

Dateien von einem Laufwerk auf das andere kopieren oder aneinanderhängen. PIP läßt eine Vielzahl von Parametern zu, um die Daten bei der Übertragung zu manipulieren.

PIP kann auf zwei Arten aufgerufen werden: einmal ohne die Angabe von Parametern, dann meldet sich PIP mit dem Zeichen „*“ und ist für eine Eingabe bereit. Wird ein Parameter angegeben, so wird dieser sofort interpretiert und ausgeführt. Die Grundform lautet:

zieldateiname = queldateiname
oder
zieldateiname = queldateiname 1,
queldateiname 2...

Als Quell- und Zielnamen können neben normalen Dateien auch logische Geräte angegeben werden. Beim Ziel sind dies z. B. „CON:“, „PUN:“, „LST:“, „PRN:“ (mit expandierten Tabulatoren und Seitenvorschub alle 60 Zeilen) und „OUT:“ (eigene Anpassung auf 106H), als Quelle CON:, RDR: und INP: (eigene Anpassung auf 109H). Hinter den Dateinamen kann ein weiterer Parameter in eckigen Klammern folgen:

B (Block mode), Dn (Löschen nach ntem Zeichen), E (echo mode), F (Formfeeds entfernen), H (Intelhexcheck) I (:00 ignorieren), L (in Kleinbuchstaben wandeln), N (Zeilennummern ausgeben), O (Ob-

jektdatei-Kopie), Pn (Seitenvorschub alle n Zeilen), Qs↑z (Kopierstop, wenn String s bis CTRL-Z gefunden), Ss↑z (Kopierstart, wenn String gefunden), Tn (Tabulatoren auf n expandieren), U (Kleinbuchstaben in große umwandeln), V (Prüflesen bei Diskettenbetrieb), Z (Paritätsbit auf 0 setzen).

Mit „PIP B:*.*=A:*.*“ werden alle Dateien von Laufwerk A auf Laufwerk B kopiert. Bild 12 zeigt ein weiteres Beispiel.

Der CP/M-Texteditor

Auch ein kleiner Texteditor wird auf der Systemdiskette mitgeliefert. Damit lassen sich die Quellprogramme von der Tastatur eingeben. Zum Editor gehört ein eigenes Handbuch und er kann hier nur grob beschrieben werden. Der Aufruf erfolgt mit ED name, also z. B. ED TEST.ASM. Es wird, falls nicht schon vorhanden, eine neue Datei angelegt. Dann meldet sich der Editor mit dem Zeichen „*“ und wartet auf Befehle. Es kann nun z. B. der Befehl I (Insert = Einfügen), gefolgt von beliebigen Textzeilen, eingegeben werden. Mit CTRL-Z wird der I-Befehl beendet.

Der Editor besitzt intern eine Textmarke; dort merkt er sich, an welcher Stelle des

Textes er gerade arbeiten soll. Nach dem I-Befehl steht dieser Zeiger am Ende der gerade eingegebenen Daten. Soll nun der Text ausgegeben werden, so kann z. B. mit dem Befehl B an den Anfang des Textes zurückgegangen werden. Mit T30 lassen sich von hier an 30 Zeilen auf der Console ausgeben. L30 verschiebt den Textzeiger um 30 Zeilen; mit -L30 kann er auch wieder zurückgesetzt werden. Bild 13 zeigt ein paar Befehle. Verläßt man den Editor mit E, so werden automatisch die eingegebenen Texte auf Disk abgespeichert. Soll erneut editiert werden, so genügt es, ED TEST.ASM aufzurufen, und mit dem Befehl 200A werden die ersten 200 Zeilen (oder soviel wie existieren, falls es weniger sind) eingelesen. Dann kann der Text modifiziert werden, und mit E wird der Editor wieder verlassen. Die Datei, die vor der Editierung existierte, hat nun die Extension .BAK erhalten und die neuerzeugte .ASM, wie es mit ED TEST.ASM angegeben wurde. Die Datei TEST.BAK dient dazu, einem Datenverlust durch fehlerhafte Bedienung vorzubeugen, so daß immer eine Vorgängerversion erhalten bleibt. Sie kann mit dem Befehl REN in TEST.ASM umbenannt werden, nachdem diese zuvor gelöscht wurde.

(Fortsetzung folgt)

```
B>aidump test.com
0000 11 0D 01 0E 09 CD 05 00 0E 00 CD 05 00 53 74 61
0010 72 74 20 65 69 6E 65 73 20 41 6E 77 65 6E 64 65
0020 72 70 72 6F 67 72 61 60 6D 73 00 0A 24 00 00 00
0030 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0040 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0050 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0060 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
0070 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

B>
Bild 9. Erzeugte .COM-Datei mit DUMP ausgegeben
```

```
B>test
Start eines Anwenderprogramms
B>

B>:a:stat test.*
      Recs  Bytes  Ext  Acc
      3      1k    1  R/W  B:TEST.ASM
      1      1k    1  R/W  B:TEST.COM
      2      1k    1  R/W  B:TEST.HEX
      5      1k    1  R/W  B:TEST.PRN
Bytes Remaining On B: 202k
B>
```

Bild 10. Start der Datei TEST.COM

Bild 11. Dateien, die für Bild 10 benötigt wurden

```
B>:a:PIP lst:=test.asm[nt8]
1: ; Kleines Testprogramm fuer den ASM
2: ;
3:          ORG 100H          ;start in TPA
4: BDOS     EQU      5
5: START:   LXI D,TEXT      ;Adresse des Textes
6:          MVI C,9         ;Funktion STRING AUS
7:          CALL BDOS      ;aufruf ueber CP/M
8:          MVI C,0         ;Warm Start
9:          CALL BDOS
10:         ;kommt nicht hierher
11:
12: TEXT:   DB 'Start eines Anwenderprogramms'
13:         DB 0DH,0AH
14:         DB '$'          ;ende des Strings
15:
16:         END
17:
18:

B>
```

Bild 12. Anwendung des PIP-Befehls

```
B>:a:ed test.asm
: *200a
1: *5t
1: ; Kleines Testprogramm fuer den ASM
2: ;
3:          ORG 100H          ;start in TPA
4: BDOS     EQU      5
5: START:   LXI D,TEXT      ;Adresse des Textes
1: *-b
: *-10t
9:          CALL BDOS
10:         ;kommt nicht hierher
11:
12: TEXT:   DB 'Start eines Anwenderprogramms'
13:         DB 0DH,0AH
14:         DB '$'          ;ende des Strings
15:
16:         END
17:
18:
: *e

B>
```

Bild 13. Eingabe eines kurzen Assembler-Quellenprogramms mit dem Texteditor ED