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.

ASM,LINK,LIB

Unter SCP1520 gab es "gleichwertige" Tools als ASM/LINK/LIB.

Eine ausführliche Beschreibung gibt es in

 asm.pdf SYSTEMHANDBUCH SCP, Anleitung fuer den Programmierer, Teil II -Assemblerprogrammierung

Ersetzt man SCP durch CP/M, ASM durch M80 usw., hat man ein sehr gutes Handbuch für das Toolpaket von Microsoft.

Beispiel

einfaches Assemblieren und Linken von DEMO.MAC:

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

Wird der Linker ohne /P: aufgerufen, wird am Codeanfang ein Sprung zum Programmbeginn eingefügt, der Code wird also um 3 Byte länger! Deshalb wird oftmals ein anderer Linker wie LINK 1.31 von Digital Research oder der in der DDR verbreitete Link/MT+ Release 5.5 genutzt, da diese etwas einfacher in der Syntax sind:

einfaches Assemblieren und Linken von DEMO.MAC mit LINK:

M80 =DEMO/L LINK DEMO

Neukompilieren von CP/A:

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:

| /0 | Ausdrucken von Oktalzahlen |
|----|---|
| /H | Ausdrucken von Hexazahlen (Standard) |
| /R | Erzwingen Objektdateierzeugung |
| /L | Erzwingen Druckdateierzeugung |
| /C | Erzwingen Symbolnachweisdatei (kann auch im Kommando angegeben werden [<prnfile>/C])</prnfile> |
| /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 .SFCOND, .LFCOND) |

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:rogbegin> - Festlegen Programmbasisadresse
                     z.B. /P:200,0TTO - 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:<globsymb>
                         Ende; <globsymb> gibt Entry an
/G oder /G:<globsymb>
                         Programm ausfuehren
/P:<addr>
                         Festlegen Programmbasisadresse (Standard: 103H)
/D:<addr>
                         Festlegen Datenadresse (Standard: vor Programm)
/U
                         Ausdrucken Anfang/Ende des Programms
                         Wie /U; zusaetzlich Globalsymbole
/M
                         Durchsuchen der LIB-Datei
/S
                         Wenn zuvor /N, Ausgabe im HEX-Format
/X
/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
```

Last update: 2011/09/25 15:38

*^C

Listing: LIB * TRANSUB.LIB/U

*TRANSUB/L

^C

CREF-80

Ausdrucken Crossreferenz-Datei vom Assembler

Kommando:

CREF80 [<listfile>]=<sourcefile>
 LST sein!

From:

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

Permanent link:

https://hc-ddr.hucki.net/wiki/doku.php/cpm/m80?rev=1316965117

Last update: 2011/09/25 15:38

