

M80,L80,CREF

Der Macroassembler von Microsoft ist ein Toolpaket mit enormen Möglichkeiten. Der Macroassembler erzeugt aus einer Assemblerquelldatei DEMO.MAC eine Maschinencode-Datei DEMO.REL. Diese wird mittels Linker in eine ausführbare Maschinencode-Datei DEMO.COM umgewandelt.

Größere Projekte bestehen aus mehreren Assemblerquelldateien. Diese werden einzeln mit M80 übersetzt und anschließend mit dem Linker in eine ausführbare Maschinencode-Datei umgewandelt. Mehrere REL-Dateien können mit LIB80 zu einer Bibliothek *.LIB zusammengefasst werden. Aus einer solchen Bibliothek entnimmt der Linker nur die benötigten Teile, es wird also nicht immer alles dazugelinkt.

einfaches Assemblieren und Linken von DEMO.MAC:

```
M80 =DEMO/L
L80 /P:0100,DEMO,DEMO/N/E
```

Neukompilieren von CP/A (Achtung: es wird ein anderer Linker genutzt!)

```
M80 BIOS.ERL=BIOS/X
LINKMT @OS=CPABAS,CCP,BDOS,BIOS/P:xxxx
```

MACRO-80

14K HS; 1000 Zeilen/min.

Dateitypen:

REL	verschiebliche Objektdatei
PRN	Druckdatei
MAC	Quellcodedatei M80 (132 Byte; kleine u. grosse Buchstaben; Zeilennumerierung erlaubt)
CRF	Symbolnachweisdatei
LIB	Modulbibliothek

Kommando:

```
M80 [[<objfile>],[<prnfile>]]=<sourcefile>[/switch[/...]]
```

Jeweils vor Filenamen Device-Angabe erlaubt:

```
A: , ... , LST: , TTY: , RDR:
```

Schalter:

```
/O Ausdrucken von Oktalzahlen
```

/H	Ausdrucken von Hexazahlen (Standard)
/R	Erzwingen Objekdateierzeugung
/L	Erzwingen Druckdateierzeugung
/C	Erzwingen Symbolnachweisdatei (kann auch im Kommando angegeben werden [<prnfile>/C])
/Z	Z80-Opcode (Standard)
/I	INTEL-8080-Opcode
/P	Anlegen eines speziellen 256-Byte-Stack (nur bei Fehlern benutzen)
/M	DS-Bereiche mit '00' vorloeschen
/X	Fehlerdruck unterdruecken (falls kein .SFCND, .LFCND)

Linker LINK-80

Kommando:

L80 inputfile[switch][,...][,outputfile[switch]]

Jeweils vor <objfilei> Device-Angabe erlaubt.

z.B.: L80 test/d:4000,mylib/s,test/n/e

Schalter:

1. Auf Kommandostufe

/G Ausfuehrung beginnen
 /M Auflisten Globalbezugnahmen
 /P:<progbegin> - Festlegen Programmbasisadresse
 z.B. /P:200,OTTO - OTTO wird auf 200H gelinkt

2. Nach Dateinamen

<file>/S - Durchsuchen LIB-Datei und Laden der erforderlichen

Bibliotheksmoduln

<file>/N - Eroeffnen <file>.COM
 <file>/E - Laden <file>.REL, Ausgabe auf <file>.COM, LINK-80-Ende

3. Am Ende der Kommandokette (bei global wirksamen Schaltern)

.../M/E - Globalbezugnahmen drucken, Retten COM-Datei, Ende

/R Reset; erneuter Start
 /E oder /E:<globalsymb> Ende; <globalsymb> gibt Entry an
 /G oder /G:<globalsymb> Programm ausfuehren
 /P:<addr> Festlegen Programmbasisadresse (Standard: 103H)
 /D:<addr> Festlegen Datenadresse (Standard: vor Programm)
 /U Ausdrucken Anfang/Ende des Programms
 /M Wie /U; zusaetzlich Globalsymbole
 /S Durchsuchen der LIB-Datei
 /X Wenn zuvor /N, Ausgabe im HEX-Format
 /Y Erzeugt SYM-Datei fuer Fehlersuchhilfen SID,ZSID

LIB80

Zusammenketten von Moduln zu einer .REL-Datei

Kommando:

```
LIB80 <libfile>=<<modullist>> | <datei><<modullist>> | <datei>
<modullist>:
- Modulname (z.B. SIN/COS)
- Modul davor/dahinter (z.B. SIN-1/SIN+1)
- bis/nach/dazwischen (..SIN/SIN../SIN..COS/SIN+1..COS-1)
```

Schalter:

```
/U - Drucken undefinierter Symbole bei Ein-Pass-Durchlauf (<libfile>/U)
/L - List-Ausgabe (z.B. auch <libfile>/L)
/C - Create Moduln und Symboldefinitionen (ueber LIB starten)
/E - LIB-80 Ende. Zerstoert Bibliothek, wenn kein Neuaufbau !!!
    Sonst ^C verwenden!
/R - Umbenennen .LIB in .REL
```

Beispiel:

```
LIB-Aufbau: LIB
            *TRANSUB=SIN,COS,TAN
            *EXP
            *^C
Listing:    LIB * TRANSUB.LIB/U
            *TRANSUB/L
            ^C
```

CREF-80

Ausdrucken Crossreferenz-Datei vom Assembler

Kommando:

```
CREF80 [<listfile>]=<sourcefile>
<listfile> muss vom Typ .LST sein!
```

From:

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

Permanent link:

<https://hc-ddr.hucki.net/wiki/doku.php/cpm/m80?rev=1280998279>

Last update: **2010/08/04 22:00**

