

TITLE 'console command processor (CCP), ver 2.0'

ASSEMBLY LANGUAGE VERSION OF THE CP/M CONSOLE COMMAND PROCESSOR

VERSION 2.2 FEBRUARY, 1980

COPYRIGHT (C) 1976, 1977, 1978, 1979, 1980

DIGITAL RESEARCH

BOX 579, PACIFIC GROVE,

CALIFORNIA, 93950

```

0000 = FALSE EQU 0000H
FFFF = TRUE EQU NOT FALSE
0000 = TESTING EQU FALSE ;TRUE IF DEBUGGING

```

```

;
; IF TESTING
; ORG 3400H
BDOSL EQU $+800H ;BDOS LOCATION
; ELSE
0000 ORG 000H
0800 = BDOSL EQU $+800H ;BDOS LOCATION
; ENDDIF
0100 = TRAN EQU 100H
0000 = TRANM EQU $
0000 = CCPLOC EQU $

```

```

*****
* BASE OF CCP CONTAINS THE FOLLOWING CODE/DATA *
* CCP: JMP CCPSTART (START WITH COMMAND) *
* JMP CCPCLEAR (START, CLEAR COMMAND) *
* CCP+6 127 (MAX COMMAND LENGTH) *
* CCP+7 COMLEN (COMMAND LENGTH = 00) *
* CCP+8 ' ... ' (16 BLANKS) *
*****
* NORMAL ENTRY IS AT CCP, WHERE THE COMMAND LINE GIVEN *
* AT CCP+8 IS EXECUTED AUTOMATICALLY (NORMALLY A NULL *
* COMMAND WITH COMLEN = 00). AN INITIALIZING PROGRAM *
* CAN BE AUTOMATICALLY LOADED BY STORING THE COMMAND *
* AT CCP+8, WITH THE COMMAND LENGTH AT CCP+7. IN THIS *
* CASE, THE CCP EXECUTES THE COMMAND BEFORE PROMPTING *
* THE CONSOLE FOR INPUT. NOTE THAT THE COMMAND IS EXE- *
* CUTED ON BOTH WARM AND COLD STARTS. WHEN THE COMMAND *
* LINE IS INITIALIZED, A JUMP TO "JMP CCPCLEAR" DIS- *
* ABLES THE AUTOMATIC COMMAND EXECUTION. *
*****

```

```

0000 C35C03 JMP CCPSTART ;START CCP WITH POSSIBLE INITIAL COMMAND
0003 C35803 JMP CCPCLEAR ;CLEAR THE COMMAND BUFFER
0006 7F MAXLEN: DB 127 ;MAX BUFFER LENGTH
0007 00 COMLEN: DB 0 ;COMMAND LENGTH (FILLED IN BY DOS)
; (COMMAND EXECUTED INITIALLY IF COMLEN NON ZERO)
COMBUF:
0008 20202020 DB ' ' ;8 CHARACTER FILL
0010 20202020 DB ' ' ;8 CHARACTER FILL
0018 434F505952 DB 'COPYRIGHT (C) 1979, DIGITAL RESEARCH ' ; 38
003E DS 128-($-COMBUF)

```

CP/M v.2.2

COPYRIGHT © 1980

DIGITAL RESEARCH, INC.

P. O. BOX 579

PACIFIC GROVE, CA 93950

SERIAL # L-756

```

; TOTAL BUFFER LENGTH IS 128 CHARACTERS
0088 0800 COMADDR:DW COMBUF ;ADDRESS OF NEXT TO CHAR TO SCAN
008A STADDR: DS 2 ;STARTING ADDRESS OF CURRENT FILLFCB REQUEST
;
0004 = DISKA EQU 0004H ;DISK ADDRESS FOR CURRENT DISK
0005 = BDOS EQU 0005H ;PRIMARY BDOS ENTRY POINT
0080 = BUFF EQU 0080H ;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 = PBUFF EQU 9 ;PRINT BUFFER FUNCTION
000A = RBUFF EQU 10 ;READ BUFFER FUNCTION
000B = BREAKF EQU 11 ;BREAK KEY FUNCTION
000C = LIFTF EQU 12 ;LIFT HEAD FUNCTION (NO OPERATION)
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 = DWRTF EQU 21 ;DISK WRITE FUNCTION
0016 = MAKEF EQU 22 ;FILE MAKE FUNCTION
0017 = RENF EQU 23 ;RENAME FILE FUNCTION
0018 = LOGF EQU 24 ;RETURN LOGIN VECTOR
0019 = CSELF EQU 25 ;RETURN CURRENTLY SELECTED DRIVE NUMBER
001A = DMAF EQU 26 ;SET DMA ADDRESS
0020 = USERF EQU 32 ;SET USER NUMBER
;
; SPECIAL FCB FLAGS
0009 = ROFILE EQU 9 ;READ ONLY FILE
000A = SYSFILE EQU 10 ;SYSTEM FILE FLAG
;
; SPECIAL CHARACTERS
000D = CR EQU 13 ;CARRIAGE RETURN
000A = LF EQU 10 ;LINE FEED
005F = LA EQU 5FH ;LEFT ARROW
001A = EOFIL EQU 1AH ;END OF FILE
;
; UTILITY PROCEDURES
PRINTCHAR:
008C 5F0E02C305 MOV E,A! MVI C,PCHARF! JMP BDOS
;
PRINTBC:
;PRINT CHARACTER, BUT SAVE B,C REGISTERS
0092 C5CD8C00C1 PUSH B! CALL PRINTCHAR! POP B! RET
;
CRLF:
0098 3E0DCD9200 MVI A,CR! CALL PRINTBC
009D 3E0AC39200 MVI A,LF! JMP PRINTBC
;
BLANK:
00A2 3E20C39200 MVI A,' '! JMP PRINTBC
;

```

COPYRIGHT © 1980
 DIGITAL RESEARCH, INC.
 P. O. BOX 579
 PACIFIC GROVE, CA 93950

SERIAL # _____

```

PRINT: ;PRINT STRING STARTING AT B,C UNTIL NEXT 00 ENTRY
00A7 C5CD9800E1    PUSH B! CALL CRLF! POP H ;NOW PRINT THE STRING
00AC 7EB7C8    PRINO: MOV A,M! ORA A! RZ ;STOP ON 00
00AF 23E5                INX H! PUSH H ;READY FOR NEXT
00B1 CD8C00E1    CALL PRINTCHAR! POP H ;CHARACTER PRINTED
00B5 C3AC00                JMP PRINO ;FOR ANOTHER CHARACTER

;
INITIALIZE:
00B8 0E0DC30500    MVI C,INITF! JMP BDOS

;
SELECT:
00BD 5F0E0EC305    MOV E,A! MVI C,SELF! JMP BDOS

;
BDOS$INR:
00C3 CD050032EE    CALL BDOS! STA DCNT! INR A! RET

;
OPEN: ;OPEN THE FILE GIVEN BY D,E
00CB 0E0FC3C300    MVI C,OPENF! JMP BDOS$INR

;
OPENC: ;OPEN COMFCB
00D0 AF32ED07    XRA A! STA COMREC ;CLEAR NEXT RECORD TO READ
00D4 11CD07C3CB    LXI D,COMFCB! JMP OPEN

;
CLOSE: ;CLOSE THE FILE GIVEN BY D,E
00DA 0E10C3C300    MVI C,CLOSEF! JMP BDOS$INR

;
SEARCH: ;SEARCH FOR THE FILE GIVEN BY D,E
00DF 0E11C3C300    MVI C,SEARF! JMP BDOS$INR

;
SEARCHN:
;SEARCH FOR THE NEXT OCCURRENCE OF THE FILE GIVEN BY D,E
00E4 0E12C3C300    MVI C,SEARNF! JMP BDOS$INR

;
SEARCHCOM:
;SEARCH FOR COMFCB FILE
00E9 11CD07C3DF    LXI D,COMFCB! JMP SEARCH

;
DELETE: ;DELETE THE FILE GIVEN BY D,E
00EF 0E13C30500    MVI C,DELF! JMP BDOS

;
BDOS$COND:
00F4 CD0500B7C9    CALL BDOS! ORA A! RET

;
DISKREAD:
;READ THE NEXT RECORD FROM THE FILE GIVEN BY D,E
00F9 0E14C3F400    MVI C,DREADF! JMP BDOS$COND

;
DISKREADC:
;READ THE COMFCB FILE
00FE 11CD07C3F9    LXI D,COMFCB! JMP DISKREAD

;
DISKWRITE:
;WRITE THE NEXT RECORD TO THE FILE GIVEN BY D,E
0104 0E15C3F400    MVI C,DWRITF! JMP BDOS$COND

;
MAKE: ;CREATE THE FILE GIVEN BY D,E

```

COPYRIGHT © 1980
 DIGITAL RESEARCH, INC.
 P. O. BOX 579
 PACIFIC GROVE, CA 93950
 SERIAL # _____

```

0109 0E16C3C300      MVI C,MAKEF! JMP BDOS$INR
;
;RENAM: ;RENAME THE FILE GIVEN BY D,E
010E 0E17C30500      MVI C,RENF! JMP BDOS
;
;GETUSER:
;RETURN CURRENT USER CODE IN A
0113 1EFF            MVI E,OFFH ;DROP THROUGH TO SETUSER
;
;SETUSER:
0115 0E20C30500      MVI C,USERF! JMP BDOS ;SETS USER NUMBER
;
;SAVEUSER:
;SAVE USER#/DISK# BEFORE POSSIBLE ^C OR TRANSIENT
011A CD1301          CALL GETUSER ;CODE TO A
011D 87878787        ADD A! ADD A! ADD A! ADD A ;ROT LEFT
0121 21EF07B6        LXI H,CDISK! ORA M ;4B=USER, 4B=DISK
0125 320400          STA DISKA ;STORED AWAY IN MEMORY FOR LATER
0128 C9              RET
;
;SETDISKA:
0129 3AEF073204      LDA CDISK! STA DISKA ;USER/DISK
012F C9              RET
;
;TRANSLATE:
;TRANSLATE CHARACTER IN REGISTER A TO UPPER CASE
0130 FE61D8          CPI 61H! RC ;RETURN IF BELOW LOWER CASE A
0133 FE7BD0          CPI 7BH! RNC ;RETURN IF ABOVE LOWER CASE Z
0136 E65FC9          ANI 5FH! RET ;TRANSLATED TO UPPER CASE
;
;READCOM:
;READ THE NEXT COMMAND INTO THE COMMAND BUFFER
;CHECK FOR SUBMIT FILE
0139 3AAB07B7CA      LDA SUBMIT! ORA A! JZ NOSUB
;SCANNING A SUBMIT FILE
;CHANGE DRIVES TO OPEN AND READ THE FILE
0140 3AEF07B73E      LDA CDISK! ORA A! MVI A,0! CNZ SELECT
;HAVE TO OPEN AGAIN IN CASE XSUB PRESENT
0149 11AC07CDCB      LXI D,SUBFCB! CALL OPEN! JZ NOSUB ;SKIP IF NO SUB
0152 3ABB073D        LDA SUBRC! DCR A ;READ LAST RECORD(S) FIRST
0156 32CC07          STA SUBCR ;CURRENT RECORD TO READ
0159 11AC07CDF9      LXI D,SUBFCB! CALL DISKREAD ;END OF FILE IF LAST RECORD
015F C29601          JNZ NOSUB
;DISK READ IS OK, TRANSFER TO COMBUF
0162 1107002180      LXI D,COMLEN! LXI H,BUFF! MVI B,128! CALL MOVE
;LINE IS TRANSFERRED, CLOSE THE FILE WITH A
;DELETED RECORD
016D 21BA073600      LXI H,SUBMOD! MVI M,0 ;CLEAR FWFLAG
0172 2335            INX H! DCR M ;ONE LESS RECORD
0174 11AC07CDDA      LXI D,SUBFCB! CALL CLOSE! JZ NOSUB
;CLOSE WENT OK, RETURN TO ORIGINAL DRIVE
017D 3AEF07B7C4      LDA CDISK! ORA A! CNZ SELECT
;PRINT TO THE 00
0184 210800CDAC      LXI H,COMBUF! CALL PRINO
018A CDC201CAA7      CALL BREAK$KEY! JZ NOREAD
0190 CDDD01C382      CALL DEL$SUB! JMP CCP ;BREAK KEY DEPRESSED

```

COPYRIGHT © 1980
DIGITAL RESEARCH, INC.

P. O. BOX 579
PACIFIC GROVE, CA 93950

SERIAL # _____

```

0196 CDD01      NOSUB: ;NO SUBMIT FILE! CALL DEL$SUB
                  ;TRANSLATE TO UPPER CASE, STORE ZERO AT END
0199 CD1A01     CALL SAVEUSER ;USER # SAVE IN CASE CONTROL C
019C 0EOA110600 MVI C,RBUFF! LXI D,MAXLEN! CALL BDOS
01A4 CD2901     CALL SETDISKA ;NO CONTROL C, SO RESTORE DISKA
                  NOREAD: ;ENTER HERE FROM SUBMIT FILE
                  ;SET THE LAST CHARACTER TO ZERO FOR LATER SCANS
01A7 21070046   LXI H,COMLEN! MOV B,M ;LENGTH IS IN B
01AB 2378B7     READCOM: INX H! MOV A,B! ORA A ;END OF SCAN?
01AE CABA017E   JZ READCOM1! MOV A,M ;GET CHARACTER AND TRANSLATE
01B2 CD30017705 CALL TRANSLATE! MOV M,A! DCR B! JMP READCOMO
                  ;
                  READCOM1: ;END OF SCAN, H,L ADDRESS END OF COMMAND
01BA 77         MOV M,A ;STORE A ZERO
01BB 2108002288 LXI H,COMBUF! SHLD COMADDR ;READY TO SCAN TO ZERO
01C1 C9        RET

;
BREAK$KEY:
;CHECK FOR A CHARACTER READY AT THE CONSOLE
01C2 0EOBCD0500 MVI C,BREAKF! CALL BDOS
01C7 B7C8       ORA A! RZ
01C9 0E01CD0500 MVI C,RCHARF! CALL BDOS ;CHARACTER CLEARED
01CE B7C9       ORA A! RET

;
CSELECT:
;GET THE CURRENTLY SELECTED DRIVE NUMBER TO REG-A
01D0 0E19C30500 MVI C,CSELF! JMP BDOS

;
SETDMABUFF:
;SET DEFAULT BUFFER DMA ADDRESS
01D5 118000     LXI D,BUFF ;(DROP THROUGH)

;
SETDMA:
;SET DMA ADDRESS TO D,E
01D8 0E1AC30500 MVI C,DMAF! JMP BDOS

;
DEL$SUB:
;DELETE THE SUBMIT FILE, AND SET SUBMIT FLAG TO FALSE
01DD 21AB077EB7 LXI H,SUBMIT! MOV A,M! ORA A! RZ ;RETURN IF NO SUB FILE
01E3 3600       MVI M,0 ;SUBMIT FLAG IS SET TO FALSE
01E5 AFCDBD00   XRA A! CALL SELECT ;ON DRIVE A TO ERASE FILE
01E9 11AC07CDEF LXI D,SUBFCB! CALL DELETE
01EF 3AEF07C3BD LDA CDISK! JMP SELECT ;BACK TO ORIGINAL DRIVE

;
SERIALIZE:
;CHECK SERIALIZATION
01F5 1128032100 LXI D,SERIAL! LXI H,BDOSL! MVI B,6 ;CHECK SIX BYTES
01FD 1ABEC2CF03 SERO: LDAX D! CMP M! JNZ BADSERIAL
0202 132305C2FD INX D! INX H! DCR B! JNZ SERO
0208 C9        RET ;SERIAL NUMBER IS OK

;
COMERR:
;ERROR IN COMMAND STRING STARTING AT POSITION
;'STADDR' AND ENDING WITH FIRST DELIMITER
0209 CD9800     CALL CRLF ;SPACE TO NEXT LINE

```

COPYRIGHT © 1980
 DIGITAL RESEARCH, INC.
 P. O. BOX 579
 PACIFIC GROVE, CA 93950
 SERIAL # _____

```

020C 2A8A00      LHL D STADDR ;H,L ADDRESS FIRST TO PRINT
                  COMERRO: ;PRINT CHARACTERS UNTIL BLANK OR ZERO
020F 7EFE20CA22  MOV A,M! CPI ' '! JZ COMERR1; NOT BLANK
0215 B7CA2202    ORA A! JZ COMERR1; NOT ZERO, SO PRINT IT
0219 E5CD8C00E1  PUSH H! CALL PRINTCHAR! POP H! INX H
021F C30F02      JMP COMERRO; FOR ANOTHER CHARACTER
                  COMERR1: ;PRINT QUESTION MARK,AND DELETE SUB FILE
0222 3E3FCD8C00  MVI A,'?'! CALL PRINTCHAR
0227 CD9800CDD    CALL CRLF! CALL DEL$SUB
022D C38203      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

```

```

0230 1AB7C8      DELIM: ;LOOK FOR A DELIMITER
                  LDAX D! ORA A! RZ ;NOT THE LAST ELEMENT
0233 FE20DA0902  CPI ' '! JC COMERR ;NON GRAPHIC
0238 C8          RZ ;TREAT BLANK AS DELIMITER
0239 FE3DC8      CPI '='! RZ
023C FE5FC8      CPI LA! RZ ;LEFT ARROW
023F FE2EC8      CPI '.'! RZ
0242 FE3AC8      CPI ':'! RZ
0245 FE3BC8      CPI ';'! RZ
0248 FE3CC8      CPI '<'! RZ
024B FE3EC8      CPI '>'! RZ
024E C9          RET ;DELIMITER NOT FOUND

```

COPYRIGHT © 1980
DIGITAL RESEARCH, INC.
P. O. BOX 579
PACIFIC GROVE, CA 93950

SERIAL # _____

```

; DEBLANK: ;DEBLANK THE INPUT LINE
024F 1AB7C8      LDAX D! ORA A! RZ ;TREAT END OF LINE AS BLANK
0252 FE20C013C3  CPI ' '! RNZ! INX D! JMP DEBLANK

```

```

; ADDH: ;ADD A TO H,L
0259 856FD0      ADD L! MOV L,A! RNC
025C 24C9        INR H! RET

```

```

; FILLFCB0:
;EQUIVALENT TO FILLFCB(0)
025E 3E00        MVI A,0

```

```

; FILLFCB:
0260 21CD07CD59  LXI H,COMFCB! CALL ADDH! PUSH H! PUSH H ;FCB RESCANNED AT END
0268 AF32F007    XRA A! STA SDISK ;CLEAR SELECTED DISK (IN CASE A:...)
026C 2A8800EB    LHL COMADDR! XCHG ;COMMAND ADDRESS IN D,E
0270 CD4F02      CALL DEBLANK ;TO FIRST NON-BLANK CHARACTER
0273 EB228A00    XCHG! SHLD STADDR ;IN CASE OF ERRORS
0277 EBE1        XCHG! POP H ;D,E HAS COMMAND, H,L HAS FCB ADDRESS
                  ;LOOK FOR PRECEDING FILE NAME A: B: ...
0279 1AB7CA8902  LDAX D! ORA A! JZ SETCuro ;USE CURRENT DISK IF EMPTY COMMAND
027E DE4047      SBI 'A'-1! MOV B,A ;DISK NAME HELD IN B IF : FOLLOWS
0281 131AFE3ACA  INX D! LDAX D! CPI ':'! JZ SETDSK ;SET DISK NAME IF :
                  ;
0288 1B          SETCuro: ;SET CURRENT DISK
                  DCX D ;BACK TO FIRST CHARACTER OF COMMAND
0289 3AEF0777C3  SETCuro: LDA CDISK! MOV M,A! JMP SETNAME

```

```

0290 7832F007      SETDSK: ;SET DISK TO NAME IN REGISTER B
0294 7013          MOV A,B! STA SDISK ;MARK AS DISK SELECTED
                   MOV M,B! INX D ;PAST THE :
                   ;
                   SETNAME: ;SET THE FILE NAME FIELD
0296 0608          MVI B,8 ;FILE NAME LENGTH (MAX)
0298 CD3002CAB9    SETNAM0: CALL DELIM! JZ PADNAME ;NOT A DELIMITER
029E 23FE2AC2A9    INX H! CPI '*'! JNZ SETNAM1 ;MUST BE ?'S
02A4 363FC3AB02    MVI M,'?'! JMP SETNAM2 ;TO DEC COUNT
                   ;
02A9 7713          SETNAM1: MOV M,A ;STORE CHARACTER TO FCB! INX D
02AB 05C29802      SETNAM2: DCR B ;COUNT DOWN LENGTH! JNZ SETNAM0
                   ;
                   ;END OF NAME, TRUNCATE REMAINDER
02AF CD3002CAC0    TRNAME: CALL DELIM! JZ SETTY ;SET TYPE FIELD IF DELIMITER
02B5 13C3AF02      INX D! JMP TRNAME
                   ;
02B9 23362005C2    PADNAME: INX H! MVI M,' '! DCR B! JNZ PADNAME
                   ;
                   SETTY: ;SET THE TYPE FIELD
02C0 0603FE2EC2    MVI B,3! CPI '.'! JNZ PADTY ;SKIP THE TYPE FIELD IF NO
02C7 13            INX D ;PAST THE ., TO THE FILE TYPE FIELD
                   SETTY0: ;SET THE FIELD FROM THE COMMAND BUFFER
02C8 CD3002CAE9    CALL DELIM! JZ PADTY! INX H! CPI '*'! JNZ SETTY
02D4 363FC3DB02    MVI M,'?' ;SINCE * SPECIFIED! JMP SETTY2
                   ;
02D9 7713          SETTY1: ;NOT A *, SO COPY TO TYPE FIELD
                   MOV M,A! INX D
02DB 05C2C802      SETTY2: ;DECREMENT COUNT AND GO AGAIN
                   DCR B! JNZ SETTY0
                   ;
                   ;END OF TYPE FIELD, TRUNCATE
02DF CD3002CAFO    TRTYP: ;TRUNCATE TYPE FIELD
                   CALL DELIM! JZ EFILL! INX D! JMP TRTYP
                   ;
02E9 23362005C2    PADTY: ;PAD THE TYPE FIELD WITH BLANKS
                   INX H! MVI M,' '! DCR B! JNZ PADTY
                   ;
                   EFILL: ;END OF THE FILENAME/FILETYPE FILL, SAVE COMMAND ADDRESS
                   ;FILL THE REMAINING FIELDS FOR THE FCB
02F0 0603          MVI B,3
02F2 23360005C2    EFILLO: INX H! MVI M,0! DCR B! JNZ EFILLO
02F9 EB228800      XCHG! SHLD COMADDR ;SET NEW STARTING POINT
                   ;
                   ;RECOVER THE START ADDRESS OF THE FCB AND COUNT ?'S
02FD E1010B00      POP H! LXI B,11 ;B=0, C=8+3
0301 237EFE3FC2    SCNQ: INX H! MOV A,M! CPI '?'! JNZ SCNQO
0308 04            ;? FOUND, COUNT IT IN B! INR B
0309 0DC20103      SCNQO: DCR C! JNZ SCNQ
                   ;
                   ;NUMBER OF ?'S IN C, MOVE TO A AND RETURN WITH FLAGS SET
030D 78B7C9        MOV A,B! ORA A! RET
                   ;
                   INTVEC:
0310 44495220      ;INTRINSIC FUNCTION NAMES (ALL ARE FOUR CHARACTERS)
                   DB      'DIR '

```

COPYRIGHT © 1980
 DIGITAL RESEARCH, INC.
 P. O. BOX 579
 PACIFIC GROVE, CA 93950

SERIAL # _____

```

0314 45524120      DB      'ERA '
0318 54595045      DB      'TYPE'
031C 53415645      DB      'SAVE'
0320 52454E20      DB      'REN '
0324 55534552      DB      'USER'
0006 =             INTLEN EQU ($-INTVEC)/4 ;INTRINSIC FUNCTION LENGTH
0328 0000000000    SERIAL: DB 0,0,0,0,0,0

```

```

;
;
;

```

INTRINSIC:

```

;LOOK FOR INTRINSIC FUNCTIONS (COMFCB HAS BEEN FILLED)
032E 2110030E00    LXI H,INTVEC! MVI C,0 ;C COUNTS INTRINSICS AS SCANNED
0333 79FE06D0      INTRINO: MOV A,C! CPI INTLEN ;DONE WITH SCAN?! RNC
                        ;NO, MORE TO SCAN
0337 11CE07        LXI D,COMFCB+1 ;BEGINNING OF NAME
033A 0604          MVI B,4 ;LENGTH OF MATCH IS IN B
033C 1ABE          INTRIN1: LDAX D! CMP M ;MATCH?
033E C24F03        JNZ INTRIN2 ;SKIP IF NO MATCH
0341 132305        INX D! INX H! DCR B
0344 C23C03        JNZ INTRIN1 ;LOOP WHILE MATCHING
;
;COMPLETE MATCH ON NAME, CHECK FOR BLANK IN FCB
0347 1AFE20C254    LDAX D! CPI ' ' ! JNZ INTRIN3 ;OTHERWISE MATCHED
034D 79C9          MOV A,C! RET ;WITH INTRINSIC NUMBER IN A
;
INTRIN2: ;MISMATCH, MOVE TO END OF INTRINSIC
034F 2305C24F03    INX H! DCR B! JNZ INTRIN2
;
INTRIN3: ;TRY NEXT INTRINSIC
0354 0C            INR C ;TO NEXT INTRINSIC NUMBER
0355 C33303        JMP INTRINO ;FOR ANOTHER ROUND

```

```

;

```

CCPCLEAR:

```

;CLEAR THE COMMAND BUFFER
0358 AF           XRA      A
0359 320700       STA      COMLEN
;DROP THROUGH TO START CCP

```

CCPSTART:

```

;ENTER HERE FROM BOOT LOADER
035C 31AB07C5     LXI SP,STACK! PUSH B ;SAVE INITIAL DISK NUMBER
; (HIGH ORDER 4BITS=USER CODE, LOW 4BITS=DISK#)
0360 791F1F1F1F   MOV A,C! RAR! RAR! RAR! RAR! ANI OFH ;USER CODE
0367 5FCD1501     MOV E,A! CALL SETUSER ;USER CODE SELECTED
;INITIALIZE FOR THIS USER, GET $ FLAG
036B CDB800       CALL INITIALIZE ;OFFH IN ACCUM IF $ FILE PRESENT
036E 32AB07       STA SUBMIT ;SUBMIT FLAG SET IF $ FILE PRESENT
0371 C1           POP B ;RECALL USER CODE AND DISK NUMBER
0372 79E60F       MOV A,C! ANI OFH ;DISK NUMBER IN ACCUMULATOR
0375 32EF07       STA CDISK ;CLEARS USER CODE NIBBLE
0378 CDBD00       CALL SELECT ;PROPER DISK IS SELECTED, NOW CHECK SUB FILES
;CHECK FOR INITIAL COMMAND
037B 3A0700B7C2   LDA COMLEN! ORA A! JNZ CCPO

```

```

;

```

CCP:

```

;ENTER HERE ON EACH COMMAND OR ERROR CONDITION
0382 31AB07      LXI SP,STACK

```

```

;ASSUME TYPED ALREADY

```

COPYRIGHT © 1980
 DIGITAL RESEARCH, INC.
 P. O. BOX 579
 PACIFIC GROVE, CA 93950

```

SERIAL ##

```

```

0385 CD9800      CALL CRLF ;PRINT D> PROMPT, WHERE D IS DISK NAME
0388 CDD001      CALL CSELECT ;GET CURRENT DISK NUMBER
038B C641CD8C00  ADI 'A'! CALL PRINTCHAR
0390 3E3ECD8C00  MVI A,'>'! CALL PRINTCHAR
0395 CD3901      CALL READCOM ;COMMAND BUFFER FILLED
                  CCP0: ;(ENTER HERE FROM INITIALIZATION WITH COMMAND FULL)
0398 118000CDD8  LXI D,BUFF! CALL SETDMA ;DEFAULT DMA ADDRESS AT BUFF
039E CDD00132EF  CALL CSELECT! STA CDISK ;CURRENT DISK NUMBER SAVED
03A4 CD5E02      CALL FILLFCBO ;COMMAND FCB FILLED
03A7 C40902      CNZ COMERR ;THE NAME CANNOT BE AN AMBIGUOUS REFERENCE
03AA 3AF007B7C2  LDA SDISK! ORA A! JNZ USERFUNC
                  ;CHECK FOR AN INTRINSIC FUNCTION
03B1 CD2E03      CALL INTRINSIC
03B4 21C103      LXI H,JMTAB ;INDEX IS IN THE ACCUMULATOR
03B7 5F16001919  MOV E,A! MVI D,0! DAD D! DAD D ;INDEX IN D,E
03BC 7E23666FE9  MOV A,M! INX H! MOV H,M! MOV L,A! PCHL
                  ;PC CHANGES TO THE PROPER INTRINSIC OR USER FUNCTION
                  JMTAB:
03C1 7704        DW      DIRECT ;DIRECTORY SEARCH
03C3 1F05        DW      ERASE  ;FILE ERASE
03C5 5D05        DW      TYPE   ;TYPE FILE
03C7 AD05        DW      SAVE   ;SAVE MEMORY IMAGE
03C9 1006        DW      RENAME ;FILE RENAME
03CB 8E06        DW      USER   ;USER NUMBER
03CD A506        DW      USERFUNC;USER-DEFINED FUNCTION
                  BADSERIAL:
03CF 21F376      LXI H,DI OR (HLT SHL 8)
03D2 2200002100  SHLD CCPLOC! LXI H,CCPLOC! PCHL
                  ;
                  ;UTILITY SUBROUTINES FOR INTRINSIC HANDLERS
READERR:
                  ;PRINT THE READ ERROR MESSAGE
03D9 01DF03C3A7  LXI B,RDMSG! JMP PRINT
03DF 5245414420  RDMSG: DB 'READ ERROR',0
                  ;
NOFILE:
                  ;PRINT NO FILE MESSAGE
03EA 01F003C3A7  LXI B,NOFMSG! JMP PRINT
03FO 4E4F204649  NOFMSG: DB 'NO FILE',0
                  ;
GETNUMBER: ;READ A NUMBER FROM THE COMMAND LINE
                  CALL FILLFCBO ;SHOULD BE NUMBER
03F8 CD5E02      LDA SDISK! ORA A! JNZ COMERR ;CANNOT BE PREFIXED
03FB 3AF007B7C2  ;CONVERT THE BYTE VALUE IN COMFCB TO BINARY
0402 21CE07010B  LXI H,COMFCB+1! LXI B,11 ;(B=0, C=11)
                  ;VALUE ACCUMULATED IN B, C COUNTS NAME LENGTH TO ZERO
0408 7EFE20CA33  CONVO: MOV A,M! CPI ' '! JZ CONV1
                  ;MORE TO SCAN, CONVERT CHAR TO BINARY AND ADD
040E 23D630FE0A  INX H! SUI '0'! CPI 10! JNC COMERR ;VALID?
0416 5778        MOV D,A ;SAVE VALUE! MOV A,B ;MULT BY 10
0418 E6E0C20902  ANI 1110$0000B! JNZ COMERR
041D 78          MOV A,B ;RECOVER VALUE
041E 070707      RLC! RLC! RLC ;*8
0421 80DA0902    ADD B! JC COMERR
0425 80DA0902    ADD B! JC COMERR ;*8+*2 = *10

```

```

0429 82DA0902      ADD D! JC COMERR ;+DIGIT
042D 4' DC20804    MOV B,A! DCR C! JNZ CONVO ;FOR ANOTHER DIGIT
0432 C9            RET
                  CONV1: ;END OF DIGITS, CHECK FOR ALL BLANKS
0433 7EFE20C209    MOV A,M! CPI ' '! JNZ COMERR ;BLANKS?
0439 230DC23304    INX H! DCR C! JNZ CONV1
043E 78C9          MOV A,B ;RECOVER VALUE! RET

```

; MOVENAME:

```

0440 0003          ;MOVE 3 CHARACTERS FROM H,L TO D,E ADDRESSES
0442 7E122313      MVI B,3
0446 05C24204      MOVE0: MOV A,M! STAX D! INX H! INX D
044A C9            DCR B! JNZ MOVE0
                  RET

```

; ADDHCF: ;BUFF + A + C TO H,L FOLLOWED BY FETCH

```

044B 2' 300081CD   LXI H,BUFF! ADD C! CALL ADDH! MOV A,M! RET

```

; SETDISK:

```

0454 AF32CD07      ;CHANGE DISKS FOR THIS COMMAND, IF REQUESTED
0458 3AF007B7C8    XRA A! STA COMFCB ;CLEAR DISK NAME FROM FCB
045D 3D21EF07BE    LDA SDISK! ORA A! RZ ;NO ACTION IF NOT SPECIFIED
0463 C3BD00        DCR A! LXI H,CDISK! CMP M! RZ ;ALREADY SELECTED
                  JMP SELECT

```

; RESETDISK:

```

0466 3AF007B7C8    ;RETURN TO ORIGINAL DISK AFTER COMMAND
046B 3D21EF07BE    LDA SDISK! ORA A! RZ ;NO ACTION IF NOT SELECTED
0471 3AEF07C3BD    DCR A! LXI H,CDISK! CMP M! RZ ;SAME DISK
                  LDA CDISK! JMP SELECT

```

;INDIVIDUAL INTRINSICS FOLLOW

DIRECT:

;DIRECTORY SEARCH

```

0477 CD5E02        CALL FILLFCB0 ;COMFCB GETS FILE NAME
047A CD5404        CALL SETDISK ;CHANGE DISK DRIVES IF REQUESTED
047D 21CE077E      LXI H,COMFCB+1! MOV A,M ;MAY BE EMPTY REQUEST
0481 FE20C28F04    CPI ' '! JNZ DIR1 ;SKIP FILL OF ??? IF NOT BLANK
                  ;SET COMFCB TO ALL ??? FOR CURRENT DISK
0486 060B          MVI B,11 ;LENGTH OF FILL ??????????.???
0488 363F2305C2    DIRO: MVI M,'?! INX H! DCR B! JNZ DIRO
                  ;NOT A BLANK REQUEST, MUST BE IN COMFCB
048F 1E00D5        DIR1: MVI E,0! PUSH D ;E COUNTS DIRECTORY ENTRIES
0492 CDE900        CALL SEARCHCOM ;FIRST ONE HAS BEEN FOUND
0495 CCEA03        CZ NOFILE ;NOT FOUND MESSAGE
0498 CA1B05        DIR2: JZ ENDIR
                  ;FOUND, BUT MAY BE SYSTEM FILE
049B 3AEE07        LDA DCNT ;GET THE LOCATION OF THE ELEMENT
049E 0F0F0FE660    RRC! RRC! RRC! ANI 110$0000B! MOV C,A
                  ;C CONTAINS BASE INDEX INTO BUFF FOR DIR ENTRY
04A4 3E0ACD4B04    MVI A,SYSFILE! CALL ADDHCF,VALUE TO A
04A9 17DA0F05      RAL! JC DIR6 ;SKIP IF SYSTEM FILE
                  ;C HOLDS INDEX INTO BUFFER
04AD D17B1CD5      ;ANOTHER FCB FOUND, NEW LINE
                  POP D! MOV A,E! INR E! PUSH D
                  ;E=0,1,2,3,...NEW LINE IF MOD

```

04B1 E603F5
 04B4 C2CC04
 04B7 CD9800
 04BA C5CDD001C1

04BF C641CD9200
 04C4 3E3ACD9200
 04C9 C3D404
 04CC CDA200
 04CF 3E3ACD9200

04D4 CDA200

04D7 0601
 04D9 78CD4B04
 04DD E67F

04DF FE20C2F904
 04E4 F1F5
 04E6 FE03C2F704
 04EB 3E09CD4B04
 04F0 E67FFE20CA

04F7 3E20

04F9 CD9200
 04FC 0478FE0CD2

0503 FE09C2D904

0508 CDA200C3D9

050E F1
 050F CDC201
 0512 C21B05
 0515 CDE400C398

051B D1
 051C C38607

ERASE:

051F CD5E02
 0522 FE0B
 0524 C24205

0527 015205CDA7
 052D CD3901
 0530 21070035C2
 0537 237EFE59C2

053E 23228800

0542 CD5404
 0545 11CD07CDEF

ANI 11B! PUSH PSW ;AND SAVE THE TEST
 JNZ DIRHDRO ;HEADER ON CURRENT LINE
 CALL CRLF
 PUSH B! CALL CSELECT! POP B
 ;CURRENT DISK IN A
 ADI 'A'! CALL PRINTBC
 MVI A,': '! CALL PRINTBC
 JMP DIRHDR1 ;SKIP CURRENT LINE HDR
 DIRHDRO:CALL BLANK ;AFTER LAST ONE
 MVI A,': '! CALL PRINTBC

DIRHDR1:
 CALL BLANK
 ;COMPUTE POSITION OF NAME IN BUFFER
 MVI B,1 ;START WITH FIRST CHARACTER OF NAME
 DIR3: MOV A,B! CALL ADDHCF ;BUFF+A+C FETCHED
 ANI 7FH ;MASK FLAGS
 ;MAY DELETE TRAILING BLANKS
 CPI ' '! JNZ DIR4 ;CHECK FOR BLANK TYPE
 POP PSW! PUSH PSW ;MAY BE 3RD ITEM
 CPI 3! JNZ DIRB ;PLACE BLANK AT END IF NOT
 MVI A,9! CALL ADDHCF ;FIRST CHAR OF TYPE
 ANI 7FH! CPI ' '! JZ DIR5
 ;NOT A BLANK IN THE FILE TYPE FIELD
 DIRB: MVI A,' ' ;RESTORE TRAILING FILENAME CHR
 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
 CALL BLANK! JMP DIR3

DIR5: ;
 ;END OF CURRENT ENTRY
 POP PSW ;DISCARD THE DIRECTORY COUNTER (MOD 4)
 DIR6: CALL BREAK\$KEY ;CHECK FOR INTERRUPT AT KEYBOARD
 JNZ ENDIR ;ABORT DIRECTORY SEARCH
 CALL SEARCHN! JMP DIR2 ;FOR ANOTHER ENTRY
 ENDIR: ;END OF DIRECTORY SCAN
 POP D ;DISCARD DIRECTORY COUNTER
 JMP RETCOM

CALL FILLFCBO ;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
 ERASEFILE:
 CALL SETDISK
 LXI D,COMFCB! CALL DELETE

COPYRIGHT © 1980
 DIGITAL RESEARCH, INC.
 P. O. BOX 579
 PACIFIC GROVE, CA 93950
 SERIAL # _____

054B 3C
054C CCEA03
054F C38607

INR A ;255 RETURNED IF NOT FOUND
CZ NOFILE ;NO FILE MESSAGE IF SO
JMP RETCOM

0552 414C4C2028

ERMSG: DB 'ALL (Y/N)?',0

TYPE:

055D CD5E02C209
0563 CD5404CDD0
0569 CAA705

CALL FILLFCB0! JNZ COMERR ;DON'T ALLOW ?'S IN FILE NAME
CALL SETDISK! CALL OPENC ;OPEN THE FILE
JZ TYPERR ;ZERO FLAG INDICATES NOT FOUND

056C CD980021F1

;FILE OPENED, READ 'TIL EOF
CALL CRLF! LXI H,BPTR! MVI M,255 ;READ FIRST BUFFER
TYPEO: ;LOOP ON BPTR

0574 21F1077EFE
057A DA8705E5

LXI H,BPTR! MOV A,M! CPI 128 ;END BUFFER
JC TYPE1! PUSH H ;CARRY IF 0,1,...,127
;READ ANOTHER BUFFER FULL

057E CDFE00E1
0582 C2A005
0585 AF77

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

0587 34
0588 218000CD59
058E 7EFE1ACA86
0594 CD8C00
0597 CDC201C286
059D C37405

INR M ;BPTR = BPTR + 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 TYPEO ;FOR ANOTHER CHARACTER

TYPEOF: ;END OF FILE, CHECK FOR ERRORS

05A0 3DCA8607
05A4 CDD903
05A7 CD6604C309

DCR A! JZ RETCOM
CALL READERR

TYPERR: CALL RESETDISK! JMP COMERR

SAVE:

05AD CDF803
05B0 F5

CALL GETNUMBER; VALUE TO REGISTER A
PUSH PSW ;SAVE IT FOR LATER

05B1 CD5E02
05B4 C20902
05B7 CD5404
05BA 11CD07D5CD
05C1 D1CD0901
05C5 CAFB05
05C8 AF32ED07
05CC F1
05CD 6F260029
05D1 110001

;SHOULD BE FOLLOWED BY A FILE TO SAVE THE MEMORY IMAGE
CALL FILLFCB0
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
JZ SAVERR ;NO DIRECTORY SPACE
XRA A! STA COMREC; CLEAR NEXT RECORD FIELD
POP PSW ;#PAGES TO WRITE IS IN A, CHANGE TO #SECTORS
MOV L,A! MVI H,0! DAD H!
LXI D,TRAN ;H,L IS SECTOR COUNT, D,E IS LOAD ADDRESS

SAVEO:

05D4 7CB5CAF105
05D9 2B
05DA E5
05DB 21800019E5
05E0 CDD801
05E3 11CD07CD04
05E9 D1E1
05EB C2FB05

;CHECK FOR SECTOR COUNT ZERO
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

05EE C3D405

JMP SAVE0 ;FOR ANOTHER SECTOR

05F1 11CD07CDDA

SAVE1: ;END OF DUMP, CLOSE THE FILE

05F7 3C

LXI D,COMFCB! CALL CLOSE

05F8 C20106

INR A; 255 BECOMES 00 IF ERROR

05FB 010706CDA7

JNZ RETSAVE ;FOR ANOTHER COMMAND

SAVERR: ;MUST BE FULL OR READ ONLY DISK

LXI B,FULLMSG! CALL PRINT

RETSAVE:

;RESET DMA BUFFER

060. CDD5C

CALL SETDMABUFF

0604 C38607

JMP RETCOM

0607 4E4F205350

FULLMSG: DB 'NO SPACE',0

;

;RENAME:

;RENAME A FILE ON A SPECIFIC DISK

0610 CD5E02C209

CALL FILLFCB0! JNZ COMERR ;MUST BE UNAMBIGUOUS

0616 3AF007F5

LDA SDISK! PUSH PSW ;SAVE FOR LATER COMPARE

061A CD5404

CALL SETDISK ;DISK SELECTED

061D CDE900

CALL SEARCHCOM ;IS NEW NAME ALREADY THERE?

0620 C27906

JNZ RENERR3

;FILE DOESN'T EXIST, MOVE TO SECOND HALF OF FCB

0623 21CD0711DD

LXI H,COMFCB! LXI D,COMFCB+16! MVI B,16! CALL MOVE0

;CHECK FOR = OR LEFT ARROW

062E 2A8800EBCD

LHLD COMADDR! XCHG! CALL DEBLANK

0635 FE3DCA3F06

CPI '='! JZ REN1 ;OK IF =

063A FE5FC27306

CPI LA! JNZ RENERR2

063F EB23228800

REN1: XCHG! INX H! SHLD COMADDR ;PAST DELIMITER

;PROPER DELIMITER FOUND

0644 CD5E02C273

CALL FILLFCB0! JNZ RENERR2

;CHECK FOR DRIVE CONFLICT

064A F147

POP PSW! MOV B,A ;PREVIOUS DRIVE NUMBER

064C 21F0077EB7

LXI H,SDISK! MOV A,M! ORA A! JZ REN2

;DRIVE NAME WAS SPECIFIED. SAME ONE?

0654 B870C27306

CMP B! MOV M,B! JNZ RENERR2

0659 70

REN2: MOV M,B ;STORE THE NAME IN CASE DRIVES SWITCHED

065A AF32CD07CD

XRA A! STA COMFCB! CALL SEARCHCOM ;IS OLD FILE THERE?

0661 CA6D06

JZ RENERR1

;

;EVERYTHING IS OK, RENAME THE FILE

0664 11CD07CDOE

LXI D,COMFCB! CALL RENAM

066A C38607

JMP RETCOM

;

RENERR1: ; NO FILE ON DISK

066D CDEA03C386

CALL NOFILE! JMP RETCOM

RENERR2: ; AMBIGUOUS REFERENCE/NAME CONFLICT

0673 CD6604C309

CALL RESETDISK! JMP COMERR

RENERR3: ; FILE ALREADY EXISTS

0679 018206CDA7

LXI B,REMSG! CALL PRINT! JMP RETCOM

0682 46494C4520

REMSG: DB 'FILE EXISTS',0

;USER:

;SET USER NUMBER

068E CDF803

CALL GETNUMBER; LEAVES THE VALUE IN THE ACCUMULATOR

0691 FE10D20902

CPI 16! JNC COMERR; MUST BE BETWEEN 0 AND 15

COPYRIGHT © 1980

DIGITAL RESEARCH, INC.

P. O. BOX 579

PACIFIC GROVE, CA 93950

SERIAL #

```

0696 5F          MOV E,A ;SAVE FOR SETUSER CALL
0697 3ACE07FE20  LDA COMFCB+1! CPI ' '! JZ COMERR
069F CD1501      CALL SETUSER ;NEW USER NUMBER SET
06A2 C38907      JMP ENDCOM

```

```

;
USERFUNC:

```

```

06A5 CDF501      CALL SERIALIZE ;CHECK SERIALIZATION
                  ;LOAD USER FUNCTION AND SET UP FOR EXECUTION
06A8 3ACE07FE20  LDA COMFCB+1! CPI ' '! JNZ USERO
                  ;NO FILE NAME, BUT MAY BE DISK SWITCH
06B0 3AF007B7CA  LDA SDISK! ORA A! JZ ENDCOM ;NO DISK NAME IF 0
06B7 3D32EF07CD  DCR A! STA CDISK! CALL SETDISKA ;SET USER/DISK
06BE CDBD00C389  CALL SELECT! JMP ENDCOM
USERO:           ;FILE NAME IS PRESENT
06C4 11D6071AFE  LXI D,COMFCB+9! LDAX D! CPI ' '! JNZ COMERR ;TYPE ' '
06CD D5CD5404D1  PUSH D! CALL SETDISK! POP D! LXI H,COMTYPE ;.COM
06D5 CD4004      CALL MOVENAME ;FILE TYPE IS SET TO .COM
06D8 CDD000CA6B  CALL OPENC! JZ USERER
                  ;FILE OPENED PROPERLY, READ IT INTO MEMORY
06DE 210001      LXI H,TRAN ;TRANSIENT PROGRAM BASE
06E1 E5          LOADO:  PUSH H ;SAVE DMA ADDRESS
06E2 EBCDD801    XCHG! CALL SETDMA
06E6 11CD07CDF9  LXI D,COMFCB! CALL DISKREAD! JNZ LOAD1
                  ;SECTOR LOADED, SET NEW DMA ADDRESS AND COMPARE
06EF E111800019  POP H! LXI D,128! DAD D
06F4 110000      LXI D,TRANM ;HAS THE LOAD OVERFLOWED?
06F7 7D937C9AD2  MOV A,L! SUB E! MOV A,H! SBB D! JNC LOADERR
06FE C3E106      JMP LOADO ;FOR ANOTHER SECTOR
;
LOAD1:           POP H! DCR A! JNZ LOADERR ;END FILE IS 1
                  CALL RESETDISK ;BACK TO ORIGINAL DISK
0701 E13DC27107  CALL FILLFCB! LXI H,SDISK! PUSH H
0706 CD6604      MOV A,M! STA COMFCB ;DRIVE NUMBER SET
0709 CD5E0221F0  MVI A,16! CALL FILLFCB ;MOVE ENTIRE FCB TO MEMO
0710 7E32CD07    POP H! MOV A,M! STA COMFCB+16
0714 3E10CD6002  XRA A! STA COMREC ;RECORD NUMBER SET TO ZERO
0719 E17E32DD07  LXI D,FCB! LXI H,COMFCB! MVI B,33! CALL MOVEO
071E AF32ED07    ;MOVE COMMAND LINE TO BUFF
0722 115C0021CD  LXI H,COMBUF
072D 210800      BMOVEO:  MOV A,M! ORA A! JZ BMOVE1! CPI ' '! JZ BMOVE1
0730 7EB7CA3E07  INX H! JMP BMOVEO ;FOR ANOTHER SCAN
073A 23C33007    ;FIRST BLANK POSITION FOUND
073E 0600118100  BMOVE1:  MVI B,0! LXI D,BUFF+1! ;READY FOR THE MOVE
0743 7E12B7CA4F  BMOVE2:  MOV A,M! STAX D! ORA A! JZ BMOVE3
                  ;MORE TO MOVE
0749 042313C343  INR B! INX H! INX D! JMP BMOVE2
BMOVE3:         ;B HAS CHARACTER COUNT
074F 78328000    MOV A,B! STA BUFF
0753 CD9800      CALL CRLF
                  ;NOW GO TO THE LOADED PROGRAM
0756 CDD501      CALL SETDMABUFF ;DEFAULT DMA
0759 CD1A01      CALL SAVEUSER ;USER CODE SAVED
                  ;LOW MEMORY DISKA CONTAINS USER CODE
075C CD0001      CALL TRAN ;GONE TO THE LOADED PROGRAM
075F 31AB07      LXI SP,STACK ;MAY COME BACK HERE
0762 CD2901CDBD  CALL SETDISKA! CALL SELECT

```

0768 C38203

JMP CCP

076B CD6604C309

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

0771 017A07CDA7

;
LOADERR: ;CANNOT LOAD THE PROGRAM

0777 C38607

LXI B,LOADMSG! CALL PRINT

077A 424144204C

JMP RETCOM

0783 434F4D

LOADMSG: DB 'BAD LOAD',0

COMTYPE: DB 'COM' ;FOR COM FILES

;
;
RETCOM: ;RESET DISK BEFORE END OF COMMAND CHECK

0786 CD6604

CALL RESETDISK

;
ENDCOM: ;END OF INTRINSIC COMMAND

0789 CD5E02

CALL FILLFCB0 ;TO CHECK FOR GARBAGE AT END OF LINE

078C 3ACE07D620

LDA COMFCB+1! SUI ' '! LXI H,SDISK! ORA M

;0 IN ACCUMULATOR IF NO DISK SELECTED AND BLANK FCB

0795 C20902

JNZ COMERR

0798 C38203

JMP CCP

;
;
;
;
DATA AREAS

079B

DS 16

;8 LEVEL STACK

STACK:

;
;
'SUBMIT' FILE CONTROL BLOCK

07AB 00

SUBMIT: DB

0

;00 IF NO SUBMIT FILE, FF IF SUBMITTING

07AC 0024242420SUBFCB:

DB

0,'\$\$\$'

;FILE NAME IS \$\$\$

07B5 5355420000

DB

'SUB',0,0

;FILE TYPE IS SUB

07BA 00

SUBMOD: DB

0

;MODULE NUMBER

07BB

SUBRC: DS

1

;RECORD COUNT FILED

07BC

DS

16

;DISK MAP

07CC

SUBCR: DS

1

;CURRENT RECORD TO READ

;
;
COMMAND FILE CONTROL BLOCK

07CD

COMFCB: DS

32

;FIELDS FILLED IN LATER

07ED

COMREC: DS

1

;CURRENT RECORD TO READ/WRITE

07EE

DCNT: DS

1

;DISK DIRECTORY COUNT (USED FOR ERROR CODES)

07EF

CDISK: DS

1

;CURRENT DISK

07F0

SDISK: DS

1

;SELECTED DISK FOR CURRENT OPERATION

;NONE=0, A=1, B=2 ...

07F1

BPTR: DS

1

;BUFFER POINTER

07F2

END

CCPLOC

COPYRIGHT © 1980

DIGITAL RESEARCH, INC.

P. O. BOX 579

PACIFIC GROVE, CA 93950

SERIAL #

0259 ADDH	044B ADDHCF	03CF BADSERIAL	0005 BDOS	00F4 BDOSCOND
00C3 BDOSINR	0800 BDOSL	00A2 BLANK	0730 BMOVE0	073E BMOVE1
0743 BMOVE2	074F BMOVE3	07F1 BPTR	000B BREAKF	01C2 BREAKKEY
0080 BUFF	0398 CCP0	0058 CCPCLEAR	0382 CCP	0000 CCPLOC
035C CCPSTART	07EF CDISK	00DA CLOSE	0010 CLOSEF	0088 COMADDR
0008 COMBUF	0209 COMERR	020F COMERRO	0222 COMERR1	07CD COMFCB
0007 COMLEN	07ED COMREC	0783 COMTYPE	0408 CONVO	0433 CONV1
000D CR	0098 CRLF	01D0 CSELECT	0019 CSELF	07EE DCNT
024F DEBLANK	00EF DELETE	0013 DELF	0230 DELIM	01DD DELSUB
0488 DIR0	048F DIR1	0498 DIR2	04D9 DIR3	04F9 DIR4
050E DIR5	050F DIR6	04F7 DIRB	0477 DIRECT	04CC DIRHRO
04D4 DIRHDR1	0004 DISKA	00F9 DISKREAD	00FE DISKREADC	0104 DISKWRITE
001A DMAF	0014 DREADF	0015 DWRTF	02F0 EFIL	02F2 EFILLO
0789 ENDCOM	051B ENDIR	001A EOFIL	051F ERASE	0542 ERASEFILE
0552 ERMSG	0000 FALSE	005C FCB	025E FILLFCB0	0260 FILLFCB
0607 FULLMSG	03F8 GETNUMBER	0113 GETUSER	000D INITF	00B8 INITIALIZE
0006 INTLEN	0333 INTRINO	033C INTRIN1	034F INTRIN2	0354 INTRIN3
032E INTRINSIC	0310 INTVEC	03C1 JMPTAB	005F LA	000A LF
000C LIFTF	06E1 LOAD0	0001 LOAD1	0771 LOADFR	077A LOADMSG
0018 LOGF	0016 MAKEF	0109 MAKE	0006 MAXLEN	0442 MOVE0
0440 MOVENAME	03EA NOFILE	03F0 NOFMSG	01A7 NOREAD	0196 NOSUB
00D0 OPENC	000F OPENF	00CB OPEN	02B9 PADNAME	02E9 PADTY
0009 PBUFF	0002 PCHARF	00AC PRINO	0092 PRINTBC	00A7 PRINT
008C PRINTCHAR	000A RBUFF	0001 RCHARF	03DF RDMSG	0139 READCOM
01AB READCOMO	01BA READCOM1	03D9 READERR	063F REN1	0659 REN2
010E RENAM	0610 RENAME	066D RENERR1	0673 RENERR2	0679 RENERR3
0017 RENF	0682 RENMSG	0466 RESETDISK	0786 RETCOM	0601 RETSAVE
0009 ROFILE	05AD SAVE	05D4 SAVE0	05F1 SAVE1	05FB SAVERR
011A SAVEUSER	0301 SCNQ	0309 SCNQ0	07F0 SDISK	00DF SEARCH
00E9 SEARCHCOM	00E4 SEARCHN	0011 SEARF	0012 SEARNF	00BD SELECT
000E SELF	01FD SERO	01F5 SERIALIZE	0328 SERIAL	0289 SETCuro
0288 SETCUR	0129 SETDISKA	0454 SETDISK	01D5 SETDMABUFF	01D8 SETDMA
0290 SETDSK	0298 SETNAM0	02A9 SETNAM1	02AB SETNAM2	0296 SETNAME
02C0 SETTY	02C8 SETTY0	02D9 SETTY1	02DB SETTY2	0115 SETUSER
07AB STACK	008A STADDR	07CC SUBCR	07AC SUBFCB	07AB SUBMIT
07BA SUBMOD	07BB SUBRC	000A SYSFILE	0000 TESTING	0100 TRAN
0000 TRANM	0130 TRANSLATE	02AF TRNAME	02DF TRTYP	FFFF TRUE
055D TYPE	0574 TYPE0	0587 TYPE1	05A0 TYPEOF	05A7 TYPERR
06C4 USER0	076B USERER	0020 USERF	068E USER	06A5 USERFUNC

COPYRIGHT © 1980
 DIGITAL RESEARCH, INC.
 P. O. BOX 579
 PACIFIC GROVE, CA 93950

SERIAL # _____