU     10    ; READ BUFFER FUNCTION
0208 -    BREAKF  EQU     11    ; BREAK KEY FUNCTION
020C -    LIFTF   EQU     12    ; LIFT HEAD FUNCTION. (SHUGART SA3900 ONLY)
020D -    INITF   EQU     13    ; INITIALIZE BDOS FUNCTION
020E -    SELF    EQU     14    ; SELECT DISK FUNCTION
020F -    OPENF   EQU     15    ; OPEN FILE FUNCTION
0210 -    CLOSEF  EQU     16    ; CLOSE FILE FUNCTION
0211 -    SEARF   EQU     17    ; SEARCH FOR FILE FUNCTION
0212 -    SEARNF  EQU     18    ; SEARCH FOR NEXT FILE FUNCTION
0213 -    DELF    EQU     19    ; DELETE FILE FUNCTION
0214 -    DREADF  EQU     20    ; DISK READ FUNCTION
0215 -    DWRITF  EQU     21    ; DISK WRITE FUNCTION
0216 -    MAKEF   EQU     22    ; FILE MAKE FUNCTION
0217 -    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,PCHARFI JMP BDOS
;
PRINTBC: 002D C5CD2700C1 ; PRINT CHARACTER, BUT SAVE B,C REGISTERS
                  PUSH B! CALL PRINTCHAR! POP B! RET
;
CRLF: 0033 3E0DCD2700 MVI A,CRI CALL PRINTCHAR
0038 3E0AC32700 MVI A,LFI JMP PRINTCHAR
;
PRINT: 003D C5CD3300D1 ; PRINT STRING STARTING AT B,C UNTIL NEXT DOLLAR SYMBOL
                  PUSH B! CALL CRLF! POP D! MVI C,PGUFI JMP BDOS
;
INITIALIZE: 0047 0E0DC38500 MVI C,INITFI JMP BDOS
;
SELECT: 004C 5F0E0EC305 MOV E,A! MVI C,SELF! JMP BDOS
;
OPEN: 0052 0E0FC0B500 ; OPENN THE FILE GIVEN. BY D,E
                  MVI C,OPENFI CALL BDOS! STA DCNT! INR A! RET
;
OPENC: 005C AF324007 ; OPEN. COMFCB
                  XRA A! STA COMREC ; CLEAR NEXT RECORD TO READ
0060 112007C352 LXI D,COMFCB! JMP OPEN
;
CLOSE: 0066 0E10CD0500 ; CLOSE THE FILE GIVEN. BY D,E
                  MVI C,CLOSEFI CALL BDOS! STA DCNT! INR A! RET
;
SEARCH: 0070 0E11CD0500 ; SEARCH FOR THE FILE GIVEN. BY D,E
                  MVI C,SEARFI CALL BDOS! STA DCNT! INR A! RET
;
SEARCHN: 007A 0E12CD0500 ; SEARCH FOR THE NEXT OCCURRENCE OF THE FILE GIVEN. BY D,E
                  MVI C,SEARNFI CALL BDOS! STA DCNT! INR A! RET
;
SEARCHCOM: 0084 112007C370 ; SEARCH FOR COMFCB FILE
                  LXI D,COMFCB! JMP SEARCH
;
DELETE: 008A 0E13C30500 ; DELETE THE FILE GIVEN. BY D,E
                  MVI C,DELFI JMP BDOS
;
DISKREAD: 009F 0E14CD0500 ; READ THE NEXT RECORD FROM THE FILE GIVEN. BY D,E
                  MVI C,DREADFI CALL BDOS! ORA A! RET
;
DISKREADC: 00A0 0E15C30500 ; 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:
> 00A3 0E16CD0500 MVI C,MAKEFI CALL BDOS! STA DCHT! RET
>
>           ;RENAME:
> 00AC 0E17C38500 MVI C,REHF! JMP BDOS
>
>           ;TRANSLATE:
>           ;TRANSLATE CHARACTER INH REGISTER A TO UPPER CASE
> 00B1 FE61D8 CPI 61H! RC ;RETURN. IF BELOW LOWER CASE A
> 00B4 FE7BD0 CPI 7BH! RNC ;RETURNNN IF ABOVE LOWER CASE Z
> 00B7 E65FC9 ANI SFH! RET ;TRANSLATED TO UPPER CASE
>
>           ;READCOM:
>           ;READ THE NEXT COMMAND INTO THE COMMAND BUFFER
>           ;CHECK FOR SUBMIT FILE
> 00BA 3AFE06B7CA LDA SUBMIT! ORA A! JZ NOSUB
>           ;SCANNING A SUBMIT FILE
>           ;CHANGE DRIVES TO OPENN AND READ THE FILE
> 00C1 3AC007873E LDA CDISK! ORA A! MVI A,0! CNZ SELECT
> 00CA 3A0E073D LDA SUBRC! DCR A ;READ LAST RECORD(S) FIRST
> 00CE 321F07 STA SUBCR ;CURRENT RECORD TO READ
> 00D1 11FF06CD08F LXI D,SUBFCB! CALL DISKREAD ;END OF FILE IF LAST RECORD
> 00D7 C2D001 JNZ NOSUB
>
>           ;DISK READ IS OK, TRANSFER TO COMBUF
> 00DA 1142072180 LXI D,COMLEN! LXI H,BUFF! MVI B,128! CALL MOVEO
>           ;LINE IS TRANSFERRED, CLOSE THE FILE WITH A
>           ;DELETED PECORD
> 00E5 210E0735 LXI H,SUBRC! DCR M ;ONE LESS RECORD
> 00E9 11FF06CD66 LXI D,SUBFCB! CALL CLOSE! JZ NOSUB
>
>           ;CLOSE WENT OK, RETURN. TO ORIGINAL DRIVE
> 00F2 3AC00787C4 LDA CDISK! ORA A! CNZ SELECT
> 00F9 0E09114307 MVI C,PBUFF! LXI D,COMBUFI CALL BDOS
> 0101 CD3301CA18 CALL BREAK$KEY! JZ NOREAD
> 0107 CD5901C3FC CALL DEL$SUB! JMP CCP ;BREAK KEY DEPRESSED
>
> 010D CD5901 NOSUB: ;NO SUBMIT FILE! CALL DEL$SUB
>           ;TRANSLATE TO UPPER CASE, STORE ZERO AT END
> 0110 0EB0A114107 MVI C,RBUFF! LXI D,MAXLEN! CALL BDOS
> NOREAD: ;ENTER HERE FROM SUBMIT FILE
>           ;SET THE LAST CHARACTER TO ZERO FOR LATER SCANS
> 0116 21420746 LXI H,COMLEN! MOV B,M ;LENGTH IS IN. B
> 011C 237887 READCOM: INX H! MOV A,B! ORA A ;END OF SCAN?
> 011F CA2B017E JZ READCOM1! MOV A,M ;GET CHARACTER AND TRANSLATE
> 0123 CDB1007705 CALL TRANSLATE! MOV M,A! DCR B! JMP READCOM
>
>           ;READCOM1: ;END OF SCAN, H,L ADDRESS END OF COMMAND
> 012B 77 MOV M,A ;STORE A ZERO
> 012C 21430722C3 LXI H,COMBUF! SHLD COMADDR ;READY TO SCANN TO ZERO
> 0132 C9 RET
>
>           ;BREAK$KEY:
>           ;CHECK FOR A CHARACTER READY AT THE CONSOLE
> 0133 0E0BC00500 MVI C,BREAKFI CALL BDOS

```

3

4

```

0138 8/C8 ORA A! RZ
013A 0E01CD0500 MVI C,RCHARFI CALL BDOS ;CHARACTER CLEARED
013F B7C9 ORA A! RET
;
LIFTHEAD:
0141 0E0CC30500 ;LIFT THE HEAD ON THE SA3900
MVI C,LIFTFI JMP BDOS
;
CSELECT:
0146 0E19C30500 ;GET THE CURRENTLY SELECTED DRIVE NUMBER TO REG-A
MVI C,CSELF1 JMP BDOS
;
MOVE:
014B 1A7713230D ;MOVE DATA FROM D,E TO H,L FOR C BYTES
LDAX D! MOV M,A! INX DI INX HI DCR C! JNZ MOVE! RET
;
SETDMA:
0154 0E1AC30500 ;SET DMA ADDRESS TO D,E
MVI C,DMAFI JMP BDOS
;
DEL$SUB:
0159 21FEB067EB7 ;DELETE THE SUBMIT FILE, AND SET SUBMIT FLAG TO FALSE
LXI H,SUBMIT! MOV A,M! ORA A! RZ ;RETURN. IF NO SUB FILE
MVI M,B ;SUBMIT FLAG IS SET TO FALSE
0161 AFC04C80 XRA A! CALL SELECT ;ON. DRIVE A TO ERASE FILE
0165 11F006CD08A LXI D,SUBFCB! CALL DELETE
016B 3AC007C34C LDA CDISK! JMP SELECT ;BACK TO ORIGINAL DRIVE
;
SERIALIZE:
0171 11A0022100 ;CHECK SERIALIZATION
LXI D,SERIAL! LXI H,B00SL! MVI B,6 ;CHECK SIX BYTES
0179 1ABEC24403 SERB: LDAX D! CMP M! JNZ BADSERIAL
017E 132305C279 INX DI INX HI DCR B! JNZ SERB
0184 C9 RET ;SERIAL NUMBER IS OK
;
COMERR:
0185 CD3300 ;ERROR IN. COMMAND STRING STARTING AT POSITION
CALL CRLF ;'STADDR' AND ENDING WITH FIRST DELIMITER
0188 2AC507 LHLD STADDR ;H,L ADDRESS FIRST TO PRINT
018B 7EFE20CA9E COMERR0: ;PRINT CHARACTERS UNTIL BLANK OR ZERO
MOV A,M! CPI '!' JZ COMERR1; NOT BLANK
0191 B7CA9E01 ORA A! JZ COMERR1; NOT ZERO, SO PRINT IT
0195 E5CD2780E1 PUSH H! CALL PRINTCHAR! POP H! INX H
0198 C38B01 JMF COMERR0; FOR ANOTHER CHARACTER
019E 3E3FC0D2780 COMERR1: ;PRINT QUESTION. MARK AND DELETE SUB FILE
01A3 CD3300CD059 MVI A,'?'! CALL PRINTCHAR
01A9 C3FC02 CALL CRLF! CALL DEL$SUB
JMP CCP ;RESTART WITH NEXT COMMAND
;
; FCB SCAN. AND FILL SUBROUTINE (ENTRY IS AT FILLCB BELOW)
; FILL THE COMFCB, INDEXED BY A (0 OR 16)
; SUBROUTINES
DELIM: ;LOOK FOR A DELIMITER
01AC 1AB7C8 LDAX D! ORA A! RZ ;NOT THE LAST ELEMENT
01AF FE20DA8501 CPI '!' JC COMERR ;NOH. GRAPHIC
01B4 C8 RZ ;TREAT BLANK AS DELIMITER
01B5 FE3DC8 CPI '='! RZ
01B8 FE5FC8 CPI LA! RZ ;LEFT ARROW
01BB FE2EC0 CPI '.'! RZ
01BE FE3AC0 CPI '..'! RZ

```

01C1 FE3BC8
 01C4 FE3CC8
 01C7 FE3EC8
 01CA C9
 ;
 01CB 1AB7C8
 01CE FE26C013C3
 ;
 01D5 656FD8
 01D8 24C9
 ;
 FILLFCB:
 ;EQUIVALENT TO FILLFCB(0)
 MV1 A,B
 ;
 FILLFCB:
 LXI H,COMFCB! CALL ADDH! PUSH H! PUSH H! ;FCB RESCANNED AT END
 XRA A! STA SDISK ;CLEAR SELECTED DISK (IN. CASE A,...)
 LHLD COMADDR! XCHG ;COMMAND ADDRESS IN. D,E
 CALL DEBLANK ;TO FIRST NON-BLANK CHARACTER
 XCHG! SHLD STADDR ;IN. CASE OF ERRORS
 XCHG! POP H ;D,E HAS COMMAND, H,L HAS FCB ADDRESS
 ;LOOK FOR PRECEDING FILE NAME A: B: ...
 LDAX DI DRA A! JZ SETCUR0 ;USE CURRENT DISK IF EMPTY COMMAND
 SBI 'A'-1! MOV B,A ;DISK NAME HELD IN. B IF : FOLLOWS
 INX DI LDAX D! CPI '!' JZ SETDSK ;SET DISK NAME IF :
 ;
 SETCUR: ;SET CURRENT DISK
 DCX D ;BACK TO FIRST CHARACTER OF COMMAND
 SETCUR0:
 LDA CDISK! MOV M,A! JMP SETNAME
 ;
 SETDSK: ;SET DISK TO NAME IN. REGISTER B
 MOV A,B! STA SDISK ;MARK AS DISK SELECTED
 MOV M,B! INX D ;PAST THE :
 ;
 SETNAME: ;SET THE FILE NAME FIELD
 MV1 B,8 ;FILE NAME LENGTH (MAX)
 SETNAME0: CALL DELINI JZ PADNAME ;NOT A DELIMITER
 INX HI CPI '!' JNZ SETNAME1 ;MUST BE ?'
 MV1 M,'?'! JMP SETNAME2 ;TO DEC COUNT
 ;
 SETNAME1: MOV M,A ;STORE CHARACTER TO FCB! INX D
 SETNAME2: DCR B ;COUNT DOWN. LENGTH! JNZ SETNAME0
 ;
 ;END OF NAME, TRUNCATE REMAINDER
 TRNAME: CALL DELIM! JZ SETTY ;SET TYPE FIELD IF DELIMITER
 INX DI JMP TRNAME
 ;
 PADNAME: INX HI MV1 M.' '! DCR B! JNZ PADNAME
 ;
 SETTY: ;SET THE TYPE FIELD
 MV1 B,3! CPI '!' JNZ PADTY ;SKIP THE TYPE FIELD IF NO .
 INX D ;PAST THE ., TO THE FILE TYPE FIELD
 SETTYB: ;SET THE FIELD FROM THE COMMAND BUFFER
 CALL DELIM! JZ PADTY1 INX HI CPI '!' JHZ SETTY1
 MV1 M,'?' ;SINCE * SPECIFIED! JMP SETTY2 5
 0255 7713
 0257 05C24482
 ;
 0258 CDAC01CA6C
 0265 23362005C2
 ;
 026C 0603
 026E 23360005C2
 0275 EB22C7:7
 0279 E1010B800
 027D 237FE3FC2
 0284 04
 0285 0DC27D82
 0289 7BB7C9
 ;
 INTVEC:
 ;INTRINSIC FUNCTION. NAMES (ALL ARE FOUR CHARACTERS)
 DB 'DIR'
 DB 'ERA'
 DB 'TYPE'
 DB 'SAVE'
 DB 'REN'
 0005 =
 02A0 0000000000
 ;
 INTRINSIC:
 ;LOOK FOR INTRINSIC FUNCTIONS (COMFCB HAS BEEN FILLED)
 LXI H,INTVEC! MV1 C,0 ;C COUNTS INTRINSICS AS SCANNED
 INTRIN0: MOV A,C! CPI INTLEN. ;DONE WITH SCAN?!! RNC
 ;NO, MORE TO SCAN
 LXI D,COMFCB+1 ;BEGINNING OF NAME
 MV1 B,4 ;LENGTH OF MATCH IS IN. B
 INTRIN1: LDAX D! CMP M ;MATCH?
 JNZ INTRIN2 ;SKIP IF NO MATCH
 INX DI INX HI DCR B
 JNZ INTRIN1 ;LOOP WHILE MATCHING
 ;
 ;COMPLETE MATCH ON. NAME, CHECK FOR BLANK IN. FCB
 LDAX D! CPI '!' JNZ INTRIN3 ;OTHERWISE MATCHED
 MOV A,C! RET ;WITH INTRINSIC NUMBER IN. A
 ;
 INTRIN2: ;MISMATCH, MOVE TO END OF INTRINSIC
 INX HI DCR B! JNZ INTRIN2
 ;
 INTRIN3: ;TRY NEXT INTRINSIC
 INR C ;TO NEXT INTRINSIC NUMBER
 JMP INTRINA ;FOR ANOTHER ROUND

```

; CHECKSUB:
; CHECK FOR A SUBMIT FILE (REGISTER A HAS DISK NAME)
02D0 B7C2E2B2 ORA A! JNZ SETSUB0 ;CLEAR SUBMIT IF NOT DISK A
; OPENH SUBMIT FILE - IF POSSIBLE
02D4 11FF06CD52 LXI D, SUBFCB1 CALL OPENH JZ SETSUB0
; SUBMIT FILE OPENED, SET SUBMIT TO TRUE
02DD 3EFFC3E302 MVI A, 0FFH! JMP SETSUB

; SETSUB0:
; XRA A ;CLEAR SUBMIT VARIABLE
02E2 AF SETSUB, STA SUBMIT
02E3 32FE06 RET

; CCPSTART:
; ENTER HERE FROM BOOT LOADER
02E7 31FE06C5C5 LXI SP, STACK! PUSH B! PUSH B ;SAVE INITIAL DISK NUMBER
02EC CD4760C1 CALL INITIALIZE! POP B ;RECOVERED DISK NUMBER TO C
02F0 F5 PUSH PSW ;SAVE $ FLAG FROM INITIALIZATION. (00 IF NO $ FILE)
02F1 79 MOV A,C ;DISK NUMBER IN. ACCUMULATOR
02F2 CD4C00 CALL SELECT ;PROPER DISK IS SELECTED. NOW CHECK SUB FILES
02F5 F1 PDP PSW ;RECOVER $ FILE FLAG FROM INITIALIZATION
02F6 C1 POP B ;ORIGINAL DISK NUMBER IS IN. C
02F7 3C81 INR A ;08 BECOMES 81, FF BECOMES B01 ORA C ;OR DISK NUMBER
;ACCUMULATOR CONTAINS 08 IF ONN DISK 0, AND $ FILE PRESENT
;OTHERWISE, THE ACCUMULATOR CONTAINS A NON-ZERO VALUE
02F9 CDD002 CALL CHECKSUB

; CCP:
; ENTER HERE ON. EACH COMMAND OR ERROR CONDITION .
02FC 31FE06 LXI SP, STACK
02FF CD3300 CALL CRLF ;PRINT D> PROMPT, WHERE D IS DISK NAME
0302 CD460132C8 CALL CSELECT! STA CDISK1
0309 C641CD2700 ADI 'A'! CALL PRINTCHAR
030D 3E3ECD2700 MVI A, 'Y'! CALL PRINTCHAR
0312 118000CD54 LXI D,BUFF1 CALL SETIMA ;DEFAULT DMA ADDRESS IS AT BUFF
0318 CD8H00CDDA CALL READCOM! CALL FILLCB0
031E C485D1 CHZ COMERR ;THE NAME CANNOT BE ANH AMBIGUOUS REFERENCE
0321 3AC90787C2 LDA SDISK1 ORA A! JNZ USERFUNC
;CHECK FOR AN. INTRINSIC FUNCTION
;CALL INTRINSIC
0323 CD4602 LXI H, JMPTAB ;INDEX IS IN. THE ACCUMULATOR
032B 213083 MOV E,A! MVI D,B! DAD D! DAD D ;INDEX INH 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:
    DW DIRECT ;DIRECTORY SEARCH
    DW ERASE ;FILE ERASE
    DW TYPE ;TYPE FILE
    DW SAVE ;SAVE MEMORY IMAGE
    DW RENAME ;FILE RENAME
    DW USERFUNC;USER-DEFINED FUNCTION

; BADSERIAL:
    LXI H,DI OR <HLT SHL 0>
    SHLD CCPLOCI LXI H, CCPLOCI PCHL

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

```

7

```

034E 015403C33D LXI B, RDMSG! JMP PRINT
0354 5245414420 RDMSG: DB 'READ ERROR$'

; NOFILE:
; PRINT NO FILE MESSAGE
035F 016503C33D LXI B, NOFMSC! JMP PRINT
0365 4E4F542046 NOFMSC: DB 'NOT FOUND$'

; CONVERT:
; CONVERT THE BYTE VALUE IN. COMFCB TO BINARY
036F 212107810B LXI H, COMFCB+1! LXI B, 11 ;(B=8, C=11)
; VALUE ACCUMULATED INH B, C COUNTS NAME LENGTH TO ZERO
0375 7FEFE20CAA0 CONVB: MOV A,M! CPI ' ' I JZ CONVI
; MORE TO SCAN, CONVERT CHAR TO BINARY AND ADD
0378 23D630FEGA INX H! SUI '0'! CPI 10! JNC COMERR ;VALID?
0383 5778 MOV D,A ;SAVE VALUE! MOV A,B ;MULT BY 10
0385 E6E0C285J1 ANI 1110$0000B! JNC COMERR
038A 78 MOV A,B ;RECOVER VALUE
038B 070707 RLCI RLCI RLC ;*8
038E 80DA85B1 ADD B! JC COMERR
0392 80DA8501 ADD B! JC COMERR ;*8+*2 = *18
0396 82DA8501 ADD D! JC COMERR ;*DIGIT
039A 470DC27503 MOV B,A! DCR C! JNZ CONVB ;FOR ANOTHER DIGIT
039F C9 RET

CONVI: ;END OF DIGITS, CHECK FOR ALL BLANKS
        MOV A,M! CPI ' ' I JZ CONVB ;BLANKS?
        INX H! DCR C! JNZ CONVI
        MOV A,B ;RECOVER VALUE! RET

; MOVENAME:
; MOVE 3 CHARACTERS FROM H,L TO D,E ADDRESSES
03AD 0603 MVI B,3
03AF 7E122313 MOVEB; MOV A,M! STAX D! INX H! INX D
03B3 05C2AF03 DCR B! JNZ MOVEB
03B7 C9 RET

; ADDHCF: ;BUFF + A + C TO H,L FOLLOWED BY FETCH
03B8 21800081CD LXI H, BUFF1 ADD C! CALL ADDH! MOV A,M! RET

; SETDISK:
; CHANGE DISKS FOR THIS COMMAND, IF REQUESTED
03C1 AF322007 KRA A! STA COMFCB ;CLEAR DISK NAME FROM FCB
03C5 3AC907B7C8 LDA SDISK1 ORA A! RZ ;NO ACTION. IF NOT SPECIFIED
03CA 3D21C807BE DCR A! LXI H, CDISK1 CMP M! RZ ;ALREADY SELECTED
03D0 C34C00 JMP SELECT

; RESETDISK:
; RETURN. TO ORIGINAL DISK AFTER COMMAND
03D3 3AC907B7C8 LDA SDISK1 ORA A! RZ ;NO ACTION. IF NOT SELECTED
03D8 3D21C807BE DCR A! LXI H, CDISK1 CMP M! RZ ;SAME DISK
03DE 3AC807C34C LDA CDISK1 JMP SELECT

; INDIVIDUAL INTRINSICS FOLLOW

; DIRECT:
; DIRECTORY SEARCH
03E4 CDDA01 CALL FILLCB0 ;COMFCB GETS FILE NAME
03E7 CDC103 CALL SETDISK ;CHANGE DISK DRIVES IF REQUESTED
03EA 2121077E LXI H, COMFCB+1! MOV A,M ;MAY BE EMPTY REQUEST
03EE FE20C2FC03 CPI ' ' I JNZ DIR1 ;SKIP FILL OF ??? IF NOT BLANK
;SET COMFCB TO ALL ??? FOR CURRENT DISK

```

8

03F3 060B
 03F5 363F2385C2
 03FF C05F03
 0400 CA5C04
 0405 CD3300
 0408 CD4681C641
 0410 3E3ACD2700
 0415 3E20CD2700
 041A 3AC707
 041D 0F20FE668
 0423 0601
 0425 78CD9803
 0429 FE20C23904
 043E 3E09CD863
 0433 3E20BECASB
 0439 CD2D00
 043C 0478FE0CD2
 0443 FE89C22504
 0448 3E20CD2D00
 0450 CD3301
 0453 C25C04
 0456 CD7A00C302
 045C C3D906
 ;
 ERASE:
 045F CDDA01
 0462 FE0B
 0464 C28784
 0467 013384CD3D
 046D C08400
 0470 21420735C2
 0477 237FEF59C2
 047E 2322C307
 0482 212007363F
 ;
 ERASEFILE:
 0487 CDC103
 048A 112007CD8A
 0490 C3D906
 ;
 0493 414C4C2046
 ;
 TYPE:
 04A4 CDDA01C285
 04AA CDC103CD5C
 04B0 CAEE04

MVI B,11 ;LENGTH OF FILL ????????.???
 DPO: 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
 JZ ENDIR
 CALL CRLF
 ;ANOTHER FCB FOUND, PRINT PROMPT
 CALL CSCLECT! 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
 RRC1 RRC1 RRC1 ANI 110\$0000B1 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 MI JZ DIR5 ;TERMINATE

DIR4:
 CALL PRINTBC ;CHAR PRINTED
 IHR 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 SEARCHNI JMP DIR2 ;FOR ANOTHER ENTRY

ENDIR: ;END OF DIRECTORY SCAN
 JMP RETCOM

;
 ERASE:
 CALL FILLCFB0 ;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 FILLCFB0! JNZ COMERR ;DON'T ALLOU ?'S IN. FILE NAME
 CALL SETDISK! CALL OPENC ;OPEN. THE FILE
 JZ TYPERR ;ZERO FLAG INDICATES NOT FOUND

> 04B3 CD330021CA
 > 04BB 21CA077EFE
 > 04C1 DACE04E5
 > 04C5 CD9600E1
 > 04C9 C2E704
 > 04CC AF77
 > 04CE 34
 > 04CF 218000CDD5
 > 04D5 7EFE1ACAD9
 > 04DB CD2700
 > 04DE CD3301C2D9
 > 04E4 C3B804

,
 04E7 3DCAD986
 04EB CD4E03
 04EE CDD303C385
 ,
 SAVE:
 04F4 CDDA01
 04F7 3AC907B7C2
 04FE CD6F03
 0501 F5

0502 CDDA01
 0505 C28501
 0508 CDC103
 0508 112007D5CD
 0512 D1CDA3B0
 0516 CD5C00
 0519 CA4105
 051C F1
 051D 6F260029
 0521 110001

0524 7CB5CA4705
 0529 2B
 052A E5
 052B 21800019E5
 0530 CD5401
 0533 112007CD9C
 0539 D1E1
 053B C24105
 053E C32405

0541 015905CD3D
 0547 112007CD66
 054D C25605
 0550 016205CD3D
 0556 C3D906
 0559 4E4F205350
 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 H ;CARRY IF B,1,...,127
 ;READ ANOTHER BUFFER FULL
 CALL DISKREADC! POP H ;RECOVER ADDRESS OF BPTR
 JNZ TYPEOF ;HARD END OF FILE
 XRA A! MOV M,A ;BPTR = 0

TYPE1: ;READ CHARACTER AT BPTR AND PRINT
 INR M ;BPTR = BPTR + 1
 LXI H,BUFF! CALL ADDH ;H,L ADDRESSES CHAR
 MOV A,M! CPI EOFLEI ;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

CALL FILLCFB0 ;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 FILLCFB0
 JNZ COMERR ;CANNOT BE AMBIGUOUS
 CALL SETDISK ;MAY BE A DISK CHANGE
 LXI D,COMFCB! PUSH D! CALL DELETE ;EXISTING FILE REMOVE
 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 HI
 LXI D,TRAN. ;H,L IS SECTOR COUNT, D,E IS LOAD ADDRESS

;CHECK FOR SECTOR COUNT ZERO
 MOV A,H! DRA 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 DISKURITE
 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

LXI B,CLOSEMSG! CALL PRINT
 JMP RETCOM

FULLMSG: DB 'NO SPACES'
 CLOSEMSG: DB 'CANNOT CLOSE'

RENAME:
 055F CDDA01C285 ;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 CD8400 CALL SETDISK ;DISK SELECTED
 057F C2D605 CALL SEARCHCOM ;IS NEW NAME ALREADY THERE?
 JNZ REHERR3
 0582 2120071130 ;FILE DOESN'T EXIST, MOVE TO SECOND HALF OF FCB
 LXI H, COMFCB! LXI D, COMFCB+16! MVI B, 16! CALL MOVEB
 ;CHECK FOR = OR LEFT ARROW
 LHLD COMADDR! XCHG! CALL DEBLANK
 CPI '=' JZ REN1 ;OK IF =
 CPI LA! JNZ RENERR2
 REN1: XCHG! INX H! SHLD COMADDR ;PAST DELIMITER
 ;PROPER DELIMITER FOUND
 CALL FILLFCB01 JNZ REHERR2
 ;CHECK FOR DRIVE CONFLICT
 POP PSW! MOV B,A ;PREVIOUS DRIVE NUMBER
 LXI H, SDISK! MOV A,M! DRA A! JZ REN2
 ;DRIVE NAME WAS SPECIFIED. SAME ONE?
 CMP B! MOV M,B! JNZ REHERR2
 REN2: MOV M,B ;STORE THE NAME IN. CASE DRIVES SWITCHED
 XRA A! STA COMFCB! CALL SEARCHCOM ;IS OLD FILE THERE?
 JZ PEHERR1
 ;
 ;EVERYTHING IS OK, RENAME THE FILE
 LXI D, COMFCB! CALL RENAM
 JMP RETCOM
 ;
 RENERR1: ;NO FILE ON DISK
 CALL NOFILE! JMP RETCOM
 RENERR2: ;AMBIGUOUS NAME/NAME CONFLICT
 CALL RESETDISK! JMP COMERR
 RENERR3: ;FILE ALREADY EXISTS
 LXI B, REHMSG! CALL PRINT! JMP RETCOM
 REHMSG: DB 'FILE EXISTS'
 ,
USERFUNC:
 05E0 CD7101 CALL SERIALIZE ;CHECK SERIALIZATION
 ;LOAD USER FUNCTION, AND SET UP FOR EXECUTION
 05F0 3A2107FE20 LDA COMFCB+1! CPI ' ' JNZ USER0
 ;NO FILE NAME, BUT MAY BE DISK SWITCH
 LDA SDISK! DRA A! JZ END00M ;NO DISK NAME IF B
 DCR A! STA CDISK! STA DISKA
 CALL SELECT! JMP END00M
 USER0: ;FILE NAME IS PRESENT
 LXI D, COMFCB+9! LDAX D! CPI ' ' JNZ COMERR ;TYPE ' '
 PUSH D! CALL SETDISK! POP D! LXI H, COMTYPE .COM
 CALL MOVENAME ;FILE TYPE IS SET TO .COM
 CALL OPEN! JZ USERER
 ;FILE OPENED PROPERLY, READ IT INTO MEMORY
 LXI H, TRAN. ;TRANSIENT PROGRAM BASE
 LOAD0: PUSH H ;SAVE DMA ADDRESS
 XCHG! CALL SETDMA
 LXI D, COMFCB! CALL DISKREAD1 JNZ LOAD1
 ;SECTOR LOADED, SET NEW DMA ADDRESS AND COMPARE
 POP H! LXI D, 128! DAD D
 LXI D, TRANM ;HAS THE LOAD OVERFLOWED? ||
 MOV A, LI SUB EI MOV A, HI SBB DI JNC LOADERR
 JMP LOAD0 ;FOR ANOTHER SECTOR

0649 E13DC2C206
 064E CDD303
 0651 CDDA0121C9
 0658 7E322807
 065C 3E10CD01
 0661 E17E323007
 0666 AF324007
 066A 115C682120
 0675 214307
 0678 7EB7CA8606
 0682 23C37806
 0686 0600118100
 0688 7E12B7CA97
 0691 042313C368
 0697 78328000
 0698 CD3300
 069E 3AC8073204
 06A4 CD41611180
 06A8 CD8001
 06B0 31FE06
 06B3 3AC807CD4C
 06B9 C3FC02
 06BC CDD303C385
 06C2 01CB06CD3D
 06C8 C3D906
 06C9 4C4F414420
 06D0 434F4D
 ;
 RETCOM: ;RESET DISK BEFORE END OF COMMAND CHECK
 CALL RESETDISK
 ;
 END00M: ;END OF INTRINSIC COMMAND
 CALL FILLFCB0 ;TO CHECK FOR GARBAGE AT END OF LINE
 LDA COMFCB+1! SUI ' ' LXI H, SDISK! DRA H
 ;B IN ACCUMULATOR IF NO DISK SELECTED, AND BLANK FCB
 JNZ COMERR
 JMP CCP

06D9 CDD303
 06DC CDDA01
 06DF 3A2107D628
 06E8 C28501
 06EB C3FC02
 ;
 ; DATA AREAS
 DS 16 ;8 LEVEL STACK
 STACK:
 ;
 ; 'SUBMIT' FILE CONTROL BLOCK
 06FE 00 SUBMIT, DB 0 ;0 IF NO SUBMIT FILE, FF IF SUBMITTING
 06FF 0024242420SUBFCB, DB 0, '\$\$' ;FILE NAME IS \$\$
 0700 5355428000 SUBRC, DB 'SUB', 0, 0, 0 ;FILE TYPE IS SUB
 070E SUBRC, DS 1 ;RECORD COUNT FILED

070F DS 15 ;DISK MAP
071F SUBCR: DS 1 ;CURRENT RECORD TO READ
;
; COMMAND FILE CONTROL BLOCK
0728 COMFCB: DS 32 ;FIELDS FILLED IN. LATER
0744 COMREC: DS 1 ;CURRENT RECORD TO READ/WRITE
;
; COMMAND BUFFER
0741 7F MAXLEN: DB 17 ;MAX BUFFER LENGTH
0742 08 COMLEN: DB 8 ;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
;
07C7 DCNT: DS 1 ;DISK DIRECTORY COUNT (USED FOR ERROR CODES)
07C8 CDISK: DS 1 ;CURRENT DISK
07C9 SDISK: DS 1 ;SELECTED DISK FOR CURRENT OPERATION
;NONE=0, A=1, B=2 ...
07CA BPTR: DS 1 ;BUFFER POINTER
07CB END CCPLOC