

# Gerätetreiber

Das OS des KC85/1 unterstützt leistungsfähig das Einbinden neuer Gerätetreiberrountinen für die sog. zeichenorientierten logischen Ein- /Ausgabekanäle. Vorbild für das im folgenden erläuterte Konzept ist das Betriebssystem CP/M.

Literatur:

- [mp 10/87 S. 311 ff.](#)
- „Betriebssystem Z9001“ Beschreibung + Quelltext des OS
- und weitere Quellen

**Kritik** (vp) Leider ist das I/O-Konzept nicht bis zum Ende durchdacht und implementiert:

- Das I/O-Byte liegt auf einer anderen Adresse als in CP/M üblich
- Es gibt keine Systemaufrufe (über CALL 5), um die Treiber direkt anzusprechen. Die Systemaufrufe nutzen nur die Treiberfunktionen Zeichenein- und -ausgabe
- Die Einbindung der Treiber ins I/O-Byte und Übergabe der Zeichenkettenadressen erfolgt nur beim Laden des Treibers, nicht über einen späteren Funktionsaufruf des Treibers
- Die Zuweisung des Treibers auf eine physische Gerätenummer kann nicht frei gewählt werden. Das macht der Treiber intern, das ist aber nirgends ersichtlich. Auch bei ASGN wird das nicht angezeigt.
- Die Adresse des aktiven Treibers ist nur über die interne Systemfunktion EXIO ermittelbar.
- Beim Warmstart (nach STOP) wird das I/O-Byte auf den Standardwert 01h zurückgesetzt.
- Im OS 1.1 ist BAT nicht nutzbar.
- Die Zeichenketten der nicht initialisierten phys. Treiber zeigen auf Adresse LOGDV (0FC24h). Dort steht 01 00, das ist zufälligerweise ein gültiger String, aber eigentlich müsste nur auf ein Nullbyte (leerer String) verwiesen werden.
- Obwohl als Zusatz zum Monitor mitgeliefert, missachtet der ZM das Assign-Konzept des OS fast vollständig (s.u.)
- So wie ASGN arbeitet und zu nutzen ist, hätte auf das Konzept des I/O-Byte und der 4 verschiedenen physischen Treiber verzichtet werden können. Mit jedem Aufruf von ASGN wird einfach der übergebene Treiber aktiv, egal, welche physische Nummer er tatsächlich hat.

## logische und physische Geräte

Das OS verwaltet logische und physische Geräte.

Logische Geräte sind z.B. die Zeichenausgabe auf Drucker. Welcher Drucker angeschlossen ist, spielt auf Anwenderprogrammebene keine Rolle, die zu druckenden Zeichen werden immer an das logische Geräte LIST: ausgegeben.

Es gibt folgende vier logische Geräte:

- CONST: Konsole (Tastatur + Bildschirm, d.h. Zeichenein- und -ausgabe),

- READER: (nur Zeicheneingabe, ursprünglich von Lochband),
- PUNCH: (nur Zeichenausgabe, ursprünglich auf Lochband),
- LIST: Drucker (nur Zeichenausgabe).

Zur Kommunikation mit diesen Geräten gibt es Systemfunktionen über CALL 5:

CALL 5, C = Rufnummer

Rufnr.	Name	Funktion
01	CONSI	Eingabe eines Zeichens von CONST
02	CONSO	Ausgabe eines Zeichens zu CONST
03	READI	Eingabe eines Zeichens von READ
04	PUNO	Ausgabe eines Zeichens zu PUNCH
05	LISTO	Ausgabe eines Zeichens zu LIST
09	PRNST	Ausgabe einer Zeichenkette zu CONST
10	RCONB	Eingabe einer Zeichenkette von CONST
11	CSTS	Abfrage Status CONST

In BASIC können diese Geräte über Kanäle #1..#4 angesprochen werden (allg. geht das nur mit verändertem Lutz-Eißner-Basic, s. [die\\_erweiterung\\_von\\_lutz\\_elsner](#))!

Für jedes logische Gerät können bis zu 4 physische Gerätetreiber geladen und im Speicher vorgehalten werden. Mit dem Systemkommando ASGN (s.u.) oder durch Setzen des I/O-Bytes (s.u.) erfolgt die Zuweisung, welcher physische Gerätetreiber aktuell einem logischen Gerät zugeordnet ist.

Mit

```
ASGN LIST:=LX86
```

wird beispielsweise der physische Druckertreiber „LX86“ aktiv dem logischen Gerät LIST: zugeordnet. Alle Druck-Ausgaben werden ab jetzt an den Druckertreiber „LX86“ weitergereicht.

## ASGN-Kommando

**ASGN** ist ein residentes Kommando des OS (von insgesamt 3). Die Aufrufform ist

```
ASGN [log. Gerät = phys. Gerät]
```

ASGN ohne Parameter gibt die aktuelle Belegung auf dem Bildschirm aus, z. B.

```
CONST:=CRT
READER:=
PUNCH:=
LIST:=
```

sofern noch keine zusätzlichen Treiber eingebunden worden sind. Die möglichen logischen Geräte wurden bereits erläutert, als phys. Geräte sind \*CRT\*, \*BAT\* oder Namen eigener Treiberrountinen möglich (max. 8 Buchstaben oder Zahlen).

Beispiel: `ASGN LIST:=LX86`

Wird eine solche Anweisung gegeben, so vollzieht sich im OS ein umfangreicher Mechanismus:

Zuerst wird eine Treiberoutine namens „LX86“ im Speicher (von oben nach unten) entsprechend den Aufrufregeln für transiente Kommandos gesucht. Ist die Suche erfolgreich, so wird die Steuerung zum Treiber übergeben (Initialisierungsroutine des Treibers), andernfalls wird angenommen, dass eine Treiberoutine vom Band nachgeladen werden soll, und es erfolgt die Ausschrift

```
start tape.
```

Nach dem Einlesen wird der Treiber an der im FCB spezifizierten Startadresse gestartet. Im allgemeinen sollte die Treiberoutine an das aktuelle Speicherende (RAM) platziert werden (über Zelle 36H zu ermitteln).

Dem OS müssen bei der Initialisierung einige Informationen übermittelt werden:

- Welchem log. Gerät darf die Treiberoutine zugeordnet werden?
- Die Startadresse des Treiberteils
- Auf welchem Platz der Zuordnungstabelle (s.u.) soll der Eintrag erfolgen (physisches Gerät 0...3)?
- Die Adresse des symbolischen Gerätenamens (8-Byte-String).

Zu beachten ist, dass physische Geräte mit der Nr. 0 jedem log. E/A-Kanal zugeordnet werden können (also auch der Console) und entsprechend aufgebaut sein müssen.

Ein physisches Gerät mit der Nr. 1 kann auch immer dem LIST-Kanal zugeordnet werden (Beispiel: residenter CRT-Treiber).

Beispielsweise wird ein Druckertreiber mit der Bezeichnung „LX86“ wie folgt in das OS eingebunden:  
Mit der Anweisung

```
ASGN LIST:=LX86
```

wird die Treiber-Routine eingebunden (ggf. von Kassette geladen) und das I/O-Byte entsprechend gestellt. Es erfolgt die Ausschrift

```
CONST:=CRT  
READER:=  
PUNCH:=  
LIST:=LX86.
```

Der Drucker kann jetzt mit CTRL/P zum Bildschirm parallel geschaltet werden (auch im BASIC) oder über den LIST-Kanal (Ruf-Nr. 5) direkt angesprochen werden.



Die meisten Treiber umgehen den Aufruf über ASGN. Statt

```
ASGN LIST:=LX86
```

wird der Druckertreiber dann mit

## LX86

geladen.

Ein guter Treiber (s.u.) sollte beide Varianten ermöglichen!

## Das I/O-Byte

Das I/O-Byte dient zur Verbindung der vier logischen mit physischen Geräten. Für jedes logische Gerät sind 4 physische Geräte wählbar. Die Zuschaltung eines dieser Geräte erfolgt über Änderung des I/O-Bytes. Zuvor müssen alle zu verwendenden Treiber zugewiesen und initialisiert werden. Das I/O-Byte ist in 4 Bereiche zu je 2 Bits unterteilt. Jeder Bereich, der einem logischen Gerät zugeordnet ist, kann einen Wert von 0 bis 3 beinhalten und damit logisches und physisches Gerät verbinden.

Aufbau des I/O-Bytes (HS-Adr. 0004, im CP/M üblicherweise Adr. 0003):

Bit	7	6	5	4	3	2	1	0
Log. Gerät	LIST	PUNCH	READER	CONST				

Das OS selbst beinhaltet zwei residente physische Treiberroutinen:

- CRT Consoltreiber
- BAT Batch-Treiber

CRT ist der Treiber für die Standard-Console, d. h. für das Fernsehgerät als Sichtgerät und die eingebaute Tastatur.

BAT ist ein Treiber, der das log. Gerät Console (CONST:) auf die Kanäle READER: (Eingabe) und LIST: (Ausgabe) aufteilt, also nur verwendbar ist, falls auch für READER: und LIST: entsprechende Treiber eingebunden wurden.

Alle anderen Plätze der Tabelle sind nach einem Kaltstart des OS (RESET oder POWER ON) mit dem Wert 0FFFFH belegt und zeigen dem OS damit einen nicht existenten Treiber an. Sollen eigene Treiberroutinen eingebracht werden, so kann das durch direktes Eintragen der Treiberadresse in den entsprechenden Tabellenplatz und zugehöriges Stellen des I/O-Bytes erfolgen.

Vom Monitor werden folgende physische Geräte realisiert:

log. Gerät	Gerätenummer	phys. Treiber
CONST:	1	CRT
	2	BAT
LIST:	1	CRT

```
System-Treiber, * ist aktiv, fehlende Treiber sind mit RET kurzgeschlossen
;      physische Gerätenummer
; log. Gerät  0    1    2    3
; -----
; CONST:  (0)      CRT*   BAT
; READER: (2)  *
```

```
; PUNCH: (4) *
; LIST: (6) * CRT
```

nach Reset ist IOByte = 01h, d.h.

```
CONST: = CRT
READER: =
PUNCH: =
LIST: =
```

Es sind jetzt folgende Zuweisungen möglich:

LIST: = CRT - Druckausgaben erfolgen auf den Bildschirm parallel zur normalen Ausgabe

Wenn Treiber für Reader und Punch installiert sind, ist noch erlaubt:

CONST: = BAT - Consoleneingabe vom Reader-Gerät, Konsolenausgabe erfolgt auf das Punch-Gerät

Die Treiberadressen der jeweiligen physischen Geräte liegen auf den Adressen EFC9H bis EFE8H. s.u. Adresstabelle der Gerätetreiber.

**Hinweis:** Es gibt symbolische Namen für die physischen Geräte, diese spielen i. Allg. keine Rolle, werden aber in der Literatur (und im Zusatzmonitor ZM) genutzt:

```
;      physische Gerätenummer
; log. Gerät   0   1   2   3
; -----
; CONST: (0)  TTY[C]  CRT[C]  BAT[C]  AUC
; READER: (2)  TTY[R]  RDR    UR1    UR2
; PUNCH: (4)  TTY[P]  PUN    UP1    UP2
; LIST: (6)   TTY[L]  CRT[L]  LST    UL
```

## Aufbau

Ein Treiber-Programm besteht aus mehreren Teilen

- OS-Kommandorahmen (enthält den Kommando-Namen, z.B. „LX86“)
- Kommando-Verteiler für die Treiber-Kommandos wie Eingabe, Statusabfrage und Initialisierung
- die interne Logik zur Umsetzung der Treiber-Kommandos (z.B. mit Sonderbehandlung von Steuerzeichen)
- die physischen Treiber zur Ansteuerung des Gerätes
- die Initialisierung der Systemzellen und die Parameterbereitstellung für ASGN

Treibereingangsparameter:

- A Kommando (siehe Tabelle) (für CONST, READER, LIST)
- C Zeichen (bei Kommando AUSGABE)
- DE Adresse bei Cursorkommandos

Treiberausgangsparameter:

- A Zeichen (bei Kommando EINGABE)
- CY Fehlerstatus
- HL,DE Adresse bei Cursorkommandos

Hinweis: Vordergrundregister müssen bei Eintritt in das Treiberprogramm nicht gerettet werden (erfolgt bereits im BOS)

Hinweis:

- TTY-Treiber (L=0) können beliebigen logischen Geräten zugewiesen werden
- CRT-Treiber (L=1) können CONST und LIST zugewiesen werden
- alle anderen nur dem im H-Register angegebenen logischen Gerät (sonst Fehlermeldung im ASGN-Kommando)

### Treiber-Kommandos

Code	Funktion	Eingang	Return
00	Abfrage Status		A Status 0 kein Zeichen bei Eingabegerät, nicht bereit bei Ausgabegerät sonst Zeichen liegt an bei Eingabegerät, (im installierten CRT-Treiber wird der Zeichencode übergeben)
01	Eingabe Zeichen		A Zeichen
02	Ausgabe Zeichen	C Zeichen	
03	Cursor löschen		
04	Cursor anzeigen		HL physische Cursoradresse
05	Abfrage logische und physische Cursoradresse		HL physische Cursoradresse DE logische Cursoradresse
06	Setzen Cursor auf logische Adresse	DE logische Cursoradresse	
07	Abfrage physische Cursoradresse		HL physische Cursoradresse
08	Setzen Cursor auf physische Adresse	DE physische Cursoradresse	
FF	Initialisieren/Rücksetzen des Gerätes		

Hinweis:

- werden verschiedene Kommandos durch den Anwender nicht benötigt, kann dafür ein Sprung zum Fehlerausgang des Treibers programmiert werden
- korrespondierende System- und Direktrufe sind dann nicht mehr verwendbar

## Treiberinitialisierung

Dem OS (bzw. dem ASGN-Kommando) müssen einige Informationen übermittelt werden:

- Welchem log. Gerät darf die Treiberroutine zugeordnet werden?
- Die Startadresse des Treiberteils
- Auf welchem Platz der Zuordnungstabelle soll der Eintrag erfolgen (physisches Gerät 0...3)?
- Die Adresse des symbolischen Gerätenamens (8-Byte-String).

Die Initialisierung sollte grundsätzlich über das ASGN-Kommando erfolgen. Eine automatische Initialisierung über die Nutzung des Kommandos und Direktzugriff auf entsprechende Systemzellen ist zu vermeiden.

Der Aufruf der Initialisierung im ASGN-Kommando erfolgt in der gleichen Weise wie der Start eines Anwenderprogrammes. Dabei kann es sich sowohl um ein transientes Kommando, als auch um ein Programm auf Kassette handeln. Ladbare Treiberprogramme auf Kassette werden durch das ASGNKommando selbständig eingelesen und zur Initialisierung gestartet. Derartige Programme sind in der Initialisierung mittels geeigneter RELOCATING-Routinen an das Ende des logischen RAM-Bereiches zu verschieben. Notwendiger Arbeitsspeicher ist gleichfalls dort einzurichten. Nach dem Verschieben ist der Zeiger des logischen RAM-Endes EOR (36H) auf eine Adresse vor dem Treiberprogramm zu stellen. Für jede Treiberroutine eines physischen Gerätes stehen im System-RAM (ab 280H) 4 Bytes Speicher zur Verfügung.

Die Treiberinitialisierung muß der ASGN-Routine folgende Parameter übergeben:

```
- CY Fehlerstatus nach Initialisierung
- H logische Gerätenummer
  0 CONST
  2 READER
  4 PUNCH
  6 LIST
- L physische Gerätenummer (0, 1 .... 3)
- BC Adresse der Treiberroutine
- DE Adresse einer Zeichenkette (symbolischer Geräteiname, wird nach der
Zuweisung durch ASGN angezeigt)
```



sollen Gerätetreiber auch für OS 1.1 nutzbar sein, muss man auf einen Bug des Systems Rücksicht nehmen!

Beispiel:

```
;PROM-Transientkommandoprogramm zur Initialisierung eines Druckers als
TTY-Gerät
    JMP     INIT
    DB     "DRUCKER "
    DA     0
INIT:   LD     HL,(EOR)      ;Adresse end of RAM (36H)
        LD     DE,RAME-RAMA ;benötigter Arbeitsspeicher
        SBC   HL,DE
        LD     (EOR),HL    ;neue log. RAM-Ende
```

```

    INC    HL
    LD     (RTTYL),HL    ;Zeiger auf Arbeitsspeicher (2B0H)
;Hardwareinitialisierung
    ....
    LD     H,6           ;log. Gerätenummer (LIST)
    LD     L,0           ;phy. Gerätenummer (TTY)
    LD     BC,OUTZ       ;Adresse der Zeichenausgabe
    LD     DE,TEXT
    or    a              ; CY=0,    kein Fehler
;
    ex    (sp), hl      ; ++ dieser Teil
    inc   hl             ; ++ ist nur bei OS Version 1.1
    inc   hl             ; ++ noetig (wegen eines Fehlers)
    ex    (sp), hl      ; ++ in der ASGN-Routine)
;
    ret
;
FEHL:    SCF            ;Fehler
        RET
;
;Zeichenausgabe
OUTZ:    ....
;
;RAM-Definitionen
RAMA:    ....
        ....
RAME:    EQU #

```

Beim eigenständigen Eintragen der Treiber unter Umgehung des ASGN-Kommandos gilt:

1. Treiberadresse muss in der Adresstabelle für Gerätetreiber eingetragen sein (0EFC9H-0EFE8H)
2. der Treiber muss eine 8 Byte lange Bezeichnung bereitstellen
3. der Treiber muss die Systemkommandos für Gerätetreiber umsetzen.

### Treiber für LIST (Druckertreiber)

Für einen Druckertreiber reichen folgende Treiber-Kommandos aus:

- Initialisieren/Rücksetzen des Gerätes (A=0FFh)
- Abfrage Status (A=0, Rückgabe 0 bei nicht bereit, sonst <> 0)
- Ausgabe Zeichen (A=2, Zeichen in C)

### Treiber für READER

Für einen Treiber für READER reichen folgende Treiber-Kommandos aus:

- Initialisieren/Rücksetzen des Gerätes (A=0FFh)
- Abfrage Status (A=0, Rückgabe 0, wenn kein Zeichen einzulesen ist, sonst <> 0)
- Eingabe Zeichen (A=1, Rückgabe Zeichen in A)



## Treiber für CONST

An die Treiber der für CONST möglichen Geräte (TTY, CRT, BAT, UC) werden besondere Anforderungen gestellt, da sie Routinen zur Eingabe, Ausgabe, Statusabfrage und Cursorsteuerung beinhalten müssen. Dem Treiber wird die zu erfüllende Aufgabe durch spezielle Kommandocodes im A - Register übergeben.

### Kommandocodetabelle

Beispiel:

```

;Eingangsverteiler für CRT-Treiber mit allen Funktionen
INC    A
JPZ    RESET    ;Initialisieren/Rücksetzen Gerät
DEC    A
JPZ    STAT     ;Status
DEC    A
JPZ    EING     ;Eingabe
DEC    A
JPZ    AUSG     ;Ausgabe
DEC    A
JPZ    LCUR     ;Löschen Cursor
DEC    A
JPZ    ACUR     ;Anzeige Cursor
DEC    A
JPZ    ADRLC    ;Abfrage log. und phy. Cursoradresse
DEC    A
JPZ    SLOGC    ;Setzen Cursor auf log. Adresse
DEC    A
JPZ    ADRPC    ;Abfrage phy. Cursoradresse
DEC    A
JPZ    SPHYC    ;Setzen Cursor auf phy. Adresse
FEHL:  SCF      ;unzulässiges Kommando
RET    ;Fehlerausgang

```

## Speicherzellen

Im Betriebssystem sind einige Speicherzellen für Treiber freigehalten. Anwenderprogramme dürfen daher den Bereich 280h..2BFh auch nicht nutzen, sondern erst ab 300h!

```

0004  IOBYT:    DB 1          ;I/O-Byte zur Zuweisung log.Gerät phys.
Gerät

;der folgende Speicherbereich ist durch den Nutzer für eigene Treiber
verwendbar
0280  RTTYC     BER 4          ;RAM TTY-Treiber für CONST
0284  RCRTC:    BER 4          ;RAM CRT-Treiber für CONST
0288  RBATC:    BER 4          ;RAM BAT-Treiber für CONST
028C  RUC:      BER 4          ;RAM UC-Treiber für CONST

```

```

0290 PTTYR: BER 4 ;RAM TTY-Treiber für READER
0294 RRDR: BER 4 ;RAM RDR-Treiber für READER
0298 RUR1: BER 4 ;RAM UR1-Treiber für READER
029C RUR2: BER 4 ;RAM UR2-Treiber für READER
02A0 RTTYP: BER 4 ;RAM TTY-Treiber für PUNCH
02A4 RPUN: BER 4 ;RAM PUN-Treiber für PUNCH
02A8 RUP1: BER 4 ;RAM UP1-Treiber für PUNCH
02AC RUP2: BER 4 ;RAM UP2-Treiber für PUNCH
02B0 RTTYL: BER 4 ;RAM TTY-Treiber für LIST
02B4 RCRTL: BER 4 ;RAM CRT-Treiber für LIST
02B8 RLST: BER 4 ;RAM LST-Treiber für LIST
02BC RUL: BER 4 ;RAM UL-Treiber für LIST

```

;Adreßtabelle der Gerätetreiber

```

EFC9 ATTYC: DA -1 ;Adresse TTY-Treiber für CONST
EFCB ACRTC: DA CRT ;Adresse CRT-Treiber für CONST
EFGD ABATC: DA BAT ;Adresse BAT-Treiber für CONST
EFCF AUC: DA -1 ;Adresse UC-Treiber für CONST
EFD1 ATTYR: DA -1 ;Adresse TTY-Treiber für READER
EFD3 ARDR: DA -1 ;Adresse RDR-Treiber für READER
EFD5 AUR1: DA -1 ;Adresse UR1-Treiber für READER
EFD7 AUR2: DA -1 ;Adresse UR2-Treiber für READER
EFD9 ATTYP: DA -1 ;Adresse TTY-Treiber für PUNCH
EFDB APUN: DA -1 ;Adresse PUN-Treiber für PUNCH
EFDD AUP1: DA -1 ;Adresse UP1-Treiber für PUNCH
EFDF AUP2: DA -1 ;Adresse UP2-Treiber für PUNCH
EFE1 ATTYL: DA -1 ;Adresse TTY-Treiber für LIST
EFE3 ACRTL: DA CRT ;Adresse CRT-Treiber für LIST
EFE5 ALST: DA -1 ;Adresse LST-Treiber für LIST
EFE7 AUL: DA -1 ;Adresse UL-Treiber für LIST

```

;Zeichenkettenadreßtabelle

```

EFE9 TXCON: DA PHYDV+2 ;Adresse einer Zeichenkette des aktuellen
CONST-Gerätes,

```

;Zeichenkette wird im ASGN-Kommando ausgegeben

```

EFEB TXRDR: DA PHYDV ;
EFED TXPUN: DA PHYDV ;entsprechend TXCON
EFEF TXLST: DA PHYDV ;

```

## BAT-Treiber

Der BAT-Treiber ist ein im OS integrierter CONST-Treiber, der das CONST-Gerät in den BATCH-Modus schaltet:

- Konsoleingabe erfolgt über das aktuelle READER-Gerät
- Konsolenausgabe erfolgt über das aktuelle LIST-Gerät
- READER und LIST müssen zuvor zugewiesen werden

Soll ein READER-Gerät Im Zusammenhang mit dem BATCH-Mode von CONST betrieben werden, so müssen im Treiber für das READER-Gerät (RDR, UR1, UR2) die Kommandos Eingabe, Statusabfrage und Initialisierung programmiert sein. Bei Treibern für das LIST-Gerät müssen die Kommandos Ausgabe, Statusabfrage und Initialisierung In jedem Falle programmiert werden.

Zuweisung

```
ASGN CONST:=BAT (Enter)
```

Die Treiber-Kommandos werden dabei wie folgt weitergeleitet:

```
Kommando A=1 → READER
A=FF → erst READER init, wenn fehlerfrei, dann LIST init
sonst → LIST
```

d.h., dass der List-Treiber die CONST-Treiber-Kommandos theoretisch alle implementieren müsste (bis auf A=1)!

## DEVICE-Kommando

2020 habe ich das Kommando DEVICE programmiert (im Mega-Flash-Paket enthalten), um die Beschränkungen von ASGN zu umgehen. Mit DEVICE erfolgt die Anzeige der Gerätetreiber-Tabelle und des I/O-Bytes. Geräte können aktiviert werden. Treiber können beliebig in der Gerätetreiber-Tabelle zugewiesen werden.

```
device                Anzeige Gerätetreiber-Tabelle
device log.gerät:=0..3  Zuweisen log. Gerät mit physischer Gerätenummer
+ Init.
                        error4 kein gültiger Treiber hinterlegt
device log.gerät:=treiber  Zuweisen Treiber, ggf. Laden (analog ASGN)
device log.gerät:=treiber 0..3 Zuweisen Treiber, dabei physische
Gerätenummer vorgeben
                        dabei ist Zuweisung an beliebiges logisches Gerät möglich
```

**Beispiel:** Anzeige der Tabelle nach Aufruf des Zusatzmonitors ZM30. Dieser installiert Treiber in der Gerätetreiber-Tabelle. Über das ZM-eigene A-Kommando oder über DEVICE gerät:=0..3 können diese aktiviert werden

```
>ZM30                Laden+Starten Zusatzmonitor ZM
  Z9001 MONITOR V2.0 (ZM30) 1986   Programmierung
=>AR=T                Zuweisung Reader:=Tape
=>AP=T                Zuweisung Punch:=Tape
=>B                   ZM Beenden
>ASGN                System-Kommando Assign. Anzeige der
Gerätezuweisungen

CONST :CRT
READER:TAPER
```

```

PUNCH :TAPEP
LIST :
>DEVICE Kommando DEVICE. Anzeige I/O-Tabelle

IOBYTE:29 0 1 2 3
CONST :01 3CE1 F8F1* F7B4 F8F1
READER:02 F8F1 30A0 311C* F8F1
PUNCH :02 F8F1 3058 3128* 30C3
LIST :00 F8F1* F8F1 30C3 3058
>DEVICE LIST:=1 Aktivierung List zugewiesenes Gerät Kanal 1

```

## CRT-Treiber

für KRT wurden von mir CRT-Treiber entwickelt. Es gab keine Vorbilder wie bei den Druckertreibern.

## ZM



der ZM ist ein schlechtes Vorbild in Sachen Gerätetreiber!

- Der ZM enthält ein eigenes ASGN-Kommando 'A'. Hier wird nicht alles wie im OS umgesetzt, so fehlen etwa Treibernamen. Obwohl nach Verlassen des ZM zugewiesene Treiber weiterhin aktiv sind, ist dies nicht bei ASGN erkennbar.
- Die Gerätenamen beim Kommando A entsprechen nicht denen des OS
- Der ZM überschreibt die komplette Adresstabelle der Gerätetreiber (ab 0EFC9h). Eigene vorher eingetragene Treiber werden damit aus dem System geworfen!
- Die Treiber unterstützen keine Treiber-Kommandos (Initialisierung, Statusabfrage etc.)
- Die Zuweisungsbeschränkung des CRT-Treibers wird nicht beachtet
- Die Treiber für TAPE sind keine zeichenorientierten Treiber und manipulieren gehörig das System (s. Sourcecode)
- Positiv, aber eben nicht zum OS-Konzept passend, ist, dass die Benutzung des I/O-Bytes logischer und mehr an CP/M angelehnt ist

Hinweis: Im Mega-Flash sind modifizierte Versionen enthalten, die sich an die Vorgaben halten und mit anderen Programmen zusammenarbeiten.

Im [Zusatzmonitor ZM](#) sind verschiedene Gerätetreiber enthalten. Den logischen Geräten Console, Reader, Punch, List können verschiedene physische Treiber zugeordnet werden. Im ZM sind physische Treiber für Tastatur mit Beep, SIF-1000 Lochbandleser und -stanzer, V24-Drucker und Kassettenzugriff enthalten.

Das Kommando A<logisches Gerät>=<physisches Gerät> dient der Zuweisung eines vorhandenen Gerätes zu einem logischen Gerät. Folgende Zuordnungen sind angebar:

<b>CONSOLE</b>	<b>READER</b>	<b>PUNCH</b>	<b>LIST</b>
AC=C für Konsole (mit Tastatur-Beep)	AR=C für Konsole	AP=C für Konsole	AL=C für Konsole
AC=V für Konsole (ohne Tastatur-Beep)	AR=P für LB-Leser (SIF-1000)	AP=P für LB-Stanzer (SIF-1000)	AL=V für Konsole
AC=B für BATCH	AR=T für Kassette (Tape)	AP=T für Kassette	AL=L für V24-Drucker (TD40)
AC=U für User-Konsole	AR=U für User-Read-Gerät	AP=U für V24-Ausgabegerät (TD40)	AL=U für SIF-1000-Ausgabegerät

From:  
<https://hc-ddr.hucki.net/wiki/> - **Homecomputer DDR**

Permanent link:  
<https://hc-ddr.hucki.net/wiki/doku.php/z9001/software/os/geraetetreiber>

Last update: **2025/03/23 11:34**

