# Programmieren

Für den Z9001 stehen diverse Programmiersprachen bereit:

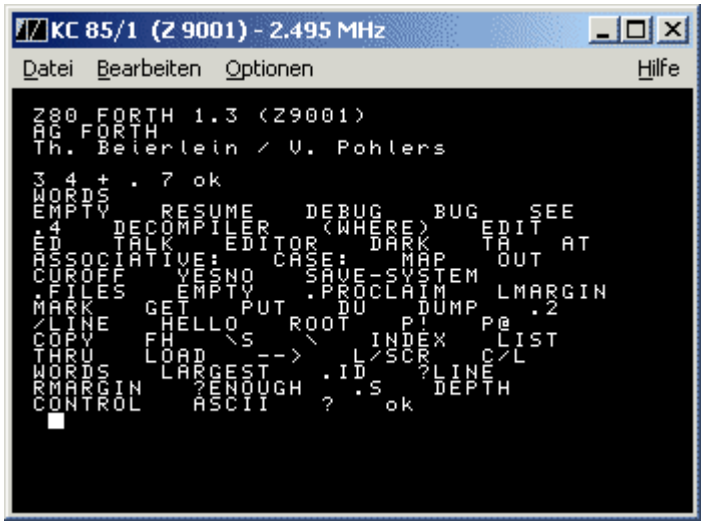
- BASIC
- FORTH
- PASCAL
- Assembler

## BASIC

Der Einsteiger sollte mit BASIC beginnen. Mit einem BASIC-ROM-Modul oder mit dem eingebauten BASIC des KC 87 2.x kann man gleich nach dem Einschalten mit der Arbeit beginnen. Das BASIC wird ausführlich im **Programmierhandbuch** und im **Anhang zum Programmierhandbuch** beschrieben (s. Handbücher und Downloads der Handbücher bei http://www.sax.de/~zander/.

## FORTH

Es gibt verschiedene FORTH-Versionen. Bekannt ist das f.i.g.-FORTH **FORTH** und das FG FORTH83 d. DDR **F83**. Beide sind im Mega-Flash-Modul enthalten. F83 ist deutlich umfangreicher und schneller.




Beim KC-Club Treffen 2012 gab es eine kleine Einführung ins FG-FORTH F83 z9001_f83.pdf.

## PASCAL

Für den KC gibt es das KC-PASCAL, eine Variante des bekannten Hisoft-PASCAL, sowie als 32K-Modul ein Turbo-Pascal-ähnliches Pascal von der TH Leipzig (s. weitere_module, das Handbuch findet man bei http://www.sax.de/~zander/).

# Assembler

Zur vollständigen und systemnahen Programmierung eignet sich Assembler. robotron bietet mit EDAS und IDAS gleich zwei Assembler an. Mit **ZSID** und **R80** stehen außerdem Debugger und Reassembler im Mega-Flash-Modul zur Verfügung.

Allerdings ist das Programmieren in Assembler um einiges komplexer und schwerer als in den „höheren“ Programmiersprachen.

Als Basis sollten unbedingt die Beschreibung des Betriebssystems incl. Betriebssystemlisting **robotron Betriebssystem KC 85/1 (Z9001)** studiert werden. Das Handbuch findet man bei http://www.sax.de/~zander/.

## Systemfunktionen

Zur systemunabhängigen Programmierung werden vom Betriebssystem 33 Systemrufe bereitgestellt. Diese werden analog CP/M über CALL 0005 aufgerufen. Die Auswahl des gewünschten Systemrufes erfolgt über das C-Register, dessen Inhalt den Systemruf adressiert. Verschiedene Systemrufe erwarten Eingabeparameter bzw. liefern Parameter zurück.

Eingabeparameter:

- Bytewerte im E -Register
- Wortwerte im DE-Register

Ausgabeparameter:

- Bytewerte im A -Register
- Wortwerte im BC-Register

Wichtige Systemrufe:

| Rufnr. | Name | Funktion |
|---|---|---|
| 01 | CONSI | Eingabe eines Zeichens von CONST |
| 02 | CONSO | Ausgabe eines Zeichens zu CONST |
| 09 | PRNST | Ausgabe einer Zeichenkette zu CONST |
| 10 | RCONB | Eingabe einer Zeichenkette von CONST |
| 11 | CSTS | Abfrage Status CONST |
| 17 | GETCU | Abfrage logische und pyhsische Cursoradresse |
| 18 | SETCU | Setzen logische Cursoradresse |

## Der OS-Rahmen

Damit eigene Programme vom OS aus gestartet werden können, wird ein spezieller Code benötigt, der sogenannte OS-Rahmen. Damit erscheinen Programme als transiente Kommandos im OS und könne über den Programmnamen aufgerufen werden. Außerdem können Parameter übergeben werden (s. z.B. Code von OS-SAVE).

Das Kommando muß auf einer integralen 100H-Grenze (300h ... 0BF00h) beginnen. Es können beliebig viele Kommandos in einem OS-Rahmen angegeben werden.

```
    ORG     xx00h

    JP      AUSF          ;Sprung zur Kommandoausführung1
    DB      'NAME     ',0    ;Kommandoname1 (im OS-Mode einzugeben)
                  ;8 Zeichen, ggf. mit Leerzeichen auffüllen, Null-Byte
    JP      AUSF2          ;Sprung zur Kommandoausführung2
    DB      'NAME2    ',0    ;Kommandoname2 (im OS-Mode einzugeben)
    ...
    DB      0          ;Kennzeichen OS-Rahmen Ende

AUSF:    ...
```

### Beispiel

Folgendes Programm gibt den Text „Hallo User!“ auf den Bildschirm aus. Das Programm wird mit dem Kommando HALLO gestartet.

```
    ORG     300h

    JP      HALLO
    DB      'HALLO    ',0
    DB      0

HALLO:  LD      DE, TEXT     ; Adr. des Textes
    LD      C, 9         ; PRNST
    CALL     5
    RET

; auszugebender Text
TEXT:   DB      'Hallo User!'
    DA      0A0DH          ; CRLF
    DB      0          ; Ende der Zeichenkette
```

# Programmerstellung am PC

Bei großen Programmen ist es leichter diese am PC zu schreiben und zu assemblieren. Ich nutze dafür den arnold-assembler. Kleine in Perl geschriebene Hilfstools unterstützen den Prozess und erzeugen z.B. gleich tap-Dateien, die im Emulator geladen werden können oder mit KCSAVE kcsave.rar als Audiosignal am realen KC geladen werden können.




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

Permanent link:
**https://hc-ddr.hucki.net/wiki/doku.php/z9001/programmieren?rev=1364312020**

Last update: **2013/03/26 15:33**