```
        page    90
        subttl  Generierparameter
;*****************************************
;*                                       *
;*       Allgemeine Vereinbarungen       *
;*       ====.   E P O S      ====       *
;*       - 01.08.89                      *
;*       - Grundversion                  *
;*                                       *
;*****************************************
;.1.08.89 Hen    -neue Formate
;                -Tastaturcode-Tabellen
; 2.08.89 Hen    -Spezify-Parameter in DPBs
; 5.10.89 Hen    -Tastatur K7672.03
;25.01.89 Hen    -Weitere Treiber eingefuegt

        cseg
        .z80
        .lall
        .tfcond

   ;                !!! Fuer den Nutzer !!!

   ;Die mit ';**' gekennzeichneten Zeilen koennen bei der Gene-
   ;rierung geaendert werden. Alle anderen Zeilen sollten unver-
   ;aendert bleiben.


;----- Version -----------

vers    equ     13
dat     equ     190
  if1
        .printx * EPOS Version 1.3   1/90 *
  endif

   ;Hier sollte nichts geaendert werden !!!

test    equ     0       ;** Testversion
   ;Fuer die Erleichterung der Inbetriebnahme ist es moeglich,
   ;die Umladeprozesse und die Initialisierungen so zu beeinflus-
   ;sen, dass das neue Betriebssystem auf die Adresse 8000h
   ;umgeladen wird. Damit ist ein Test mittels DU (ZSID)
   ;moeglich.
  if    test
  if1
        .printx * Test-Version *
  endif

  endif


;----- Kalt- und Warmstart-Kommandos  -----
;-----       und Startprogramm         -----
```



```
offsec  equ     0       ;** Groesse des Startprogramm
                        ;Fuer NLOAD: 11
```

```
   ;offsec ist die Anzahl der 256-Byte-Sektoren der Starthilfe
   ;(siehe auch 2.1.). Im folgenden Macro muss anstelle von
   ;NLOAD.COM der Name der Starthilfe eingeschrieben werden.
   ;(Ist nur bei EPROM-Urlader, sinnvoll, siehe unten)

stoff   equ     offsec*256
                        ;Anzahl Byte fuer Startprog.

stprg   macro           ;** Name des Start-Programmes
x1pl:   db      x1len
  if    offsec
        db      'SAVE '
        db      offsec/10+30h,offsec mod 10+30h
        db      ' NLOAD.COM'

        else    ;offsec

   ;    db      'DIENST'        ;**
   ;Hier kann ein beliebiges Kaltstart-Kommando eingetragen werden.
   ;Dieses wird nicht umgeladen, sondern ueber CCP aufgerufen.
  endif
        ;offsec
x1len   equ     $-x1pl-1

        endm    ;stprg


wbcom   macro           ;** Name des Warmstartstart-
                        ;** Kommando
x2pl:   db      x2len
   ;    db      'DIR'   ;** Hier kann ein Warmstart-
                        ;** Kommando eingetragen werden
   ;Diese Kommando funktioniert nur wenn der CCP nicht resi-
   ;dent ist.
x2len   equ     $-x2pl-1
        ds      128-x2len,0
        endm    ;wbcom


;---- Adressen und Konfiguration ------
  iff   test
tpa     equ     0d000h  ;** TPA-Ende (CCP Beginn !)
ramend  equ     0f700h  ;** Ende des nutzbaren RAMs
   ;Hier werden die Beginn-(tpa) und die End-(ramend) Adres-
   ;sen des Betriebssystems eingetragen. tpa bezeichnet den
   ;Beginn von CCP.
  else  ;test
tpa     equ     0300h
ramend  equ     3800h
  endif
        ;test

inst    equ     200h    ;** Maximale Laenge EPOSINST
```



```
   ;Falls ein umfangreicheres EPOSINST verwendet werden soll,
   ;ist, muss hier geaendert werden.

biosl   aset    1200h   ;** Maximale Laenge EPOSBIOS
   ;Hier kann die maximale Laenge des Code-Bereiches (ohne
   ;RAM) eingetragen werden. Diese Angabe wird fuer die
   ;Berechnung der effektiven Laenge beim Umladen benoetigt.
  if    biosl gt ramend-tpa-800h-0e00h-inst
```

```
biosl   aset    ramend-tpa-800h-0e00h-inst
  endif
```

3/

```
ccpr    equ     1           ;** CCP resident oder auf
                            ;** (RAM-) Floppy
                            ;   1= resident, 0= nicht res.
   ;Dieses Flag legt fest, ob der CCP fest im RAM installiert
   ;wird, oder ob er bei jedem Warmstart vom (RAM-)Floppy
   ;nachgeladen wird. Ist eine RAM-Floppy installiert, so wird
   ;der CCP  auf die RAM-Floppy geschrieben und beim Warmstart
   ; von dort geladen, ansonsten steht er ja auf den
   ;Systemspuren der Diskette.
   ;Wenn der CCP als resident gwaehlt wird, hat man weniger
   ;Hauptspeicher (2kByte), aber der Warmstart geht schneller.
   ;Ausserdem kann beim Test weniger nicht gehen.

tmon    equ     1           ;** Testmonitor einbinden
                            ;   0= kein Monitor
   ;Es ist moeglich einen Testmonitor in das BIOS einzubinden.
   ;Dieser kann mit der Taste (physischer Tastencode in
   ;EPOSCONx TTMON) aufgerufen werden.
   ;(nur bei EPOSCON1 und EPOSCON2)

wbkey   equ     0           ;** Sonder-Taste fuer Warmstart-
                            ;** Ausloesung (0: keine)
   ;Die Warmstart-Taste funktioniert unabhaengig von ^C
   ;und solange der Systeminterrupt laeuft.
   ;Der Tastencode steht in EPOSCONx (TWBKEY).
   ;(nur bei EPOSCON1 und EPOSCON2)

 ;----- Interrupt ------

sysint  aset    1           ;** System-Interrupt fuer
                            ;   Tastatur und Floppy
sysctc  equ     0ffh        ;** CTC fuer Systemtakt (25ms)
   ;Hier kann man sich wuenschen, ob das System mit Interrupt
   ;arbeiten soll oder nicht. Der CTC wird aber auch benoetigt
   ;wenn Floppy-Disk-Laufwerke angeschlossen sind.

;----- Urlader -------
dul     equ     1           ;** Urlader-Typ
                            ;   1= Diskette, 0= EPROM
   ;Es gibt zwei Moeglichkeiten den Urlader zu realisieren:
   ;1. Das System wird von Diskette geladen
   ;2. Das System wird von einer Eprom-Bank in den
   ;   RAM geladen
```



```
uldos   equ     0           ;** MS-DOS-Lader einbinden
                            ;   1= einbinden, 0= nicht
   ;Es wird ein MS-DOS-Urlader eingebunden, um bei Vor-
   ;handensein einer PZ86 direkt von einer DOS-Diskette
   ;zu realisieren

uladr   aset    0c800h      ;** Urlader-Anfang
   ;Auf dieser Adresse wird der Disk-Urlader abgearbeitet.
   ;Er laedt sich selbst dorthin.

epanf   equ     4000h       ;** Eprom-Anfang
eplen   equ     3000h       ;** Eprom-Laenge
   ;das sind die Daten der EPROM-Bank. Sie werden von EPOSU
```

```
   ;gebraucht

  if     test
uladr    aset     100h
  endif
         ;test

stpro    aset     inst+800h+0e00h+biosl
stpro    aset     (stpro+3ffh) and 0fc00h
                       ;Platz fuer Startprogramm

;----- Monitor   ---------

scrtyp   equ      2        ;** TYP
   ;Im Moment gibt es 3 Typen
   ;1:   Normal, Invers wird durch die Hardware gesteuert
   ;     (ROBOTRON-Video-Platten). Die Steuerzeichen (84h,85h,
   ;     86h,87h) wirken entsprechend.
   ;2:   Hier laesst die Hardware nur eine Moeglichkeit zu,
   ;     etwas hervorzuheben (BIT 7 gesetzt).Die Steuerzeichen
   ;     85h,86h,87h schalten dieses Bit ein, 84h aus.
   ;3:   Genau wie scrtyp 2. Zusaetzlich lassen sich mit den
   ;     Steuerzeichen 88h und 89h noch Sonderzeichen im Zei-
   ;     chensatz darstellen (dieser Treiber wird nicht
   ;     mitgeliefert).

scbeg    equ      0f800h   ;** Begin BWS
linmb    equ      24       ;** Anzahl der Zeilen
zlen     equ      80       ;** Zeilenlaenge
   ;Hier werden die Bildschirmdaten eingetragen.

softcu   aset     0;30     ;** Software-Kursor
cuchar   equ      80h+20h  ;** Kursor-Zeichen
   ;Wenn man das Bit 7 nicht hat oder nicht zur Realisierung
   ;des Kursors benutzen will, kann man auch einen Software-
   ;Kursor einstellen. Er blink mit der angegebenen System-
   ;taktzahl. Dargesellt wird das Zeichen auf der BS-Position
   ;im Wechsel mit 'cuchar'.

  if     softcu and (not sysint)
         :printx * Fuer Softkursor fehlt Systemtakt (sysint) *
```



```
softcu   aset     0
  endif ;softcu and (not sysint)

;----- Tastatur  ---------

ttypp    equ      1        ;** Tastatur-Typ (physisch)
   ;1:   IFSS-Tastatur ueber SIO
   ;2:   ASCII-Tastatur ueber PIO (Bit 7 = Taste aktiv)
   ;3:   Parallele Tastatur ueber ATS
   ;4:   K7672 (Scan-Mode) an IFSS
   ;5:   K7672 (fuer EPOSCON3)
   ;6:   K7672 (fuer EPOSCON3 mit SIO-Interrupt)


ttypl    equ      2        ;** Tastatur-Typ (logisch)
   ;Auch 2 Typen
   ;1:   nur das notwendigste fuer EPOS
   ;2:   ein bischen Komfort (Stringtastendefinition)
   ;3:   komfortables Tastaturdressing (EPOSCON3, wird
```

```
   ;      aber auch nicht mitgeliefert)

tport   equ     0a1h    ;** Daten-Port der Tastatur
tportc  equ     tport+1 ;** Steuer-Port der Tastatur
tctc    equ     0a6h    ;** Takt-CTC
tbufl   equ     16      ;** Tastaturpufferlaenge ( >2 !)
   ;Hier werden die Portadressen und die Tastaturpufferlaenge
   ;eingetragen. Diese muss fuer ttypl equ 2 mindestens 10 Byte
   ;lang sein.

tcodtab equ     1       ;** Code-Tabelle fuer Tastaturtyp
   ;Da die Tastaturen alle verschiedene Codes (zumindest
   ;die Sondertasten) muessen auch unterschiedliche Codetabellen
   ;verwendet werden
   ; 1=  K7637.67
   ; 2=  K7634.54
   ; 3=  K7634.51
   ; 4=  K7672.03
  if2
  if    (tbufl lt 2) or ((ttypl eq 2) and (tbufl lt 10))
        .printx * !!! Tastaturpuffer zu klein !!! *
  endif ;tbufl
  endif ;if2

;----- Disk-System -------

maxdsk  equ     4       ;** Groesste Anzahl von log. LWen (max. 5)
   ;Beispiel: Es sind ein Ram-Floppy und zwei Diskettenlauf-
   ;werke (Kanaele) angeschlossen. Das zweite Laufwerk soll
   ;unter zwei Namen mit unterschiedlichen Formaten ange-
   ;sprochen werden. Summasummarumm:
   ;A: Ram-Floppy
   ;B: Diskette 1 (Kanal 0) Format 1
   ;C:      "   2 (  "   1) Format 1
   ;D:      "   2 (  "   1) Format 2
```



```
   ;maxdisk=4

maxkap  equ     400     ;** Maximale FD-Kapazitaet [Bloecke]
   ;Ist fuer 800k-Disketten ausreichend. (wird fuer die Fest-
   ;legung der Laenge der Allocationsmaps (EPOSRAM 2.4.16) be-
   ;noetigt).

maxdir  equ     128     ;** Maximale Anzahl Dir.-Eintraege
   ;Ist fuer 800k-Disketten ausreichend. (wird fuer die Fest-
   ;legung der Laenge der Checkvektoren (EPOSRAM ) be-
   ;noetigt)

diskanz equ     1       ;** Anzahl phys. LWe
   ;Anzahl der Kanaele, ohne RAM-Floppy

plwa    equ     0;80h   ;** Zuordnung Laufwerk-phys. Kanal
plwb    equ     0       ;** Kanal 0..3
plwc    equ     0       ;** (80h = Ram-Floppy)
plwd    equ     0
plwe    equ     80h
pklw    equ     0       ;** Kaltstartlaufwerk
   ;Hier wird die Zuordnung Laufwerk(name) = Kanal (bzw RAM-
   ;Floppy) getroffen. pklw ist nur fuer EPOSUL wichtig.

flwa    equ     6       ;** Formate (nur fuer Disk)
flwb    equ     5       ;** 1=  5 x 1024 x 80 x 2
```

```
                                ;** 2= 16 x  256 x 80 x 2
flwd    equ     1               ;** 3= 10 x  512 x 80 x 2
flwe    equ     6               ;** 4=  9 x  512 x 80 x 2
                                ;** 5= 16 x  256 x 40 x 1 (1.6)
                                ;** 6= 16 x  256 x 40 x 1 (1.2)
                                ;** 7=  5 x 1024 x 40 x 1 (190 K)*
                                ;** 8=  5 x 1024 x 80 x 2 (800 K)*

   ;Festlegung der Formate, die Angabe beim RAM-Floppy-Laufwerk
   ;ist bedeutungslos.

tlwa    equ     0               ;** Laufwerkstyp (nur fuer Disk)
tlwb    equ     0               ;** 1: 8''- Laufwerk
tlwc    equ     0               ;** 0: 5 1/4 ''- Laufwerke
tlwd    equ     0               ;**
tlwe    equ     0               ;**
tklw    equ     0               ;** Kaltstartlaufwerk
   ;Festlegung der Typen, die Angabe beim RAM-Floppy-Laufwerk
   ;ist bedeutungslos (noch nicht installiert).

;----- Ram-Floppy --------
   ;Wenn bei tlw(x)(x=a,b,c..) eine 80h auftaucht, wird ein
   ;RAM-Floppy-Treiber mit den folgenden Daten installiert.
   ;Im Moment gibt es 4 Typen: 192k, 256k, 512k, 448k.

                                ;** wenn bei tlw(x) angegeben
rfkap   equ     192             ;** Kapazitaet der Ram-Floppy
ioa     equ     0c0h            ;** I/O-Adresse
ioa2    equ     0c8h            ;** I/O-Adresse der 2.Platte
```



```
wind    equ     0f700h          ;** Fenster-Adresse

rlw     aset    'Z'
rflop   aset    0
  if    plwa eq 80h
rlw     aset    'A'
rflop   aset    1
  endif

  if    plwb eq 80h
rlw     aset    'B'
rflop   aset    1
  endif

  if    plwc eq 80h
rlw     aset    'C'
rflop   aset    1
  endif

  if    plwd eq 80h
rlw     aset    'D'
rflop   aset    1
  endif

  if    plwe eq 80h
rlw     aset    'E'
rflop   aset    1
  endif


  if maxdsk-rflop
len     equ     stpro+stoff-1024
```

```
  else ; maxdsk-rflop
len     equ     stpro+stoff
  endif ; maxdsk-rflop


ccptrk  equ     0       ;Mit diesen Angaben wird der CCP
ccpsec  equ     0       ;vom RAM-Floppy geladen.

;----- Floppy-Disk -------

updprt  equ     92h     ;** Steuerport FDC-Karte
updcmd  equ     updprt+2        ;Kommandoport FDC
upddat  equ     updcmd+1        ;Datenport FDC
   ;Das sind die I/O-Adressen der FDC-Karte.

;---- LAN -----

niu     equ     0a0h    ;** Adr NIU
geadr   equ     42      ;** eigene Adresse
dbadr   equ     92      ;** Datenbankadresse
ulw     equ     'A'-41h ;** Kaltstart-Laufwerk beim Server
mpassw  macro
        db

        endm
   ;Diese Angaben werden fuer EPOSNUL und bestimmte
   ;LAN-Anwendungen gebraucht.

;---- Printer ----

ptypp   equ     0       ;** Printer-Typ
pport   equ     0f8h    ;** I/O-Adr
pportc  equ     pport+1 ;**
pctc    equ     0fch    ;** CTC
   ;Es sind 2 Typen moeglich (beide ueber SIO)
   ;1:   DC1/DC3-Protokoll (i.a. IFSS)
   ;2:   DTR-Protokoll (i.a. V.24)

;---- Serielle Schnittstelle ----
   ;ist fuer spaeter
stypp   equ     0       ;** Serial-Typ
sport   equ     0       ;** I/O-Adr
sctc    equ     0       ;** CTC

;---- Kassette ----
   ;wird auch nicht benutzt
ktypp   equ     0       ;** Kass-Typ
kport   equ     0       ;** I/O-Adr
kctc    equ     0       ;** CTC
rmask   equ     0       ;** Maske Lesen
wmask   equ     0       ;** Maske Schreiben
smask   equ     0       ;** Maske Steuer


;---- ZRE ----
   ;Dies sind auch Daten fuer den Urlader. Sie beziehen sich
   ;auf das Umladen der Urlader-EPROMs und Aktivieren des RAMs.
   ;Wer hier 'sonst.' ankreuzt, sollte sich EPOSUMLD Urladers
   ;nochmal genau angucken.

zretyp  equ     1       ;** ZRE-Typ  1: NANOS
                        ;**          2: KRED
```

```
                                ;                PIO B1 wenn Ramtyp = 2)
                                ;**            3: sonst.
ramtyp  equ     2               ;** Ram-Typ    1: Ram-Fl.(256-k-DRam)
                                ;**            2: 64-k=Ram
                                ;**            3: 64k+448k Ram-Fl.(2x256-k-DRam)
                                ;**            4: sonst.

  ;**** So, der Rest ist nicht mehr interessant. ****
;---- Daten zum Blockumladen -----
  ;interne Daten !!

pha1    equ     tpa             ;CCP
max1    equ     800h
pha2    equ     tpa+800h        ;BDOS
max2    equ     0e00h
pha3    equ     tpa+1600h       ;BIOS
pha4    equ     tpa-inst        ;ladr
```



```
max4    equ     inst
  if    dul
pha5    equ     uladr           ;URLAD
max5    aset    800h
pha6    equ     uladr+800h      ;Test-Exp.
max6    equ     800h
  else  ;dul
pha5    equ     0               ;URLAD
max5    aset    100h
pha6    equ     800h            ;Test-Exp.
max6    equ     800h
  endif
        ;dul

ccp     equ     pha1
bdos    equ     pha2
bios    equ     pha3

;----- Macros    ---------

;Spezify-Parameter fuer 1.6 und 1.2 LW
;-----------------------------------------
spez16  macro
        db      0efh            ;(8) SRT= 4 ms, HUT= 480 ms
        db      003h            ;(9) HLT= 4 ms, non DMA
        endm

spez12  macro
        db      08fh            ;(8) SRT= 16 ms, HUT= 480 ms
        db      01fh            ;(9) HLT= 64 ms, non DMA
        endm
;Format-Uebersicht fuer Report
;-----------------------------------------
mfmte   macro
        .printx         * Format 1=  5 x 1024 x 80 x 2 (780 K)*
        .printx         *        2= 16 x  256 x 80 x 2 (624 K)*
        .printx         *        3= 10 x  512 x 80 x 2 (780 K)*
        .printx         *        4=  9 x  512 x 80 x 2 (720 K)*
        .printx         *        5= 16 x  256 x 40 x 1 (1.6) (148 K)*
        .printx         *        6= 16 x  256 x 40 x 1 (1.2) (148 K)*
        .printx         *        7=  5 x 1024 x 40 x 1 (190 K)*
        .printx         *        8=  5 x 1024 x 80 x 2 (800 K)*

        endm
```

```
;Format-Anzeige in Konfigurationszeile
;-----------------------------------------------------------
mfma    macro   fmtnr,plw
  if    fmtnr eq 1
        db      '5*1024*80*2/'
        db      plw+30h
        db      ' (780 k)'
  endif
        ;fmtnr eq 1
```



```
  if    fmtnr eq 2
        db      '16*256*80*2/'
        db      plw+30h
        db      ' (624 k)'
endif
        ;fmtnr eq 2

  if    fmtnr eq 3
        db      '10*512*80*2/'
        db      plw+30h
        db      ' (780 k)'
  endif
        ;fmtnr eq 3

  if    fmtnr eq 4
        db      '9*512*80*2/'
        db      plw+30h
        db      ' (720 k)'

  endif
        ;fmtnr eq 4

  if    fmtnr eq 5
        db      '16*256*40*1/'
        db      plw+30h
        db      ' (148 k)'
  endif
        ;fmtnr eq 5

  if    fmtnr eq 6
        db      '16*256*40*1/'
        db      plw+30h
        db      ' (148 k)'
  endif
        ;fmtnr eq 6

  if    fmtnr eq 7
        db      '5*1024*40*1/'
        db      plw+30h
        db      ' (190 k)'
  endif
        ;fmtnr eq 7

  if    fmtnr eq 8
        db      '5*1024*80*2/'
        db      plw+30h
        db      ' (800 k)'
  endif
        ;fmtnr eq 8
```

```
        endm    ;mfma

; Disk-Parameter-Bloecke
```



```
;-------------------------

format  macro   fmtnr,tlw,plw

  if    fmtnr eq 1
   ;logischer Teil (bdos)
        dw      80              ;spt
        db      4               ;bsh
        db      15              ;blm
        db      0               ;exm
        dw      389             ;dsm
        dw      127             ;drm
        db      0c0h            ;al0
        db      00              ;al1
        dw      32              ;cks
        dw      2               ;off

   ;Dieser Teil ist nicht CP/M-typisch
        db      plw             ;device

   ;physischer Teil (diskio)
        db      41h-tlw         ;(0) initbyte:
                                ;bit0 =1: mini-floppy
                                ;      0: 8''-floppy
                                ;bit4 =0:1 step/track lw dd, disk sd
                                ;bit4 =1:2 step/track lw = disk
                                ;bit5    (nicht benutzt )
                                ;bit6 =0:Praekomp. ein ab prae (1.2)
                                ;bit6 =1:keine Praekomp.
                                ;bit7 =0:lw nicht initialis.
        db      040h            ;(1) mt, mfm-(fm)-cmd
        db      3               ;(2) n (voreinstellung 1kbyte
        db      6               ;(3) eot (voreinstellung 1kbyte
        db      0eh             ;(4) gpl (voreinstellung 1kbyte .
        db      30h             ;(5) gpl3
        db      0e5h            ;(6) Daten beim Formatieren
        db      25              ;(7) prae
        spez16

  endif
        ;fmtnr eq 1

  if    fmtnr eq 2
   ; Format 16 x 256 x 80 x 2
        dw      64              ;spt
        db      4               ;bsh
        db      15              ;blm
        db      0               ;exm
        dw      311             ;dsm
        dw      127             ;drm
        db      0c0h            ;al0
        db      00              ;al1
        dw      32              ;cks
        dw      2               ;off
```

7

```
        db      plw                     ;device

        db      41h-tlw                 ;(0) initbyte:
        db      040h                    ;(1) mt, mfm-(fm)-cmd
        db      1                       ;(2) n (voreinstellung 256 byte
        db      17                      ;(3) eot (voreinstellung 256 byte
        db      0eh                     ;(4) gpl (voreinstellung 256 byte
        db      32h                     ;(5) gpl3
        db      0e5h                    ;(6) Daten beim Formatieren
        db      25                      ;(7) prae
        spez16

endif
        ;fmtnr eq 2

if      fmtnr eq 3
 ; Format 10 x 512 x 80 x 2
        dw      80                      ;spt
        db      4                       ;bsh
        db      15                      ;blm
        db      0                       ;exm
        dw      389                     ;dsm
        dw      127                     ;drm
        db      0c0h                    ;al0
        db      00                      ;al1
        dw      32                      ;cks
        dw      2                       ;off

        db      plw                     ;device

        db      41h-tlw                 ;(0) initbyte:
        db      040h                    ;(1) mt, mfm-(fm)-cmd
        db      2                       ;(2) n
        db      11                      ;(3) eot
        db      08h                     ;(4) gpl
        db      20h                     ;(5) gpl3
        db      0e5h                    ;(6) Daten beim Formatieren
        db      25                      ;(7) prae
        spez16

endif
        ;fmtnr eq 3


if      fmtnr eq 4
 ; Format 9 x 512 x 80 x 2
        dw      72                      ;spt
        db      4                       ;bsh
        db      15                      ;blm
        db      0                       ;exm
        dw      350                     ;dsm
        dw      127                     ;drm
        db      0c0h                    ;al0
        db      00                      ;al1
        dw      32                      ;cks
```

```
        dw      2                       ;off


        db      plw                     ;device

        db      41h-tlw                 ;(0) initbyte:
        db      040h                    ;(1) mt, mfm-(fm)-cmd
        db      2                       ;(2) n
        db      10                      ;(3) eot
        db      02ah                    ;(4) gpl
        db      50h                     ;(5) gpl3
        db      0e5h                    ;(6) Daten beim Formatieren
        db      25                      ;(7) prae
        spez16

endif
        ;fmtnr eq 4

if      fmtnr eq 5
 ; Format 16 x 256 x 40 x 1 (auf 1.6)
        dw      32                      ;spt
        db      4                       ;bsh
        db      15                      ;blm
        db      1                       ;exm
        dw      73                      ;dsm
        dw      63                      ;drm
        db      080h                    ;al0
        db      00                      ;al1
        dw      16                      ;cks
        dw      3                       ;off

        db      plw                     ;device

        db      51h-tlw                 ;(0) initbyte:
        db      040h                    ;(1) mt, mfm-(fm)-cmd
        db      1                       ;(2) n
        db      17                      ;(3) eot
        db      0eh                     ;(4) gpl
        db      32h                     ;(5) gpl3
        db      0e5h                    ;(6) Daten beim Formatieren
        db      25                      ;(7) prae
        spez16

endif
        ;fmtnr eq 5

if      fmtnr eq 6
 ; Format 16 x 256 x 40 x 1 (auf 1.2)
        dw      32                      ;spt
        db      4                       ;bsh
        db      15                      ;blm
        db      1                       ;exm
        dw      73                      ;dsm
        dw      63                      ;drm
        db      080h                    ;al0
```



```
        db      00                      ;al1
        dw      16                      ;cks
```

```
        dw      3                       ;off

        db      plw                     ;device

        db      01h-tlw                 ;(0) initbyte:
        db      040h                    ;(1) mt, mfm-(fm)-cmd
        db      1                       ;(2) n
        db      17                      ;(3) eot
        db      0eh                     ;(4) gpl
        db      32h                     ;(5) gpl3
        db      0e5h                    ;(6) Daten beim Formatieren
        db      25                      ;(7) prae
        spez12

endif
        ;fmtnr eq 6

if      fmtnr eq 7
 ; Format 5 x 1024 x 40 x 1
        dw      40                      ;spt
        db      3                       ;bsh
        db      7                       ;blm
        db      0                       ;exm
        dw      189                     ;dsm
        dw      63                      ;drm
        db      080h                    ;al0
        db      00                      ;al1
        dw      16                      ;cks
        dw      2                       ;off

        db      plw                     ;device

        db      41h-tlw                 ;(0) initbyte:
        db      040h                    ;(1) mt, mfm-(fm)-cmd
        db      3                       ;(2) n (voreinstellung 1024 byte
        db      6                       ;(3) eot (voreinstellung 1024 byte
        db      0eh                     ;(4) gpl (voreinstellung 1024 byte
        db      30h                     ;(5) gpl3
        db      0e5h                    ;(6) Daten beim Formatieren
        db      25                      ;(7) prae
        spez12

endif
        ;fmtnr eq 7

if      fmtnr eq 8
 ; Format  5 x 1024 x 80 x 2  (800 K)
        dw      80                      ;spt
        db      4                       ;bsh
        db      15                      ;blm
        db      0                       ;exm
        dw      399                     ;dsm
        dw      127                     ;drm
```



```
        db      0c0h                    ;al0
        db      00                      ;al1
        dw      32                      ;cks
        dw      0                       ;off

        db      plw                     ;device

        db      41h-tlw                 ;(0) initbyte:
```

```
        db      040h            ;(1) mt, mfm-(fm)-cmd
        db      3               ;(2) n (voreinstellung 256 byte
        db      6               ;(3) eot (voreinstellung 256 byte
        db      0eh             ;(4) gpl (voreinstellung 256 byte
        db      30h             ;(5) gpl3
        db      0e5h            ;(6) Daten beim Formatieren
        db      25              ;(7) prae
        spez16

  endif
        ;fmtnr eq 8

        endm    ;format

;----- Protokollierung Assemblerlauf -----

include macro   file
   if1
        .printx * include file *
   endif
        maclib  file
        endm

printe  macro   txt,v1,i
        .printx * txt v1 i *
        endm

printh  macro   txt,v1
  if2
        .radix  16
        printe <txt>,%(v1),!H
        .radix  10
  endif

        endm

printd  macro   txt,v1
  if2
        printe <txt>,%(v1),!
  endif

        endm

size    macro   x
        local   plen,pmax,pvon,pbis
plen    aset    end&x-beg&x
pmax    aset    max&x
```



```
pvon    aset    pha&x
pbis    aset    pha&x+plen
        printh   <Laenge: >,plen
        printh   <Max:    >,pmax
        printh   <von     >,pvon
        printh   <bis     >,pbis
  if    plen gt pmax
        .printx *    zu gross !!!    *
        end
  endif

        endm
```