

Raoul O. Koerber

NDR-Klein-Computer und CP/M 3

Mehr Komfort für den Z80-Ausbau

CP/M ist für 8-Bit-Rechner sicherlich das universellste und am weitesten verbreitete Disketten-Betriebssystem. Mit seiner neuen Plus-Version (CP/M 3) bietet es neben allen bekannten Möglichkeiten doch noch verschiedene Vorteile wie z. B. größere Geschwindigkeit und größere TPA. Alleine der wesentlich verbesserte Befehlszeilen-Editor ist ein Umrüsten auf CP/M 3 wert.

Gleich zu Anfang: Die besprochenen Assemblerprogramme sind so lang, daß deren Abdruck in mc zu viel Platz weggenommen hätte. Wer die Listings haben möchte, kann sie beim Elektronikladen in Detmold anfordern. Sie kosten als Ringordner 29.80 DM.

Bei der Implementation von CP/M 3 auf dem NDR-Klein-Computer waren einige Dinge zu berücksichtigen, die die Angelegenheit nicht ganz so einfach machten. Allem voran mußte die Kompatibilität zu CP/M 2 gewahrt bleiben. Dazu mußte der Boot-Monitor Flomon (von Rolf-Dieter Klein) beibehalten werden. Dies gilt ab Version 1.6 von Flomon (in welcher eine unpassende Speicherveränderung im Bereich 0F000H-0F400H beseitigt wurde). Mit diesem Boot-Programm kann CP/M 2 wie vorgesehen gebootet werden.

Die Unterprogramme von Flomon, die über die Sprungleiste ab 0F000H aufgerufen werden können, sind leider nur sehr bedingt oder garnicht CP/M-3-tauglich. Der Grund liegt alleine darin, daß keinerlei Maßnahmen zur Bankumschaltung beim Einspung in den EPROM-Bereich vorgesehen wurden, auch die Diskettenroutinen nehmen mit ihrem Interrupt darauf keine Rücksicht. Um dies alles zu umgehen, wurde ein Hilfsmonitor geschrieben, der alle Voraussetzungen für einwandfreies Arbeiten gewährleistet: Elmon

Elmon wird als Zusatz-EPROM in den zweiten Sockel der Bank-Boot-Karte (ab

Adresse 2000H) gesteckt und aus Flomon mit dem Befehl <3> aufgerufen. Elmon transferiert seinen Ein/Ausgabeteil (MONIO – Monitor Input-Output) nach Aufruf in den Speicherbereich 0FC00H..0FFFFH. Wie bei Flomon werden auch hier die Tastatur-Eingabe und die Video-Ausgabe über die Flomon-Routinen (im EPROM) getrieben. Die Konsole ist also kompatibel zur CP/M-2-Version (natürlich auch im Grafik-Teil). Da der Ein-/Ausgabetreiber MONIO nur knapp 1 KByte groß ist, wurde zur besseren EPROM-Ausnutzung neben der reinen Konsolen-Anpassung noch ein kompletter Monitor-Teil eingebunden, mit dem Speicherplätze betrachtet, verändert, verschoben oder geprüft, mit dem der Inhalt von Ports gelesen oder verändert und mit dem Programmteile disassembliert, im Einzelschritt durchgeprüft und auch neu assembliert werden können.

Dieser Monitorteil belegt den Speicherbereich 0F400H...0FBFFH. Der Speicherbereich 0F000H...0F3FFH wird als Diskettenpuffer verwendet, während der Bereich 0E000H...0EFFFH als Fenster zum Nachladen des eingebauten Assembler- bzw. Debuggerbauteiles verwendet wird.

Die Befehle von Elmon sind weitgehend SID-kompatibel (das ist der Debugger, den Digital Research mit CP/M 3 liefert) und so aufgebaut, daß sie auch vom Anfänger leicht zu handhaben sind, wenn er in die 'Geheimnisse' der Assembler-

sprache eindringen will, aber auch den Könner unterstützen, wenn er schnell einmal neue Systemkarten zum Laufen bringen will, aber ihm aus irgendwelchen Gründen CP/M nicht zur Verfügung steht.

Der Befehlssatz von Elmon:

A aaaa

Assembliere ab Adresse aaaa.

B

Boote CP/M.

C aaaa

Aufruf (call) des Programmes mit Startadresse aaaa.

D aaaa

Dump (Speicherausdruck) ab Adresse aaa0 hexadezimal und in ASCII.

F aaaa eeee bb

Füllen des Speicherbereiches aaaa bis eeee mit dem hexadezimalen Wert bb.

G aaaa brk brk

Go (gehe) nach Adresse aaaa.

H aaaa eeee

Hexmathematik, ergibt Summe und Differenz von aaaa und eeee hexadezimal und binär.

I pp

Input, liest Port pp und gibt Inhalt binär auf Konsole aus.

L aaaa

Listet Befehlsfolge ab aaaa.

M aaaa eeee zzzz

Move (verlagere) den Inhalt des Speicherbereiches aaaa bis eeee nach zzzz.

O pp bb

Output (Ausgabe) des Hexwertes bb auf Port pp.

P nn

Umschalten auf Page (Bank) nn.

Q

Quick Check, schneller zerstörungsfreier Speichertest.

S aaaa

Setzen der Speicherstelle aaaa (und folgende) auf Hex- oder ASCII-Werte.

T

Trace (Verfolgung in Einzelschritten), dazu zuerst mit g aaaa brk1 initialisieren! Nur im RAM anwendbar.

U
Wie T-Befehl, es werden jedoch keine CALLs verfolgt.

V aaaa eeee zzzz
Vergleichen des Speicherraumes aaaa bis eeee mit dem Speicherraum ab zzzz.

X (Registername Wert)
Registerinhalt ausgeben oder ändern (nach Trace oder Haltepunkt).

Z bb1 bb2 bb3
Zap, sucht im gesamten RAM-bereich (außer im eigenen Puffer) nach der Hex-Folge bb1 bb2 bb3..., mit Z ASCII-String wird wie oben nach der ASCII-Zeichenfolge ASCII-String gesucht.

Zusätzlich kann mit CTRL-P die Druckerschnittstelle zugeschaltet werden. Es können so Assemblerprotokolle und Disassemblerausdrucke sowie Speicherinhalte mitgedruckt werden.

Unter CP/M 3 wird von Elmon nur noch der Ein-/Ausgabe-Teil (Mono) mit seiner Sprungleiste ab Adresse 0FC00H verwendet. Die Konsolen-Treiber werden nach korrekter Bankbehandlung in gleicher Weise wie bei Flomon direkt aus dem EPROM-Bereich aufgerufen, die GDP-Treiberroutine benötigt dazu auch nach wie vor einen RAM-Pufferbereich auf der Bank-Boot-Karte. Um das Blinken des Cursors zu gewährleisten, werden auch die Konsolen-Status- sowie Konsolen-Eingabe-Routinen aus dem EPROM-Bereich verwendet.

Die Einsprungleiste im MONIO-Teil von Elmon ist nicht kompatibel zu Flomon, da die Seitenadresse (0FCH anstelle 0FOH) geändert wurde, bestand hierzu auch keine Veranlassung mehr. Anwender von CP/M 2 finden selbstverständlich ebenfalls alle Routinen, um ihr System ebenfalls an den MONIO-Teil anbinden zu können. Alle notwendigen Einsprungadressen und die dazugehörenden Unterprogramme sind aus dem Listing zu ersehen (MONIO.PRN).

Leider gibt es in der Zwischenzeit für den NDR-Klein-Computer viele Programme, die direkt auf die Flomon-Sprungleiste zugreifen. Es ist schon schlimm genug, wenn unnötig auf die BIOS-Sprungleiste zugegriffen wird, aber Zugriffe auf Monitor-Routinen sind von vorneherein immer CP/M-unkompatibel und sollten tunlichst vermieden werden. Derartige Programme können unter CP/M 3 und unter Elmon natürlich nicht mehr funktionieren – hier kann

nur ein manuelles Suchen der entsprechenden Einsprünge und entsprechenden Abändern helfen.

Die Hardware-Voraussetzungen zum Betrieb von CP/M 3 mit dem NDR-Klein-Computer

Die minimalen Hardware-Voraussetzungen für ein CP/M-3-System mit Karten aus der NDR-Klein-Computer-Serie sehen wie folgt aus:

Vollausbau-CPU-128k, dynamisch RAM (bis 1MByte möglich), mit 8 KByte COMMON-Bereich oder zwei Stück ROA64 (128 KByte RAM) mit CP/M-3-Anpassung. Bank-Boot-Karte, KEY, TAST, GDP und FDC oder FLO2 (FLO2 mit kleinen Änderungen). Unterstützt werden darüberhinaus: Je eine SER(ielle Karte) für AUX und die serielle Druckerschnittstelle sowie einmal SER als Terminalschnittstelle (anstelle oder zusätzlich zu KEY und GDP), eine Centronics-Schnittstelle.

Angepaßt wird CP/M 3 für Mini-Laufwerke (5 Zoll) und 3,5-Zoll-Laufwerke mit 80-Track auf zwei Seiten (800 KByte). Als weitere Laufwerke (jedoch als A-Laufwerk nur in Sonderversion) sind 80-Spur-, einseitig und 40-Spurlaufwerke ein- oder zweiseitig einfach einzubinden. Darüberhinaus kann ein 8-Zoll-Laufwerk einfacher Dichte (Standard-CP/M) eingebunden werden. Stehen mehr als zwei Speicherbänke von je 64 KByte Speicher zur Verfügung, kann eine RAM-Disk eingebunden werden.

Die wesentlichste Einschränkung (bezüglich kompatibler Hardware) ist der 8-KByte-Common-Bereich für die benutzten Speicherkarten. Es ist darunter ein Speicherbereich (von 0E000H-0FFFFH) zu verstehen, der auf allen eingeschalteten Bänken erhalten bleibt, da über diesen Bereich der (gebankte) Programmtransfer stattfindet bzw. in diesem Bereich Programmteile liegen, auf die von allen Bänken aus zugegriffen wird. Es besteht prinzipiell die Möglichkeit,

Jeder macht es anders

Es gibt verschiedene Möglichkeiten der Seitenumschaltung bei zweiseitigen Laufwerken. Ebenso wie bei der Sektorgröße und dem benutzten Skew-Faktor wurden hier nie Versuche einer Einigung zwischen den verschiedenen Geräteherstellern unternommen. So liegt hier nach wie vor ein beliebtes Betätigungsfeld der Systementwickler, um den Diskettenzugriff optimal schnell zu bekommen (das ist immerhin legal) oder auch um ein Lesen von Disketten in Fremdsystemen möglichst zu erschweren, falls es nicht gar darum geht, Programme vor dem Kopieren zu schützen.

Die derzeit verbreitetste Art der Seitenumschaltung geht nach dem Motto, alle (logisch) geraden Spuren liegen auf Seite 0 und alle (logisch) ungeraden Spuren liegen auf Seite 1 einer Diskette. Die Berechnung, welche Seite dran ist, wird (durch einfaches Dividieren der Spurnummer) mit dieser Methode recht einfach. Beim Aufbau des DPBs ist aber zu berücksichtigen, daß jede Spur (bei einem zwei-seitigen Laufwerk) zweimal vorhanden ist, aus einem 80-Spur-Laufwerk wird also ein 160-Spur-Laufwerk und aus einem Offset von drei Spuren wird ein Offset von sechs. Dies ist eine sehr beliebte Fehlerquelle beim Aufbau der DPBs, aber auch Benutzer von Disketteninspektionsprogrammen (z.B. DU,DUTIL,DISKED usw...) können da schon etwas ins Schleudern kommen, wenn sie das Inhaltsverzeichnis (nach Angaben von STAT bzw. SHOW) auf Spur 2 suchen, aber erst auf Spur 4 finden, denn die letzte Entscheidung, was ein logischer und was ein physikalischer Track ist, wird erst in der Schreib-/Lese-Routine des BIOS getroffen.

Was den Skew angeht, so läßt sich auch hier viel Schlimmes anstellen. Viele Diskettenkopierprogramme (die angeblich dutzende von Formaten können) kommen hier (durch falsche Anwendung) zu dem Schluß, daß ein Skew-Faktor, der bereits beim Formatieren eingegeben wurde (und mit READ-TRACK zurück gelesen wurde), durch einen Skew in der XLT-Tabelle ersetzt werden kann. Ich wünsche, diese Leute müßten alle ihre Programme auf der Zielmaschine wieder selber rückwandeln und dem armen Käufer die ärgerlichen Stunden ersetzen, die sie durch solcherlei Tun angerichtet haben.

auch mit einem gemeinsamen Bereich von 4 KByte auszukommen. Effektiv werden derzeit 5 KByte benutzt – es stehen also 59 KByte TPA zur Verfügung. Dies ist mehr als ein 64-KByte-CP/M-2-System bieten würde. Diese Möglichkeit würde jedoch einen wesentlich langsameren Bank-zu-Bank-Transfer nach sich ziehen, würde nur 2 Laufwerke zulassen und die Anzahl der Ein-/Ausgabe-Einheiten beschränken, denn nur die Treiber und Tabellen zu diesen Systemkomponenten könnten ausgedünnt werden um Programmcode zu kürzen.

Aufbau des CP/M-3-Systems im NDR-Klein-Computer.

Nach dem Bootbefehl (unter Elmon) wird der Inhalt von Spur 0, Sektor 1 in den Speicherbereich 0F000H-0F3FFH (1 KByte) geladen. In diesem Sektor steht der Track-0-Lader, der seinerseits vom (logischen) Sektor 2 beginnend, den sogenannten CPM-LADER nach Adresse 01004 in Bank 0 lädt. Das ist ein Rumpf-CP/M mit einem Rumpf-BIOS, dessen einzige Aufgabe es ist, das eigentliche CP/M, das sich unter einer Datei namens CPM3.SYS (komplett mit dem BIOS) verbirgt, nach Bank 0 (wo sich auch der CPM-Lader befindet) zu laden. Das Generierungsprogramm (GENCPM.COM) hat bei der Generierung CPM3.SYS bereits so aufgeteilt, daß beim Laden die Startadressen des gebankten Teiles (BNKBDOS und BNKBIOS) ebenso richtig liegen, wie die Adressen der BDOS- und BIOS-Teile, die im gemeinsamen Speicherbereich liegen.

Ist der CPM-Lader geladen, wird die Kaltstart-Adresse des BIOS-Teiles angesprungen und von dort der CCP (Konsol-Befehls-Interpreter) in die TPA-Startadresse (0100H) auf Bank 1 (der TPA-Bank) sowie nach Bank 0 in den CCP-Puffer geladen. Aus dem CCP heraus meldet sich das System dann mit dem typischen

A>

```

dw   fd$write      ;Adresse der Schreibroutine.
dw   fd$read       ;Adresse der Leseroutine.
dw   fd$loginit    ;Adresse der Loginroutine.
dw   fd$init       ;Adresse der INIT-Routine.
db   00b           ;Steprate des Laufwerkes.
db   sela+minis    ;(phys) Select des Laufwerkes.
fda: dphx1t5,dpb80d ;Angabe der (phys) Translate-
                    ;Tabelle und Adresse des DPB.

```

Bild 1. Ein Muster-XDPH

Es müssen also insgesamt 4 Leseoperationen stattfinden, bis CP/M 3 geladen ist: Spur 0, Sektor 1 mit dem Track-0-Lader; mit diesem der CPM-Lader; durch diesen CPM3.SYS; und zum Abschluß CCP.COM. Dieser ganze Vorgang dauert schon einen kurzen Augenblick, erst dann zeigen sich die Vorteile von CP/M 3: Nach CNTRL-C oder Warmstart wird nur der CCP aus Bank 0 nachgeladen – also nicht von Diskette. Dies hat den großen Vorteil, daß jederzeit eine Diskette eingelegt werden kann, die weder CPM3.SYS noch CCP.COM beinhaltet, die auch keine Bootsektoren (den Track-0-Lader und den CPM3-Lader) enthält. Jeder Warmstart geht schnell und geräuschlos über die Bühne.

Bei Systemen, die mehrere Formate unterstützen, kann der Benutzer Disketten mit anderen Formaten einlegen, ohne daß sich das System aufhängt. Es wird einfach als neu erkannt und eingeloggt, wenn das Bios entsprechend geschrieben wurde, oder als nicht lesbar abgewiesen – ohne Systemabsturz.

CP/M 3 bietet dem Benutzer noch weitere Annehmlichkeiten, wobei sicherlich die angenehmste die Vielfalt der zuschaltbaren Ein-/Ausgabe-Einheiten ist. Es können mit einem einfachen Hilfsprogramm (DEVICE.COM) 13 Einheiten ein-, aus- oder zugeschaltet werden (praktisch sind es 16 Einheiten, aber DRI hat sich die Weisungsbefugnis für 3 Geräte vorbehalten). Im BIOS-Unterprogramm CHARIO.Z80 (siehe Listing) sind in der ausgelieferten Version 6 Einheiten (devices) vorgesehen: KEY, die Tastatur (fest parallel), GDP, das Video-Display (fest über BUS), TERM als alternative Konsole mit variabler Baudrate über RS-232C (SER), AUX mit variabler Baudrate über RS-232C (SER). Als Drucker sind vorgesehen: SPRINT mit variabler Baudrate über RS-232C (SER); PPRINT mit paralleler Schnittstelle. Alle diese Einheiten können beliebig kombiniert werden. Anstelle von TERM kann selbstverständlich auch ein Terminal angeschlossen werden und an AUX ein MODEM.

Diskettenformate unter CP/M 3 im NDR-Klein-Computer

Eingebunden in das Diskettentreiber-Unterprogramm DISKIO.Z80 (siehe Listing) ist das Standard-NDR-Format. Andere Formate (soweit diese die gleiche Track-Umschaltung haben) können beliebig eingebunden werden. Die Diskettenparameter werden dabei im sogenannten XDPH (eXtended Disk Parameter Header) eingetragen. Notwendig sind hier nur wenige Daten (Bild 1):

fda ist dabei der Name des XDPH. Die Steprate kann in dieser Konfiguration für jedes (physikalische) Laufwerk getrennt angegeben werden – es ist also eine beliebige Mischung von Laufwerkstypen möglich. Bei Select sind unterschiedliche Kriterien anzuwenden. Bit 0 bis 3 wählen ein Laufwerk physikalisch aus. Es sind bis zu vier Laufwerke möglich.

Bit 4 auf 1 bedeutet einfache, auf 0 doppelte Dichte. Bit 5 auf 1 bedeutet 5.25-Zoll-Laufwerke, auf 0 große Laufwerke. Bit 7 entscheidet, ob eine Diskette als einseitig (0) oder zweiseitig (1) behandelt wird.

Es ist ohne weiters möglich, einem physikalischen Laufwerk (bestimmt durch Bit 0...3) verschiedene logische Laufwerke zuzuweisen. Logische Laufwerke unterscheiden sich prinzipiell nur durch unterschiedliche XDPH-Adressen. Es können bis zu 16 logische Laufwerke zugewiesen werden, entsprechend den Laufwerksbezeichnungen A...P. In der CP/M-3-Anpassung für den NDR-Klein-Computer sind jedoch folgenden Einschränkungen (aus Kompatibilitätsgründen) gemacht worden: E ist einer RAM-Disk, F einer Harddisk zugeordnet, ob diese Einheiten nun verwendet werden oder nicht. Alle anderen (logischen) Laufwerke außer Laufwerk A (Bootlaufwerk) können beliebig zugeteilt werden. Als Beispiel soll folgende Kombination gelten: Physikalisch besitze Laufwerk A 80 Spuren zweiseitig, Laufwerk B ebenfalls, Laufwerk C 40 Spuren zweiseitig. Die XDPHs könnten dann wie folgt aussehen (Bild 2).

Es fehlen nun nur noch die Daten für XLT- und DPB, diese werden, ebenso wie die kompletten Daten des XDPHs von Macros aufgebaut, die einzugebenden Daten sind in Bild 3 gezeigt.

Die Zahlenanordnung in den verschiedenen Macros hat dabei folgende Bedeutung (und wird entsprechend aufgelöst).

dpb: Name des Macros. Erste Ziffer: physikalische Sektorgröße in Byte (dezimal). Zweite Ziffer: Anzahl der phys. Sektoren pro Spur. Dritte Ziffer: Anzahl der (logischen) Spuren pro Laufwerk (Anmerkung). Vierte Ziffer: Blockgröße in Bytes. Fünfte Ziffer: Anzahl der Directory-Einträge. Sechste Ziffer: Anzahl der (vom Booter) reservierten Spuren.

Skew: Name des Makros. Erste Ziffer: Skew-Faktor. Zweite Ziffer: Anzahl der phys. Sektoren pro Spur. Dritte Ziffer: Start mit diesem Sektor.

Besonderheiten der CP/M-3-Anpassung für den NDR-Klein-Computer

Die CP/M-3-Anpassung für den NDR-Klein-Computer unterscheidet sich (auch in Elmon) von der CP/M-2-Anpassung eigentlich nur im Diskettentreiber

```
dpb80d: dpb1024,5,160,2048,256,4; 5" 80 Spuren zweiseitig NDR
dpb40d: dpb1024,5,80,2048,128,1; 5" 40 Spuren zweiseitig
dpb40s: dpb1024,5,40,1024,64,3; 5" 40 Spuren einseitig NDR
dpb80e: dpb 1024,5,160,2048,128,2 ; 5" 80 Spuren zweiseitig
dpb40q: dpb 256,16,80,2048,128,8 ;Epson QX10 zweiseitig 40
;Spuren.
```

```
xlt5: skew 1,5,1
xlt16: skew 1,16,1
```

Bild 3. DPB- und XLT-Daten

(DISKIO), wenn man einmal vom CP/M-3-spezifischen absieht. Im Gegensatz zur CP/M-2-Anpassung ist unter CP/M 3 keine Interruptbehandlung möglich (zumindest nicht unter den gegebenen Hardware-Voraussetzungen). Ein Interrupt mit RST 8 benötigt die Adresse 38H. Diese Adresse gibt es natürlich auch unter CP/M 3 – aber sie müsste dann schon in jeder Bank initialisiert werden. Das wäre zwar möglich, würde aber Platz und Zeit kosten. Es wurde daher auf die Methode des Pollens (laufende

Abfrage des entsprechenden Status) zurückgegriffen.

Diese Methode ist übrigens genauso schnell wie die Interruptmethode (und auch hier können 8-Zoll-Laufwerke nicht mit doppelter Dichte betrieben werden, zumindest nicht bei 4 MHz)! Ein schneller Bank-zu-Bank-Transfer wurde durch Einführung eines speziellen 516-Byte-Puffers von Adresse 0FA00H bis 0FBFFH ermöglicht. Dieser Pufferbereich bringt auch Tempo in eine RAM-Disk. Es bleibt dem Benutzer überlassen, diesen Bereich durch Verdoppelung oder Vervielfachung (zuungunsten der TPA) zu vergrößern, um hier Geschwindigkeit herauszuholen. Eine weitere (sinnvolle) Möglichkeit für schnellere Disketten-Zugriffe ist es, einen geeigneten Skew-Faktor einzuführen. Dies sollte jedoch sinnvollerweise bereits beim Formatieren einer Diskette geschehen, da dies der einzige Weg ist, auch nach Einführung eines Skews mit anderen NDR-Klein-Computer-Benutzern kompatibel zu bleiben.

In einem zusätzlichen Programmpaket kann eine Formatierungs-Anpassung für (beinahe) beliebige Formate erworben werden, sowie ein sehr schneller Z80-Assembler (SRL80ASM) für den alle Programme des BIOS geschrieben wurden (M80 ab Version 3.44 tats auch ... aber viel langsamer)

Sicherlich ist die CP/M-3-Anpassung für den NDR-Klein-Computer nicht die allerschnellste auf dem Markt, dies liegt vor allem am gewählten Diskettenformat (das ja kompatibel bleiben sollte), aber auch am Treiber für die GDP. Ein typischer Grafikprozessor, wie es die GDP nun einmal ist, kann sicher schnelle Grafik zaubern, aber für die Textverarbeitung würde man sich doch etwas anderes wünschen. Soll das System vor allem für Textverarbeitung benutzt werden, so ist es empfehlenswert, über die (eingebundene) Terminal-Schnittstelle ein Terminal anzuschließen oder eine schnelle Terminal-Emulationskarte.

```
dw fd$write
dw fd$read
dw fd$login
dw fd$init
db 00b ;home mit 3ms step
db sela+minis ;Laufwerk A
fda:dphxlt5,dpb80d ;NDR-Standard

dw fd$write
dw fd$read
dw fd$login
dw fd$init
db 00b ;home mit 3ms step
db selb+minis ;Laufwerk B
fdb:dphxlt5,dpb80d ;NDR-Standard

dw fd$write
dw fd$read
dw fd$login
dw fd$init
db 01b ;home mit 6ms step
db selc+minis ;Laufwerk C
fdc:dphxlt5,dpb40s ;NDR-40 Track 1-seitig

dw fd$write
dw fd$read
dw fd$login
dw fd$init
db 00b ;home mit 3ms step
db selb+minis ;phys B; log D;
fdd:dphxlt5,dpb80e ;Format des Eurocom

dw fd$write
dw fd$read
dw fd$login
dw fd$init
db 01b ;home mit 6ms step
db selc+minis ;phys C; log G;
fdg:dphxlt5,dpb40d ;2-seitig 40Track

dw fd$write
dw fd$read
dw fd$login
dw fd$init
db 01b ;home mit 6ms step
db selc+minis ;phys C; log H;
fdh:dphxlt16,dpb40x ;EPSON QX10
```

Bild 2. Die XDPHs für verschiedene Laufwerke