Programme und Kommandos

Speicheraufteilung

Der U880 (Z80) adressiert einen Speicherbereich von 0000h - 0FFFFh (64 kByte). Den Speicher teilen sich Betriebssystem, Bildwiederholspeicher, Anwendungsprogramme und Daten.

Der Z9001 verwaltet den Speicher nicht selbst. Der Anwender muss selbst die Speicheraufteilung und -nutzung im Blick haben. Die Programme haben feste Speicherbereiche (Programmstandorte und Arbeitsspeicher). Gerade bei der Nutzung von Treiber-Programmen ist deshalb darauf zu achten, dass sich verschiedene Treiber und die restlichen Programme nicht in den Speicherbereichen überschneiden!

Standard-Aufteilung mit 2 gesteckten 16K-RAM-Modulen:

<ditaa noedgesep>

```
FFFF          +-----------+
              | BOS       |       Z9001 Betriebssystem
F000          +-----------+
              | BWS +FAS  |       Bild- +Farbattributspeicher
E800          +-----------+
              |           |
              | ROM-Modul |       ROM-Basic (oder IDAS-Modul o.a.)
              |           |
C000          +-----------+
              |           |
              | RAM-Modul |       16K-RAM-Modul
              |           |
8000          +-----------+
              |           |
              | RAM-Modul |       16K-RAM-Modul
              |           |
4000          +-----------+
              |           |
              | RAM       |       System-RAM
              |           |
0300          +-----------+
              | System    |       BOS-Arbeitsbereich
0000          +-----------+
```

</ditaa>

Wichtige Programme und ihre Speicherbereiche:

```
ZM30        3000-3DFF
GRAF        8E60-A7FF    (Grafik-Treiber f. BASIC)
EPSON       A400-AFFF
```

```
ZMA8        A800-B5FF
CRT40       B000-B2FF
USBX        B600-BFFF
V24     B800-BBFF
DOSX        BD00-BFFF    + 4000-57FF (Hintergrund-RAM)
```

Man erkennt z.B., dass ZMA8 und CRT40 nicht zusammen genutzt werden können.

Dateien

Programme können in ROM-Modulen enthalten sein (IDAS-Modul, ASM-Modul, BASIC-Modul u.a.). Diese Programme sind sofort nach dem Einschalten des Kleincomputers verfügbar und brauchen nicht erst geladen zu werden.

Sonstige Programme werden von Kassette (oder von anderen externen Speichermedien wie Diskette oder USB-Stick) in den Speicher geladen. Programme und Daten sind in diesem Fall in Dateien organisiert.

Das „Dateisystem“ einer Kassette ist ganz einfach aufgebaut: alle Dateien liegen hintereinander auf der Kassette. Der Anwender muss selbst den Überblick behalten, an welcher Position der Kassette welches Programm abgelegt ist. Dieses ist üblicherweise auf den Kassetteneinlegern vermerkt.

KLEINCOMPUTER robotron KC 87
Programmkassette R 0112

A
R + HANOI
R + FPLOT
R + MOSAIK
ZM 30
EPROM 2 A
EPROM 6 A
EPROM A 2
R + DEMO 1
R + DEMO 2
R + DEMO 3

B
R + DEMO 1
Text für R + DEMO 1
R + DEMO 2
Text für R + DEMO 2
R + DEMO 3
Text für R + DEMO 3

Piste 1

PROGRAMMKASSETTE R 0112

Auf der Seite A sind die ersten 7 Programme zweimal in der angegebenen Reihenfolge aufgezeichnet. Anschließend folgen die 3 Demo-Programme.

Kassetten-Dateien haben einen 128 Byte großen Anfang, der nicht zum Programm gehört (Block 0, = FCB), s. Kassetteninterface. In diesem sogenannten Kopfblock stehen Programmname, Typ und der Speicherbereich, in den das Programm geladen werden soll. Ausführbare Maschinencode-Programme haben den Typ COM, Assembler-Quelltexte den Typ TXT.

Hinweis: BASIC-Programme und -Daten haben ein eigene Kassettenroutinen, ein eigenes Dateiformat und verzichten auf diesen Kopfblock!

Laden

Um ein Programm von Kassette zu laden, gibt man einfach den Programmnamen (ohne Typ) am Prompt ein:

```
>ZM30
```

Es erscheint die Aufforderung

```
start tape
```

Nach Enter wird auf Kassette der nächste Kopfblock geladen. Steht in ihm derselbe Programmname wie eingegeben und als Typ COM, wird das Laden fortgesetzt. Andernfalls gibt es eine Fehlermeldung.

Nach dem Laden startet das Programm automatisch!

Kommandos

Um nicht jedesmal ein Programm von Kassette laden zu müssen, um es zu starten, oder um Programme zu starten, die im ROM-Modul stehen, enthalten Maschinencode-Programme ein oder mehrere **Kommandos**, die vom OS aus durch Eingabe des Kommandonamens gestartet werden. Nach Laden des Programms stehen die enthaltenen Kommandos dauerhaft bereit, bis der Speicherbereich durch ein anderes Programm überschrieben wird.

Meist sind der Programmname und (einziger) Kommandoname identisch. Es gibt praktisch keinen Unterschied zwischen Programm und Starten des Kommandos. Beim erstmaligen Aufruf wird das Programm geladen und gestartet, beim erneuten Aufruf wird das gleichnamige Kommando im Speicher gefunden und gestartet.

Programme wie Assembler von Kassette R 0121 u.a. enthalten dagegen mehrere Kommandos, die erst nach dem Laden des Programms zur Verfügung stehen. Hier muss zweistufig gearbeitet werden:

```
>ASM       Laden des Assembler-Programmpakets in den Speicher
start tape
...
>EDIT      Dieses Kommando ist erst nach Laden des Assemblers verfügbar. Es
startet den Editor.
>ASM       Dieses Kommando ist erst nach Laden des Assemblers verfügbar. Es
startet den eigentlichen Assembler.
```

Kommando ASM und Programmname ASM sind gleich[1)], bewirken aber unterschiedliches.

Wenn ein eingegebenes Kommando nicht im Speicher gefunden wird, versucht das Betriebssystem, ein gleichnamiges Programm mit Typ COM von Kassette zu laden.

Manchmal sind Programmname und Kommando NICHT gleich. Damit kann gezielt ein Laden des Programms vom Start des Kommandos unterschieden werden. Der Zusatzmonitor wird z.B. mit

```
>ZM30
```

von Kassette geladen, das Kommando zum erneuten Start ist jedoch

```
>ZM
```

Kommandos können **Parameter** übergeben werden. Diese sind durch mindestens ein Leerzeichen vom Kommandoname getrennt. Beispiel ist das System-Kommando TIME: Ohne Parameter wird die Zeit angezeigt, mit Parameter die Uhrzeit gesetzt.

```
>TIME 07:30    Uhrzeit setzen
>TIME          Uhrzeit anzeigen
```

[1)]

eigentlich ist der volle Programmname ASM.COM und damit nicht gleich dem Kommando

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

Permanent link:
https://hc-ddr.hucki.net/wiki/doku.php/z9001/software/os/kommandos?rev=1597043702

Last update: **2020/08/10 07:15**