

0000'  
0000'

aseg  
Title CSLOAD  
SUBTTL Version 02.02.83  
.280

```

;*****
;##                               ##
;## CP/M COLD START LOADER      ##
;##                               ##
;*****
    
```

```

;(0080) : LADEADRESSE
;(0082) : ZAHL DER LESEVERSUCHE
    
```

```

FFF0      SEL      EQU      OFFF0H
FFFC      STATR    EQU      OFFFCH
FFFC      CMDREG   EQU      OFFFCH
FFFD      TRKREG   EQU      OFFFDH
FFFE      SECREG   EQU      OFFFEH
FFFF      DATREG   EQU      OFFFFH
FC0B      PREAD    EQU      OFC0BH
F600      BIOS     EQU      OF600H
FC11      DSPLMS   EQU      OFC11H
    
```

.PHASE OH

```

-----
0000 21 DF80      START: LD      HL,0DF80H      ;Abspeicheradresse
0003 22 0080      LD      (0080H),HL
0006 3E 02        LD      A,02H          ;Befehl: 'Restore'
0008 02           LD      (BC),A
0009 18 39        JR      NBUSY
000B 3E 01        L00002: LD     A,01H
000D 32 FFFE      L00003: LD     (SECREG),A      ;Sector 1
0010 3E 01        LD      A,01H
0012 32 FFF0      LD      (SEL),A          ;Drive A
0015 3E 10        LD      A,10H          ;10 Leseversuche
0017 32 0082      LD      (0082H),A
001A 2A 0080      L00004: LD     HL,(0080H)
001D CD FC0B      CALL    PREAD          ;einen Sector lesen
0020 28 08        JR      Z,L00005      ;wenn fehlerfrei
0022 21 0082      LD      HL,0082H
0025 35           DEC     (HL)
0026 20 F2        JR      NZ,L00004      ;erneut lesen
0028 18 D6        JR      START          ;neu starten
002A 22 0080      L00005: LD     (0080H),HL
002D 3A FFFE      LD      A,(SECREG)
0030 3C           INC     A          ;Sector Nr. incrementieren
0031 FE 13        CP      13H
0033 20 D8        JR      NZ,L00003
0035 3A FFFD      LD      A,(TRKREG)
0038 3C           INC     A          ;Track Nr. incrementieren
0039 FE 03        CP      03H
003B CA 004F      JP      Z,END          ;nach 2 Tracks: Meldung
003E 32 FFFF      LD      (DATREG),A
0041 3E 13        LD      A,13H
0043 02           LD      (BC),A          ;Befehl: 'Track lesen'
    
```

```

;-----
0044 3E 04          NBSY: LD    A,04H          ;Warten auf 'NOT BUSY'
0046 3D             LOOP: DEC   A          ;Warteschleife
0047 20 FD          JR     NZ,LOOP
0049 0A             L00008: LD   A,(BC)
004A 0F             RRCA
004B 38 FC          JR     C,L00008
004D 18 BC          JR     L00002
;-----
004F 21 0058        END:  LD   HL,TEXT          ;Display Meldung
0052 CD FC11        CALL DSPLMS
0055 C3 F600        JP    BIOS          ;CP/M-Kaltstart
;-----
0058 0D             TEXT:  DEFB 0DH          ;Text
0059 1C             DEFB 1CH
005A 1F             DEFB 1FH
005B 36 33 4B 20    DEFM '63K CP/M VERS 2.3'
005F 43 50 2F 4D
0063 20 56 45 52
0067 53 20 32 2E
0068 33
006C 0D             DEFB 0DH
006D 0A             DEFB 0AH
006E 32 2E 32 2E    DEFM '2.2.83 EG3003'
0072 38 33 20 45
0076 47 33 30 30
007A 33
007B 0D             DEFB 0DH
007C 0A             DEFB 0AH
END      START

```

Macros:

Symbols:

BIOS	F600	CMDREG	FFFC	DATREG	FFFF	DSPLMS	FC11
END	004F	L00002	000B	L00003	000D	L00004	001A
L00005	002A	L0000B	0049	LOOP	0046	NBUSY	0044
PREAD	FC0B	SECREG	FFFE	SEL	FFF0	START	0000
STAIR	FFFC	TEXT	005B	TRKREG	FFFD		

No Fatal error(s)

0000'

Title BIOS63

.SALL

;MARCH 21, 1980  
 ;Revised February 2, 1983  
 ;by R. Vomrath  
 ;for use with Video Genie System EG 3003

; THIS PROGRAM CAN BE ASSEMBLED WITH  
 ; THE M80 MACRO ASSEMBLER.  
 ; MEMSIZ SHOULD BE CHANGED AS NECESSARY  
 ; OMIKRON CAN SUPPLY CPM SYSTEMS FOR  
 ; OTHER MEMORY SIZES FROM STOCK.

; BE VERY CAREFUL IF CHANGES ARE MADE.

; THE LIST ROUTINE IS PLACED LAST FOR  
 ; POSSIBLE MEMORY PATCHES.

; SET MINISWITCH EQUAL TO TRUE FOR MINI DRIVES  
 ; IN THE CBIOS FOR THE MINI DRIVES,  
 ; DRIVES A AND B ARE MINI,  
 ; DRIVES C AND D ARE LARGE

; SET MINISWITCH EQUAL TO FALSE FOR 8" DRIVES  
 ; IN THE CBIOS FOR 8" DRIVES,  
 ; DRIVES A AND B ARE 8",  
 ; DRIVES C AND D ARE MINI.

.LIST  
 SUBTTL Version 02.02.83

0000'

ASEG

003F	MEMSIZ EQU 63
FFFF	TRUE EQU 0FFFFH
0000	FALSE EQU NOT TRUE
FFFF	MINISWITCH EQU TRUE
0000	FORTYTRACK EQU FALSE
AC00	CBASE EQU (MEMSIZ-20)*1024
	.PHASE CBASE+4A00H
C810	BIAS EQU 0C810H
0028	RETRIES EQU 40
FFF8	PTR EQU 37EBH+BIAS
FC08	PREAD EQU 0FC08H
FC0E	PWRITE EQU 0FC0EH
FC02	PCONIN EQU 0FC02H
FC05	PCONOUT EQU 0FC05H
FC08	PCONST EQU 0FC08H
FC14	PREADY EQU 0FC14H
E806	BDS EQU CBASE+3C06H
E000	CPMB EQU CBASE+3400H
FFFC	CMD EQU 37ECH+BIAS
FFFC	STAT EQU 37ECH+BIAS
FFFD	TRKREG EQU 37EDH+BIAS
FFFE	SECREG EQU 37EEH+BIAS
FFFF	DATA EQU 37EFH+BIAS
FFF0	SEL EQU 37E0H+BIAS

```

F000
F000 C3 F052          JMP      ROOT
F003 C3 F05C          JWBOOT: JMP      WBOOT
F006 C3 FC08          JMP      PCONST
F009 C3 FC02          JMP      PCONIN
F00C C3 FC05          JMP      PCONOUT
F00F C3 F8CB          JMP      LIST
F012 C3 F6C5          JMP      PUNCH
F015 C3 F8C8          JMP      READER
F018 C3 F6C6          JMP      HOME
F01B C3 F6E6          JMP      SELDSK
F01E C3 F748          JMP      SETTRK
F021 C3 F74D          JMP      SETSEC
F024 C3 F752          JMP      SETDMA
F027 C3 F757          JMP      READ
F02A C3 F75D          JMP      WRITE
F02D C3 F641          JMP      LISTST
F030 C3 F04C          JMP      SECTRAM
F033 00              PRNTST: DB      0
F034              ERRCT: DS      1
F035 00              TEST:  DB      0
F036 00              FFTEST: DB      0
F037 42 42          PTRAM:  DB      66,66
F039 03              MSTEPR: DB      01 ;STEP RATE FOR MINI DRIVE = 40 MS.
F03A 00              LSTEPR: DB      0 ;STEP RATE FOR 8" DRIVES = 6 MS.
F03B 00 01 82 83    IF      MINISWITCH
DRTAB: DB      0,1,82H,83H
ELSE
DRTAB: DB      80H,81H,2,3
ENDIF
F03F 00              READY: DB      0
IF      MINISWITCH
F040 12              SECTRZ: DB      18
ELSE
SECTRZ: DB      26
ENDIF
F041 DB FD          LISTST: IN      0FDH ;GET PRINTER STATUS
F043 E6 F0          ANI      0F0H
F045 FE 30          CPI      30H
F047 3E FF          MVI      A,0FFH
F049 CB            RZ
F04A 3C            INR      A
F04B C9            RET
F04C 06 00          SECTRAM: MVI      B,0
F04E EB            XCHG
F04F 09            DAD      B
F050 6E            MOV      L,M
F051 C9            RET
F052 4F            BOOT:  XRA      A
F053 32 0004        STA      4 ;INITIALIZE DEFAULT DRIVE
F056 32 0003        STA      IOBYTE ;& IOBYTE
F059 F3            DI
JR      J3
F05A 18 36          +
F05C 31 0100        JWBOOT: LXI      SP,100H
F05F 21 0F80        LXI      H,CPMB-80H
F062 22 F7F5        SHLD   DMA
F065 0E 00          MVI      C,0
F067 CD F6E6        CALL   SELDSK
F06A CD F6C6        CALL   HOME
F06D 0E 01          MVI      C,01

```

```

F66F CD F74D          CALL SETSEC
F672 06 2D           MVI B,45
F674 C5              J1:  PUSH B
F675 CD F757         CALL READ
                          JRNZ WBOOT

F678 20 E2           +
F67A 22 F7F5         SHLD DMA
F67D 3A F7F7         LDA SECTR
F680 21 F640         LXI H,SECTRZ
F683 BE              CMP M
                          JRC J2

F684 38 05           +
F686 21 F7F8         LXI H,TRACK
F689 34              INR M
F68A AF              XRA A
F68B 3C              J2:  INR A
F68C 32 F7F7         STA SECTR
F68F C1              POP B
                          DJNZ J1

F690 10 E2           +
F692 3E C3           J3:  MVI A,0C3H
F694 21 F603         LXI H,JWBOOT
F697 32 0000         STA 0
F69A 22 0001         SHLD 1
F69D 21 E806         LXI H,BDOS } C3E806 mach 0005 H
F6A0 32 0005         STA 5H
F6A3 22 0006         SHLD 6
F6A6 32 0045         STA 45H ;ENABLE LOWERCASE DRIVER
F6A9 3A 0004         LDA 4
F6AC 4F              MOV C,A
F6AD C3 E000         JMP CPMB
F6B0 3E 04           DELAY: MVI A,4 ;50 MS. DELAY FOR VALID STATUS
F6B2 3D              DLOOP: DCR A
                          JRNZ DLOOP

F6B3 20 FD           +
F6B5 C9              RET
F6B6                WINT: EXX ;25 MS. DELAY TIMES A REG VALUE
F6B6 D9           +
F6B7 4F              MOV C,A
F6B8 11 06A4         TLOOP1: LXI D,1700
F6BB 1B              TLOOP2: DCX D
F6BC 7B              MOV A,E
F6BD B2              ORA D
                          JRNZ TLOOP2

F6BE 20 FB           +
F6C0 0D              DCR C
                          JRNZ TLOOP1

F6C1 20 F5           +
                          EXX

F6C3 D9           +
F6C4 C9              RET
F6C5 C9              PUNCH: RET
F6C6 AF              HOME: XRA A ;DELAY HOME UNTIL DISK ACCESS
F6C7 32 F7FB         STA TRACK
F6CA C9              RET
F6CB 3A F63F         TREADY: LDA READY ;SELECTS DRIVE AND STROBES TIMEOUT
F6CE B7              ORA A
                          JRZ RLOOP ;IF NEW DRIVE TEST FOR READY

F6CF 28 0E           +
F6D1 CD F6DF         CALL RLOOP
F6D4 CD FC14         CALL PREADY ;PROM ROUTINE TO TEST DISK SPINNING
F6D7 3E 03           MVI A,3 ;HEADLOAD DELAY

```

```

F6D9 CD F6B6          CALL WINT ;ALLOW EXTRA TIME FOR MINI HEAD SETTLE
F6DC 32 F63F          STA  READY
F6DF 3A F7FC          RLOOP: LDA  MASK
F6E2 32 FFF0          STA  SEL
F6E5 C9              RET
F6E6 3A F7F9          SELDSK: LDA ODISK ;GET PRESENT DISK NO.
F6E9 B9              CMP  C ;SEE IF SAME
                   JRZ  SKIP1

F6EA 28 05          +
F6EC 3E FF          MVI  A,OFFH ;NOTE THAT A NEW DRIVE IS REQUESTED
F6EE 32 F63F          STA  READY
F6F1 79              SKIP1: MOV  A,C
F6F2 E6 03          ANI  3
F6F4 32 F7F9          STA  ODISK ;UPDATE ODISK NO.
F6F7 21 F63B          LXI  H,DRTAB
F6FA 85              ADD  L
F6FB 6F              MOV  L,A
F6FC 4E              MOV  C,M
F6FD 21 F7EB          LXI  H,0BYTE
                   BIT  7,C

0001                +
F700 CB 79          +
F702 F5              PUSH PSW
F703 CC F729          CZ   MINI
F706 F1              POP  PSW
F707 C4 F736          CNZ  LARGE
                   RES  7,C
                   RES 1,C

0001                +
F70A CB B9          +
F70C 21 F7ED          LXI  H,TBASE ;SET HL TO BASE OF TRACK TABLE
F70F 06 00          MVI  B,0
F711 C5              PUSH B
F712 09              DAD  B
F713 22 F7FA          SHLD TPOINTER ;STORE HL POINTING TO CURRENT TRACK
F716 7E              MOV  A,M
F717 32 FFFD          STA  TRYREG
F71A 3E 04          MVI  A,4
F71C 85              ADD  L
F71D 6F              MOV  L,A
F71E 7E              MOV  A,M
F71F 32 F7FC          J21: STA  MASK ;COMPUTE SELECT MASK
F722 E1              POP  H
F723 29              DAD  H
F724 29              DAD  H
F725 29              DAD  H
F726 29              DAD  H
F727 19              DAD  D
F728 C9              RET
F729 3E 80          MINI: MVI  A,80H ;SETS CLOCK RATE TO 1 MHZ
F72B CD F738          CALL LS
F72E 3A F639          LDA  MSTEPR
F731 77              MOV  M,A
F732 11 F83E          LXI  D,DPBASE+64
F735 C9              RET
F736 3E C0          LARGE: MVI  A,0C0H ;SETS 2 MHZ CLOCK RATE
F738 32 FFFE          LS:  STA  SECREG ;OUTPUT TO CONTROLLER
F73B 32 0044          STA  44H ;STORE DRIVE SIZE IN MEMORY
F73E 77              MOV  M,A ;ALSO SAVE IN MEMORY FOR WRITE
F73F 23              INX  H ;POINT TO STEPR
F740 3A F63A          LDA  LSTEPR
F743 77              MOV  M,A
F744 11 F7FE          LXI  D,DPBASE
    
```

F747	C9		RET		
F748	79		SETTRK: MOV	A.C	
F749	32 F7FB		STA	TRACK	
F74C	C9		RET		
F74D	79		SETSEC: MOV	A.C	
F74E	32 F7F7		STA	SECTR	
F751	C9		RET		
F752			SETDMA: SBCD	DMA	
F752	ED 43	+			
F754	F7F5	+			
F756	C9		RET		
F757			READ: LXIY	J26	
F757	FD 21	+			
F759	F7AC	+			
			JR	J22	
F758	18 04	+			
F75D			WRITE: LXIY	J27	
F75D	FD 21	+			
F75F	F7B1	+			
F761	3A FFFC		J22: LDA	STAT	:TEST FOR TIMEOUT
F764	07		RLC		
			JRNC	J23	:SKIP IF NOT TIMED OUT
F765	30 08	+			
F767	CD F6CB		J50: CALL	TREADY	:CHECKS FOR DISK IN DRIVE
F76A	3E 28		MVI	A.40	:1 SEC DELAY FOR MINI MOTOR
F76C	CD F6B6		CALL	WINT	
F76F	AF		J23: XRA	A	
F770	32 F634		STA	ERRCT	
F773	CD F6CB		J33: CALL	TREADY	
F776	2A F7FA		LHLD	TPOINTER	
F779	3A F7F7		LDA	SECTR	
F77C	32 FFFE		STA	SECREG	
F77F	01 FFFC		LXI	B.CMD	
F782	11 FFFF		LXI	D.DATA	
F785	3E FF		MVI	A.OFFH	
F787	BE		CMP	M	
			JRZ	SEEKH	
F788	28 4A	+			
F78A	3A F7FB		LDA	TRACK	:GET DESIRED TRACK
F78D	BE		CMP	M	:TEST IF SEEK NECESSARY
			JRZ	J34	
F78E	28 14	+			
F790	12		STAX	D	
F791	77		MOV	M.A	
F792	3A F7EC		LDA	STEPR	:GET STEP RATE
F795	F6 10		ORI	10H	:ADD SEEK COMMAND
F797	02		STAX	B	
F798	CD F6B0		CALL	DELAY	
F79B	0A		J24: LDAX	B	
F79C	0F		RRC		
			JRC	J24	
F79D	38 FC	+			
F79F	3E 03		MVI	A.3	:STEP SETTLE TIME
F7A1	CD F6B6		CALL	WINT	
F7A4	CD F6DF		J34: CALL	RLOOP	
F7A7	2A F7F5		LHLD	DMA	
			PCIY		
F7AA	FD E9	+			
F7AC	CD FC0B		J26: CALL	PREAD	
			JR	J30	
F7AF	18 0D	+			
F7B1	3A F7EB		J27: LDA	DBYTE	



```

                                BIT    6.A    :SET FLAGS FOR WRITE
0007      +
F7B4  CB 77      +
F7B6  CD F0E     CALL    PWRITE  :PROM ROUTINE FOR DISK WRITE
                                BIT    5.A
0007      +
F7B9  CB 77      +
                                JRNZ   J50
F7BB  20 AA      +
F7BD  B7         ORA    A
F7BE  CB         J30:  RZ
                                EXAF
F7BF  08         +
F7C0  21 F634   LXI    H,ERRCT
F7C3  34         INR    M
F7C4  7E         MOV    A,M
F7C5  FE 28     CPI    RETRIES
                                JRZ    J31
F7C7  28 20     +
F7C9  FE 23     CPI    RETRIES-5
                                JRZ    SEEKH
F7CB  28 07     +
                                JRC    J34
F7CD  38 D5     +
                                EXAF
F7CF  08         +
                                BIT    4.A    :TEST FOR RECORD NOT FOUND
0007      +
F7D0  CB 67     +
                                JRZ    J34
F7D2  28 D0     +
F7D4  CD F6DF   SEEKH: CALL   RLOOP
F7D7  3A F7EC   LDA    STEPR  :GET HOME COMMAND
F7DA  02         STAX   B
F7DB  CD F6B0   CALL   DELAY
F7DE  0A         J19:  LDAX  B
F7DF  0F         RRC
                                JRC    J19
F7E0  38 FC     +
F7E2  2A F7FA   LHLD  TPOINTER
F7E5  36 00     MVI    M,0    :UPDATE TRACK RECORD
                                JR     J33
F7E7  18 8A     +
F7E9  3C         J31:  INR    A
F7EA  C9         RET
F7EB         DBYTE: DS    1    :C0 = LARGE 80 = MINI
F7EC         STEPR: DS    1    :MUST FOLLOW DBYTE.CONTAINS STEP RATE
                                IF    MINISWITCH
F7ED  02 FF FF FF TBASE: DB    2.OFFH.OFFH.OFFH    :TRACK FOR EACH DRIVE
                                ENDIF
                                IF    NOT MINISWITCH
F7ED         TBASE: DB    1.OFFH.OFFH.OFFH
                                ENDIF
F7F1  01 02 04 08 MSKTAB: DB    1.2.4.8
F7F5         DMA:  DS    2
F7F7         SECTR: DS    1
F7FB  00         TRACK: DB    0
F7F9  00         ODISK: DB    0
F7FA         TPOINTER: DS    2
F7FC         MASK:  DS    1
F7FD         LFMASK: DS    1
                                .XALL

```

```

DISKS 8
0008 + NDISKS SET 8
F7FE + DPBASE EQU $ :base of disk parameter blocks
F7FE F88D 0000 + DPE%0: DW XLT%0,0000H :translate table
F802 0000 0000 + DW 0000H,0000H :scratch area
F806 F980 F87E + DW DIRBUF.DPB%0 :dir buff, parm block
F80A FA1F FA00 + DW CSV%0,ALV%0 :check, alloc vector
F80E F88D 0000 + DPE%1: DW XLT%1,0000H :translate table
F812 0000 0000 + DW 0000H,0000H :scratch area
F816 F980 F87E + DW DIRBUF.DPB%1 :dir buff, parm block
F81A FA4E FA2F + DW CSV%1,ALV%1 :check, alloc vector
F81E F88D 0000 + DPE%2: DW XLT%2,0000H :translate table
F822 0000 0000 + DW 0000H,0000H :scratch area
F826 F980 F87E + DW DIRBUF.DPB%2 :dir buff, parm block
F82A FA7D FA5E + DW CSV%2,ALV%2 :check, alloc vector
F82E F88D 0000 + DPE%3: DW XLT%3,0000H :translate table
F832 0000 0000 + DW 0000H,0000H :scratch area
F836 F980 F87E + DW DIRBUF.DPB%3 :dir buff, parm block
F83A FA8C FA8D + DW CSV%3,ALV%3 :check, alloc vector
F83E F8B6 0000 + DPE%4: DW XLT%4,0000H :translate table
F842 0000 0000 + DW 0000H,0000H :scratch area
F846 F980 F8A7 + DW DIRBUF.DPB%4 :dir buff, parm block
F84A FAD1 FABC + DW CSV%4,ALV%4 :check, alloc vector
F84E F8B6 0000 + DPE%5: DW XLT%5,0000H :translate table
F852 0000 0000 + DW 0000H,0000H :scratch area
F856 F980 F8A7 + DW DIRBUF.DPB%5 :dir buff, parm block
F85A FAF6 FAE1 + DW CSV%5,ALV%5 :check, alloc vector
F85E F8B6 0000 + DPE%6: DW XLT%6,0000H :translate table
F862 0000 0000 + DW 0000H,0000H :scratch area
F866 F980 F8A7 + DW DIRBUF.DPB%6 :dir buff, parm block
F86A FB1B FB06 + DW CSV%6,ALV%6 :check, alloc vector
F86E F8B6 0000 + DPE%7: DW XLT%7,0000H :translate table
F872 0000 0000 + DW 0000H,0000H :scratch area
F876 F980 F8A7 + DW DIRBUF.DPB%7 :dir buff, parm block
F87A FB40 FB2B + DW CSV%7,ALV%7 :check, alloc vector

DISKDEF 0,1,26,6,1024,243,64,64,2
0019 + SECMAX SET 26-(1)
001A + SECTORS SET SECMAX+1
001E + ALS%0 SET (243)/8
001F + ALS%0 SET ALS%0+1
0010 + CSS%0 SET (64)/4
0008 + BLKVAL SET 1024/128
0000 + BLKSHF SET 0
0000 + BLKMSK SET 0
0001 + BLKSHF SET BLKSHF+1
0001 + BLKMSK SET (BLKMSK SHL 1) OR 1
0004 + BLKVAL SET BLKVAL/2
0002 + BLKSHF SET BLKSHF+1
0003 + BLKMSK SET (BLKMSK SHL 1) OR 1
0002 + BLKVAL SET BLKVAL/2
0007 + BLKSHF SET BLKSHF+1
0007 + BLKMSK SET (BLKMSK SHL 1) OR 1
0001 + BLKVAL SET BLKVAL/2
0001 + BLKVAL SET 1024/1024
0000 + EXTMSK SET 0
0040 + DIRREM SET 64
0020 + DIRBKS SET 1024/32
0000 + DIRBLK SET 0
8000 + DIRBLK SET (DIRBLK SHR 1) OR 8000H
0020 + DIRREM SET DIRREM-DIRBKS
C000 + DIRBLK SET (DIRBLK SHR 1) OR 8000H
0000 + DIRREM SET 0
    
```

FB7E	FB7E	001A	+	DPR&0	EDU	\$
FB7E	001A	+			DW	SECTORS :SEC PER TRACK
FB80	03	+			DB	BLKSHF :BLOCK SHIFT
FB81	07	+			DB	BLKMSK :BLOCK MASK
FB82	00	+			DB	EXTMSK :EXTENT MASK
FB83	00F2	+			DW	(243)-1 :DISK SIZE - 1
FB85	003F	+			DW	(64)-1 :DIRECTORY MAX
FB87	C0	+			DB	DIRBLK SHR 8 :ALLOCO
FB88	00	+			DB	DIRBLK AND OFFH :ALLOCI
FB89	0010	+			DW	(64)/4 :CHECK SIZE
FB8B	0002	+			DW	2 :OFFSET
0000		+		NXTSEC	SET	0
0000		+		NXTBAS	SET	0
001A		+		GCDM	SET	SECTORS
0006		+		GCDM	SET	6
0000		+		GCDR	SET	0
0004		+		GCDX	SET	GCDM/GCDM
0002		+		GCDR	SET	GCDM-GCDX*GCDM
0006		+		GCDM	SET	GCDM
0002		+		GCDM	SET	GCDR
0003		+		GCDX	SET	GCDM/GCDM
0000		+		GCDR	SET	GCDM-GCDX*GCDM
000D		+		NELTST	SET	SECTORS/GCDM
000D		+		NELTS	SET	NELTST
FB8D		+		1L1&0	EDU	\$ :translate table
FB8D	01	+			DB	NXTSEC+(1)
0006		+		NXTSEC	SET	NXTSEC+(6)
000C		+		NELTS	SET	NELTS-1
FB8E	07	+			DB	NXTSEC+(1)
000C		+		NXTSEC	SET	NXTSEC+(6)
0008		+		NELTS	SET	NELTS-1
FB8F	0D	+			DB	NXTSEC+(1)
0012		+		NXTSEC	SET	NXTSEC+(6)
000A		+		NELTS	SET	NELTS-1
FB90	13	+			DB	NXTSEC+(1)
0018		+		NXTSEC	SET	NXTSEC+(6)
0009		+		NELTS	SET	NELTS-1
FB91	19	+			DB	NXTSEC+(1)
001E		+		NXTSEC	SET	NXTSEC+(6)
0004		+		NXTSEC	SET	NXTSEC-SECTORS
0008		+		NELTS	SET	NELTS-1
FB92	05	+			DB	NXTSEC+(1)
000A		+		NXTSEC	SET	NXTSEC+(6)
0007		+		NELTS	SET	NELTS-1
FB93	08	+			DB	NXTSEC+(1)
0010		+		NXTSEC	SET	NXTSEC+(6)
0006		+		NELTS	SET	NELTS-1
FB94	11	+			DB	NXTSEC+(1)
0016		+		NXTSEC	SET	NXTSEC+(6)
0005		+		NELTS	SET	NELTS-1
FB95	17	+			DB	NXTSEC+(1)
001C		+		NXTSEC	SET	NXTSEC+(6)
0002		+		NXTSEC	SET	NXTSEC-SECTORS
0004		+		NELTS	SET	NELTS-1
FB96	03	+			DB	NXTSEC+(1)
0008		+		NXTSEC	SET	NXTSEC+(6)
0003		+		NELTS	SET	NELTS-1
FB97	09	+			DB	NXTSEC+(1)
000E		+		NXTSEC	SET	NXTSEC+(6)
0002		+		NELTS	SET	NELTS-1
FB98	0F	+			DB	NXTSEC+(1)
0014		+		NXTSEC	SET	NXTSEC+(6)

0001		+	NELTS	SET	NELTS-1	
F899	15	+		DB	NXTSEC+(1)	
001A		+	NXTSEC	SET	NXTSEC+(6)	
0000		+	NXTSEC	SET	NXTSEC-SECTORS	
0000		+	NELTS	SET	NELTS-1	
0001		+	NXTBAS	SET	NXTBAS+1	
0001		+	NXTSEC	SET	NXTBAS	
000D		+	NELTS	SET	NELTST	
F89A	02	+		DB	NXTSEC+(1)	
0007		+	NXTSEC	SET	NXTSEC+(6)	
000C		+	NELTS	SET	NELTS-1	
F89B	08	+		DB	NXTSEC+(1)	
000D		+	NXTSEC	SET	NXTSEC+(6)	
000B		+	NELTS	SET	NELTS-1	
F89C	0E	+		DB	NXTSEC+(1)	
0013		+	NXTSEC	SET	NXTSEC+(6)	
000A		+	NELTS	SET	NELTS-1	
F89D	14	+		DB	NXTSEC+(1)	
0019		+	NXTSEC	SET	NXTSEC+(6)	
0009		+	NELTS	SET	NELTS-1	
F89E	1A	+		DB	NXTSEC+(1)	
001F		+	NXTSEC	SET	NXTSEC+(6)	
0005		+	NXTSEC	SET	NXTSEC-SECTORS	
0008		+	NELTS	SET	NELTS-1	
F89F	06	+		DB	NXTSEC+(1)	
000B		+	NXTSEC	SET	NXTSEC+(6)	
0007		+	NELTS	SET	NELTS-1	
F8A0	0C	+		DB	NXTSEC+(1)	
0011		+	NXTSEC	SET	NXTSEC+(6)	
0006		+	NELTS	SET	NELTS-1	
F8A1	12	+		DB	NXTSEC+(1)	
0017		+	NXTSEC	SET	NXTSEC+(6)	
0005		+	NELTS	SET	NELTS-1	
F8A2	18	+		DB	NXTSEC+(1)	
001D		+	NXTSEC	SET	NXTSEC+(6)	
0003		+	NXTSEC	SET	NXTSEC-SECTORS	
0004		+	NELTS	SET	NELTS-1	
F8A3	04	+		DB	NXTSEC+(1)	
0009		+	NXTSEC	SET	NXTSEC+(6)	
0003		+	NELTS	SET	NELTS-1	
F8A4	0A	+		DB	NXTSEC+(1)	
000F		+	NXTSEC	SET	NXTSEC+(6)	
0002		+	NELTS	SET	NELTS-1	
F8A5	10	+		DB	NXTSEC+(1)	
0015		+	NXTSEC	SET	NXTSEC+(6)	
0001		+	NELTS	SET	NELTS-1	
F8A6	16	+		DB	NXTSEC+(1)	
001B		+	NXTSEC	SET	NXTSEC+(6)	
0001		+	NXTSEC	SET	NXTSEC-SECTORS	
0000		+	NELTS	SET	NELTS-1	
0002		+	NXTBAS	SET	NXTBAS+1	
0002		+	NXTSEC	SET	NXTBAS	
000D		+	NELTS	SET	NELTST	
					DISKDEF 1,0	
F87E		+	DPB&1	EQU	DPB&0	;equivalent parameters
001F		+	ALS&1	EQU	ALS&0	;same allocation vector size
0010		+	CSS&1	EQU	CSS&0	;same checksum vector size
F88D		+	XLT&1	EQU	XLT&0	;same translate table
					DISKDEF 2,0	
F87E		+	DPB&2	EQU	DPB&0	;equivalent parameters
001F		+	ALS&2	EQU	ALS&0	;same allocation vector size
0010		+	CSS&2	EQU	CSS&0	;same checksum vector size

```

F86D      +      XLT&2 EQU      XLT&0 :same translate table
          DISKDEF 3.0
F87E      +      DPB&3 EQU      DPB&0 :equivalent parameters
001F      +      ALS&3 EQU      ALS&0 :same allocation vector size
0010      +      CSS&3 EQU      CSS&0 :same checksum vector size
F88D      +      XLT&3 EQU      XLT&0 :same translate table
          DISKDEF 4.1.18.4.1024.166,64.64.3
0011      +      SECMAX SET     18-(1)
0012      +      SECTORS SET     SECMAX+1
0014      +      ALS&4 SET     (166)/8
0015      +      ALS&4 SET     ALS&4+1
0010      +      CSS&4 SET     (64)/4
0008      +      BLKVAL SET     1024/128
0000      +      BLKSHF SET     0
0000      +      BLKMSK SET     0
0001      +      BLKSHF SET     BLKSHF+1
0001      +      BLKMSK SET     (BLKMSK SHL 1) OR 1
0004      +      BLKVAL SET     BLKVAL/2
0002      +      BLKSHF SET     BLKSHF+1
0003      +      BLKMSK SET     (BLKMSK SHL 1) OR 1
0002      +      BLKVAL SET     BLKVAL/2
0003      +      BLKSHF SET     BLKSHF+1
0007      +      BLKMSK SET     (BLKMSK SHL 1) OR 1
0001      +      BLKVAL SET     BLKVAL/2
0001      +      BLKVAL SET     1024/1024
0000      +      EXTMSK SET     0
0040      +      DIRREM SET     64
0020      +      DIRBKS SET     1024/32
0000      +      DIRBLK SET     0
8000      +      DIRBLK SET     (DIRBLK SHR 1) OR 8000H
0020      +      DIRREM SET     DIRREM-DIRBKS
C000      +      DIRBLK SET     (DIRBLK SHR 1) OR 8000H
0000      +      DIRREM SET     0
F8A7      +      DPB&4 EQU      $
F8A7      0012      +      DW      SECTORS :SEC PER TRACK
F8A9      03        +      DB      BLKSHF :BLOCK SHIFT
F8AA      07        +      DB      BLKMSK :BLOCK MASK
F8AB      00        +      DB      EXTMSK :EXTENT MASK
F8AC      00A5      +      DW      (166)-1 :DISK SIZE - 1
F8AE      003F      +      DW      (64)-1 :DIRECTORY MAX
F8B0      C0        +      DB      DIRBLK SHR 8 :ALLOCO
F8B1      00        +      DB      DIRBLK AND OFFH :ALLOCI
F8B2      0010      +      DW      (64)/4 :CHECK SIZE
F8B4      0003      +      DW      3 :OFFSET
0000      +      NXTSEC SET     0
0000      +      NXTBAS SET     0
0012      +      GCDM SET     SECTORS
0004      +      GCDN SET     4
0000      +      GCDR SET     0
0004      +      GCDX SET     GCDM/GCDN
0002      +      GCDR SET     GCDM-GCDX*GCDN
0004      +      GCDM SET     GCDN
0002      +      GCDN SET     GCDR
0002      +      GCDX SET     GCDM/GCDN
0000      +      GCDR SET     GCDM-GCDX*GCDN
0009      +      NELTST SET     SECTORS/GCDN
0009      +      NELTS SET     NELTST
F8B6      +      XLT&4 EQU      $ :translate table
F8B6      01        +      DB      NXTSEC+(1)
0004      +      NXTSEC SET     NXTSEC+(4)
0008      +      NELTS SET     NELTS-1
F8B7      05        +      DB      NXTSEC+(1)

```

(83)-1 für 40 Track CP/M.

0008		+	NXTSEC	SET	NXTSEC+(4)
0007		+	NELTS	SET	NELTS-1
F888	09	+		DB	NXTSEC+(1)
000C		+	NXTSEC	SET	NXTSEC+(4)
0006		+	NELTS	SET	NELTS-1
F8B9	0D	+		DB	NXTSEC+(1)
0010		+	NXTSEC	SET	NXTSEC+(4)
0005		+	NELTS	SET	NELTS-1
F8BA	11	+		DB	NXTSEC+(1)
0014		+	NXTSEC	SET	NXTSEC+(4)
0002		+	NXTSEC	SET	NXTSEC-SECTORS
0004		+	NELTS	SET	NELTS-1
F8BB	03	+		DB	NXTSEC+(1)
0006		+	NXTSEC	SET	NXTSEC+(4)
0003		+	NELTS	SET	NELTS-1
F8BC	07	+		DB	NXTSEC+(1)
000A		+	NXTSEC	SET	NXTSEC+(4)
0002		+	NELTS	SET	NELTS-1
F8BD	0B	+		DB	NXTSEC+(1)
000E		+	NXTSEC	SET	NXTSEC+(4)
0001		+	NELTS	SET	NELTS-1
F8BE	0F	+		DB	NXTSEC+(1)
0012		+	NXTSEC	SET	NXTSEC+(4)
0000		+	NXTSEC	SET	NXTSEC-SECTORS
0000		+	NELTS	SET	NELTS-1
0001		+	NXTBAS	SET	NXTBAS+1
0001		+	NXTSEC	SET	NXTBAS
0009		+	NELTS	SET	NELTST
F8BF	02	+		DB	NXTSEC+(1)
0005		+	NXTSEC	SET	NXTSEC+(4)
0008		+	NELTS	SET	NELTS-1
F8C0	06	+		DB	NXTSEC+(1)
0009		+	NXTSEC	SET	NXTSEC+(4)
0007		+	NELTS	SET	NELTS-1
F8C1	0A	+		DB	NXTSEC+(1)
000D		+	NXTSEC	SET	NXTSEC+(4)
0006		+	NELTS	SET	NELTS-1
F8C2	0E	+		DB	NXTSEC+(1)
0011		+	NXTSEC	SET	NXTSEC+(4)
0005		+	NELTS	SET	NELTS-1
F8C3	12	+		DB	NXTSEC+(1)
0015		+	NXTSEC	SET	NXTSEC+(4)
0003		+	NXTSEC	SET	NXTSEC-SECTORS
0004		+	NELTS	SET	NELTS-1
F8C4	04	+		DB	NXTSEC+(1)
0007		+	NXTSEC	SET	NXTSEC+(4)
0003		+	NELTS	SET	NELTS-1
F8C5	08	+		DB	NXTSEC+(1)
000B		+	NXTSEC	SET	NXTSEC+(4)
0002		+	NELTS	SET	NELTS-1
F8C6	0C	+		DB	NXTSEC+(1)
000F		+	NXTSEC	SET	NXTSEC+(4)
0001		+	NELTS	SET	NELTS-1
F8C7	10	+		DB	NXTSEC+(1)
0013		+	NXTSEC	SET	NXTSEC+(4)
0001		+	NXTSEC	SET	NXTSEC-SECTORS
0000		+	NELTS	SET	NELTS-1
0002		+	NXTBAS	SET	NXTBAS+1
0002		+	NXTSEC	SET	NXTBAS
0009		+	NELTS	SET	NELTST
					DISKDEF 5.4
F8A7		+	DPB&5	EQU	DPB&4 :equivalent parameters

```

0015      +   ALS&5 EQU   ALS&4 ;same allocation vector size
0010      +   CSS&5 EQU   CSS&4 ;same checksum vector size
F8B6      +   XLT&5 EQU   XLT&4 ;same translate table
                DISKDEF 6,4
F8A7      +   DPB&6 EQU   DPB&4 ;equivalent parameters
0015      +   ALS&6 EQU   ALS&4 ;same allocation vector size
0010      +   CSS&6 EQU   CSS&4 ;same checksum vector size
F8B6      +   XLT&6 EQU   XLT&4 ;same translate table
                DISKDEF 7,4
F8A7      +   DPB&7 EQU   DPB&4 ;equivalent parameters
0015      +   ALS&7 EQU   ALS&4 ;same allocation vector size
0010      +   CSS&7 EQU   CSS&4 ;same checksum vector size
F8B6      +   XLT&7 EQU   XLT&4 ;same translate table

```

```

.FALL
F8C8      3E 1A      READER: MVI   A,1AH
F8CA      C9                RET

```

```

;-----
;
;   Line printer driver
;
;-----

```

```

F8CB      3A F924      LIST:  LDA   UMSCH
F8CE      B7                ORA   A       ;war letztes Zeichen '01H'
                JRNZ  CTRLA ;ja
F8CF      20 09      +
F8D1      3E 01                MVI   A,01H ;nein, testen ob jetzt '01H'
F8D3      B9                CMP   C
                JRNZ  LIST1 ;nein, normal weiter
F8D4      20 09      +
F8D6      32 F924      STA   UMSCH ;merken
F8D9      C9                RET       ;nichts ausgeben
F8DA      AF      CTRLA: XRA   A       ;Zeichen nach CTRL A
F8DB      32 F924      STA   UMSCH ;merken
F8DE      C9                RET       ;nichts ausgeben
F8DF      2A F925      LIST1: LHLD  LCNTR ;LINE COUNTER IN HL
F8E2      3A 0003      LDA   IOBYTE
F8E5      E6 40                ANI   FFBIT ;LIEGT TOF AN?
                JRNZ  LSTCHR ;NEIN
F8E7      20 0D      +
F8E9      3E 0C                MVI   A,FF  ;JA, FORM FEED VERLANGT?
F8EB      B9                CMP   C
                JRZ   TOF   ;JA, IGNORIEREN
F8EC      2B 29      +
F8EE      3A 0003      LDA   IOBYTE ;MERKEN, DASS TOF NICHT
F8F1      F6 40                ORI   FFBIT ;MEHR ANLIEGT
F8F3      32 0003      STA   IOBYTE
F8F6      DB FD      LSTCHR: IN   PRINTER ;DRUCKERSTATUS ABFRAGEN
F8F8      E6 F0                ANI   0F0H
F8FA      FE 30                CPI   30H  ;READY?
                JRNZ  LSTCHR
F8FC      20 FB      +
F8FE      79                MOV   A,C
F8FF      D3 FD                OUT  PRINTER ;ZEICHEN AUSGEBEN
F901      FE 0A                CPI   LF   ;LINE FEED?
                JRNZ  FF?   ;NEIN
F903      20 0F      +

```

```

F905 25          DCR  H      ;ZEILENZAEHLER RUNTERZAEHLEN
F906 22 F925    SHLD LCNTR
F909 C0         RNZ

F90A 3A 0003    LDA  IOBYTE ;SEITENVORSCHUB?
F90D E6 80     ANI  LCBIT
F90F C0         RNZ          ;NEIN

F910 0E 0C     MVI  C,FF   ;JA, SEITENVORSCHUB AUSFUEHREN
          JR    LSTCHR

F912 18 E2     +

F914 FE 0C     FF?: CPI  FF   ;FORM FEED?
F916 C0         RNZ          ;NEIN, ALLES KLAR

F917 3A 0003    TOF: LDA  IOBYTE
F91A E6 BF     ANI  NOT FFBIT ;TOF-BIT ZURUECKSETZEN
F91C 32 0003    STA  IOBYTE
F91F 65         MOV  H,L
F920 22 F925    SHLD LCNTR ;ZEILENZAEHLER INITIALISIEREN
F923 C9         RET
F924          UMSCH: DS   1    ;01H, wenn letztes Zeichen 01H war, sonst 00H

F925 42 42     LCNTR: DB   66,66
0003          IOBYTE EQU   3
000C          FF   EQU   12
000A          LF   EQU   10
00FD          PRINTER EQU 0FDH
0040          FFBIT EQU 01000000B
0080          LCBIT EQU 10000000B
  
```

```

.XALL
.DEPHASE
ORG 4D80H-4A00H
.PHASE CBASE+4D80H
ENDEF
  
```

```

F980 + BEGDAT EQU $
F980 + DIRBUF: DS 128 ;directory acces buffer
FA00 + ALV&0: DS ALS&0
FA1F + CSV&0: DS CSS&0
FA2F + ALV&1: DS ALS&1
FA4E + CSV&1: DS CSS&1
FA5E + ALV&2: DS ALS&2
FA7D + CSV&2: DS CSS&2
FA8D + ALV&3: DS ALS&3
FAAC + CSV&3: DS CSS&3
FABC + ALV&4: DS ALS&4
FAD1 + CSV&4: DS CSS&4
FAE1 + ALV&5: DS ALS&5
FAF6 + CSV&5: DS CSS&5
FB06 + ALV&6: DS ALS&6
FB1B + CSV&6: DS CSS&6
FB2B + ALV&7: DS ALS&7
FB40 + CSV&7: DS CSS&7
FB50 + ENDDAT EQU $
01D0 + DATSIZ EQU $-BEGDAT
          END ;
  
```



Macros:

BIT	DOB	DDW	DEFDS	DISKDE	DISKS	DJNZ	DPBHDR
DSKHDR	ENDEF	EXAF	EXX	GCD	JR	JRC	JRNC
JRNZ	JRZ	LDS	LXIY	PCIY	RES	SBCD	

Symbols:

ALS0	001F	ALS1	001F	ALS2	001F	ALS3	001F
ALS4	0015	ALS5	0015	ALS6	0015	ALS7	0015
ALV0	FA00	ALV1	FA2F	ALV2	FA5E	ALV3	FA8D
ALV4	FABC	ALV5	FAE1	ALV6	FB06	ALV7	FB2B
BDS	EB06	BEGDAT	F980	BIAS	C810	BLKMSK	0007
BLKSHF	0003	BLKVAL	0001	BOOT	F652	CBASE	AC00
CMD	FFFC	CPMB	E000	CSS0	0010	CSS1	0010
CSS2	0010	CSS3	0010	CSS4	0010	CSS5	0010
CSS6	0010	CSS7	0010	CSV0	FA1F	CSV1	FA4E
CSV2	FA7D	CSV3	FAAC	CSV4	FAD1	CSV5	FAF6
CSV6	FB1B	CSV7	FB40	CTRLA	FBDA	DATA	FFFF
DATSIZ	01D0	DBYTE	F7EB	DELAY	F6B0	DIRBKS	0020
DIRBLK	C000	DIRBUF	F980	DIRREM	0000	DLOOP	F6B2
DMA	F7F5	DPB0	F87E	DPB1	F87E	DPB2	F87E
DPB3	F87E	DPB4	F8A7	DPB5	F8A7	DPB6	F8A7
DPB7	F8A7	DPBASE	F7FE	DPE0	F7FE	DPE1	F80E
DPE2	FB1E	DPE3	F82E	DPE4	F83E	DPE5	F84E
DPE6	F85E	DPE7	F86E	DRTAB	F63B	ENDDAT	FB50
ERRCT	F634	EXTMSK	0000	FALSE	0000	FF	000C
FF?	F914	FFBIT	0040	FFTEST	F636	FORTYT	0000
GCDM	0004	GCDN	0002	GCDR	0000	GCDX	0002
HOME	F6C6	IOBYTE	0003	J1	F674	J19	F7DE
J2	F68B	J21	F71F	J22	F761	J23	F76F
J24	F79B	J26	F7AC	J27	F7B1	J3	F692
J30	F7BE	J31	F7E9	J33	F773	J34	F7A4
J50	F767	JWBOOT	F603	LARGE	F736	LCBIT	0080
LCNTR	F925	LF	000A	LFMASK	F7FD	LIST	F8CB
LIST1	F8DF	LISTST	F641	LS	F738	LSTCHR	F8F6
LSTEPR	F63A	MASK	F7FC	MEMSIZ	003F	MINI	F729
MINISM	FFFF	MSKTAB	F7F1	MSTEPR	F639	NDISKS	0008
NELTS	0009	NELTST	0009	NXTBAS	0002	NXTSEC	0002
ODISK	F7F9	PCONIN	FC02	PCONOU	FC05	PCONST	FC08
PREAD	FC0B	PREADY	FC14	PRINTE	00FD	PRNTST	F633
PTR	FFF8	PTRAM	F637	PUNCH	F6C5	PWRITE	FC0E
READ	F757	READER	F8C8	READY	F63F	REG	0007
RETRIE	0028	RLOOP	F6DF	SECMAX	0011	SECREG	FFFE
SECTOR	0012	SECTR	F7F7	SECTRA	F64C	SECTRZ	F640
SEEKH	F7D4	SEL	FFF0	SELDISK	F6E6	SETDMA	F752
SETSEC	F74D	SETTRK	F748	SKIP1	F6F1	STAT	FFFC
STEPR	F7EC	TBASE	F7ED	TEST	F635	TLOOP1	F6B8
TLOOP2	F6BB	TDF	F917	TPOINT	F7FA	TRACK	F7F8
TREADY	F6CB	TRKREG	FFFD	TRUE	FFFF	UMSCH	F924
WBOOT	F65C	WINT	F6B6	WRITE	F75D	XLT0	F88D
XLT1	F88D	XLT2	F88D	XLT3	F88D	XLT4	F886
XLT5	F8B6	XLT6	F8B6	XLT7	F8B6		

No Fatal error(s)

0000'  
0000'

Aseq  
Title CBIOS63  
SUBTTL Version 02.02.83  
.Z80

```

:      ;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
:      ;;                                ;;
:      ;;      EPROM - CP/M      ;;
:      ;;                                ;;
:      ;!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

```

:Belegung der Speicherplaetze:

```

;(0040): Cursorstand                                PCONOUT
;(0041): Cursorstand
;(0042): von Cursor verdecktes Zeichen
;(0043): Bit 0 : letztes Zeichen: 'Esc'
;         Bit 1 : letzte Zeichen : 'Esc' - '='
;         Bit 2 : letzte Zeichen : 'Esc' - '=' - 'Line Nr'
;(0044): Bit 0 : '01H' als letztes Zeichen auf den Bildschirm
;(0045): Bit 0 : Kleinschrift in Grossschrift umwandeln
;         Bit 1 : toggelt nach Eingabe von CTRL C (bei '0' nur Grossschreibung)
;         Bit 3 : Graphik- in ASCII-Zeichen umwandeln
;(0046): von Tastatur eingegebenes Zeichen (nach PCONST)      PCONIN
;(0047): Tastaturzeile 3801H bei Abfrage von Inkey
;(0048):          3802H
;(0049):          3804H
;(004A):          3808H
;(004B):          3810H
;(004C):          3820H
;(004D):          3840H
;(004E): 'FF': echo deleted Characters : '00': not echo deleted Characters
;         Bit 2 : Taste gedrueckt
;(004F): von Tastatur gelesenes Zeichen (nach PCONIN)

```

```

0000      BANKE EQU 0H
0001      BANKA EQU 1H
0003      EPRFCX EQU 3H
FFFF0     SEL EQU 0FFF0H
FFFC      STATR EQU 0FFFCH
FFFC      CMDREG EQU 0FFFCH
FFFD      TRKREG EQU 0FFFDH
FFFE      SECRES EQU 0FFFEH
FFFF      DATREG EQU 0FFFFH

```

```

.PHASE 0FC00H
Page 66

```

FC00

```

:*****
FC00 18 17      L00001: JR      L00002                      ;EINSPRUNGADRESSEN
FC02  C3 FEB5      JP      PCDNIN
FC05  C3 FD09      JP      PCNDU
FC08  C3 FE91      JP      PCNST
FC0B  C3 FCA2      JP      PREAD
FC0E  C3 FCC5      JP      PWRITE
FC11  C3 FF71      JP      DSPLMS
FC14  C3 FFB1      JP      PREADY
FC17  18 15      JR      KALTST
    
```

```

FC19  18 4B      L00002: JR      NMI                      ;Warteschleife
FC1B  21 0029     L00003: LD      HL,UML-0FC00H  ;Eprom verlagern
FC1E  11 F000      LD      DE,0F000H
FC21  01 0005      LD      BC,0005H
FC24  ED 80      LDIR
FC26  C3 F000      JP      0F000H
    
```

```

FC29  D3 03      UML:  OUT      (EPRFCX).A
FC2B  C3 FC2E      JP      KALTST
    
```

```

FC2E  31 0100     KALTST: LD      SP,100H                      ;CP/M Kaltstart
FC31  1E 01      LD      E,1H
FC33  CD FC7b     CALL     DRVSEL                      ;Drive Select
FC3b  18 1F      JR      L00015
    
```

```

FC38  3E 03      L00013: LD      A,03H  ;RESTORE                      ;Cold Start Loader laden
FC3A  CD FC6F     CALL     CMDFDC                      ;Track 00 suchen
FC3D  0A          L00014: LD      A,(BC)
FC3E  0F          RRCA
FC3F  38 FC      JR      C,L00014
FC41  1E 01      LD      E,01H                      ;select drive A
FC43  CD FC7E     CALL     L00012
FC46  32 FFFE     LD      (SECREG),A                  ;Sector 1
FC49  21 0000     LD      HL,0000H                    ;ab Adresse 0000 abspeichern
FC4C  CD FCA2     CALL     PREAD                        ;Lesen eines Sectors
FC4F  CA 0000     JP      Z,0000H                      ;falls fehlerfrei Sorung zum Cold Start Loader
FC52  3A 0080     LD      A,(0080H)
FC55  CB 77      BIT      0,A                          ;8'' - Diskette
FC57  3E C0      L00015: LD      A,0C0H
FC59  28 01      JR      Z,L00016
FC5B  07          RLCA
FC5C  32 0080     L00016: LD      (0080H),A              ;C0 fuer 8'', 80 fuer Mini-Diskette
FC5F  32 FFFE     LD      (SECREG),A                  ;FDC auf 2 MHz umschalten
FC62  18 D4      JR      L00013                      ;Leseversuch mit Drive A als 8''- Laufwerk
FC64  00          NOP
FC65  00          NOP
    
```

```

FC66  06 C0      NMI:  LD      B,0C0H                      ;Warteschleife
FC68  0B          LOOP1: DEC     BC
FC69  79          LD      A,C
FC6A  80          OR      B
FC6B  20 FB      JR      NZ,LOOP1
FC6D  18 AC      JR      L00003                      ;Eprom verlagern
    
```

```

FC6F  02          CMDFDC: LD      (BC),A                  ;Kommando an FDC          ;Kommando an FDC + Warteschleife
FC70  3E 04      LD      A,4H                      ;Warteschleife
FC72  3D          LOOP2: DEC     A
FC73  20 FD      JR      NZ,LOOP2
FC75  C9          RET
    
```

```

-----
FC76  CD FC7E      DRVSEL: CALL  L00012      ;select Drive      ;Select Drive
FC79  06 C0        LD      B.0COH
FC7B  CD FF6B      CALL  LOOP3      ;Warteschleife
FC7E  7B          L00012: LD      A.E
FC7F  32 FFF0      LD      (SEL).A
FC82  01 FFFC      LD      BC.STATR
FC85  C9          RET
-----
FC86  D9          WPRTXT: EXX          ;'PROTECTED TYPE C' ausgeben
FC87  21 FF9A      LD      HL.TEXT1      ;Text:'PROTECTED'
FC8A  CD FF71      L00018: CALL  DSPLMS
FC8D  21 FFA8      LD      HL,TEXT2      ;Text:'TYPE C'
FC90  CD FF71      CALL  DSPLMS
FC93  CD FEB5      L00019: CALL  PCONIN
FC96  FE 03        CP      03H          :CTRL C
FC98  28 94        JR      Z.KALTST      ;Kaltstart bei CTRL C
FC9A  E6 5F        AND     5FH
FC9C  FE 43        CP      43H          :C
FC9E  20 F3        JR      NZ,L00019     ;falsche Eingabe, neuer Versuch
FCA0  D9          EXX
FCA1  C9          RET
-----
FCA2  3E 88        PREAD: LD      A.88H          ;Sector lesen
FCA4  CD FC6F      CALL  CMDFDC          ;Befehl:'READ'
FCA7  11 FFFF      LD      DE,DATREG
FCAA  0A          L00021: LD      A.(BC)
FCAB  E6 03        AND     03H
FCAD  E2 FCAA      JP      PD,L00021     ;Warten auf DRQ
FCB0  1A          L00022: LD      A.(DE)      ;Byte laden und abspeichern
FCB1  77          LD      (HL),A
FCB2  23          INC     HL
FCB3  0A          L00023: LD      A.(BC)
FCB4  E6 02        AND     02H          :DRQ
FCB6  20 F8        JR      NZ,L00022
FCB8  0A          LD      A.(BC)
FCB9  E6 02        AND     02H          :DRQ
FCBB  20 F3        JR      NZ,L00022
FCBD  0A          LD      A.(BC)
FCBE  0F          RRCA
FCBF  38 F2        JR      C.L00023      ;wenn 'busy'
FCC1  0A          LD      A.(BC)      ;wenn Befehl abgearbeitet (busy='0')
FCC2  E6 9F        AND     9FH          ;Fehlertest
FCC4  C9          RET
-----
FCC5  08          PWRITE: EX     AF.AF'      ;A-Reg und Flags retten ;Sector schreiben
FCC6  3E A8        LD      A.0A8H
FCC8  CD FC6F      CALL  CMDFDC          ;Befehl: 'WRITE'
FCCB  08          EX     AF.AF'      ;A-Reg und Flags zurueckholen
FCCC  20 0C        JR      NZ,LARGE      ;bei 8''-Laufwerk
FCCD  0A          MINI: LD      A.(BC)      ;'not busy'?          ;Mini Diskette
FCCF  0F          RRCA
FCD0  30 2D        JR      NC.WRPROT     ;bei 'not busy' Abfrage ob WRITE PROTECT
FCD2  0F          RRCA
FCD3  30 F9        JR      NC.MINI      ;warten auf DRQ
FCD5  7E          LD      A.(HL)
FCD6  12          LD      (DE).A      ;Byte aus Speicher auf Diskette
FCD7  23          INC     HL
FCD8  18 F4        JR      MINI
FCDA  0A          LARGE: LD     A.(BC)      ;8''-Diskette
FCDB  E6 03        AND     03H
FCDD  E2 FCDA      JP      PD,LARGE      ;Warten auf DRQ

```

```

FCE0 7E          LD      A.(HL)
FCE1 12          LD      (DE).A      ;Byte von Speicherplatz auf Diskette
FCE2 3E E0       LD      A.0E0H
FCE4 32 FFFE     LD      (SECREG).A   ;FDC-Clock auf 2MHz umsch.+CPU-Halt vorbereiten
FCE7 23          INC     HL
FCE8 3E 28       LD      A.28H
FCEA CD FC72     CALL   LOOP2          ;Warteschleife
FCED 0A          LD      A.(BC)      ;CPU anhalten bis DRQ
FCEE 7E          LD      A.(HL)
FCEF 12          LD      (DE).A      ;von Speicherplatz auf Diskette
FCF0 23          INC     HL
FCF1 0A          LD      A.(BC)
FCF2 7E          L00027: LD     A.(HL)
FCF3 12          LD      (DE).A      ;von Speicherplatz auf Diskette solange 'busv'='1'
FCF4 23          INC     HL
FCF5 7E          LD      A.(HL)
FCF6 04          LD      A.(BC)
FCF7 0F          RRCA
FCFB 38 F8       JR      C.L00027   ;solange 'busv' = '1'
FCFA 3E C0       LD      A,0C0H
FCFC 32 FFFE     LD      (SECREG).A   ;FDC-Clock auf 2MHz + CPU-Halt 'disablen'
-----
FCFF 0A          WRPROT: LD     A.(BC)      ;Abfrage, ob 'WRITE PROTECT'
FD00 6F          LD      L.A
FD01 E6 40       AND     40H
FD03 C4 FCB6     CALL   NZ.WPRTXT      ;Text: 'PROTECTED TYPE C' ausgeben
FD06 7D          LD      A.L
FD07 B7          OR      A
FD08 C9          RET
-----
FD09 DD 21 0044  PCONDU: LD     IX.0044H      ;Video-Treiber
FD0D DD CB 00 46 BIT     0.(IX)      ;war letztes Zeichen '01H' ?
FD11 20 10       JR      NZ.L00030      ;wenn ja
FD13 2A 004E     LD      HL.(004EH)      ;'echo deleted characters'
FD16 CB 45       BIT     0.L
FD18 20 09       JR      NZ.L00030      ;geloeschtes Zeichen wiederholen
FD1A 7C          LD      A.H              ;(004FH) ins A-Reg
FD1B FE 7F       CP      7FH      ;Delete
FD1D 20 04       JR      NZ.L00030      ;nein
FD1F B9          CP      C          ;ja
FD20 C8          RET     Z          ;bei 'Delete'
FD21 0E 08       LD      C.08H      ;Backsp ;sonst
FD23 2A 0040     L00030: LD     HL.(0040H)   ;Cursorstand ;altes Zeichen auf aktuellen Cursorstand bringen
FD26 7C          LD      A.H
FD27 E6 3F       AND     3FH      ;Cursor in Bildschirmbereich zurueckfuehren
FD29 F6 3C       OR      3CH
FD2B 57          LD      H.A
FD2C 3A 0042     LD      A.(0042H)      ;altes Zeichen
FD2F D3 00       OUT   (BANKE).A
FD31 77          LD      (HL).A      ;auf aktuellen Cursorstand
FD32 D3 01       OUT   (BANKA).A
FD34 79          LD      A.C          ;neues Zeichen
FD35 DD CB 00 46 BIT     0.(IX)      ;war letztes Zeichen '01H'
FD39 20 49       JR      NZ.GRAPH2      ;wenn ja
FD3B EB          EX      DE.HL
FD3C CD FE4E     CALL   ESCAPE        ;Esc = Shift-(Break) ?
FD3F FE 20       CP      20H
FD41 38 68       JR      C.STEUER      ;falls Steuerzeichen (<20H)
FD43 21 0045     LD      HL.0045H
-----
FD46 FE 5B       CP      5BH          ;'5B' in '5E' umwandeln
FD48 20 04       JR      NZ.L00031

```

```

FD4A 3E 5E          LD  A.5EH
FD4C 18 06          JR  L00032

FD4E FE 5E          L00031: CP  5EH          ;'5E' in '5B' umwandeln
FD50 20 02          JR  NZ.L00032
FD52 3E 5B          LD  A.5BH

FD54 CB 5E          L00032: BIT  3,(HL)      ;Graphikzeichen auf Bildschirm ?
FD56 28 02          JR  Z,L00033          ;Graphik zugelassen
FD58 E6 7F          AND  7FH             ;Graphik disabled : in ASCII umwandeln
FD5A FE C0          L00033: CP  0C0H
FD5C 30 35          JR  NC.FX3X          ;Graphik-Zeichen von C0H bis FFH
FD5E CB 46          L00034: BIT  0,(HL)
FD60 20 0A          JR  NZ.L00035          ;Kleinschrift auf Bildschirm zugelassen
FD62 FE 80          CP  80H
FD64 30 06          JR  NC.L00035          ;Graphik
FD66 FE 60          CP  60H
FD68 38 02          JR  C.L00035          ;Grosschrift bzw. Zahlen
FD6A E6 5F          AND  5FH             ;Kleinschrift in Grosschrift umwandeln
FD6C EB            L00035: EX  DE,HL      ;HL: Cursorstand. DE: 0045H
FD6D CD FDF5        CALL AUS6             ;Ausgabe + Scroll, wenn noetig
;-----
FD70 7C            RUECK: LD  A.H          ;Cursor setzen + Cursorstand abspeichern
FD71 E6 3F          AND  3FH
FD73 F6 3C          OR   3CH
FD75 67            LD  H.A              ;in Bildschirmbereich zurueckfuehren
FD76 D3 00          OUT  (BANKE).A
FD78 7E            LD  A,(HL)
FD79 36 5F          LD  (HL),5FH         ;Cursor setzen
FD7B D3 01          OUT  (BANKA),A
FD7D 32 0042        LD  (42H),A          ;von Cursor verdecktes Zeichen
FD80 22 0040        LD  (0040H),HL      ;Cursorstand
FD83 C9            RET
;-----
FD84 DD CB 00 86    GRAPH2: RES  0,(IX)      ;nach '01H' werden alle Zeichen ausgegeb
FD8B DD 21 FD70     LD  IX,RUECK
FD8C DD E5          PUSH IX             ;nach RET Rueckkehr nach RUECK
FD8E E6 1F          AND  1FH             ;in Controlzeichen umwandeln
FD90 C3 FDF5        JP   AUS6             ;Zeichen auf jeden Fall in Bildschirmsp. schreiben
;-----
FD93 E6 3F          FX3X: AND  3FH          ;C0H...FFH in 00H...3FH umwandeln
FD95 18 C7          JR  L00034
;-----
FD97 21 3C00        L00036: LD  HL,03C00H   ;CLS + Umsch. auf 64 Z/Zeile ;Sonderzeichendecodierung
FD9A 18 75          JR  CLEAR
FD9C 19            L00037: ADD  HL,DE      ;Zeilenvorschub
FD9D C9            RET
FD9E 2B            L00038: DEC  HL        ;Cursor zurueck
FD9F C9            RET
FDA0 23            L00039: INC  HL        ;Cursor vor
FDA1 C9            RET
FDA2 21 3C00        HOME: LD  HL,3C00H
FDA5 C9            RET
FDA6 2B            L00040: DEC  HL        ;loescht letztes Zeichen
FDA7 D3 00          OUT  (BANKE).A
FDA9 36 20          LD  (HL),20H
FDAB D3 01          OUT  (BANKA),A
FDAD C9            RET
;-----
FDAE EB            STEUER: EX  DE,HL      ;Steuerzeichen
FAF 11 FD70        LD  DE,RUECK        ;nach RET Ruecksprung zu RUECK
FDB2 D5            PUSH DE

```

```

FDB3 B7 OR A
FDB4 C8 RET Z
FDB5 FE 1A CP 1AH ;Down-Arrow.CTL-Z
FDB7 11 0040 LD DE.0040H
FDBA 28 E0 JR Z.L00037
FDBC FE 17 CP 17H ;Up-Arrow
FDBE 11 FFC0 LD DE.0FFC0H
FDC1 28 D9 JR Z.L00037
FDC3 FE 08 CP 08H ;loescht letztes Zeichen
FDC5 28 DF JR Z.L00040
FDC7 FE 0A CP 0AH ;LF
FDC9 28 72 JR Z.LF
FDCB FE 0D CP 0DH ;CR
FDCD 28 69 JR Z,CR
FDCF FE 18 CP 18H ;Left Arrow
FDD1 28 CB JR Z.L00038
FDD3 FE 19 CP 19H ;Right Arrow
FDD5 28 C9 JR Z.L00039
FDD7 FE 1F CP 1FH ;Clear
FDD9 28 36 JR Z.CLEAR
Fddb FE 1E CP 1EH ;loescht Rest der Zeile
FDDd 28 47 JR Z,CLR LIN
FDDf FE 1C CP 1CH ;Clear Bildschirm + Cassette ausschalten
FDE1 28 B4 JR Z.L00036
FDE3 FE 1D CP 1DH ;Home, zurueck zum Bildschirm anfang
FDE5 28 BB JR Z.HOME
FDE7 FE 01 CP 01H ;Strichgraphik-Steuerzeichen
FDE9 28 01 JR Z.GRAPH1
FDEB C9 RET

;-----
FDEC DD 21 0044 GRAPH1: LD IX,0044H
FDF0 DD CB 00 C6 SET 0,(IX)
FDF4 C9 RET

;-----
FDF5 D3 00 AUSS: OUT (BANKE),A ;Zeichen auf Bildschirm
FDF7 77 LD (HL),A
FDF8 D3 01 OUT (BANKA),A
FDFA 23 INC HL
FDFB 7C LD A,H
FDFC FE 40 CP 40H ;Bildschirmende
FDFE C0 RET NZ ;wenn o.k., sonst Scroll

;-----
FDFf 11 3C00 SCROLL: LD DE.3C00H ;Scroll Bildschirm
FE02 21 3C40 LD HL,3C40H
FE05 01 03C0 LD BC,03C0H
FE08 D3 00 OUT (BANKE),A
FE0A ED B0 LDIR ;gesamten Bildschirm um eine Zeile nach oben verschieben
FE0C D3 01 OUT (BANKA),A
FE0E 21 3FC0 LD HL,3FC0H ;Beginn der letzten Zeile

;-----
heFE11 E5 CLEAR: PUSH HL ;von Cursor (HL) an Bildschirm lo
FE12 D3 00 OUT (BANKE),A
FE14 11 4000 LD DE,4000H ;Bildschirmende
FE17 36 20 L00046: LD (HL),20H ;Zeichen loeschen
FE19 23 INC HL
FE1A 7C LD A,H
FE1B BA CP D
FE1C 20 F9 JR NZ.L00046 ;noch nicht am Bildschirmende
FE1E 7D LD A,L
FE1F BB CP E
FE20 20 F5 JR NZ.L00046 ;noch nicht am Bildschirmende
FE22 D3 01 OUT (BANKA),A

```

```

FE24 E1          POP HL
FE25 C9          RET          ;wenn alles geloescht
;-----
FE26 E5          CLRLIN: PUSH HL          ;Rest der Zeile loeschen
FE27 7D          LD A,L
FE28 E6 3F       AND 3FH
FE2A 03 00       OUT (BANKE).A
FE2C 36 20       L00048: LD (HL),20H      ;Zeichen loeschen
FE2E 23          INC HL
FE2F 3C          INC A
FE30 FE 40       CP 40H
FE32 20 F8       JR NZ,L00048      ;noch nicht am Zeilenende
FE34 03 01       OUT (BANKA).A
FE36 E1          POP HL
FE37 C9          RET          ;wenn Zeile geloescht
;-----
FE38 7D          CR: LD A,L          ;Wagenruecklauf
FE39 E6 C0       AND 0C0H
FE3B 6F          LD L,A
FE3C C9          RET
;-----
FE3D E5          LF: PUSH HL          ;Zeilenvorschub
FE3E 11 0040     LD DE,0040H
FE41 19          ADD HL,DE      ;Zeilenvorschub
FE42 7C          LD A,H
FE43 FE 40       CP 40H          ;Bildschirmende
FE45 20 05       JR NZ,L00051     ;noch nicht Bildschirmende
FE47 CD FDFE     CALL SCROLL    ;wenn am Bildschirmende
FE4A E1          POP HL
FE4B C9          RET
;-----
FE4C D1          L00051: POP DE
FE4D C9          RET
;-----
FE4E 21 0043     ESCAPE: LD HL,0043H      ;Shift-(Break) = 'Esc'
FE51 CB 46       BIT 0,(HL)
FE53 20 09       JR NZ,L00054      ;wenn letztes Zeichen ein Shift-(Break) war
FE55 FE 1B       CP 1BH          ;Shift-(Break)
FE57 C0          RET NZ          ;wenn jetzt nicht Shift-(Break) gedrueckt
FE58 CB C6       SET 0,(HL)      ;wenn Shift-(Break) gedrueckt
FE5A C9          RET
;-----
FE5B 78          CHOME: LD A,B          ;Escape-String decodieren
FE5C 18 30       JR L0005B      ;Esc-String abgearbeitet
;-----
FE5E CB 4E       L00054: BIT 1,(HL)
FE60 20 14       JR NZ,L00055      ;wenn die letzten Zeichen 'Esc' - '=' waren
FE62 06 1C       LD B,1CH          ;Bildschirm loeschen
FE64 FE 2A       CP 2AH          ;'Esc' - '=' - String : Cursor home + Bildschirm loeschen
FE66 28 F3       JR Z,CHOME
FE68 06 1E       LD B,1EH          ;Rest der Zeile loeschen
FE6A FE 54       CP 54H          ;T
FE6C 28 ED       JR Z,CHOME      ;Rest der Zeile loeschen
FE6E FE 3D       CP 3DH          ;'Esc'-'='-x-x - String positioniert Cursor
FE70 20 1C       JR NZ,L0005B     ;falsches Zeichen nach 'Esc'
FE72 CB CE       SET 1,(HL)      ;letzte Zch: 'Esc'-'='
FE74 AF          XOR A          ;kein Zeichen auf Bildschirm ausgeben
FE75 C9          RET
FE76 D6 1F       L00055: SUB 1FH      ;eingeg. Code war Cursorposition + 20H
FE78 CB 56       BIT 2,(HL)      ;wenn letzte Zch 'Esc'-'='-'Line Nr' waren
FE7A 20 0E       JR NZ,L00057
FE7C CB D6       SET 2,(HL)      ;aktuelles Zeichen bezeichnet Line Nr

```



```

FE7E 11 0040          LD    DE.0040H          ;Laenge einer Bildschirmzeile
FE81 21 3BC0          LD    HL.3BC0H          ;Bildschirmanfang -40H
FE84 19              L00056: ADD   HL.DE          ;Cursor Zeilenvorschuebe
FE85 3D              DEC    A
FE86 20 FC          JR    NZ.L00056          ;solange A > 0 : Cursor steht zu Beginn gewuenschter Zeile
FE88 EB              EX    DE.HL
FE89 C9              RET
FE8A 3D              L00057: DEC    A          ;Cursor an Beginn der adressierten Zeile
FE8B 83              ADD    A.E              ;Cursor innerhalb Zeile positionieren
FE8C 5F              LD    E,A              ;in DE : Cursorposition
FE8D AF              XOR    A
FE8E 36 00          L00058: LD    (HL).00H          ;Shift-(Break)-String abgearbeitet
FE90 C9              RET

;-----
FE91 21 004E          PCONST: LD    HL,004EH          ;Abfrage Tastaturstatus
FE94 D3 00              OUT   (BANKE).A
FE96 3A 387F          LD    A,(387FH)          ;Abfrage, ob Taste gedrueckt
FE99 D3 01              OUT   (BANKA).A
FE9B B7              OR    A
FE9C 20 0A          JR    NZ,L00060          ;wenn ja
FE9E CB 56              BIT   2,(HL)
FEA0 C8              RET    Z
FEA1 CB 96              RES   2,(HL)
FEA3 CD FF69          CALL  L0007B          ;Warteroutine
FEA6 18 02              JR    L00061
FEA8 CB D6              L00060: SET   2,(HL)
FEAA CD FED7          L00061: CALL  INKEY          ;Tastaturdecodierung
FEAD B7              OR    A
FEAE 32 0046          LD    (0046H),A          ;eingegebenes Zeichen
FEB1 C8              RET    Z              ;kein Zeichen eingegeben: A='00'
FEB2 3E FF          LD    A,0FFH
FEB4 C9              RET              ;wenn Zeichen eingegeben: A='FF'

;-----
FEB5 21 0046          PCONIN: LD    HL,0046H          ;von Tastatur eingeg. Zch.      ;Eingabe von Tastatur
FEB8 7E              LD    A,(HL)
FEB9 B7              OR    A
FEBA 20 05          JR    NZ,L00063          ;wenn Zch eingegeben
FEBC CD FE91          CALL  PCONST          ;Tastatur neu abfragen
FEBF 18 F4              JR    PCONIN          ;warten bis Eingabe

FEC1 36 00          L00063: LD    (HL).00H          ;Zeichen ist in Bearbeitung
FEC3 F2 FED3          JP    P,L00064          ;wenn nicht CTRL 0
FEC6 21 0045          LD    HL,0045H          ;wenn CTRL 0
FEC9 CB 46              BIT   0,(HL)
FECB 28 EB          JR    Z,PCONIN          ;Kleinschrift nicht in Grossschrift umwandeln: neues Zeichen
FECD 7E              LD    A,(HL)          ;Kleinschrift in Grossschrift umwandeln
FECE EE 02          XOR    02H          ;Bit '1' von (0045H) wird getoggelt: '0' nur Grossschreibung
FED0 77              LD    (HL).A
FED1 18 E2          JR    PCONIN          ;neues Zeichen

FED3 32 004F          L00064: LD    (004FH),A          ;von Tastatur gelesenes Zeichen
FED6 C9              RET

;-----
FED7 21 0047          INKEY: LD    HL,0047H          ;Lesen e. Zeile von Tastatur      ;Tastatur-Treiber
FEDA 01 3801          LD    BC,3801H          ;Tastaturadresse nach BC
FEDD 16 00          LD    D,00H          ;Zeilenzaehler
FEDF D3 00          L00066: OUT   (BANKE).A
FEE1 0A              LD    A,(BC)          ;eine Tastaturzeile laden
FEE2 D3 01          OUT   (BANKA).A
FEE4 5F              LD    E,A          ;nach E
FEE5 AE              XOR    (HL)          ;noch selbe Taste
FEE6 73              LD    (HL),E

```

```

FEE7 A3 AND E
FEE8 20 08 JR NZ,L00067 ;wenn in aktueller Zeile Bit gesetzt (Taste gedruickt)
FEEA 14 INC D ;Zeilenzaehler + 1
FEEB 2C INC L ;Statuszeiger + 1
FEEC CB 01 RLC C ;naechste Zeilenadresse
FEEE F2 FEDF JP P.L00066 ;bis alle Zeilen abgearbeitet
FEF1 C9 RET ;wenn keine oder immer noch die selbe Taste gedruickt

;-----
FEF2 5F L00067: LD E,A ;Zeileninfo nach E
FEF3 C5 PUSH BC
FEF4 CD FF69 CALL L00078 ;Warteroutine (Entprellung)
FEF7 C1 POP BC
FEF8 D3 00 OUT (BANKE),A
FEFA 0A LD A,(BC) ;Tastaturzeile nochmals laden
FEFB D3 01 OUT (BANKA),A
FEFD A3 AND E ;gleiche Taste noch immer gedruickt
FEFE CB RET Z ;nein

FEFF 7A LD A,D ;Zeilennummer & 8 (ergibt Bit 3...5 des ASCII-Codes)
FF00 07 RLCA
FF01 07 RLCA
FF02 07 RLCA
FF03 57 LD D,A

FF04 0E 01 LD C,01H ;Bit 0 in C setzen ;Stelle des ges. Bits in C ermitteln
FF06 79 L00068: LD A,C
FF07 A3 AND E ;Bit in C = Bit in Zeileninfo ?
FF08 20 05 JR NZ,L00069 ;ja
FF0A 14 INC D
FF0B CB 01 RLC C ;naechstes Bit in C
FF0D 18 F7 JR L00068 ;weiter

FF0F D3 00 L00069: OUT (BANKE),A
FF11 3A 3880 LD A,(3880H) ;Shift Taste abfragen: wenn gedruickt, A0 = 0
FF14 D3 01 OUT (BANKA),A
FF16 47 LD B,A ;nach B
FF17 7A LD A,D
FF18 C6 60 ADD A,60H ;zum vorlaeufigen Code 60H add. (60H...8FH)
FF1A FE 8A CP 8AH ;Buchstaben bzw Ziffern ?
FF1C 30 28 JR NC,L00072 ;nein (8A...8F)
FF1E 21 3840 LD HL,3840H ;CTRL Taste
FF21 FE 80 CP 80H ;Buchstaben (60H...7FH) ?
FF23 D3 00 OUT (BANKE),A
FF25 38 0A JR C,L00070 ;ja
FF27 CB 66 BIT 4,(HL) ;Control-Taste gedruickt ?
FF29 D3 01 OUT (BANKA),A
FF2B 28 19 JR Z,L00072 ;nein
FF2D D6 70 SUB 70H ;fuer CTRL-(1...9) 10H...19H generieren
FF2F 18 2F JR L00075 ;Code fuer CTRL-(Ziffern) ueber Tabelle

FF31 CB 66 L00070: BIT 4,(HL) ;Control-Taste gedruickt ?
FF33 D3 01 OUT (BANKA),A
FF35 20 30 JR NZ,L00077 ;ja
FF37 21 0045 LD HL,0045H
FF3A CB 4E BIT 1,(HL) ;nur Grossschritt ?
FF3C 28 04 JR Z,L00071 ;ja
FF3E CB 08 RRC B ;Shift-Taste gedruickt ? ;Buchstaben
FF40 30 24 JR NC,L00076 ;nein : Ende
FF42 D6 20 L00071: SUB 20H ;Kleinschrift in Grosschrift umwandeln
FF44 18 20 JR L00076 ;Ende

FF46 D6 90 L00072: SUB 90H

```

```

FF48 30 10      JR      NC.L00074      :wenn Controlzeichen (pos)
FF4A C6 40      ADD      A.40H          :wenn Sonderzeichen (3AH...3FH)
FF4C FE 3C      CP        3CH          :<
FF4E 38 02      JR      C,L00073      :Zahlen bzw Sonderzeichen
FF50 EE 10      XOR      10H
FF52 CB 08      L00073: RRC      B              ;Shift gedruickt ?      ;Zahlen bzw Sonderzeichen
FF54 30 10      JR      NC.L00076      :nein : Ende
FF56 EE 10      XOR      10H          :ja : Sonderzeichen umwandeln
FF58 18 0C      JR      L00076          ;Ende

FF5A 07         L00074: RLCA          :Tastaturcode * 2      ;Controllasten
FF5B CB 08      RRC      B              ;Shift gedruickt
FF5D 30 01      JR      NC.L00075      :nein : endgueltigen Code erzeugen
FF5F 3C         INC      A              ;ja
FF60 21 FF80    L00075: LD      HL,TAB4
FF63 85         ADD      A,L
FF64 6F         LD      L,A          :richtige Stelle in Tabelle
FF65 7E         LD      A,(HL)       :ASCII - Code
FF66 57         L00076: LD      D,A
FF67 7A         L00077: LD      A,D          :ASCII - Code nach A
FF68 C9         RET

;-----
FF69 06 06      L00078: LD      B.06H          :Warterroutine (Entprellung)
FF6B 08         LOOP3: DEC     BC
FF6C 78         LD      A,B
FF6D B1         OR      C
FF6E 20 FB      JR      NZ,LOOP3
FF70 C9         RET

;-----
FF71 AF         DSPLMS: XOR     A              :Textausgabe auf Bildschirm
FF72 32 0043    LD      (0043H),A      ;Escape-Status zuruecksetzen
FF75 4E         LD      C,(HL)
FF76 B1         OR      C
FF77 C8         RET      Z          :wenn Zeichen '00'
FF78 E5         PUSH     HL
FF79 CD F009    CALL    PCONQU        ;Ausgabe auf Bildschirm
FF7C E1         POP      HL
FF7D 23         INC      HL
FF7E 18 F1      JR      DSPLMS        :naechstes Zeichen

;-----
FF80 00         TAB4:  DEFB     0DH          :CR                      :Code Tabelle fuer Tastatortreibe
FF81 0A         DEFB     0AH          ;Shift-(ENTER) = LF
FF82 13         DEFB     13H         :CLEAR = CTRL S
FF83 5E         DEFB     5EH         ;Shift-(CLEAR = Up Arrow
FF84 08         DEFB     08H         :BREAK = Backspace
FF85 1B         DEFB     1BH         :Shift-(BREAK) = Escape
FF86 1A         DEFB     1AH         :Up Arrow = CTRL Z
FF87 5C         DEFB     5CH         :Shift-(Up Arrow) = Backslash
FF88 00         DEFB     00H
FF89 00         DEFB     00H
FF8A 7F         DEFB     7FH         :Left Arrow = Delete
FF8B 5B         DEFB     5BH         :Shift-(Left Arrow) = Left Bracket
FF8C 09         DEFB     09H         ;Right Arrow = TAB
FF8D 5D         DEFB     5DH         :Shift-(Right Arrow) = Right Bracket
FF8E 20         DEFB     20H
FF8F 20         DEFB     20H
FF90 80         DEFB     80H          :CTRL 0
FF91 5F         DEFB     5FH          :CTRL 1
FF92 1C         DEFB     1CH          :CTRL 2
FF93 1D         DEFB     1DH          :CTRL 3
FF94 1E         DEFB     1EH          :CTRL 4
FF95 1F         DEFB     1FH          :CTRL 5
    
```

```

FF96 7B          DEFB 7BH    ;CTRL 6
FF97 7C          DEFB 7CH    ;CTRL 7
FF98 7D          DEFB 7DH    ;CTRL 8
FF99 7E          DEFB 7EH    ;CTRL 9
;-----
FF9A 0D          TEXT1: DEFB 0DH          ;Text: 'PROTECTED'
FF9B 0A          DEFB 0AH
FF9C 50 52 4F 54 DEFM 'PROTECTED'
FFA0 45 43 54 45
FFA4 44
FFA5 0D          DEFB 0DH
FFA6 0A          DEFB 0AH
FFA7 00          DEFB 00H
;-----
FFA8 54 59 50 45 TEXT2: DEFM 'TYPE C'          ;Text: 'TYPE C'
FFAC 20 43
FFAE 0D          DEFB 0DH
FFAF 0A          DEFB 0AH
FFB0 00          DEFB 00H
;-----
FFB1 01 FFFC    PREADY: LD BC,STATR          ;dreht Diskette sich ?
FFB4 08          EX AF,AF'
FFB5 0A          L00086: LD A,(BC)          ;Status-Register
FFB6 0F          RRCA
FFB7 3B FC      JR C,L00086          ;warten auf 'not busy'
FFB9 3E D8      LD A,0DBH
FFBB CD FC6F    CALL CMDFDC          ;Befehl:'Force Interrupt every 10 ms'
FFBE 0A          LD A,(BC)
FFBF E6 02      AND 02H          ;Index-Marke
FFC1 57          LD D,A
FFC2 26 4E      LD H,4EH          ;Vorwahl: Zahl der Versuche
FFC4 2B          L00087: DEC HL
FFC5 7C          LD A,H
FFC6 B5          OR L          ;letzter Versuch
FFC7 28 08      JR Z,L00088          ;Fehlermeldung wenn immer unverändert
FFC9 0A          LD A,(BC)
FFCA -- E6 02   AND 02H          ;Index-Marke
FFCC BA          CP D          ;hat sich Index-Bit geändert?
FFCD 57          LD D,A
FFCE 28 F4      JR Z,L00087          ;wenn nein
FFD0 C9          RET          ;wenn ja (-> Diskette dreht sich)
FFD1 21 FFDE    L00088: LD HL,TEXT3
FFD4 CD FC8A    CALL L00018          ;Text:'NOT READY TYPE C' ausgeben + Eingabe abwarten
FFD7 08          EX AF,AF'
FFD8 5F          LD E,A
FFD9 CD FC76    CALL DRVSEL          ;Drive Select
FFDC 1B D3      JR PREADY          ;neuer Versuch
;-----
FFDE 0D          TEXT3: DEFB 0DH          ;Text: 'NOT READY'
FFDF 0A          DEFB 0AH
FFE0 4E 4F 54 20 DEFM 'NOT READY'
FFE4 52 45 41 44
FFE8 59
FFE9 0D          DEFB 0DH
FFEA 0A          DEFB 0AH
FFEB 00          DEFB 0
END

```

Macros:

Symbols:

AUSG	FD5	BANKA	0001	BANKE	0000	CHOME	FE5B
CLEAR	FE11	CLRLIN	FE26	CMDFDC	FC6F	CMDREG	FFFC
CR	FE38	DATREG	FFFF	DRVSEL	FC76	DSPLMS	FF71
EPRFCX	0003	ESCAPE	FE4E	FX3X	FD93	GRAPH1	FDEC
GRAPH2	FD84	HOME	FDA2	INKEY	FED7	KALTST	FC2E
L00001	FC00	L00002	FC19	L00003	FC1B	L00012	FC7E
L00013	FC38	L00014	FC3D	L00015	FC57	L00016	FC5C
L00018	FC8A	L00019	FC93	L00021	FCAA	L00022	FCB0
L00023	FCB3	L00027	FCF2	L00030	FD23	L00031	FD4E
L00032	FD54	L00033	FD5A	L00034	FD5E	L00035	FD6C
L00036	FD97	L00037	FD9C	L00038	FD9E	L00039	FDA0
L00040	FDA6	L00046	FE17	L00048	FE2C	L00051	FE4C
L00054	FE5E	L00055	FE76	L00056	FE84	L00057	FEBA
L00058	FE8E	L00060	FEA8	L00061	FEAA	L00063	FEC1
L00064	FED3	L00066	FEDF	L00067	FEF2	L00068	FF06
L00069	FF0F	L00070	FF31	L00071	FF42	L00072	FF46
L00073	FF52	L00074	FF5A	L00075	FF60	L00076	FF66
L00077	FF67	L00078	FF69	L00086	FFB5	L00087	FFC4
L00088	FFD1	LARGE	FCDA	LF	FE3D	LOOP1	FC68
LOOP2	FC72	LOOP3	FF6B	MINI	FCCE	NMI	FC66
PCDNIN	FE85	PCDNOU	FD09	PCONST	FE91	PREAD	FCA2
PREADY	FFB1	PWRITE	FCC5	RUECK	FD70	SCROLL	FDFF
SECREG	FFFE	SEL	FFF0	STATR	FFFC	STEUER	FDAE
TAB4	FF80	TEXT1	FF9A	TEXT2	FFA8	TEXT3	FFDE
TRKREG	FFFD	UML	FC29	WPRTXT	FC86	WRPROT	FCFF

No Fatal error(s)

0000'  
0000'

Aseq  
Title SETUP63  
SUBTTL Version 02.02.83  
.Z80

```

;*****
; **
; ** SETUP63.COM **
; **
;*****
    
```

```

;Programm fuer OMIKRON-CP/M 48k-Version
;jetzt jedoch auch lauffaehig in
;63k-CP/M - System
    
```

0005

```

BDOS EQU 0005H
.PHASE 100H
    
```

```

-----
0100 3A 00B2          LD    A,(00B2H)                ;ABFRAGEN
0103 FE 44          CP    44H
0105 CA 0165        JP    Z,L00001
0108 21 058C        LD    HL,TEXT14
010B CD 065E        CALL DSPLMS
010E 21 0497        LD    HL,TEXT9
0111 CD 065E        CALL DSPLMS
0114 21 04D9        LD    HL,TEXT10
0117 CD 065E        CALL DSPLMS
011A 21 0370        LD    HL,TEXT3
011D CD 065E        CALL DSPLMS
0120 CD 028D        CALL YESNO
0123 32 02A1        LD    (02A1H),A
0126 21 03D4        LD    HL,TEXT5
0129 CD 065E        CALL DSPLMS
012C CD 028D        CALL YESNO
012F 32 02A2        LD    (02A2H),A
0132 21 0437        LD    HL,TEXT7
0135 CD 065E        CALL DSPLMS
0138 CD 028D        CALL YESNO
013B 32 02A3        LD    (02A3H),A
013E 21 033E        LD    HL,TEXT2
0141 CD 065E        CALL DSPLMS
0144 CD 028D        CALL YESNO
0147 32 02A4        LD    (02A4H),A
014A 21 051E        LD    HL,TEXT12
014D CD 065E        CALL DSPLMS
0150 CD 028D        CALL YESNO
0153 32 02A5        LD    (02A5H),A
0156 CD 0183        CALL L00002
0159 21 058C        LD    HL,TEXT15
015C CD 065E        CALL DSPLMS
015F CD 028D        CALL YESNO
0162 32 02A7        LD    (02A7H),A
-----
0165 CD 0232        L00001: CALL L00012                ;HAUPTPROGRAMM
0168 21 0516        LD    HL,TEXT11
    
```

```

016B CD 065E CALL DSPLMS
016E CD 01AB CALL L00004
0171 CD 01E1 CALL L00006
0174 CD 01F2 CALL L00007
0177 CD 0212 CALL L00009
017A CD 01C6 CALL L00005
017D CD 019B CALL L00003
0180 C3 0000 JP 0000H
    
```

```

;-----
0183 2A 0001 L00002: LD HL,(0001H) ;CURSOR BLINKEN
0186 2E 37 LD L,37H ;(HL) = PTRAM (1# BIOS)
0188 7E LD A,(HL)
0189 3C INC A
018A 3C INC A
018B FE 04 CP 04H
018D D0 RET NC
018E 21 062F LD HL,TEXT17
0191 CD 065E CALL DSPLMS
0194 CD 028D CALL YESNO
0197 32 02A6 LD (02A6H),A
019A C9 RET
    
```

```

;-----
019B 2A 0001 L00003: LD HL,(0001H) ;CURSOR NICHT BLINKEN
019E 2E 37 LD L,37H ;(HL) = PTRAM (1# BIOS)
01A0 7E LD A,(HL)
01A1 3C INC A
01A2 3C INC A
01A3 FE 04 CP 04H
01A5 D0 RET NC
01A6 3A 02A6 LD A,(02A6H)
01A9 77 LD (HL),A
01AA C9 RET
    
```

```

;-----
01AB 3A 02A4 L00004: LD A,(02A4H) ;EN. LOWER CASE
01AE E6 03 AND 03H
01B0 F5 PUSH AF
01B1 4F LD C,A
01B2 3A 0045 LD A,(0045H)
01B5 E6 FC AND 0FCH ;Bits 0,1 = '1' : Lower Case enabled
01B7 B1 OR C
01B8 32 0045 LD (0045H),A
01BB F1 POP AF
01BC CA 0231 JP Z,L00011
01BF 21 02A8 LD HL,TEXT1
01C2 CD 065E CALL DSPLMS
01C5 C9 RET
    
```

```

;-----
01C6 3A 02A5 L00005: LD A,(02A5H) ;DISABLE GRAPHICS
01C9 E6 08 AND 08H
01CB F5 PUSH AF
01CC 4F LD C,A
01CD 3A 0045 LD A,(0045H)
01D0 E6 F7 AND 0F7H ;Bit 3 = '1' : Graphics disabled
01D2 B1 OR C
01D3 32 0045 LD (0045H),A
01D6 F1 POP AF
01D7 CA 0231 JP Z,L00011
01DA 21 055D LD HL,TEXT13
01DD CD 065E CALL DSPLMS
01E0 C9 RET
    
```

```

;-----
01E1 3A 02A1 L00006: LD A,(02A1H) ;ECHO DELETED CHARACTERS
    
```

```

01E4 32 004E      LD      (004EH).A      ;'echo deleted characters on screen'
01E7  B7          OR      A
01EB  CA 0231      JP      Z,L00011
01EB  21 03A6      LD      HL,TEXT4
01EE  CD 065E      CALL   DSPLMS        ;Text: 'Deleted characters will echo on the screen'
01F1  C9          RET

```

```

-----
01F2 0E 0C          L00007: LD      C,0CH          ;PRINTER LF
01F4  CD 0005      CALL   BDOS          ;Version Number (L = '00' fuer Version < 2.0)
01F7  7D          LD      A,L
01FB  B7          OR      A
01F9  2A 0001      LD      HL,(0001H)
01FC  2E 35          LD      L,35H        ;(HL) = TEST (im BIOS)
01FE  C2 0203      JP      NZ,L00008
0201  2E 2F          LD      L,2FH        ;Version < 2.0
0203  3A 02A2      L00008: LD      A,(02A2H)
0206  77          LD      (HL),A
0207  B7          OR      A
0208  CA 0231      JP      Z,L00011
020B  21 0408      LD      HL,TEXT6
020E  CD 065E      CALL   DSPLMS
0211  C9          RET

```

```

-----
0212 0E 0C          L00009: LD      C,0CH          ;EN. LINE COUNTER
0214  CD 0005      CALL   BDOS          ;Version Number (L = '00' fuer Version < 2.0)
0217  7D          LD      A,L
0218  B7          OR      A
0219  2A 0001      LD      HL,(0001H)
021C  2E 36          LD      L,36H        ;(HL) = FFTEST (im BIOS)
021E  C2 0223      JP      NZ,L00010
0221  2E 2D          LD      L,2DH        ;Version < 2.0
0223  3A 02A3      L00010: LD      A,(02A3H)
0226  77          LD      (HL),A
0227  B7          OR      A
0228  CA 0231      JP      Z,L00011
022B  21 0474      LD      HL,TEXT8
022E  CD 065E      CALL   DSPLMS
0231  C9          L00011: RET

```

```

-----
0232 3A 02A7      L00012: LD      A,(02A7H)          ;MINISWITCH
0235  B7          OR      A
0236  C8          RET      Z
0237  21 05F2      LD      HL,TEXT16
023A  CD 065E      CALL   DSPLMS
023D  CD 028D      CALL   YESNO
0240  B7          OR      A
0241  C8          RET      Z
0242  AF          XOR      A
0243  21 0004      LD      HL,0004H
0246  0E 0C          LD      C,0CH
0248  CD 0005      CALL   BDOS          ;Version Number (L = '00' fuer Version < 2.0)
024B  7D          LD      A,L
024C  B7          OR      A          ;Version < 2.0
024D  2A 0001      LD      HL,(0001H)
0250  C2 026C      JP      NZ,L00015      ;Version >= 2.0
0253  2E 24          LD      L,24H        ;Version < 2.0
0255  7E          LD      A,(HL)
0256  FE 18          CP      18H
0258  C0          RET      NZ
0259  2E 32          LD      L,32H
025B  CD 0277      CALL   L00016      ;Aenderungen DRTAB (im BIOS)
025E  7E          LD      A,(HL)

```



```

025F 2E 26          LD      L.26H
0261 E6 80          L00013: AND    80H          ;Aenderungen SECTRZ (im BIOS)
0263 3E 1A          LD      A.1AH
0265 C2 026A        JP      NZ.L00014
0268 3E 12          LD      A.12H
026A 77             L00014: LD      (HL).A
026B C9             RET
026C 2E 3B          L00015: LD      L.3BH          ;(HL) = DRTAB (im BIOS)
026E CD 0277        CALL   L00016
0271 7E             LD      A.(HL)
0272 2E 40          LD      L.40H          ;(HL) = SECTRZ (im BIOS)
0274 C3 0261        JP      L00013
0277 E5             L00016: PUSH   HL          ;Aenderungen DRTAB im BIOS (Miniswitch umschalten)
0278 5E             LD      E,(HL)
0279 23             INC     HL
027A 56             LD      D.(HL)
027B 23             INC     HL
027C 4E             LD      C.(HL)
027D 23             INC     HL
027E 46             LD      B.(HL)
027F E1             POP     HL
0280 E5             PUSH   HL
0281 71             LD      (HL).C
0282 23             INC     HL
0283 70             LD      (HL).B
0284 23             INC     HL
0285 73             LD      (HL).E
0286 23             INC     HL
0287 72             LD      (HL).D
0288 23             INC     HL
0289 36 FF          LD      (HL),OFFH
028B E1             POP     HL
028C C9             RET

```

```

-----
028D CD 0671        YESNO: CALL   PCONIN          ;TASTATURABFRAGE ('FF' fuer Yes,'00' fuer
0290 E6 5F          AND      5FH          ;Kleinschrift in Grosschrift
0292 4F             LD      C,A
0293 FE 59          CP      59H          ;Y
0295 3E FF          LD      A.OFFH        ;A='FF' fuer YES
0297 C8             RET      Z
0298 79             LD      A.C
0299 FE 4E          CP      4EH          ;N
029B 3E 00          LD      A,00H        ;A='00' fuer NO
029D C8             RET      Z
029E C3 028D        JP      YESNO          ;falls falsche Eingabe

```

```

-----
02A1             DEFS    7          ;TAB. + TEXTE
02A8 54 48 45 20    TEXT1: DEFM  'THE KEYBOARD WILL NOW SEND LOWER CASE'
02AC 48 45 59 42
02B0 4F 41 52 44
02B4 20 57 49 4C
02B8 4C 20 4E 4F
02BC 57 20 53 45
02C0 4E 44 20 4C
02C4 4F 57 45 52
02C8 20 43 41 53
02CC 45
02CD 0D             DEFB    0DH
02CE 0A             DEFB    0AH
02CF 20 55 4E 4C    DEFM  ' UNLESS THE SHIFT KEY IS PRESSED'
02D3 45 53 53 20
02D7 54 48 45 20

```

02DB 53 48 49 46  
02DF 54 20 48 45  
02E3 59 20 49 53  
02E7 20 50 52 45  
02EB 53 53 45 44  
02EF 0D  
02F0 0A  
02F1 0A  
02F2 54 48 45 20  
02F6 56 49 44 45  
02FA 4F 20 44 52  
02FE 49 56 45 52  
0302 20 57 49 4C  
0306 4C 20 4E 4F  
030A 57 20 44 49  
030E 53 50 4C 41  
0312 59 20  
0314 0D  
0315 0A  
0316 4C 4F 57 45  
031A 52 20 43 41  
031E 53 45 20 49  
0322 46 20 54 48  
0326 45 20 22 4D  
032A 4F 44 22 20  
032E 49 53 20 49  
0332 4E 53 54 41  
0336 4C 4C 45 44  
033A 0D  
033B 0A  
033C 0A  
033D 00  
033E 2A 2A 2A 2A  
0342 2A 20 44 4F  
0346 20 59 4F 55  
034A 20 57 41 4E  
034E 54 20 54 48  
0352 45 20 4C 4F  
0356 57 45 52 20  
035A 43 41 53 45  
035E 20 45 4E 41  
0362 42 4C 45 44  
0366 20 59 2F 4E  
036A 20 3F  
036C 0D  
036D 0A  
036E 0A  
036F 00  
0370 2A 2A 2A 2A  
0374 2A 20 44 4F  
0378 20 59 4F 55  
037C 20 57 41 4E  
0380 54 20 44 45  
0384 4C 45 54 45  
0388 44 20 43 48  
038C 41 52 41 43  
0390 54 45 52 53  
0394 20 54 4F 20  
0398 45 43 48 4F  
039C 20 59 2F 4E  
03A0 20 3F  
03A2 0D

DEFB 0DH  
DEFB 0AH  
DEFB 0AH  
DEFM 'THE VIDEO DRIVER WILL NOW DISPLAY '

DEFB 0DH  
DEFB 0AH  
DEFM 'LOWER CASE IF THE "MOD" IS INSTALLED'

DEFB 0DH  
DEFB 0AH  
DEFB 0AH  
DEFB 0H  
TEXT2: DEFM '\*\*\*\*\* DO YOU WANT THE LOWER CASE ENABLED Y/N ?'

DEFB 0DH  
DEFB 0AH  
DEFB 0AH  
DEFB 0  
TEXT3: DEFM '\*\*\*\*\* DO YOU WANT DELETED CHARACTERS TO ECHO Y/N ?'

DEFB 0DH

03A3 0A  
03A4 0A  
03A5 00  
03A6 44 45 4C 45  
03AA 54 45 44 20  
03AE 43 48 41 52  
03B2 41 43 54 45  
03B6 52 53 20 57  
03BA 49 4C 4C 20  
03BE 45 43 48 4F  
03C2 20 4F 4E 20  
03C6 54 48 45 20  
03CA 53 43 52 45  
03CE 45 4E  
03D0 0D  
03D1 0A  
03D2 0A  
03D3 00  
03D4 2A 2A 2A 2A  
03D8 2A 20 44 4F  
03DC 45 53 20 59  
03E0 4F 55 52 20  
03E4 50 52 49 4E  
03EB 54 45 52 20  
03EC 52 45 51 55  
03F0 49 52 45 20  
03F4 4C 49 4E 45  
03F8 46 45 45 44  
03FC 53 20 20 59  
0400 2F 4E 20 3F  
0404 0D  
0405 0A  
0406 0A  
0407 00  
0408 54 48 45 20  
040C 50 52 49 4E  
0410 54 45 52 20  
0414 57 49 4C 4C  
0418 20 4E 4F 57  
041C 20 52 45 43  
0420 45 49 56 45  
0424 20 41 4C 4C  
0428 20 43 48 41  
042C 52 41 43 54  
0430 45 52 53  
0433 0D  
0434 0A  
0435 0A  
0436 00  
0437 2A 2A 2A 2A  
043B 2A 20 44 4F  
043F 20 59 4F 55  
0443 20 57 41 4E  
0447 54 20 54 48  
044B 45 20 50 52  
044F 49 4E 54 45  
0453 52 20 4C 49  
0457 4E 45 20 43  
045B 4F 55 4E 54  
045F 45 52 20 45  
0463 4E 41 42 4C  
0467 45 44 20 20

DEFB 0AH  
DEFB 0AH  
DEFB 0  
TEXT4: DEFB 'DELETED CHARACTERS WILL ECHO ON THE SCREEN'

DEFB 0DH  
DEFB 0AH  
DEFB 0AH  
DEFB 0  
TEXT5: DEFB '\*\*\*\*\* DOES YOUR PRINTER REQUIRE LINEFEEDS Y/N ?'

DEFB 0DH  
DEFB 0AH  
DEFB 0AH  
DEFB 0  
TEXT6: DEFB 'THE PRINTER WILL NOW RECEIVE ALL CHARACTERS'

DEFB 0DH  
DEFB 0AH  
DEFB 0AH  
DEFB 0  
TEXT7: DEFB '\*\*\*\*\* DO YOU WANT THE PRINTER LINE COUNTER ENABLED Y/N ?'

```
046B 59 2F 4E 20
046F 3F
0470 0D DEF B 0DH
0471 0A DEF B 0AH
0472 0A DEF B 0AH
0473 00 DEF B 0
0474 54 48 45 20 TEXT8: DEF M 'THE LINE COUNTER IS NOW ENABLED'
0478 4C 49 4E 45
047C 20 43 4F 55
0480 4E 54 45 52
0484 20 49 53 20
0488 4E 4F 57 20
048C 45 4E 41 42
0490 4C 45 44
0493 0D DEF B 0DH
0494 0A DEF B 0AH
0495 0A DEF B 0AH
0496 00 DEF B 0
0497 54 48 45 20 TEXT9: DEF M 'THE COMMAND "SETUP63 D" WILL ENABLE THE SAVED DEFAULT VALUES'
049B 43 4F 4D 4D
049F 41 4E 44 20
04A3 20 22 53 45
04A7 54 55 50 36
04AB 33 20 44 22
04AF 20 20 57 49
04B3 4C 4C 20 45
04B7 4E 41 42 4C
04BB 45 20 54 48
04BF 45 20 53 41
04C3 56 45 44 20
04C7 44 45 46 41
04CB 55 4C 54 20
04CF 56 41 4C 55
04D3 45 53
04D5 0D DEF B 0DH
04D6 0A DEF B 0AH
04D7 0A DEF B 0AH
04D8 00 DEF B 0
04D9 54 59 50 45 TEXT10: DEF M 'TYPE "SAVE 6 SETUP63.COM" TO STORE THE FOLLOWING CHOICES'
04DD 20 20 22 53
04E1 41 56 45 20
04E5 36 20 53 45
04E9 54 55 50 36
04ED 33 2E 43 4F
04F1 4D 22 20 54
04F5 4F 20 53 54
04F9 4F 52 45 20
04FD 54 48 45 20
0501 46 4F 4C 4C
0505 4F 57 49 4E
0509 47 20 43 48
050D 4F 49 43 45
0511 53
0512 0D DEF B 0DH
0513 0A DEF B 0AH
0514 0A DEF B 0AH
0515 00 DEF B 0
0516 0D TEXT11: DEF B 0DH
0517 0A DEF B 0AH
0518 0A DEF B 0AH
0519 0A DEF B 0AH
051A 0A DEF B 0AH
```

051B	0A	DEFB	0AH
051C	0A	DEFB	0AH
051D	00	DEFB	0
051E	2A 2A 2A 2A	TEXT12: DEFB	'***** DO YOU WANT THE VIDEO DRIVER GRAPHICS DISABLED Y/N ?'
0522	2A 20 44 4F		
0526	20 59 4F 55		
052A	20 57 41 4E		
052E	54 20 54 48		
0532	45 20 56 49		
0536	44 45 4F 20		
053A	44 52 49 56		
053E	45 52 20 47		
0542	52 41 50 48		
0546	49 43 53 20		
054A	44 49 53 41		
054E	42 4C 45 44		
0552	20 20 59 2F		
0556	4E 20 3F		
0559	0D	DEFB	0DH
055A	0A	DEFB	0AH
055B	0A	DEFB	0AH
055C	00	DEFB	0
055D	47 52 41 50	TEXT13: DEFB	'GRAPHIC CHARACTERS WILL HAVE BIT 7 SET TO 0'
0561	48 49 43 20		
0565	43 48 41 52		
0569	41 43 54 45		
056D	52 53 20 57		
0571	49 4C 4C 20		
0575	48 41 56 45		
0579	20 42 49 54		
057D	20 37 20 53		
0581	45 54 20 54		
0585	4F 20 30		
0588	0D	DEFB	0DH
0589	0A	DEFB	0AH
058A	0A	DEFB	0AH
058B	00	DEFB	0
058C	4F 4D 49 48	TEXT14: DEFB	'OMIKRON SYSTEMS VERSION 1.3 JULY 13, 1980'
0590	52 4F 4E 20		
0594	53 59 53 54		
0598	45 4D 53 20		
059C	20 20 56 45		
05A0	52 53 49 4F		
05A4	4E 20 31 2E		
05A8	33 20 20 4A		
05AC	55 4C 59 20		
05B0	31 33 2C 20		
05B4	31 39 38 30		
05B8	0D	DEFB	0DH
05B9	0A	DEFB	0AH
05BA	0A	DEFB	0AH
05BB	00	DEFB	0
05BC	2A 2A 2A 2A	TEXT15: DEFB	'***** DO YOU WANT TO SWAP DRIVE ASSIGNMENTS Y/N ?'
05C0	2A 20 44 4F		
05C4	20 59 4F 55		
05C8	20 57 41 4E		
05CC	54 20 54 4F		
05D0	20 53 57 41		
05D4	50 20 44 52		
05D8	49 56 45 20		
05DC	41 53 53 49		
05E0	47 4E 4D 45		

05E4 4E 54 53 20  
 05EB 20 59 2F 4E  
 05EC 20 3F  
 05EE 0D  
 05EF 0A  
 05F0 0A  
 05F1 00  
 05F2 2A 2A 2A 2A  
 05F6 2A 20 48 41  
 05FA 56 45 20 59  
 05FE 4F 55 20 20  
 0602 22 53 59 53  
 0606 47 45 4E 45  
 060A 44 22 20 20  
 060E 41 20 53 59  
 0612 53 54 45 4D  
 0616 20 4F 4E 20  
 061A 44 52 49 56  
 061E 45 20 20 22  
 0622 43 22 20 20  
 0626 59 2F 4E 20  
 062A 3F  
 062B 0D  
 062C 0A  
 062D 0A  
 062E 00  
 062F 2A 2A 2A 2A  
 0633 2A 20 44 4F  
 0637 20 59 4F 55  
 063B 20 57 41 4E  
 063F 54 20 54 48  
 0643 45 20 43 55  
 0647 52 53 4F 52  
 064B 20 54 4F 20  
 064F 42 4C 49 4E  
 0653 48 20 20 59  
 0657 2F 4E 3F  
 065A 0D  
 065B 0A  
 065C 0A  
 065D 00

DEFB 0DH  
 DEFB 0AH  
 DEFB 0AH  
 DEFB 0  
 TEXT16: DEFM '\*\*\*\*\* HAVE YOU "SYSGENED" A SYSTEM ON DRIVE "C" Y/N ?'

DEFB 0DH  
 DEFB 0AH  
 DEFB 0AH  
 DEFB 0  
 TEXT17: DEFM '\*\*\*\*\* DO YOU WANT THE CURSOR TO BLINK Y/N?'

065E 7E  
 065F 23  
 0660 B7  
 0661 C8  
 0662 CD 066B  
 0665 C3 065E  
 0668 0E 02  
 066A 5F  
 066B E5  
 066C CD 0005  
 066F E1  
 0670 C9  
 0671 0E 01  
 0673 CD 0005  
 0676 C9

-----  
 DSPLMS: LD A,(HL) ;auszugebendes Zeichen ;Ausgabe einer Zeichenkette  
 INC HL ;naechstes  
 OR A ;Kette zu Ende ?  
 RET Z ;wenn ja  
 CALL CHAROU ;Ausgabe eines Zeichens  
 JP DSPLMS ;naechstes Zeichen

CHAROU: LD C,02H ;Befehl : Zeichenausgabe auf Bildschirm  
 LD E,A ;auszugebendes Zeichen  
 PUSH HL  
 CALL BOOS ;Console Output  
 POP HL  
 RET

-----  
 PCONIN: LD C,01H ;Bef: Eingabe abwarten ;Eingabe  
 CALL BOOS ;Console Input  
 RET  
 END

Macros:

Symbols:

BDOS	0005	CHARDU	0668	DSPLMS	065E	L00001	0165
L00002	0183	L00003	019B	L00004	01AB	L00005	01C6
L00006	01E1	L00007	01F2	L00008	0203	L00009	0212
L00010	0223	L00011	0231	L00012	0232	L00013	0261
L00014	026A	L00015	026C	L00016	0277	PCDNIN	0671
TEXT1	02A8	TEXT10	04D9	TEXT11	0516	TEXT12	051E
TEXT13	055D	TEXT14	058C	TEXT15	058C	TEXT16	05F2
TEXT17	062F	TEXT2	033E	TEXT3	0370	TEXT4	03A6
TEXT5	03D4	TEXT6	0408	TEXT7	0437	TEXT8	0474
TEXT9	0497	YESNO	02BD				

No Fatal error(s)

0000'  
0000'

Aseg  
Title MFORM63  
SUBTTL Version 02.02.83  
.Z80

```

:*****
:##                                     ##
:##  MFORMAT fuer 63k-CP/M           ##
:##                                     ##
:*****
    
```

```

FC14  PREADY EQU 0FC14H
FFFO  SEL EQU 0FFFOH
FFFC  STATR EQU 0FFFCB
FFFC  CMDREG EQU 0FFFCB
FFFD  TRKREG EQU 0FFFDH
FFFE  SECREG EQU 0FFFEH
FFFF  DATREG EQU 0FFFFH
03AD  STACK EQU 03ADH
      .PHASE 100H
    
```

```

0100  C3 04FC
0103  3A 0082
0106  CD 010C
0109  C3 012A
    
```

```

-----
          JP CHANGE ;auf 48k - Version umschreiben ;Anfang
START: LD A,(00B2H) ;Drive Nr bei MFORM63 *
      CALL L00001
      JP L00003
    
```

```

010C  FE 41
010E  CA 0121
0111  FE 42
0113  CA 0121
0116  FE 43
0118  CA 0121
011B  FE 44
011D  CA 0121
0120  C9
    
```

```

-----
L00001: CP 41H ;A ;Eingabetest
      JP Z,L00002
      CP 42H ;B
      JP Z,L00002
      CP 43H ;C
      JP Z,L00002
      CP 44H ;D
      JP Z,L00002
      RET
    
```

```

0121  32 022F
0124  DE 40
0126  32 01EC
0129  C9
    
```

```

-----
L00002: LD (DRVASC),A ;A.B.C,D ;Drive-Nummer abspeichern
      SBC A,40H
      LD (DRVNR),A ;1,2,3,4
      RET
    
```

```

012A  21 01BE
012D  CD 018A
0130  3E 80
0132  32 FFFE
0135  CD 017C
0138  21 0273
013B  CD 018A
013E  CD 0153
0141  21 0216
0144  CD 018A
0147  21 0231
014A  CD 018A
014D  CD 0196
0150  C3 03AD
    
```

```

-----
L00003: LD HL,TEXT1 ;Hauptprogramm
      CALL KETAUS
      LD A,80H
      LD (SECREG),A ;FDC-Controller auf 1 MHz umschalten (=Minidisk)
      CALL L00008 ;Drive Nr testen
      LD HL,TEXT8
      CALL KETAUS
      CALL TRWAHL
      LD HL,TEXT4
      CALL KETAUS
      LD HL,TEXT5
      CALL KETAUS
      CALL L00010 ;Warten auf 'RETURN'
      JP L00029 ;formatieren
    
```

```

0153  0E 01
    
```

```

-----
TRWAHL: LD C,01H ;Befehl : Eingabe ;Eingabe von 0.5,7
    
```



```

0155 CD 0005 CALL 0005H ;Console Input
0158 FE 30 CP 30H ;0
015A CA 016A JP Z.L00005
015D FE 35 CP 35H ;5
015F CA 0170 JP Z.L00006
0162 FE 37 CP 37H ;7
0164 CA 0176 JP Z.L00007
0167 C3 0153 JP TRWAHL
;-----
016A 3E 27 L00005: LD A,27H ;40 Tracks
016C 32 018A LD (TRKMAX).A
016F C9 RET
;-----
0170 3E 22 L00006: LD A,22H ;35 Tracks
0172 32 018A LD (TRKMAX).A
0175 C9 RET
;-----
0176 3E 4C L00007: LD A,4CH ;77 Tracks
0178 32 018A LD (TRKMAX).A
017B C9 RET
;-----
017C 3A 01EC L00008: LD A,(DRVNR) ;Drive Nr testen
017F B7 OR A
0180 C0 RET NZ ;wenn fehlerfrei
0181 21 029C LD HL,TEXT9
0184 CD 018A CALL KETAUS ;Fehlermeldung
0187 C3 0000 JP 0000H ;Warnstart CP/M
;-----
018A 7E KETAUS: LD A,(HL) ;Ausgabe einer Zeichenkette
018B 23 INC HL
018C B7 OR A
018D CA 01A6 JP Z.CRLF ;Ende der Kette
0190 CD 0181 CALL ZCHAUS
0193 C3 018A JP KETAUS ;naechstes Zeichen
;-----
0196 0E 01 L00010: LD C,01H ;Befehl : Eingabe ;warten auf 'RETURN'
0198 CD 0005 CALL 0005H ;Console Input
019B FE 03 CP 03H ;CTRL C
019D CA 0000 JP Z.0000H ;Warnstart
01A0 FE 0D CP 0DH ;RETURN
01A2 C2 0196 JP NZ.L00010
01A5 C9 RET
;-----
01A6 3E 0D CRLF: LD A,0DH ;CR ;Ausgabe von CR und LF
01A8 CD 0181 CALL ZCHAUS
01AB 3E 0A LD A,0AH ;LF
01AD CD 0181 CALL ZCHAUS
01B0 C9 RET
;-----
01B1 0E 02 ZCHAUS: LD C,02H ;Befehl : Ausgabe ;Textausgabe
01B3 5F LD E,A ;auszugebendes Zeichen
01B4 E5 PUSH HL
01B5 CD 0005 CALL 0005H ;Console Output
01B8 E1 POP HL
01B9 C9 RET
;-----
01BA TRKMAX: DEFS 1 ;27H fuer 40, 22H f. 35, 4CH f. 77 Tracks
01BB TRKNR: DEFS 1 ;aktuelle Track Nr.
01BC DRVBIN: DEFS 1 ;01H fuer Drive A, 02H f. B, 04H f. C. 08H f. D
01BD DEFS 1
;-----
01BE 46 4F 52 4D TEXT1: DEFW 'FORMAT PROGRAM FOR THE OMIKRON CPM MINI DRIVE' ;Texte
    
```

01C2 41 54 20 50  
01C6 52 4F 47 52  
01CA 41 4D 20 46  
01CE 4F 52 20 54  
01D2 48 45 20 4F  
01D6 4D 49 4B 52  
01DA 4F 4E 20 43  
01DE 50 4D 20 4D  
01E2 49 4E 49 20  
01E6 44 52 49 56  
01EA 45  
01EB 00  
01EC  
01ED 54 59 50 45  
01F1 20 52 20 54  
01F5 4F 20 52 45  
01F9 50 45 41 54  
01FD 00  
01FE 54 59 50 45  
0202 20 52 45 54  
0206 55 52 4E 20  
020A 54 4F 20 43  
020E 4F 4E 54 49  
0212 4E 55 45  
0215 00  
0216 49 4E 53 45  
021A 52 54 20 4E  
021E 45 57 20 44  
0222 49 53 4B 20  
0226 49 4E 20 44  
022A 52 49 56 45  
022E 20  
022F 2A  
0230 00  
0231 54 59 50 45  
0235 20 52 45 54  
0239 55 52 4E  
023C 00  
023D 49 4E 53 45  
0241 52 54 20 53  
0245 59 53 54 45  
0249 4D 20 44 49  
024D 53 4B 20 54  
0251 4F 20 52 45  
0255 42 4F 4F 54  
0259 00  
025A 44 49 53 4B  
025E 20 49 53 20  
0262 57 52 49 54  
0266 45 20 50 52  
026A 4F 54 45 43  
026E 54 45 44 20  
0272 00  
0273 54 59 50 45  
0277 20 35 20 46  
027B 4F 52 20 33  
027F 35 20 54 52  
0283 41 43 4B 53  
0287 2C 30 20 46  
028B 4F 52 20 34  
028F 30 2C 20 37  
0293 20 46 4F 52

DEFB 00H  
DRVNR: DEFB 1 :DRIVE NUMBER  
TEXT2: DEFM 'TYPE R TO REPEAT'

DEFB 00H  
TEXT3: DEFM 'TYPE RETURN TO CONTINUE'

DEFB 00H  
TEXT4: DEFM 'INSERT NEW DISK IN DRIVE '

DRVASC: DEFM '\*' :Drive-Nr in ASCII  
DEFB 00H  
TEXT5: DEFM 'TYPE RETURN'

DEFB 00H  
TEXT6: DEFM 'INSERT SYSTEM DISK TO REBOOT'

DEFB 00H  
TEXT7: DEFM 'DISK IS WRITE PROTECTED '

DEFB 00H  
TEXT8: DEFM 'TYPE 5 FOR 35 TRACKS.0 FOR 40. 7 FOR 77 '

0297 20 37 37 20  
 0298 00  
 029C 4D 46 4F 52  
 02A0 4D 36 33 20  
 02A4 41 2C 42 2C  
 02AB 43 2C 4F 52  
 02AC 20 44 20 57  
 02B0 49 4C 4C 20  
 02B4 53 45 4C 45  
 02B8 43 54 20 43  
 02BC 48 4F 53 45  
 02C0 4E 20 44 52  
 02C4 49 56 45  
 02C7 00  
 02C8 4C 4F 53 54  
 02CC 20 44 41 54  
 02D0 41 20 46 4C  
 02D4 41 47 20 53  
 02D8 45 54 20 20  
 02DC 20 20 20 20  
 02E0 52 45 50 45  
 02E4 41 54 21  
 02E7 00

DEFB 00H  
 TEXT9: DEFM 'MFORM63 A.B.C.OR D WILL SELECT CHOSEN DRIVE'

DEFB 00H  
 TEXT10: DEFM 'LOST DATA FLAG SET REPEAT!'

DEFB 00H

02EB CD 0300  
 02EB 3E D8  
 02ED 32 FFFC  
 02F0 CD 0316  
 02F3 3A FFFC  
 02F6 E6 40  
 02F8 C9

-----  
 L00013: CALL NBUSY ;warten auf 'NOT BUSY' ;Force Interupt  
 LD A.0DBH  
 LD (CMDREG),A ;FORCE INTERRUPT every 10 ms  
 CALL WAIT2  
 LD A.(STATR)  
 AND 40H ;WRITE PROTECT - Flag  
 RET

02F9 3A 01BC  
 02FC 32 FFF0  
 02FF C9

-----  
 DRVSEL: LD A,(DRVBIN) ;select Drive  
 LD (SEL).A ;Drive select  
 RET

0300 3A FFFC  
 0303 0F  
 0304 DA 0300  
 0307 C9

-----  
 NBUSY: LD A.(STATR) ;warten auf 'NOT BUSY'  
 RRCA  
 JP C.NBUSY  
 RET

0308 11 06A4  
 030B 1B  
 030C 7B  
 030D B2  
 030E C2 030B  
 0311 0D  
 0312 C2 030B  
 0315 C9

-----  
 WAIT1: LD DE,06A4H ;Warteschleife  
 LOOP1: DEC DE  
 LD A.E  
 OR D  
 JP NZ,LOOP1  
 DEC C  
 JP NZ,WAIT1  
 RET

0316 3E 04  
 0318 3D  
 0319 C2 0318  
 031C C9

-----  
 WAIT2: LD A,04H ;Warteschleife  
 LOOP2: DEC A  
 JP NZ,LOOP2  
 RET

031D CD 02F9  
 0320 3E 03  
 0322 32 FFFC  
 0325 CD 0316  
 0328 CD 0300  
 032B C9

-----  
 L00020: CALL DRVSEL ;Select Drive and Restore  
 LD A.03H  
 LD (CMDREG),A ;Befehl : RESTORE  
 CALL WAIT2  
 CALL NBUSY ;warten auf 'NOT BUSY'  
 RET

```

032C 79          SEEK: LD  A.C          ;Track suchen
032D 32 FFFF    LD  (DATREG).A
0330 3E 13      LD  A.13H          ;SEEK
0332 32 FFFC    LD  (CMDREG),A    ;aktuellen Track suchen
0335 CD 0316    CALL WAIT2
0338 CD 0300    CALL NBUSY
033B CD 0316    CALL WAIT2
033E 0E 02      LD  C,02H
0340 CD 0308    CALL WAIT1
0343 C9        RET

;-----
0344 11 0000    WAIT3: LD  DE.0000H          ;Warteschleife
0347 1B        LOOP3: DEC  DE
0348 7B        LD  A.E
0349 B2        OR   D
034A C2 0347    JP   NZ,LOOP3
034D C9        RET

;-----
034E 3E 01      L00024: LD  A,01H          ;Drive Nr binar darstellen
0350 0C        INC  C
0351 0D        L00025: DEC  C
0352 CA 0359    JP   Z,L00026
0355 07        RLCA
0356 C3 0351    JP   L00025
0359 32 01BC   L00026: LD  (DRVBIN),A    ;01H,02H,04H,08H, fuer A,B,C,D
035C C9        RET

;-----
035D 21 025A    L00027: LD  HL,TEXT7      ;Text: DISK IS WRITE PROTECTED
0360 CD 018A    CALL KETAUS
0363 21 0231    LD  HL,TEXT5          ;Text : 'TYPE RETURN'
0366 CD 018A    CALL KETAUS
0369 CD 0196    CALL L00010          ;warten auf 'RETURN'
036C C9        RET

;-----
036D          DEFS  40H          ;STACK

;-----
03AD 3A 01EC    L00029: LD  A,(DRVNR)      ;Drive testen
03B0 3D        DEC  A
03B1 4F        LD  C,A          ;Drive A = '00' usw
03B2 CD 034E    CALL L00024          ;Drive Nr binar darstellen
03B5 CD 02F9    CALL DRVSEL          ;Drive Nr waehlen
03B8 CD 0344    CALL WAIT3
03BB 3A 01BC   LD  A,(DRVBIN)
03BE CD FC14    CALL PREADY          ;dreht Diskette sich ?
03C1 CD 02EB    CALL L00013          ;Force Interrupt
03C4 CA 03CD    JP   Z,L00030          ;wenn nicht Write Protect
03C7 CD 035D    CALL L00027          ;wenn Write Protect
03CA C3 03AD    JP   L00029          ;neuer Versuch

;-----
03CD CD 031D    L00030: CALL L00020          ;Drive Select + Restore ;Diskette formatieren
03D0 AF        XOR  A
03D1 32 01BB    LD  (TRKNR),A
03D4 CD 02F9    CALL DRVSEL
03D7 CD 0344    CALL WAIT3
03DA 21 1000    L00031: LD  HL,1000H
03DD CD 0431    CALL L00036
03E0 CD 03F1    CALL WRTRK
03E3 21 01BB    LD  HL,TRKNR
03E6 3A 01BA   LD  A,(TRKMAX)
03E9 BE        CP   (HL)
03EA CA 04B9    JP   Z,L00040          ;alle Tracks formatiert
03ED 34        INC  (HL)

```

```

03EE  C3 03DA                JP      L00031      ;naechsten Track
;-----
03F1  3A 01BB      WRTRK: LD      A,(TRKNR)                ;Track schreiben
03F4  4F                LD      C,A
03F5  CD 032C                CALL     SEEK
03F8  CD 02F9                CALL     DRVSEL
03FB  CD 0316                CALL     WAIT2
03FE  CD 042B                CALL     WRTRK1      ;Befehl : Track schreiben
0401  21 1000                LD      HL,1000H
0404  01 FFFF                LD      BC,DATREG
0407  11 FFFC                LD      DE,STATR
040A  1A                MINI: LD      A,(DE)      ;'NOT BUSY'          ;warten auf 'NOT BUSY'
040B  0F                RRCA
040C  D2 0419                JP      NC,L00034     ;bei 'NOT BUSY' Abfrage ob 'DATA LOST' oder fertig
040F  0F                RRCA
0410  D2 040A                JP      NC.MINI      ;warten auf DRQ
0413  7E                LD      A,(HL)
0414  02                LD      (BC),A      ;abspeichern
0415  23                INC     HL
0416  C3 040A                JP      MINI
;-----
0419  3A FFFC      L00034: LD      A,(STATR)      ;Abfrage, ob 'DATA LOST' oder fertig
041C  E6 04                AND     04H      ;'DATA LOST' - Bit
041E  C8                RET     Z          ;wenn fehlerfrei
041F  21 02C8                LD      HL,TEXT10   ;Text: 'LOST DATA FLAG SET'
0422  CD 018A                CALL     KETAUS
0425  C3 012A                JP      L00003      ;von vorne beginnen
;-----
0428  CD 0300      WRTRK1: CALL     NBUSY                ;Track schreiben
042B  3E F4                LD      A,0F4H
042D  32 FFFC                LD      (CMDREG),A  ;Befehl : Write Track
0430  C9                RET
;-----
0431  0E 11      L00036: LD      C,11H                ;Track Abbild in Speicher generieren
0433  06 FF                LD      B,0FFH
0435  CD 04AF                CALL     L00039
0438  1E 01                LD      E,01H
043A  0E 06      L00037: LD      C,06H
043C  06 00                LD      B,00H
043E  CD 04AF                CALL     L00039
0441  0E 01                LD      C,01H
0443  06 FE                LD      B,0FEH
0445  CD 04AF                CALL     L00039
0448  0E 01                LD      C,01H
044A  3A 01BB                LD      A,(TRKNR)
044D  47                LD      B,A
044E  CD 04AF                CALL     L00039
0451  0E 01                LD      C,01H
0453  06 00                LD      B,00H
0455  CD 04AF                CALL     L00039
0458  0E 01                LD      C,01H
045A  43                LD      B,E
045B  CD 04AF                CALL     L00039
045E  0E 01                LD      C,01H
0460  06 00                LD      B,00H
0462  CD 04AF                CALL     L00039
0465  0E 01                LD      C,01H
0467  06 F7                LD      B,0F7H
0469  CD 04AF                CALL     L00039
046C  0E 08                LD      C,08H
046E  06 FF                LD      B,0FFH
0470  CD 04AF                CALL     L00039
    
```

```

0473 0E 06 LD C,06H
0475 06 00 LD B,00H
0477 CD 04AF CALL L00039
047A 0E 01 LD C,01H
047C 06 FB LD B,0FBH
047E CD 04AF CALL L00039
0481 0E 80 LD C,80H
0483 06 E5 LD B,0E5H
0485 CD 04AF CALL L00039
0488 0E 01 LD C,01H
048A 06 F7 LD B,0F7H
048C CD 04AF CALL L00039
048F 0E 09 LD C,09H
0491 06 FF LD B,0FFH
0493 CD 04AF CALL L00039
0496 7B LD A,E
0497 FE 12 CP 12H
0499 CA 04A0 JP Z,L00038
049C 1C INC E
049D C3 043A JP L00037

```

```

-----
04A0 0E FF L00038: LD C,0FFH
04A2 06 FF LD B,0FFH
04A4 CD 04AF CALL L00039
04A7 0E FF LD C,0FFH
04A9 06 FF LD B,0FFH
04AB CD 04AF CALL L00039
04AE C9 RET

```

```

-----
04AF 79 L00039: LD A,C ;in Speicher
04B0 FE 00 CP 00H
04B2 C8 RET Z
04B3 70 LD (HL),B
04B4 23 INC HL
04B5 0D DEC C
04B6 C3 04AF JP L00039

```

```

-----
04B9 CD 01A6 L00040: CALL CRLF ;neue Formatierung bzw Warmstart
04BC 21 01ED LD HL,TEXT2 ;Text: 'TYPE R TO REPEAT'
04BF CD 018A CALL KETAUS
04C2 CD 01A6 CALL CRLF
04C5 21 01FE LD HL,TEXT3 ;Text: 'TYPE RETURN TO CONTINUE'
04CB CD 018A CALL KETAUS
04CC CD 01A6 CALL CRLF ;Ausgabe von CR und LF
04CE 0E 01 L00041: LD C,01H ;Befehl : Eingabe
04D0 CD 0005 CALL 0005H ;Console Input
04D3 E6 5F AND 5FH
04D5 FE 52 CP 52H ;R ;Repeat
04D7 CA 012A JP Z,L00003 ;Beginn von Anfang an
04DA FE 0D CP 0DH
04DC C2 04CE JP NZ,L00041 ;falsche Eingabe
04DF 0E 00 LD C,00H ;Drive A
04E1 CD 034E CALL L00024 ;Drive Nr binar darstellen
04E4 CD 01A6 CALL CRLF ;Ausgabe von CR und LF
04E7 21 023D LD HL,TEXT6 ;Text: 'INSERT SYSTEM DISK TO REBOOT'
04EA CD 018A CALL KETAUS
04ED CD 01A6 CALL CRLF
04F0 21 0231 LD HL,TEXT5 ;Text: 'TYPE RETURN'
04F3 CD 018A CALL KETAUS
04F6 CD 0196 CALL L00010 ;warten auf RETURN
04F9 C3 0000 JP 0000H ;Warmstart
-----

```

```

04FC 31 03AD      CHANGE: LD  SP,STACK      ;Stackpointer setzen ;Aenderungsroutine
04FF 21 0558      LD  HL,TEXT11           ;Text:'Format Program for 63k and 48k CP/M - Version'
0502 CD 01BA      CALL KETAUS
0505 CD 01A6      CALL CRLF
0508 CD 01A6      CALL CRLF
050B 3A 0002      LD  A,(0002H)
050E FE F6        CP  0F6H                ;63k - Version
0510 CA 0103      JP  Z,START            ;keine Aenderung bei 63k - Version
;-----
0513 DD 21 053F    CHFDC: LD  IX,TAB        ;zu aendernde Speicherstellen ;FDC-Adressen
0517 DD 66 00      LOOP: LD  H,(IX)
051A DD 6E 01      LD  L,(IX+1)
051D 3E EF        LD  A,0EFH
051F A6             AND  (HL)                ;niederwertiges Byte
0520 77           LD  (HL),A
0521 23           INC  HL
0522 3E F7        LD  A,0F7H
0524 A6           AND  (HL)                ;hoeherwertiges Byte
0525 77           LD  (HL),A
0526 DD 23        INC  IX
0528 DD 23        INC  IX
052A DD 7E 00     LD  A,(IX)
052D 87           OR  A
052E 28 02       JR  Z,CHEPR            ;Aenderung beendet
0530 18 E5       JR  LOOP              ;naechste Adresse
;-----
0532 3E C0        CHEPR: LD  A,0C0H        ;Einsrung Eprom
0534 32 03C0      LD  (03C0H),A          ;hoeherwertiges Byte
0537 3E 1F        LD  A,1FH
0539 32 03BF      LD  (03BFH),A          ;niederwertiges Byte
053C C3 0103      JP  START              ;Start des eigentlichen Programmes
;-----
053F 01 33 02 EE    TAB:  DEFB  01H,033H,2H,0EEH,2H,0F4H,2H,0FDH,3H,1H,3H,23H,3H,2EH
0543 02 F4 02 FD
0547 03 01 03 23
054B 03 2E
054D 03 33 04 05    DEFB  3H,33H,4H,05H,4H,8H,4H,1AH,4H,2EH,00H
0551 04 08 04 1A
0555 04 2E 00
0558 18 2A          TEXT11: DEFB 18H,2AH
055A 46 6F 72 6D    DEFM  'Format Program for 63k and 48k CP/M - Version'
055E 61 74 20 50
0562 72 6F 67 72
0566 61 6D 20 66
056A 6F 72 20 36
056E 33 68 20 61
0572 6E 64 20 34
0576 38 6B 20 43
057A 50 2F 4D 20
057E 2D 20 56 65
0582 72 73 69 6F
0586 6E
0587 00           DEFB  00H
                        END

```

Macros:

Symbols:

CHANGE	04FC	CHEPR	0532	CHFDC	0513	CMDREG	FFFC
CRLF	01A6	DATREG	FFFF	DRVASC	022F	DRVBIN	01BC
DRVNR	01EC	DRVSEL	02F9	KETAUS	018A	L00001	010C
L00002	0121	L00003	012A	L00005	016A	L00006	0170
L00007	0176	L00008	017C	L00010	0196	L00013	02E8
L00020	031D	L00024	034E	L00025	0351	L00026	0359
L00027	035D	L00029	03AD	L00030	03CD	L00031	03DA
L00034	0419	L00036	0431	L00037	043A	L00038	04A0
L00039	04AF	L00040	04B9	L00041	04CE	LOOP	0517
LOOP1	030B	LOOP2	0318	LOOP3	0347	MINI	040A
NBUSY	0300	PREADY	FC14	SECREG	FFFE	SEEK	032C
SEL	FFF0	STACK	03AD	START	0103	STATR	FFFC
TAB	053F	TEXT1	01BE	TEXT10	02C8	TEXT11	0558
TEXT2	01ED	TEXT3	01FE	TEXT4	0216	TEXT5	0231
TEXT6	023D	TEXT7	025A	TEXT8	0273	TEXT9	029C
TRKMAX	01BA	TRKNR	01BB	TRKREG	FFFD	TRWAHL	0153
WAIT1	0308	WAIT2	0316	WAIT3	0344	WRTRK	03F1
WRTRK1	0428	ZCHAUS	01B1				

No Fatal error(s)



0000'  
0000'

Aseg  
Title LFORM63  
SUBTTL Version 02.02.83  
.Z80

```

; *****
; **                                     **
; **      LFORMAT fuer 63k-Version      **
; **                                     **
; *****

```

```

FC14      PREADY EQU 0FC14H
FFF0      SEL     EQU 0FFFF0H
FFFC      STATR  EQU 0FFFC0H
FFFC      CMDREG EQU 0FFFC0H
FFFD      TRKREG EQU 0FFFD0H
FFFE      SECRES EQU 0FFFE0H
FFFF      DATREG EQU 0FFFF0H
037E      STACK  EQU 037E0H

```

.PHASE 100H

```

;-----
0100      C3 04E2      JP      CHANGE      ;auf 48k-Version umschr. ;Anfang
0103      3A 0082      START: LD      A,(00B2H)      ;Drive Nr bei LFORMAT #
0106      CD 010C      CALL     L00001
0109      C3 012A      JP      L00003
;-----
010C      FE 41        L00001: CP      41H      ;A                      ;Eingabetest
010E      CA 0121      JP      Z,L00002
0111      FE 42        CP      42H      ;B
0113      CA 0121      JP      Z,L00002
0116      FE 43        CP      43H      ;C
0118      CA 0121      JP      Z,L00002
011B      FE 44        CP      44H      ;D
011D      CA 0121      JP      Z,L00002
0120      C9          RET
;-----
0121      32 0205      L00002: LD      (DRVASC),A      ;A,B,C,D          ;Drive-Nummer abspeichern
0124      DE 40        SBC     A,40H
0126      32 01BC      LD      (DRVNR),A      ;1,2,3,4
0129      C9          RET
;-----
012A      21 01BF      L00003: LD      HL,TEXT1      ;Hauptprogramm
012D      CD 015B      CALL     KETAUS
0130      3E C0        LD      A,0C0H
0132      32 FFFE      LD      (SECRES),A      ;FDC-Controller auf 2 MHz umschalten
0135      CD 014D      CALL     L00004      ;Drive Nr testen
0138      21 0000      LD      HL,0000H
0138      21 01EA      LD      HL,TEXT4
013E      CD 015B      CALL     KETAUS
0141      21 0207      LD      HL,TEXT5
0144      CD 015B      CALL     KETAUS
0147      CD 0167      CALL     L00006      ;warten auf 'RETURN'

```

```

014A C3 037E JP L00024 ;formatieren
;-----
014D 3A 01BC L00004: LD A,(DRVNR) ;Drive Nr testen
0150 B7 OR A
0151 C0 RET NZ ;wenn fehlerfrei
0152 21 0269 LD HL,TEXT8
0155 CD 015B CALL KETAUS ;Fehlermeldung
0158 C3 0000 JP 0000H ;Warmstart CP/M
;-----
015B 7E KETAUS: LD A,(HL) ;Ausgabe einer Zeichenkette
015C 23 INC HL
015D B7 OR A
015E CA 0177 JP Z,CRLF
0161 CD 0182 CALL ZCHAUS
0164 C3 015B JP KETAUS
;-----
0167 0E 01 L00006: LD C,01H ;Befehl : Eingabe ;warten auf 'RETURN'
0169 CD 0005 CALL 0005H ;Console Input
016C FE 03 CP 03H ;CTRL C
016E CA 0000 JP Z,0000H ;Warmstart
0171 FE 0D CP 0DH ;RETURN
0173 C2 0167 JP NZ,L00006
0176 C9 RET
;-----
0177 3E 0D CRLF: LD A,0DH ;CR ;Ausgabe von CR und LF
0179 CD 0182 CALL ZCHAUS
017C 3E 0A LD A,0AH :LF
017E CD 0182 CALL ZCHAUS
0181 C9 RET
;-----
0182 0E 02 ZCHAUS: LD C,02H ;Befehl : Ausgabe ;Textausgabe
0184 5F LD E,A ;auszugebendes Zeichen
0185 E5 PUSH HL
0186 CD 0005 CALL 0005H ;Console Output
0189 E1 POP HL
018A C9 RET
;-----
018B TRKNR: DEFS 1 ;aktuelle Track Nr
018C 00 DEFB 00H
018D 00 DEFB 00H
018E DRVBIN: DEFS 1 :01H,02H,04H,08H fuer Drive A,B,C,D
;-----
018F 20 46 4F 52 TEXT1: DEFM ' FORMAT PROGRAM FOR THE OMIKRON CPM 8" DRIVE'
0193 4D 41 54 20
0197 50 52 4F 47
019B 52 41 4D 20
019F 46 4F 52 20
01A3 54 48 45 20
01A7 4F 4D 49 4B
01AB 52 4F 4E 20
01AF 43 50 4D 20
01B3 38 22 20 44
01B7 52 49 56 45
01BB 00 DEFB 00H
01BC 00 DRVNR: DEFB 00H
01BD 20 20 54 59 TEXT2: DEFM ' TYPE R TO REPEAT'
01C1 50 45 20 52
01C5 20 54 4F 20
01C9 52 45 50 45
01CD 41 54
01CF 00 DEFB 00H
01D0 20 20 54 59 TEXT3: DEFM ' TYPE RETURN TO CONTINUE'

```

01D4 50 45 20 52  
01D8 45 54 55 52  
01DC 4E 20 54 4F  
01E0 20 43 4F 4E  
01E4 54 49 4E 55  
01EB 45  
01E9 00  
01EA 20 20 49 4E  
01EE 53 45 52 54  
01F2 20 4E 45 57  
01F6 20 44 49 53  
01FA 4B 20 49 4E  
01FE 20 44 52 49  
0202 56 45 20  
0205 2A  
0206 00  
0207 20 20 54 59  
0208 50 45 20 20  
020F 52 45 54 55  
0213 52 4E 20 28  
0217 4F 52 20 43  
021B 4F 4E 54 52  
021F 4F 4C 20 43  
0223 20 54 4F 20  
0227 52 45 42 4F  
022B 4F 54 29  
022E 00  
022F 20 20 49 4E  
0233 53 45 52 54  
0237 20 53 59 53  
023B 54 45 4D 20  
023F 44 49 53 4B  
0243 20 54 4F 20  
0247 52 45 42 4F  
024B 4F 54  
024D 00  
024E 20 20 44 49  
0252 53 4B 20 49  
0256 53 20 57 52  
025A 49 54 45 20  
025E 50 52 4F 54  
0262 45 43 54 45  
0266 44 20  
0268 00  
0269 20 20 4C 46  
026D 4F 52 4D 36  
0271 33 20 41 2C  
0275 42 2C 43 20  
0279 4F 52 20 44  
027D 20 57 49 4C  
0281 4C 20 53 45  
0285 4C 45 43 54  
0289 20 43 48 4F  
028D 53 45 4E 20  
0291 44 52 49 56  
0295 45  
0296 00  
0297 20 20 4C 4F  
029B 53 54 20 44  
029F 41 54 41 20  
02A3 46 4C 41 47  
02A7 20 53 45 54

DEFB 00H  
TEXT4: DEFM ' INSERT NEW DISK IN DRIVE '

DRVASC: DEFM '1'  
DEFB 00H  
TEXT5: DEFM ' TYPE RETURN (OR CONTROL C TO REBOOT)'

DEFB 00H  
TEXT6: DEFM ' INSERT SYSTEM DISK TO REBOOT'

DEFB 00H  
TEXT7: DEFM ' DISK IS WRITE PROTECTED '

DEFB 00H  
TEXT8: DEFM ' LFORM63 A,B,C OR D WILL SELECT CHOSEN DRIVE'

DEFB 00H  
TEXT9: DEFM ' LOST DATA FLAG SET REPEAT!'

```

02AB 20 20 20 20
02AF 20 20 52 45
02B3 50 45 41 54
02B7 21
02B8 00                                DEFB 00H
;-----
02B9 CD 02D1    L00009: CALL NBUSY          ;warten auf 'NOT BUSY' ;Force Interrupt
02BC 3E 08      LD A,0D8H
02BE 32 FFFC    LD (CMDREG),A      ;FORCE INTERRUPT every 10 ms
02C1 CD 02E7    CALL WAIT2
02C4 3A FFFC    LD A,(STATR)
02C7 E6 40      AND 40H           ;WRITE PROTECT - Flag
02C9 C9        RET
;-----
02CA 3A 018E    DRVSEL: LD A,(DRVBIN)          ;select Drive
02CD 32 FFF0    LD (SEL),A           ;Drive Select
02D0 C9        RET
;-----
02D1 3A FFFC    NBUSY: LD A,(STATR)          ;warten auf 'NOT BUSY'
02D4 0F        RRCA
02D5 DA 02D1    JP C,NBUSY
02D8 C9        RET
;-----
02D9 11 06A4    WAIT1: LD DE,06A4H          ;Warteschleife
02DC 1B        Loop1: DEC DE
02DD 7B        LD A,E
02DE B2        OR D
02DF C2 02DC    JP NZ,Loop1
02E2 0D        DEC C
02E3 C2 02D9    JP NZ,WAIT1
02E6 C9        RET
;-----
02E7 3E 04      WAIT2: LD A,04H           ;Warteschleife
02E9 3D        L00015: DEC A
02EA C2 02E9    JP NZ,L00015
02ED C9        RET
;-----
02EE CD 02CA    L00016: CALL DRVSEL          ;Select Drive and Restore
02F1 3E 00      LD A,00H
02F3 32 FFFC    LD (CMDREG),A      ;Befehl : RESTORE
02F6 CD 02E7    CALL WAIT2
02F9 CD 02D1    CALL NBUSY          ;warten auf 'NOT BUSY'
02FC C9        RET
;-----
02FD 79        SEEK: LD A,C             ;Track suchen
02FE 32 FFFF    LD (DATREG),A
0301 3E 10      LD A,10H           ;SEEK
0303 32 FFFC    LD (CMDREG),A      ;aktuellen Track suchen
0306 CD 02E7    CALL WAIT2
0309 CD 02D1    CALL NBUSY
030C CD 02E7    CALL WAIT2
030F 0E 02      LD C,02H
0311 CD 02D9    CALL WAIT1
0314 C9        RET
;-----
0315 11 0000    WAIT3: LD DE,0000H
0318 1B        LOOP3: DEC DE
0319 7B        LD A,E
031A B2        OR D
031B C2 0318    JP NZ,LOOP3
031E C9        RET
;-----

```

```

031F 3E 01          L00020: LD    A.01H          ;Drive Nr binær darstellen
0321 0C              INC    C
0322 0D              L00021: DEC    C
0323 CA 032A        JP     Z.L00022
0326 07              RLCA
0327 C3 0322        JP     L00021
032A 32 018E        L00022: LD    (DRVBIN).A    :01H,02H,04H,08H fuer A,B,C,D
032D C9              RET

;-----
032E 21 024E        L00023: LD    HL.TEXT7      ;Text: Disk 15 Write Protected
0331 CD 015B        CALL   KETAUS
0334 21 0207        LD     HL.TEXT5          ;Text : 'Type Return'
0337 CD 015B        CALL   KETAUS
033A CD 0167        CALL   L00006           ;warten auf RETURN
033D C9              RET

;-----
033E                DEFS   40H          ;Stack

;-----
037E 3A 018C        L00024: LD    A.(DRVNR)      ;Drive testen
0381 3D              DEC    A
0382 4F              LD     C,A              ;Drive A= '00' usw
0383 CD 031F        CALL   L00020           ;Drive Nr binær darstellen
0386 CD 02CA        CALL   DRVSEL           ;Drive Nr wæhlen
0389 CD 0315        CALL   WAIT3
038C 3A 018E        LD     A.(DRVBIN)
038F CD FC14        CALL   PREADY          ;dreht Diskette sich
0392 CD 02B9        CALL   L00009          ;Force Interrupt
0395 CA 039E        JP     Z.L00025        ;wenn nicht Write Protect
0398 CD 032E        CALL   L00023          ;wen Write Protect
039B C3 037E        JP     L00024          ;neuer Versuch

;-----
039E CD 02EE        L00025: CALL   L00016      ;Drive Select + Restore ;Diskette formatieren
03A1 AF              XOR    A
03A2 32 018B        LD     (TRKNR).A
03A5 CD 0315        CALL   WAIT3
03A8 21 1000        L00026: LD    HL.1000H
03AB CD 0419        CALL   L00032
03AE CD 03BE        CALL   WRTRK
03B1 21 018B        LD     HL.TRKNR
03B4 3E 4C          LD     A.4CH           ;Anzahl Tracks
03B6 BE              CP     (HL)
03B7 CA 04A1        JP     Z.L00036        ;alle Tracks formatiert
03BA 3A              INC    (HL)
03BB C3 03AB        JP     L00026          ;naechsten Track

;-----
03BE 3A 018B        WRTRK: LD    A,(TRKNR)    ;Track schreiben
03C1 4F              LD     C.A
03C2 CD 02FD        CALL   SEEK
03C5 CD 02CA        CALL   DRVSEL
03C8 CD 02E7        CALL   WAIT2
03CB 21 FFFF        LD     HL.DATREG
03CE 01 FFFC        LD     BC.STATR
03D1 1E FF          LD     E.OFFH
03D3 16 02          LD     D.02H
03D5 CD 0410        CALL   L00031
03D8 CD 02E7        CALL   WAIT2
03DB 0A              L00028: LD    A.(BC)
03DC E6 03          AND    03H
03DE E2 03DB        JP     PO.L00028
03E1 73              LD     (HL),E
03E2 0A              L00029: LD    A.(BC)
03E3 A2              AND    D

```

```

03E4 28 FC          JR      Z.L00029
03E6 73            LD      (HL),E
03E7 3E E0         LD      A,0E0H
03E9 32 FFFE       LD      (SECREG),A
03EC 0A            LD      A,(BC)
03ED 73            LD      (HL),E
03EE 11 1000       LD      DE,1000H
03F1 0A            LD      A,(BC)
03F2 1A            L00030: LD      A,(DE)
03F3 77            LD      (HL),A
03F4 13            INC     DE
03F5 00            NOP
03F6 00            NOP
03F7 0A            LD      A,(BC)
03F8 0F            RRCA
03F9 DA 03F2       JP      C.L00030
03FC 3E C0         LD      A,0C0H
03FE 32 FFFE       LD      (SECREG),A

0401 3A FFFC       LD      A,(STATR) ;Abfrage, ob 'Data Lost' oder fertig
0404 E6 04         AND     04H      : 'DATA LOST' - Bit
0406 C8            RET     Z        : wenn fehlerfrei
0407 21 0297       LD      HL,TEXT9 : Text: LOST DATA FLAG SET
040A CD 015B       CALL   KETAUS
040D C3 012A       JP      L00003   ; von vorne beginnen

;-----
0410 CD 02D1       L00031: CALL   NBUSY ; Track schreiben
0413 3E F4         LD      A,0F4H
0415 32 FFFC       LD      (CMDREG),A : Befehl : Write Track
0418 C9            RET

;-----
0419 0E 32         L00032: LD      C,32H ; Track Abbild in Speicher generieren
041B 06 FF         LD      B,0FFH
041D CD 0497       CALL   L00035
0420 1E 01         LD      E,01H
0422 0E 06         L00033: LD      C,06H
0424 06 00         LD      B,00H
0426 CD 0497       CALL   L00035
0429 0E 01         LD      C,01H
042B 06 FE         LD      B,0FEH
042D CD 0497       CALL   L00035
0430 0E 01         LD      C,01H
0432 3A 018B       LD      A,(TRKNR)
0435 47            LD      B,A
0436 CD 0497       CALL   L00035
0439 0E 01         LD      C,01H
043B 06 00         LD      B,00H
043D CD 0497       CALL   L00035
0440 0E 01         LD      C,01H
0442 43            LD      B,E
0443 CD 0497       CALL   L00035
0446 0E 01         LD      C,01H
0448 06 00         LD      B,00H
044A CD 0497       CALL   L00035
044D 0E 01         LD      C,01H
044F 06 F7         LD      B,0F7H
0451 CD 0497       CALL   L00035
0454 0E 0B         LD      C,0BH
0456 06 FF         LD      B,0FFH
0458 CD 0497       CALL   L00035
045B 0E 06         LD      C,06H
045D 06 00         LD      B,00H

```

```

045F CD 0497 CALL L00035
0462 OE 01 LD C.01H
0464 06 FB LD B.0FBH
0466 CD 0497 CALL L00035
0469 OE 80 LD C.80H
046B 06 E5 LD B.0E5H
046D CD 0497 CALL L00035
0470 OE 01 LD C.01H
0472 06 F7 LD B.0F7H
0474 CD 0497 CALL L00035
0477 OE 1B LD C.1BH
0479 06 FF LD B.0FFH
047B CD 0497 CALL L00035
047E 7B LD A.E
047F FE 1A CP 1AH
0481 CA 048B JP Z,L00034
0484 1C INC E
0485 C3 0422 JP L00033

```

```

-----
0488 OE FF L00034: LD C.OFFH
048A 06 FF LD B.OFFH
048C CD 0497 CALL L00035
048F OE FF LD C.OFFH
0491 06 FF LD B.OFFH
0493 CD 0497 CALL L00035
0496 C9 RET

```

```

-----
0497 70 L00035: LD A.C ;in Soeicher
0498 FE 00 CP 00H
049A C8 RET Z
049B 70 LD (HL).B
049C 23 INC HL
049D 0D DEC C
049E C3 0497 JP L00035

```

```

-----
04A1 CD 0177 L00036: CALL CRLF ;neue Formatierung bzw Warmstart
04A4 21 01B0 LD HL.TEXT2 ;Text: 'Type R to Repeat'
04A7 CD 015B CALL KETAUS
04AA CD 0177 CALL CRLF
04AD 21 01D0 LD HL.TEXT3 ;Text: 'Type Return to Continue'
04B0 CD 015B CALL KETAUS
04B3 CD 0177 CALL CRLF ;Ausgabe von CR und LF
04B6 OE 01 L00037: LD C.01H ;Befehl : Eingabe
04BB CD 0005 CALL 0005H ;Console Input
04BB FE 52 CP 52H ;R ;Repeat
04BD CA 012A JP Z,L00003 ;Beginn von Anfang an
04C0 FE 0D CP 0DH
04C2 C2 04B6 JP NZ,L00037 ;falsche Eingabe
04C5 OE 00 LD C.00H ;Drive A
04C7 CD 031F CALL L00020 ;Drive Nr binar darstellen
04CA CD 0177 CALL CRLF ;Ausgabe von CR und LF
04CD 21 022F LD HL.TEXT6 ;Text: 'Insert System Disk to Reboot'
04D0 CD 015B CALL KETAUS
04D3 CD 0177 CALL CRLF
04D6 21 0207 LD HL.TEXT5 ;Text: 'Type RETURN'
04D9 CD 015B CALL KETAUS
04DC CD 0167 CALL L00006 ;warten auf RETURN
04DF C3 0000 JP 0000H ;Warmstart

```

```

-----
04E2 31 037E CHANGE: LD SP.STACK ;Stackpointer setzen ;Aenderunsroutine
04E5 21 0542 LD HL.TEXT11 ;Text:'Format Program for 63k and 48k CP/M - Version'
04EB CD 015B CALL KETAUS

```

```

04EB CD 0177 CALL CRLF
04EE CD 0177 CALL CRLF
04F1 3A 0002 LD A,(0002H)
04F4 FE F6 CP 0F6H ;63k - Version
04F6 CA 0103 JP Z,START ;keine Aenderung bei 63k - Version
;-----
04F9 DD 21 0525 CHFDC: LD IX,TAB ;zu aendernde Speicherstellen ;FDC-Adressen
04FD DD 66 00 LOOP: LD H,(IX)
0500 DD 6E 01 LD L,(IX+1)
0503 3E EF LD A,0EFH
0505 A6 AND (HL) ;niederwertiges Byte
0506 77 LD (HL),A
0507 23 INC HL
0508 3E F7 LD A,0F7H
050A A6 AND (HL) ;hoeherwertiges Byte
050B 77 LD (HL),A
050C DD 23 INC IX
050E DD 23 INC IX
0510 DD 7E 00 LD A,(IX)
0513 B7 OR A
0514 2B 02 JR Z,CHEPR ;Aenderung beendet
0516 1B E5 JR LOOP ;naechste Adresse
;-----
0518 3E C0 CHEPR: LD A,0C0H ;Einsprung Eprom
051A 32 0391 LD (0391H),A ;hoeherwertiges Byte
051D 3E 1F LD A,1FH
051F 32 0390 LD (0390H),A ;niederwertiges Byte
0522 C3 0103 JP START ;Start des eigentlichen Programmes
;-----
0525 01 33 02 BF TAB: DEFB 1H,33H,2H,0BFH,2H,0C5H,2H,0CEH,2H,0D2H,2H,0F4H,2H,0FFH,3H,4H
0529 02 C5 02 CE
052D 02 D2 02 F4
0531 02 FF 03 04
0535 03 CC 03 CF DEFB 3H,0CCH,3H,0CFH,3H,0EAH,3H,0FFH,4H,2H,4H,16H,0H
0539 03 EA 03 FF
053D 04 02 04 16
0541 00
0542 1B 2A TEXT11: DEFB 1BH,2AH
0544 46 6F 72 6D DEFM 'Format Program for 63k and 48k CP/M-Version'
0548 61 74 20 50
054C 72 6F 67 72
0550 61 6D 20 66
0554 6F 72 20 36
0558 33 6B 20 61
055C 6E 64 20 34
0560 3B 6B 20 43
0564 50 2F 4D 2D
0568 56 65 72 73
056C 69 6F 6E
056F 00 DEFB 00H
END

```



Macros:

Symbols:

CHANGE	04E2	CHEPR	0518	CHFDC	04F9	CMDREG	FFFC
CRLF	0177	DATREG	FFFF	DRVASC	0205	DRVBIN	018E
DRVNR	01BC	DRVSEL	02CA	KETAUS	015B	L00001	010C
L00002	0121	L00003	012A	L00004	014D	L00006	0167
L00009	02B9	L00015	02E9	L00016	02EE	L00020	031F
L00021	0322	L00022	032A	L00023	032E	L00024	037E
L00025	039E	L00026	03A8	L00028	03DB	L00029	03E2
L00030	03F2	L00031	0410	L00032	0419	L00033	0422
L00034	0488	L00035	0497	L00036	04A1	L00037	0486
LOOP	04FD	LOOP1	02DC	LOOP3	0318	NBUSY	02D1
PREADY	FC14	SECREG	FFFE	SEEK	02FD	SEL	FFF0
STACK	037E	START	0103	STATR	FFFC	TAB	0525
TEXT1	018F	TEXT11	0542	TEXT2	018D	TEXT3	01D0
TEXT4	01EA	TEXT5	0207	TEXT6	022F	TEXT7	024E
TEXT8	0269	TEXT9	0297	TRKNR	018B	TRKREG	FFFD
WAIT1	02D9	WAIT2	02E7	WAIT3	0315	WRTRK	03BE
ZCHAUS	0182						

No Fatal error(s)