

Unser Treiber kann nicht formatieren, DRIVER.SYS allein kann das Double-Stepping nicht unterdrücken. Folglich bleibt nur ein mühsamer (Um-) Weg.

Man baue sich wieder einen Treiber-Aufruf für DRIVER.SYS in CONFIG.SYS ein, und zwar exakt mit denselben Parametern wie eben beschrieben. Dann lege man eine 80-Spur-Diskette ins Laufwerk und schließe die Laufwerksklappe. Man starte DEBUG und setze die bewußte Status-Zelle (40:90 oder 91) auf 54h. Jetzt DEBUG verlassen und FORMAT (ohne Parameter) für das logische Laufwerk aufrufen, auf das DRIVER.SYS installiert ist.

Wenn es mal nicht auf Anhieb klappt, machen Sie zur Sicherheit einen Warmstart und wiederholen Sie die Prozedur. Und vor allem: öffnen Sie die Laufwerksklappe nicht, bevor die bewußte Diskette formatiert ist, denn sonst wird Ihre Debug-Arbeit jedesmal wieder zunichte gemacht. Da die geschilderte Prozedur für jede einzelne 80-Spur-Scheibe wiederholt werden muß (man muß ja beim Disk-Wechsel leider die Klappe öffnen), empfiehlt es sich, eine Batch-Datei anzulegen, die ihrerseits eine Kommando-Datei in den DEBUG schickt. Die Batch-Datei enthält nur die Zeile

DEBUG<FORMIN

und die Datei FORMIN nur die beiden Zeilen

E40:90 54

q

beziehungsweise 91, wenn das zweite Laufwerk umgestellt wird. Wenn Sie allerdings ein Programm wie DOS-EDIT im Hintergrund geladen haben, funktioniert die hier angewendete Methode der Input-Redirection nach DEBUG nicht mehr.

Mit Pin 34

Über eine uns erst kürzlich aufgefallene Fähigkeit von DRIVER.SYS unter PC DOS 3.2/3.3 sei noch berichtet. Richtig flexibel wird die Parameter-Eingabe erst, wenn man für den F-Parameter '7' ('others') angibt, denn nur dann werden alle folgenden Parameter auch wirklich ausgewertet. Bei anderen Laufwerkstypen klappt das nicht in jeder Kombination.

Viel wichtiger ist aber, daß sich über '/F:7' auch die einzige Chance bietet, Nicht-MF-Laufwerke, die Signale auf Pin 34 führen, korrekt im AT zu bedienen, ohne diesen Pin abzuklemmen. Sogar bei einem MF-Laufwerk findet sich dann plötzlich 93h in der Disk-Status-Zelle, wenn man alle anderen Parameter gemäß '80-Spur normal' einstellt.

Das heißt aber nur, oh bittere Ironie, daß auf dem MF-Laufwerk zwar kein Double-Stepping mehr stattfindet, dafür aber statt dessen eine Datenrate von 250 kBit pro Sekunde eingestellt ist, die absolut nicht zur Standarddrehzahl eines MF-Laufwerks im AT (360 Umdrehungen pro Minute) paßt.

Aber auch aus einem zweiten Grund ist unser Treiber keinesfalls überflüssig geworden, und die Empfehlung, Pin 34 lahmzulegen, bleibt ebenfalls bestehen; denn nur dadurch können alle von IBM unterstützten Formate – außer natürlich beim Formatieren – über den üblichen Laufwerksbuchstaben erreicht werden, während es sonst immer nur über den Treiber ginge.

Um die Verwirrung komplett zu machen, sei noch nachgetragen, daß bei PC DOS 3.3 '/F:7' nunmehr für das neue High-Density-Format bei 3,5-Zoll-Scheiben mit der Kapazität 1,44 MByte (PS/2 Modell 50 bis 80) vorgesehen ist.

Ob dieses Format auch auf normalen ATs verwendbar sein wird, konnten wir noch nicht testen. Es spricht eigentlich nichts dagegen. Allerdings darf man sich nicht davon irritieren lassen, daß der Versuch, dieses Format etwa durch Anwendung eines Treiberaufrufes mit den Parametern '/F:1/S:18...' auf ATs mit 5,25-MF-Drives zu erzeugen, unter harten FORMAT-Fehlermeldungen scheitert. Die neuen 3,5-Zoll-Laufwerke mit 1,44 MByte rotieren nämlich auch im HD-Modus nur noch mit 300 Umdrehungen, die 5,25er (nicht nur im AT!) in diesem Modus aber immer mit 360. Da die Übertragungsraten in beiden Fällen 500 kBit/s beträgt, paßt auf die 3,5-Zoll-Scheiben also mehr drauf. Wir werden zu gegebener Zeit über den Anschluß der neuen Drives an 'die alten Rechner' berichten.

Fortschritt und so

Eigentlich ist ein Teil des Aufwandes, den wir mit unserem Treiber treiben, eher dem Rückschritt gewidmet, da wir ja wieder auf das alte Formaterkennungsverfahren mit dem Media-Byte zurückgekommen sind. Die berühmten Änderungen, die dem Fortschritt dienen, sind dennoch da, nur liegen sie ganz woanders.

schon den DOS-Versionen 2.x und 3.x machte uns dabei das Leben schwer (siehe Tabelle). Mit DOS 3.x lief es prima, mit 2.x nur, wenn zwischen Name und Parameter mindestens zwei Leerzeichen lagen. DOS 3.x interpretiert nämlich die CONFIG-Datei und wandelt die Bezeichner wie DEVICE oder SHELL in Token um (der von DEVICE lautet zum Beispiel DC:). Gleichzeitig fügt dieser

DOS 2.x

Offset	Text (device=)
00	erstes Zeichen des Treibernamens
.	.
n	letztes Zeichen des Treibernamens
n+1	00 als Trennzeichen
n+2	erstes Zeichen der Parameterkette
.	.
n+m	letztes Zeichen der Parameterkette

DOS 3.x

Offset	Text (device=)
00	erstes Zeichen des Treibernamens
.	.
n	letztes Zeichen des Treibernamens
n+1	20h als Trennzeichen
n+2	20h als Trennzeichen
n+3	erstes Zeichen der Parameterkette
.	.
n+m+1	letztes Zeichen der Parameterkette

Die beiden DOS-Generationen liefern die Treiberaufrufzeilen innerhalb von CONFIG.SYS unterschiedlich aufbereitet an den Anwender.

Zum einen ist der Code des Treibers gegenüber seiner Ursprungsversion weitgehend neu arrangiert worden, damit alle nur für die Initialisierung nötigen Teile nach selbiger überschrieben werden können. Obwohl also das SYS-File des Treibers länger als die ältere Version ist, verbraucht der Treiber in installierter Form weniger RAM. Das ist ja nicht unwichtig, denn mitunter ist er zwei- bis dreimal geladen, und die Formatierhilfe DRIVER.SYS möglicherweise auch.

Weiterhin sind jetzt alle beschriebenen neuen Fähigkeiten eingebaut. Die Beschränkung der Ursprungsversion, daß der Treibername nur fünf Buchstaben vor der Extension haben durfte und keine Pfadnamen im Aufruf erlaubt waren, ist entfallen. Dazu wurde die Routine zur Parameterauswertung überarbeitet. Jetzt kann ein beliebig langer Leerraum zwischen Name und Parameter bestehen. Ein kleiner Unterschied zwi-

Interpreter noch ein zusätzliches Leerzeichen zwischen Name und Parameter ein.

DOS 2.x hingegen wandelt den Text zwar nicht in eine interne Darstellung um, aber statt des Leerzeichens in der Zeile wird 00h als Trennzeichen verwendet, wonach unsere Routine natürlich nicht gesucht hat. Erst das zweite Leerzeichen kam als solches durch.

Des weiteren wurden die Startmeldungen deutlich erweitert. Der Treiber gibt jetzt alle eingestellten Parameter im Klartext aus und teilt auch mit, für welches physikalische Laufwerk er bestimmt ist und welchen logischen Laufwerksbuchstaben er sich erkoren hat.

Um die Universalität nicht zu schmälern, sollte der Treiber unverändert auf PCs, ATs und dem c't86 laufen. Daher ist im Initialisierungsteil eine Abfrage enthalten, die prüft, um welchen Rechner es sich handelt. Schließlich muß in PCs an