

## JOYCE-Disk mal 2

### PCW mit zweimal 720-kByte-Diskettenkapazität

Nach längerer Pause kommt aus Nürnberg mal wieder eine Hardware-Bastelei, die es in sich hat: Der gute alte PCW wird auf eine Diskettenkapazität von zweimal 720 kByte (!) aufgerüstet und stellt somit jetzt zwei "große" Laufwerke zur Verfügung. Der von der Enge auf seinen Disketten geplagte PCW-Benutzer ist damit nicht nur ein Platzproblem los, sondern kann endlich auch "große" Disketten in einem Zug kopieren, ohne Diskettenwechsel und ohne Umweg über die RAM-Disk. Eine Fortsetzung des Berichts wird zeigen, wie man vom großen Diskettenformat sogar booten kann. Wie das alles mit einem Aufwand von deutlich unter 300 DM möglich ist? Sehen Sie selbst.

Eines der bekannten Murphy'schen Gesetze besagt: Software dehnt sich in einem Rechnersystem stets so lange aus, bis aller verfügbare Platz restlos voll ist (Software always expands to fill the available amount of space). Ich persönlich glaube nicht unbedingt an Murphys Behauptungen, aber auch bei mir war vor einiger Zeit der Punkt erreicht, an dem die vorhandene Diskettenkapazität einfach nicht mehr ausreichen wollte.

Meine RAM-Floppy war bei der Arbeit stets randvoll mit Programmen (zum Beispiel beim Schreiben von Texten mit dem vollen Umfang des WORDSTAR/WS-TUNER-Pakets) und einigen ständig benötigten Hilfsprogrammen wie SUBMIT und SWEEP. Im Laufwerk B: steckte die Textdiskette. Das so knapp bemessene Laufwerk A: zu benutzen, bin ich schon seit geraumer Zeit leid. Die "kleinen" Disketten speichern einfach zu wenig, um damit

sinnvoll zu arbeiten. So liegen meine Programme und Texte alle auf "großen" Disketten, die nicht so schnell ausgenutzt sind und auf denen sich zusammengehörige Dateien viel praktischer speichern lassen. Was nützt mir ein Laufwerk A:, welches die "großen" Disketten nicht einmal lesen kann?

### PCW-Disketten – Wunsch und Wirklichkeit

Mein Wunschtraum wäre folgende Vision: Ich benötige während meiner Arbeit noch ein weiteres Programm, greife mir einfach die Programmdiskette und schiebe sie in ein zusätzliches Laufwerk. Danach wird das Programm ohne langwierige Hantiererei mit anderen Disketten gestartet, wie es sich bei nur einem (großen) Laufwerk zwangsläufig ergibt. Das kenne ich nun aus eigener leidvoller "Diskjockey"-Erfahrung

zur Genüge: Textdiskette raus, Programmdiskette rein, Programm starten. Natürlich soll das Programm aber nicht eines der anderen Programme auf der RAM-Disk bearbeiten, sondern einen auf der Textdiskette liegenden Text, welche ich zu meinem Entsetzen eben aus dem Laufwerk herausgenommen habe. Schließlich legt man wichtige Texte nicht so gerne auf die voll ausgenutzte RAM-Disk, denn der Mut zum Risiko hilft nun auch nichts mehr. Also: Das Programm von der Programmdiskette starten, Programmdiskette wieder entfernen, damit die Textdiskette wieder eingelegt werden kann. So kann man dies weiterhin fortsetzen.

### PCWs Diskettensystem – mit Fehlern leben?

Bevor dies allerdings bewerkstelligt ist, gibt es auf dem PCW einige konstruktive Grenzen zu überwinden. Ein Überschreiten der Grenzen wurde bei der Konstruktion des PCW wahrscheinlich aus Kostengründen nicht vorgesehen. Im Unterschied zu dem PCW 9512, in dem bereits zwei "große" Laufwerke eingebaut sind, haben sich die Konstrukteure des "alten" PCW zunächst entschlossen, schon per Hardware nur zwei Laufwerke gleichzeitig zu unterstützen. Daran ist auch nichts zu ändern. Wer mehr als zwei Laufwerke betreiben will, muß also zu dem Trick greifen, einen Schalter einzubauen, mit dem die Select-Leitung des Floppy-Busses von den Originallaufwerken auf das gerade gewünschte Zusatzlaufwerk umgeschaltet werden kann.

Außerdem hat man aber auch die Laufwerksformate bereits bei der Zusammenstellung des Betriebssystems grundsätzlich festgelegt. Der hardwareabhängige Kern des PCW-Betriebssystems, das XBIOS, ist so gestaltet, daß sich ohne schwerwiegenden Eingriff als Laufwerk A: nur ein einseitiges Modell mit 40 Spuren betreiben läßt. Das eventuell vorhandene Laufwerk B: muß dann unbedingt 80 Spuren pro Seite (im Normalfall auch zwei Seiten) in dem "großen" 720-kByte-Format haben. Beim PCW 9512 wurde das wohl endlich als unpraktisch eingesehen und abgeändert. Muß man aber bei dem "alten" Modell wirklich mit allen diesen Einschränkungen leben?

Da vom Hersteller des PCW und auch von Händlern zumindest in Deutschland bei mehreren Anfragen in der Ver-

gangenheit nichts zu diesem Thema zu erfahren war, habe ich es schließlich einfach einmal ausprobiert, ein zweites "großes" Laufwerk an einen PCW anzuschließen. Nach einigen Rückschlägen stellte sich wirklich auch ein Erfolg ein.

## Laufwerkswahl

Als zusätzliches Laufwerk wurde ein 3,5-Zoll-Laufwerk gewählt, weil dieses Diskettenformat bedeutend gängiger ist als das "große" PCW-3-Zoll-Format und dadurch auch die Disketten viel billiger zu haben sind. Mancher PCW-Plus-Benutzer mag schon neidisch auf die Preise der 3,5-Zoll-Disketten für 720-kByte-Kapazität geschickt haben, die bei bester Markenqualität gerade ein Viertel bis ein Fünftel des Preises der CF2-DD-Disketten kosten und sogar noch billiger sind als "normale" CF-2-Disketten. Aus eigener Erfahrung wird mit einem 3,5-Zoll-Laufwerk außerdem der Datenaustausch zwischen PCW und anderen Rechnern, die diese Disketten verwenden, ziemlich problemlos, da es innerhalb des 3,5-Zoll-Formats kaum noch Varianten gibt. Man braucht allerdings ein Programm wie MSCOPY, um die andere Directorystruktur auf diesen Disketten verarbeiten zu können; somit steht dem Lesen und Schreiben solcher Fremddisketten nichts mehr im Weg. Qualitativ hochwertige Diskettenlaufwerke selbst bekommt man inzwischen im guten Versandhandel und einigen Kaufhausketten deutlich unter 300 DM. Aus diesen Gründen habe ich auch schon beim Aufrüsten meines PCW zum PCW PLUS ein 3,5-Zoll-Laufwerk verwendet.

Getestet wurden ein NEC FD 1035 in "halber" – aus heutiger Sicht doppelter – Bauhöhe, (inzwischen nicht mehr erhältlich) und zwei moderne "low power"-Laufwerke, wie sie auch in Laptops zum Einsatz kommen, ein NEC FD 1037 A und ein Mitsubishi-Laufwerk.

## Experimente und ihre Folgen

Zunächst wurde festgestellt, ob diese Laufwerke überhaupt am PCW laufen. Das war eigentlich zu erwarten, nachdem an meinem PCW ein Shugart-Bus-Anschluß geschaffen worden war (siehe PCW-Sonderheft 2), da die getesteten Laufwerke sich laut Hersteller an den Quasistandard des Shugart-Busses halten. Das FD 1035 lief zunächst als Laufwerk B auf Antrieb, während

die modernen Laufwerke erstaunlicherweise Probleme machten. Solange sie nicht angesprochen wurden, verhielten sich zwar beide – im Test jeweils abwechselnd angeschlossen – wie von anderen Laufwerken am PCW gewohnt und erwartet. Beim Ansprechen eines Laufwerks dagegen, bei bereits im Laufwerk befindlicher Diskette, meldete PCW spontan "Laufwerk nicht bereit". Noch toller kam es, wenn zunächst keine Diskette eingelegt wurde. Das "moderne" Laufwerk lief an, setzte sich "ohrenscheinlich" auf Spur 0 zurück und versuchte die nicht vorhandene Diskette zu lesen. In dieser Situation kommt vom PCW-Betriebssystem normalerweise sofort die Meldung "Laufwerk nicht bereit". Nicht hier: Das leere Laufwerk lief endlos weiter, bis es per Hardware (über den Select-Schalter) vom Floppybus abgehängt oder aber eine Diskette eingelegt wurde. Letzteres mit der noch erstaunlicheren Folge, daß nun sogleich die Meldung erschien: "Laufwerk nicht bereit". Darüber hinaus brachte dieses Laufwerk zuweilen den gesamten Floppybus restlos durcheinander.

Da das jeweilige Laufwerk genau das entgegengesetzte Verhalten von dem zeigte, was zu erwarten gewesen wäre, wurde zunächst ein Laufwerksfehler vermutet (READY-Signal invertiert durch internen Fehler auf einem der eingebauten Chips?). Nach mehrmaligem, erfolglosem Austausch durch die Vertragswerkstatt, die anscheinend keinerlei Fehlersuche unternahm, führte ein Tip eines anderen Händlers

schließlich zu der Entdeckung, daß überhaupt kein Defekt vorlag, sondern die 3,5-Zoll-Laufwerke der neuen Generation offenbar mit dem READY-Signal allgemein ziemlich schlampig umgehen. Das READY-Signal der Testlaufwerke zeigte jedenfalls beim Durchmessen nicht das in den technischen Unterlagen beschriebene Verhalten. Den MS-DOS-Rechnern, an die diese Laufwerke normalerweise angeschlossen werden, macht dies anscheinend nichts aus, so daß die Hersteller keinem Druck ausgesetzt sind, klare Verhältnisse zu schaffen. Was aber nun am PCW?

## Simulierte Bereitschaft

Hier blieb nur der Weg, ein ordentliches oder zumindest von PCW akzeptiertes READY-Signal künstlich zu erzeugen. Ein Eingriff in die Laufwerke selbst kam mangels genauerer Unterlagen und angesichts der mit einfachen Mitteln kaum noch zu beherrschenden filigranen Technik nicht in Frage. Dem PCW-Floppytreiber mußte also von außerhalb des Laufwerks vorgetäuscht werden, das Laufwerk sei stets bei Selektion bereit. Normalerweise wartet das PCW-XBIOS nach dem Ansprechen eines Laufwerks von sich aus eine gewisse Hochlaufzeit ab, bevor es annimmt, das Laufwerk müsse jetzt bereit sein. Man führt dann eine Sicherheitsabfrage durch, ob das Laufwerk tatsächlich ein READY-Signal auf den Bus liefert. Das READY-Signal dient hier vor allem zur Sicherstellung, daß

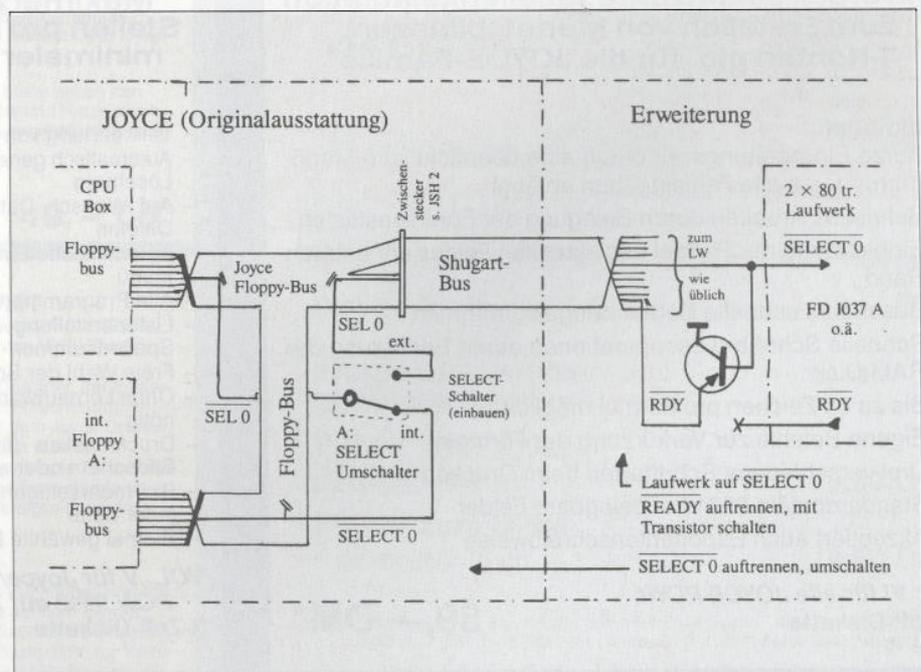


Bild 1: Stromlaufplan: Anschluß des "großen" Laufwerks A: an PCW

```

;lwagross.sub
;C.Frederking & DMV, 1990
;
;Stellt Laufwerk A: fest auf "großes" Format ein
;
sid
<sFF4A
<24, <00, <04, <0F, <00, <64, <01, <FF, <00, <F0,
<00, <40, <00, <01, <00, <02, <03, <81, <50, <09,
<01, <00, <02, <2A, <52, <60, <FF, <. , <g0

(Kommata sind durch <RETURN> zu ersetzen)

```

```

;lwanorm.sub
;C.Frederking & DMV, 1990
;
;Stellt Laufwerk A: auf "normalen" Betrieb ein
;Diskettenformate werden automatisch erkannt
;
sid
<sFF64
<00
<.
<g0

```

Bild 2: Listing: Setzen des XDPB A: für das "große" Format (fest) und zurück auf "Automatik"

das Laufwerk seine Solldrehzahl erreicht hat und damit wirklich gültige Daten lesen und schreiben kann. Schaltet man die Wirkung dieser Sicherheitsabfrage dadurch aus, daß der Bus ständig mit einem READY-Signal belegt wird, so muß der Floppytreiber so lange warten, bis man dessen auch ohne Abfrage sicher sein kann.

Glücklicherweise ist dies im PCW – sogar einstellbar – vorgesehen, während von den Laufwerksherstellern in den technischen Unterlagen eine maximale Hochlaufzeit garantiert wird, die sogar kürzer ist als bei den PCW-Originallaufwerken. Man kann daher annehmen, daß ein Laufwerk, welches die garantierten Daten einhält, ohne Abfrage des READY-Signals auskommt. Es wird allerdings nicht mehr automatisch erkannt, wenn das Laufwerk etwa wegen eines – theoretisch möglichen – Defekts oder Diskettenklemmers nach Ablauf der Wartezeit seine Solldrehzahl noch nicht erreicht hat. Ein Defekt, der bei simuliertem READY-Signal dazu führt, daß ein zu langsam laufendes Laufwerk auf eine Diskette schreibt, wäre allerdings für die Daten auf der Diskette fatal.

Die Wahrscheinlichkeit, daß hier etwas passiert, ist jedoch gering. Es scheint gut möglich zu sein, mit diesem Restrisiko zu leben. Die Wahrscheinlichkeit, daß Daten durch einen Benutzerfehler zerstört werden, ist weitaus größer. "Gewöhnliche" Floppy-Fehler (Schreib/Lese-Fehler), meldet der Floppytreiber

ohnehin nach wie vor; diese haben mit dem manipulierten READY-Signal nichts zu tun. Erzeugt wird das "vorge-täuschte" READY-Signal durch einen einfachen Universaltransistor (pnp), der ohne Vorwiderstände an den Leitungen des Shugart-Busses und an Masse angeschlossen wird. Das Massepotential muß man dazu irgendwo herkriegen. Wenn nichts anderes zur Verfügung steht, tut es im allgemeinen eine der Masseadern auf dem Floppybus. Man sollte nicht gerade die Ader neben WRITE DATA oder WRITE GATE verwenden. Der Anschluß des Transistors muß so nahe am Laufwerk erfolgen, daß er nur durchgeschaltet wird, wenn tatsächlich das betreffende Laufwerk angesprochen wird, somit vom PCW aus gesehen erst hinter einem eventuellen Select-Umschalter. Der tatsächliche READY-Ausgang des Laufwerks wird dabei überflüssig; er wird einfach vom Bus getrennt, also an keine Verbindungsleitung angeschlossen, und bleibt unbeachtet.

### Und sie fliegen doch!

Nach Überwindung dieser Anfangshindernisse konnten die Laufwerke nun "richtig" angeschlossen werden. Der Haupterfolg zeigte sich aber beim Ansprechen des neuen A-Laufwerks mit einer 3,5-Zoll-Diskette im Schacht, nachdem in dem B-Laufwerk mit DISCKIT auf "PCW CF 2 DD" formatiert wurde. Zuerst schien es eher ein

Mißerfolg zu sein. Denn das erste, was ich danach zu sehen und zu hören bekam, war ein ärgerliches Piepen und die Meldung des XBIOS in der Statuszeile: "Laufwerk A – Falsche Diskette! Wiederholen, Ignorieren, Abbrechen?" Wiederholen oder Abbrechen führt hier natürlich zu keinem vernünftigen Ergebnis. Aber, man staune, wenn man diese Meldung einfach "ignoriert", wird die für das Laufwerk zuständige Tabelle (XDPB) im PCW-BIOS ohne weitere Komplikationen auf das eben noch beanstandete Diskettenformat eingestellt. PCW benimmt sich so artig, als wäre das Lesen von "großen" Disketten in Laufwerk A: schon immer völlig normal gewesen! Bild 5 zeigt den Erfolg handgreiflich auf dem Bildschirm.

Das stumme Warten auf nicht vorhandene Disketten in den "modernen" Laufwerken ist dem PCW allerdings nicht abzugewöhnen, da es ja auch ein Konstruktionsfehler der Laufwerke ist. Man muß sich eben daran gewöhnen, daß nicht gleich reklamiert wird, wenn man vergißt, eine Diskette einzulegen, sondern PCW nun still und geduldig darauf wartet, daß man es nachholt. Vielleicht ja auch nicht so schlecht. Man sollte freilich nach dem Umbau eines bedenken: Wenn man sich irgendwann darüber wundert, daß nichts geschieht, so ist es wahrscheinlich nicht das System, das sich "aufgehängt" hat, sondern nur ein vergessenes Laufwerk, das ohne einen Mucks, aber beharrlich auf "seine" Diskette wartet. Diese Eigenschaft des aufgerüsteten Systems zu vergessen, hätte mich selbst schon einmal fast zur Verzweiflung gebracht. Die Motoren dieser Laufwerke sind nämlich so leise, daß man sie im Betrieb glatt überhören kann.

Diese kleine Macke kann aber den Erfolg letztlich nicht verderben, denn der Betrieb mit den zwei "großen" Laufwerken funktioniert einfach vorzüglich. Wenn man sich das alte "kleine" Laufwerk A: wegdenkt, hat man beinahe einen PCW 9512 vor sich. Vorbei die Zeit, in der das Kopieren von DD-Disketten den Umweg über die RAM-Disk nehmen mußte; Quelldiskette in A:, Kopie in B:, und los geht's.

Das PCW-Diskettenkopierprogramm DISCKIT ist leider nicht so flexibel wie das Betriebssystem selbst. Es kennt einfach nur "kleine" A- und "große" B-Floppies. Unabhängig davon, was tatsächlich an Möglichkeiten vorhanden ist, muß man sich zunächst auf das dateiweise Kopieren bis zur Veröffentlichung

lichung eines Disk-to-Disk-Kopierprogramms für "große" Disketten beschränken. Für die tägliche Datensicherung nach der Arbeit ist das wohl auch der Normalfall, denn dabei erzeugt oder ändert man in der Regel Dateien und nicht komplette Disketten. DISCKIT reagiert überhaupt etwas unbeholfen auf die "große" A-Floppy; es behauptet auch im leeren Laufwerk eine Diskette zu finden und fordert beim Programmstart auf, sie herauszunehmen. Offenbar verwendet DISCKIT beim Feststellen des Laufwerkstatus das READY-Signal, das nach unserem unfreiwilligen Eingriff in die Laufwerk-Rechner-Kommunikation ja bei Laufwerk-Selektion stets aktiv ist.

Wie ich zufällig entdeckte, läuft PCW dafür vorläufig noch nach dem Booten von einer "normalen" kleinen Diskette auch mit dem "großen" Laufwerk A: alleine; dann wird die Laufwerkskennung vom XBIOS umgeschaltet, wie vom "kleinen" PCW gewohnt ("Laufwerk ist A:/"Laufwerk ist B:."), aber bei voller Kapazität von netto 706 kByte.

## Software

Zum Betrieb der Laufwerkserweiterung in dem bis hierher geschilderten Rahmen benötigt man im Grunde weder zusätzliche Software noch irgendeinen Patch im Betriebssystem. Um das Laufwerk A: fest auf das große Format einzustellen, kann ein kleines Programm für SID verwendet werden. Die Formateinstellung sollte allerdings bis zum Erscheinen der Fortsetzung dieses Artikels erst nach dem Booten vorgenommen werden, das in Bild 2 wiedergegeben ist. Einige Bemerkungen hierzu werden noch folgen.

## Montage

Einen detaillierten Stromlaufplan, in dem besonders die ungewöhnlichen Details dieses Floppy-Einbaus beziehungsweise Anbaus dargestellt sind, zeigt Bild 1. Für die routinemäßigen Einzelheiten der Montage und des elektrischen Anschlusses der Laufwerke möchte ich den Leser auf frühere Veröffentlichungen in dieser Zeitschrift hinweisen, in denen diese Dinge schon sehr detailliert dargestellt wurden, insbesondere auf die ausführliche praktische Umbauanleitung in PCW-Sonderheft 2 (siehe Kasten "Literaturhinweise"). Abgesehen von Details wie der Tatsache, daß hier ein zweites "kleines" statt eines zweiten "großen"

Laufwerks angeschlossen wird, ist die benötigte Hardware-Information in beiden Fällen identisch.

Selbstverständlich braucht man im Unterschied zu dem angeführten Artikel für das 3,5-Zoll-Laufwerk keine Step-Untersetterschaltung und keinen 40-Spur (CF2)-, sondern einen 160-Spur (CF2DD)-XDPB. Damit sind die Unterschiede aber auch schon erledigt. Nicht getestet habe ich allerdings, ob sich das 3,5-Zoll-Laufwerk (oder gar zwei nebeneinander?) auch (knapp) in den Haltekäfig für das zweite PCW-Einbaulaufwerk einsetzen läßt, wenn man noch keines hat. Der Einbaukäfig jedenfalls müßte, wie ein Nürnberger Händler meinte, wohl auch einzeln zu bekommen sein.

Ein zusätzliches Netzteil braucht man für das Extralaufwerk übrigens auch als Drittlaufwerk wohl nicht gleich, wenn man ein modernes "low power"-Laufwerk verwendet. Das Netzteil eines "kleinen" PCW dürfte sogar ohne weiteres die Leistungsaufnahme von zwei solcher Laufwerke zusätzlich zu dem einen vorhandenen Einbaulaufwerk verkraften. Bei meinem eigenen PCW hängen allerdings sämtliche nachgerüsteten Floppies an einem Extranetzteil, mit dem zusammen sie auch in einem eigenen PC-Gehäuse untergebracht sind (Bild 3), weil unter diesen Laufwerken sich ein schwergewichtiger alter "5,25-Zoll-Hobel" (BASF 6106) befindet, der das eingebaute PCW-Netzteil auf die Dauer natürlich überfordert.

Bild 4 zeigt, wie in einem solchen Fall die Verbindung zwischen PCW und den externen Laufwerken aussehen kann. Die Signalleitungen des Floppybusses wurden auf einem Centronics-Steckverbinder sauber zusammengefaßt, der den Vorzug hat, in allen (!)

Varianten preisgünstig erhältlich zu sein, und der einfach in die Gehäusewand des PCW eingepaßt wurde. Das dafür erforderliche Loch kann man etwas ungalant, aber wirkungsvoll und genau in einem gut gelüfteten Raum mit einem kräftigen FeinlötKolben erzeugen. Die freien Pins des Centronics-Steckers, der ja mehr Verbindungsadern aufnehmen kann, als für einen Floppybus erforderlich, können zum Beispiel dazu verwendet werden, ein nicht allzu stromgieriges Relais in der externen Floppybox anzusteuern, so daß sie beim Einschalten des Rechners von selbst anspringt. Man sollte aber darauf achten, daß auch bei den zusätzlichen Leitungen zwischen zwei Adern immer eine Masseverbindung liegt, um die Verbindungen gegen elektromagnetische Einstrahlungen abzuschirmen. Wegen der Abschirmung ist auch die Länge des Flachbandkabels von der Zentraleinheit im Rechner bis zu den Floppies begrenzt. Die maximale Länge beträgt üblicherweise je nach Floppytyp ein bis drei Meter.

## Ruhe im XBIOS!

Bei der Bedienung des aufgerüsteten PCW ist noch etwas störend. Das XBIOS unterbricht bei jedem Neuzugriff auf eine "große" Diskette in A: erst einmal warnend, wie das ja bei einem "normalen" PCW auch vernünftig wäre, "Falsche Diskette in A:". Bekanntlich identifiziert das XBIOS das Format jeder eingelegten Diskette an einer Eintragung auf dem Bootsektor und stellt sich intern auf dieses Format ein. Dies geschieht ungefähr nach jedem Zurücksetzen des Diskettensystems von CP/M aus und damit mindestens nach einem erkannten Diskettenwechsel. Dabei wird auch geprüft, ob



Bild 3: Die Floppy-Erweiterung, hier im Ausbau als selbständige Einheit

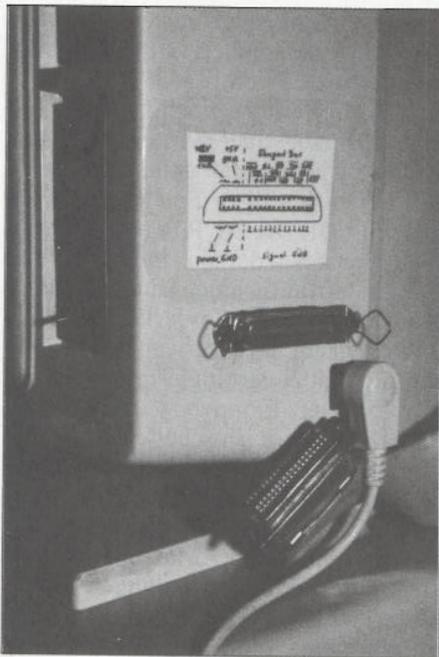


Bild 4: Vorschlag für den externen Anschluß der Floppyeinheit

das erkannte Format im aktuellen Laufwerk überhaupt zu verarbeiten ist. Das XBIOS weiß natürlich nichts von dem unverhofften Laufwerkwechsel, und das kleine Originallaufwerk, das sich normalerweise unter A: befindet, kann nun einmal mit dem DD-Format tatsächlich nichts anfangen. Seine Aufpassernatur kann man dem XBIOS aber abgewöhnen, indem man im XDPB A: von vornherein die richtigen Parameter für das Format sowie das FREEZE-Flag setzt. PCW prüft dann beim Diskettenzugriff nicht mehr das Diskettenformat, sondern geht einfach davon aus, daß die eingelegte Diskette dem intern eingestellten Format entspricht. Diese Einstellung wird auf Wunsch von dem kleinen SUBMIT-Programm in Bild 2 erledigt mit dem hoffentlich selbsterklärenden Kürzelnamen LW A GROSS (ohne Blanks einzugeben!); damit es läuft, muß außer SUBMIT auch das Programm SID für das System verfügbar sein.

Wenn man das FREEZE-Flag nicht setzt, hat das allerdings auch den einen Vorteil, daß das Diskettenformat weiterhin automatisch überprüft wird und man dadurch das notwendige Umstellen beim Wechsel zwischen verschiedenen "A:-Laufwerken nicht vergessen kann; denn natürlich kann sich PCW im festgestellten Zustand auf das "kleine" Diskettenformat nicht mehr richtig einstellen. Man bekommt dann unsinnige Meldungen wie "Spur 1, Sektor 7: Keine Daten". Ohne FREEZE kann das nicht passieren, für diesen Komfort muß man dann eben ab

und zu einmal "Ignorieren" drücken. Übrigens: Das Listing im zweiten Teil von Bild 2 (LW A NORM) setzt PCW wieder in den Ursprungszustand zurück.

### Trau, schau, wem...

Leider erwies es sich nicht als möglich, mit dem "großen" Laufwerk A: Disketten im entsprechenden Format auch zu beschreiben. Auch dies liegt an einer Vorsichtsmaßnahme innerhalb des XBIOS, das hier in ganz besonderer Weise niemandem traut. Es wäre allerdings auch fatal für die Daten auf einer CF-2-DD-Diskette, wenn man mit der völlig anderen Spurlage eines normalen CF-2-Laufwerks auf ihr herum-schreiben würde. Um beim Standard PCW PLUS zu vermeiden, daß dies durch eine Verwechslung der Disketten doch einmal geschieht, wurde offensichtlich sehr tief in der Floppy-Schreibroutine des BIOS eine Überprüfung daraufhin eingebaut, ob das aktuelle Laufwerk (oder das, was das BIOS als solches zu verwalten glaubt) physikalisch zu dem entsprechenden zuletzt von Diskette gelesenen beziehungsweise intern eingestellten Format paßt.

Diese Sicherung liegt so tief im BIOS, daß sie auch bei direktem Zugriff über die XBIOS-Programmierschnittstelle wirksam ist (man erhält einen nicht dokumentierten XBIOS-Fehlerstatus). Das betrifft natürlich auch DISCKIT. Man kann "große" Disketten also auch nur in Laufwerk B: formatieren. Um dies zu ändern, müßte man Eingriffe im Kern des Betriebssystems im Bereich des Diskettentreibers vornehmen, von dem bisher nicht einmal eine halbwegs vollständige Dokumentation vorliegt. Ich habe deshalb darauf verzichtet, nachdem sich die Unbequemlichkeit, für Sicherungskopien eventuell die Disketten in A: und B: vertauschen zu müssen, dann aber alles in einem Zug automatisch ablaufen lassen zu können, als durchaus tragbar erwiesen hat. Vielleicht führt ja jemand schon in einem der nächsten Hefte vor, wie es noch einfacher geht?

### Resümee und Ausblick

Im übrigen kann man diese Einschränkung ja durchaus auch als Erleichterung nutzen. "Große" Disketten, die in A: liegen, können grundsätzlich nicht beschrieben werden, also kann man auch nicht versehentlich beim Anferti-

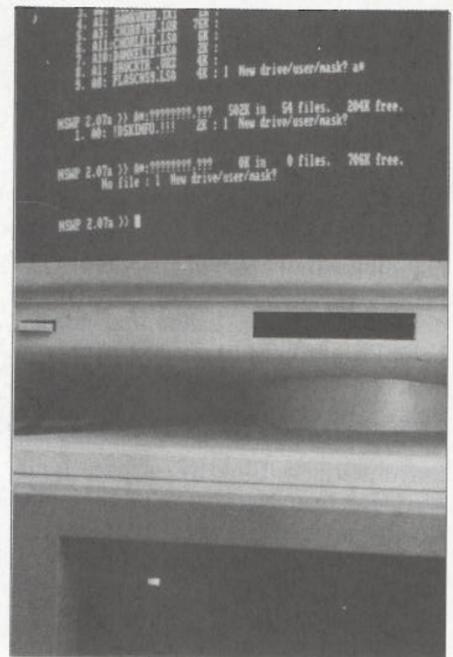


Bild 5: Geschäft!

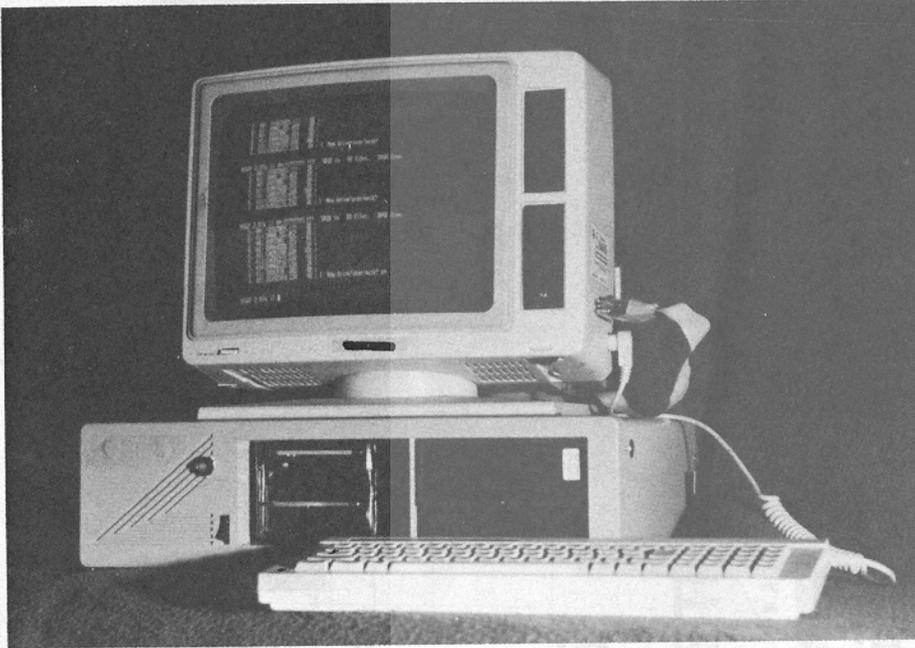
gen von Sicherungskopien seine Daten "in die falsche Richtung sichern". Fürs erste ist es ein Bombenerfolg, daß man jetzt

- mit Laufwerk A: alle in Laufwerk B: geschriebenen Disketten lesen kann.
  - Daten zwischen "großen" Disketten in einem Zug kopieren kann.
  - auf Laufwerk A: so viel Platz hat, daß es möglich wird, Utility-Disketten zusammenzustellen, die man bei der Arbeit einfach nebenbei einlegt und kaum zu wechseln braucht, weil praktisch alles für eine bestimmte Sorte Arbeit zusätzlich benötigte jetzt auf eine einzelne Diskette paßt.
- PCW booten mußte man auch bis hierher immer noch von einer Diskette im ganz normalen "kleinen" Format, da sich der Boot-Treiber (in einem ROM) leider nicht so einfach überlisten läßt wie das flexible, weil im RAM liegende, komplette Betriebssystem. Mit etwas Geschick kann man aber auch eine voll bootfähige Diskette erzeugen, die wenigstens annähernd die vollen 706 Netto-kByte zur Verfügung stellt. Mehr darüber wird in der Fortsetzung dieses Artikels verraten. Bis dahin wünsche ich viel Spaß beim Aufrüsten – und den "Profis" unter den Lesern vielleicht beim Umstricken des Floppytreibers!
- (Christian Frederking/rs)

#### Literaturhinweis:

PCW-Zweit-Floppy  
PCW Sonderheft Nr. 2, S. 76 ff  
DMV-Verlag 1987

Die Verwandlung – Mehr Speicher und so weiter am  
PCW, c't 7/86 Seite 85  
Heise Verlag 1986



## Software für den Super-JOYCE

### 720-kByte-Laufwerk A: wird bootfähig

**720 kByte auf Laufwerk A: ? Nein, kein Druckfehler, Sie lesen richtig. In der vorausgegangenen Ausgabe wurde ein kostengünstiger Ausbau des JOYCE auf zwei 'große' Laufwerke A: und B: mit jeweils 720 kByte Speicherkapazität vorgestellt, die auch mühelos untereinander Disketten austauschen können.**

**Wir zeigen diesmal, wie Sie mit dem so aufgerüsteten JOYCE von dem 'vergrößerten' Laufwerk A: auch booten können. Ihr JOYCE wird dadurch endlich ein Computer mit einem vollwertigen Laufwerkspaar, der auf die exotischen kleinen Originaldisketten nicht mehr angewiesen ist.**

Wie in dem schon erwähnten Vorläuferartikel ausführlich berichtet, wurden für den PCW-Ausbau hardwaremäßig zwei Laufwerke im 3,25-Zoll-Format verwendet.

Gegenüber dem 'normalen' 3-Zoll-Format haben sowohl diese Laufwerke als auch die Disketten den Vorteil weit aus größerer Verbreitung und zugleich erheblich niedrigerer Preise.

Die Verwendung zweier Laufwerke in demselben Format befreit Sie als PCW-Anwender zudem endlich von dem Zwang, immer darauf achten zu müssen, daß jede Diskette auch in das Laufwerk mit dem für sie richtigen Format (CF2 oder DD) eingelegt wird.

Damit können Sie jetzt auch Daten von einer 'großen' Diskette ohne Umwege und Tricks auf eine andere kopieren, ohne mittendrin Disketten wechseln zu müssen.

Um das einwandfreie Arbeiten des PCW-Betriebssystems mit den zwei großen Laufwerken zu erreichen, mußten nach der Aufrüstung zunächst einige kleine Tricks angewendet werden, denn konstruktionsbedingt erwartet das XBIOS, unter der Laufwerksbezeichnung A: immer das einseitige, 'kleine' JOYCE-Originallaufwerk vorzufinden.

Beim ersten Ansprechen einer im Laufwerk eingelegten Diskette überprüft das XBIOS, ob das vorgefundene Diskettenformat mit den angenommenen Werten des Laufwerks übereinstimmt. Findet sich in Laufwerk A: eine 'großformatige' Diskette, warnt das XBIOS den Benutzer vor einem Bedienungsfehler ('falsche Diskette') und weigert sich, Operationen mit dieser Diskette auszuführen, da es von dem Umbau ja nichts wissen kann. Mit einem Patch

im XDPB wurden diese überflüssig gewordenen Kontrollmechanismen 'überlistet', so daß Daten und Programme nun auch von dem großen Laufwerk A: in vollem doppelseitigem Format einwandfrei gelesen bzw. geladen und ausgeführt werden können.

### Der Trick: Dem PCW eine 3-Zoll-Disk vorgaukeln

Was bisher noch fehlte, war vor allem die Möglichkeit, das 'aufgebohrte' Laufwerk A: auch zum Systemstart einzusetzen und somit dem platzknappen und exotischen 3-Zoll-Format Ade sagen zu können. Vernünftigerweise möchte man ja gerne alle Dateien, einschließlich der Startsoftware, in einem einheitlichen Diskettenformat vorliegen haben, um sie möglichst einfach bearbeiten zu können.

Leider läßt sich das PCW-Betriebssystem beim Thema Booten nicht so leicht überlisten wie im normalen Betrieb, denn die dafür zuständige Software steht notwendigerweise nicht wie der XDPB im Speicher, der ja beim Booten gerade erst geladen wird! Das 'Wissen' darüber, wie der Rechner nach dem Einschalten an sein Betriebssystem auf der Diskette kommt, ist vielmehr auf einem ROM fest in PCW eingebaut. Und hier wird – wie sollte es anders sein – derjenige Laufwerkstyp zugrundegelegt, der vom Hersteller als Bootlaufwerk in den PCW eingebaut wird, eben das unwillkommene kleine CF-2-Format. Lediglich die Zahl der angeblich beschriebenen Spuren auf der Diskette kann durch einen Patch im Bootsektor verändert werden. Andere Möglichkeiten, den Bootvorgang zu beeinflussen, sind nicht vorhanden oder jedenfalls nicht bekannt. Um die Bootfähigkeit von einem anderen Format zu erreichen, muß also ein anderer Weg beschritten werden – der in unserem Fall tatsächlich gangbar ist.

### Die Überlistung der Hardware

Die Überlegung, die diesem Lösungsansatz zugrunde liegt, ist im Prinzip ganz einfach: Wenn die Software im Boot-ROM unbedingt eine Diskette im CF-2-Format sehen will, dann kriegt sie eben eine! Da es grundsätzlich keine Möglichkeit gibt, softwaremäßig festzustellen, welche räumlichen Abmessungen die Diskette hat, auf die ein bestimmtes Format aufgebracht ist, be-

deutet die Festlegung des Formats nämlich noch lange nicht, daß man dann auf die 3,25-Zoll-Diskette verzichten muß. Schließlich muß man bei einer Diskette immer zwischen ihrem 'hardwaremäßigen' Format – den Abmessungen – und dem 'Software'-Format, also der Art und Weise, in der die Daten auf ihr organisiert sind, unterscheiden. Formatiert man also eine 3,25-Zoll-Diskette einfach mit DISCKIT in dem großen Laufwerk A: als CF-2-Diskette, so läßt sie sich ohne weiteres als solche behandeln und auch als Startdiskette verwenden. Auch DISCKIT stört sich nicht daran, daß das aufgerüstete 720-kByte-Laufwerk A: dabei nur ein Viertel seiner Kapazität ausnutzt; es werden eben einfach nur 40 Spuren auf der Vorderseite der Diskette formatiert, und die Fähigkeit des Laufwerks, auch größere Formate zu bearbeiten, fällt schlicht unter den Tisch.

#### ● Booten von 3,25 Zoll

Ein Teilerfolg ist damit schon erreicht – auf das PCW-Originallaufwerk mit seinen 3-Zoll-Disketten kann verzichtet werden, und die 'getürkte' CF-2-Startdiskette kann – ein entsprechend modernes Laufwerk vorausgesetzt – dauerhaft in dem neuen Startlaufwerk verbleiben.

#### ● Booten von 720 kByte

Etwas unbefriedigend bleibt noch, daß die Diskette in diesem Format (CF-2) nur mit 40 Spuren beschrieben wird, obwohl sie 160 Spuren aufnehmen könnte. Aber auch für diesen Engpaß gibt es noch Abhilfe! Hierfür muß man allerdings in der Struktur der Diskette ins Eingemachte gehen – aber immerhin, es geht.

### Kleine Diskettenkunde

Die Daten auf einer Diskette sind unter CP/M bekanntlich in Spuren und Sektoren organisiert; die Spuren liegen als konzentrische Ringe auf der Diskette und sind wie Tortenstücke jeweils in mehrere (bei JOYCE: 9) Abschnitte oder Sektoren eingeteilt. Ein Sektor ist die kleinste Einheit, in der CP/M Daten auf der Diskette ablegen kann. Eine Datei besteht daher aus einer Kette von Sektoren, von denen jeder durch eine Spur- und eine Sektornummer eindeutig identifiziert ist.

Um nun nicht jeden Sektor einer Datei einzeln im Inhaltsverzeichnis der Dis-

kette vermerken zu müssen, faßt CP/M die Sektoren der Reihe nach zu sogenannten Blöcken zusammen und verwendet beim Speichern einer Datei immer mindestens einen kompletten Block, wenn dieser nicht ausreicht, noch einen weiteren dazu und so weiter. Dadurch wird zwar etwas Speicherplatz verschwendet, dafür wird aber das Inhaltsverzeichnis klein gehalten und kann damit schneller gelesen und leichter in einem Pufferbereich von CP/M Plus gespeichert werden.

### Unterschiede

Wenn Sie sich die Inhaltsverzeichnisse Ihrer Disketten mit DIR.COM ansehen, können Sie leicht feststellen, daß die verschiedenen großen Diskettenformate des PCW auch unterschiedliche Clustergößen haben. So arbeitet CP/M in dem 'großen' Format mit 2-kByte-Blöcken, in dem kleinen nur mit Blöcken von 1 kByte. Die Blockstrukturen beider Formate sind also nicht identisch. Hinzu kommt, daß in dem großen Format ja auch die Rückseite der Diskette unmittelbar zur Verfügung steht. Um beim Lesen und Schreiben die größtmögliche Geschwindigkeit zu erzielen, nummeriert CP/M Plus dabei die Spuren der Vorder- und Rückseite wechselweise durch: Spur 0 ist die äußerste Spur auf der Diskettenvorderseite, Spur 1 die entsprechende Spur auf der Rückseite, Spur 2 liegt wieder auf der Vorderseite und so weiter. Auf den kleinen Disketten liegen dagegen alle Spuren direkt hintereinander auf der Vorderseite als der einzigen Seite, die mit dem einen Schreib-/Lesekopf des Laufwerks erreicht werden kann. (Bei dem 'getürkten' CF-2-Format auf dem in Wirklichkeit viel größeren aufgerüsteten Laufwerk wird der zweite Schreib-/Lesekopf ignoriert.) Beide Formate sind also keineswegs "aufzeichnungskompatibel" zueinander.

Eine Aufstellung der Eigenschaften, die eine 'funktionsfähige' 720-kByte-Startdiskette haben muß, liest sich so zunächst wie eine Unmöglichkeit: Sie muß einerseits beim Systemstart als einseitige CF-2-Diskette fungieren können, auf der die zum Start notwendigen Dateien des Betriebssystems gespeichert sind, und andererseits volle 2 x 80 formatierte Spuren aufweisen, um Dateien bis zur Grenze der tatsächlichen Speicherkapazität aufnehmen zu

können. Die Aufzeichnung dieser 'zusätzlichen' Spuren soll dabei keinen faulen Kompromiß darstellen, sondern uneingeschränkt die Struktur des 'großen' PCW-Diskettenformats aufweisen, damit die Diskette sich in das Konzept, nur noch mit einem Aufzeichnungsformat zu arbeiten, einfügt. Wie lassen sich nun beide so gegensätzlichen Strukturen vereinen?

### Vortäuschen falscher Tatsachen

Die Lösung liegt in der Erzeugung einer Struktur auf der Diskette, die trotz aller Widrigkeiten beiden Formaten gerecht wird. Die Konstruktion des PCW-CP/M macht dies möglich: Es genügt hier, die beiden unterschiedlichen Strukturen nur vorzutäuschen. Das bedeutet, daß im wesentlichen gar nicht die Diskette selbst verändert wird, sondern das CP/M wird durch entsprechende Einstellung per Software dazu gebracht, das, was es physikalisch auf der Diskette findet (nämlich Spuren und Sektoren), einmal im Sinne des kleinen und einmal im Sinne des großen Formats zu interpretieren. Welche Organisation einer Diskette unterstellt wird, hängt auf dem PCW nämlich ausschließlich vom inneren Betriebszustand des CP/M ab: Solange der XDPB für das Laufwerk A: die Daten für das CF-2-Format enthält, wird die Diskette als einseitige CF-2-Diskette aufgefaßt; enthält er die Parameter des 'großen' Formats, so werden alle 160 Spuren der Diskette benutzt und die entsprechend anderen Block- und Inhaltsverzeichnisstrukturen angewandt. In jedem Fall betrachtet CP/M die ganze Diskette als Diskette in dem jeweils eingestellten Format, so daß sich die Strukturen der beiden 'gedachten' Formate überlagern.

### Physikalisches

Um dies zu verstehen, muß man wissen, daß die Zusammenfassung der Sektoren zu Blöcken, die das CP/M vornimmt, auf der Diskette nicht fest vorhanden ist; sie ergibt sich nur aus der Art, wie das System die Sektoren beim Zugriff 'betrachtet'. Auf der physikalischen Ebene haben beide Formate viel mehr Gemeinsamkeiten: Eine formatierte Diskette hat immer 9 Sektoren pro Spur und 512 Bytes pro Sektor, und die Spurnummern auf einer Diskettenseite sind immer fortlaufend; die Um-

rechnung der physikalischen Spurnummern 0 bis 79 mit den Seitennummern 0 und 1 in die logischen Spurnummern 0 bis 159 bei den 'großen' Disketten geschieht erst intern in CP/M beim Diskettenzugriff. Auch die Numerierung der physikalischen Sektoren untereinander stimmt in beiden Formaten überein.

Dagegen sind Lage, Größe und Aufbau des Inhaltsverzeichnisses und die Organisation der Cluster in beiden Formaten völlig verschieden. Das große Format bietet zum Beispiel Platz für 256 Einträge im Inhaltsverzeichnis, das kleine nur für 64, und die Darstellung der Clusternummern geschieht in dem 'kleinen' Format mit 8 und in dem 'großen' mit 16 Bit. Wenn man eine Datei in einem der beiden Formate abspeichert, liefert daher die Interpretation des entsprechenden Inhaltsverzeichniseintrags im anderen Format unsinnige Werte, und die Datei kann selbstverständlich auch nicht etwa geladen und ausgeführt werden (der Versuch führt zum Systemabsturz). Mit anderen Worten: Es ist nicht möglich, ein und denselben Platz auf der Diskette in beiden Formaten zu nutzen.

Um die Verhältnisse auf der Diskette für das Betriebssystem beherrschbar zu

machen, müssen wir also eine Unterteilung in zwei Bereiche vornehmen, von denen einer ausschließlich im 'kleinen' und der andere nur im 'großen' Format bearbeitet wird, und dafür sorgen, daß das System immer nur auf den Bereich zugreift, auf dessen Format es gerade eingestellt ist. Wie die Aufteilung der Diskette aussieht, zeigt Bild 1.

**“Divide et impera“  
(Teile und herrsche)**

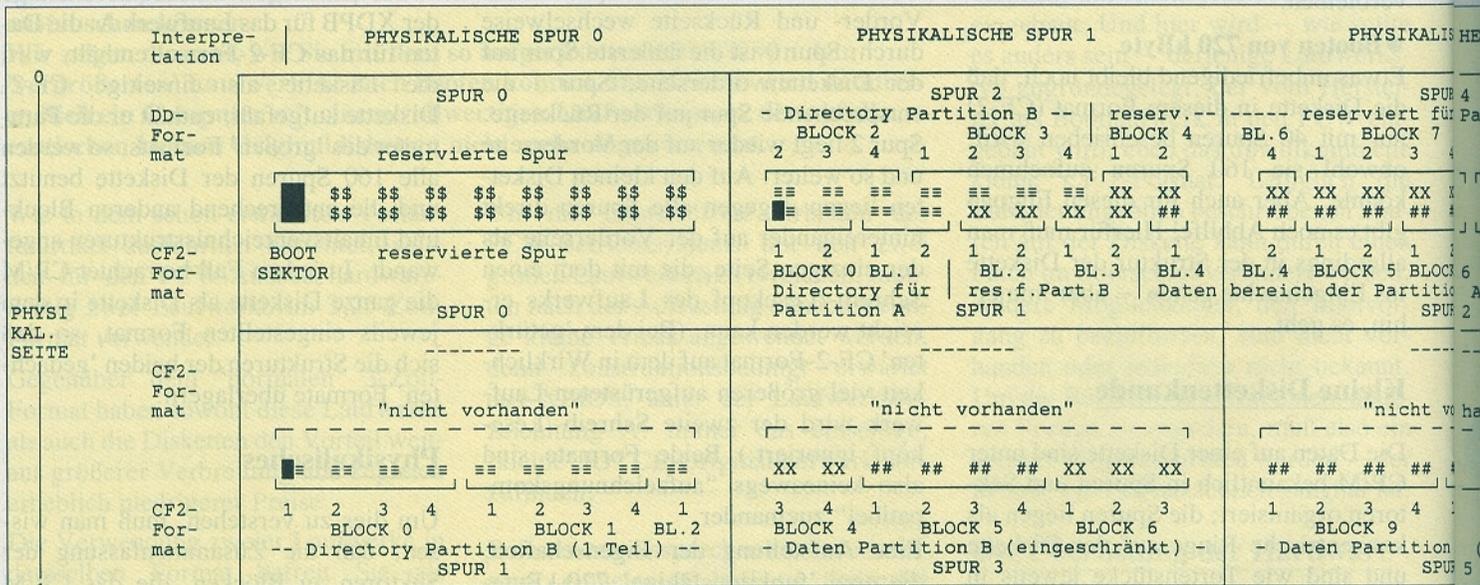
Auf dem Diagramm ist zu erkennen, daß die Datenblöcke des CF-2-Formats ausschließlich auf der Vorderseite der Diskette (Seitennummer 0) liegen und dort fortlaufend angeordnet sind, während die Blöcke des doppelseitigen Formats die Diskette in der Reihenfolge der logischen Spuren auffüllen, wie sie oben beschrieben wurde.

Dabei ist zu beobachten, daß bei beiden Formaten auch die reservierte Spur (jeweils eine) auf der Diskette identisch ist: Beide Formate reservieren die Spur 0, die immer auf der Vorderseite der Diskette liegt – das CF-2-Format hält dort gegebenenfalls den Bootsektor, im doppelseitigen Format sind nur die ersten 10 Bytes für den DPH (siehe [2])

interessant. Das Inhaltsverzeichnis beginnt jeweils am Anfang der Spur 1. Diese allerdings liegt im CF-2-Format auf der Vorderseite der Diskette, im doppelseitigen Format dagegen auf der Rückseite. Der eigentliche Datenbereich schließt sich an das Inhaltsverzeichnis an. Die unterschiedliche Lage der Inhaltsverzeichnisse ist für unsere Problemlösung von entscheidender Bedeutung, denn aus der strikten Trennung der beiden 'verschiedenformatigen' Bereiche auf der Diskette und aus der Verschiedenheit der dazugehörigen Verzeichnisstrukturen ergibt sich unmittelbar die Notwendigkeit, für jeden Bereich ein eigenes Inhaltsverzeichnis zu haben. Lägen beide an derselben Stelle, so wäre es unmöglich, Einträge in einem Format zu schreiben, ohne Daten des anderen zu zerstören.

**Gegenseitiger Zugriffsschutz**

Wie errichten wir nun die notwendigen Zugriffsschranken auf der Diskette, um eine Bearbeitung jener Bereiche zu verhindern, mit denen das System 'nicht umgehen' kann, weil es gerade auf das jeweils andere Format eingestellt ist?



Erläuterung: Im CF2-Format sind die Blöcke 0 und 1 das Inhaltsverzeichnis und alle weiteren sind Dateibereich. Das ergibt ns  
Im DD -Format sind die Blöcke 0 bis 3 das Inhaltsverzeichnis und alle weiteren sind Dateibereich. Das ergibt ns  
Deutlich ist die Überlappung auf Seite 0 Spur 1 und der Schutz des überhängenden Teils des "B"-Inhaltsverzeichnisses im Da er

- präpariert (modifiziert)
- # normaler Sektor
- = Directory-Sektor
- \$ reservierte-Spur-Sektor
- X reserviert (für jeweils andere Partition)

Abbildung 1a: Überlagerung der logischen Spurzuordnungen und der Cluster der verschiedenen Formate ("Nahaufnahme")

Eine wirkungsvolle Methode, die vor allem völlig ohne Eingriffe in das Betriebssystem auskommt, besteht darin, die 'formatfremden' Bereiche ganz einfach für belegt zu erklären. Nun ist eine Diskette in CP/M genau dann belegt, wenn sämtliche verfügbaren Cluster mit gespeicherten Daten belegt sind, und dasselbe gilt natürlich auch für einzelne Bereiche. Wir brauchen also nur eine 'künstliche' Datei im Inhaltsverzeichnis des jeweiligen Formatbereichs zu erzeugen, welche genau diejenigen Cluster belegt, die für das CP/M in diesem Format tabu sind. Jeder Bereich schützt somit jeweils den anderen, indem er ihn als eine besondere Datei betrachtet, auf die keine Zugriffe erlaubt sind. Für die Einhaltung des Zugriffsverbots sind Sie als Benutzer allerdings selbst verantwortlich, denn natürlich kann diese 'Pseudodatei' genauso gelöscht oder überschrieben werden wie jeder andere Eintrag im Inhaltsverzeichnis, was in diesem Fall äußerst unerwünschte Folgen hätte.

Um Sie dabei zu unterstützen, erzeugen wir diese Dateien, soweit möglich, in dem selten benutzten Userbereich 15 und versehen sie mit dem Read-Only- und System-Attribut sowie mit Datei-

namen, die deutlich aus dem Rahmen fallen und von CP/M nicht einmal verarbeitet werden können. (NSWEEP allerdings kann diese Dateien dennoch löschen, wenn man so unvorsichtig ist, es dazu aufzufordern.) Die Nummern der Cluster, die geschützt werden sollen, müssen, ausgehend von der Betrachtungsweise des jeweiligen Formats, von Hand ausgesucht und – zum Beispiel mit Hilfe eines Diskettenmonitors – im Inhaltsverzeichnis eingetragen werden. Diese ausgesprochene Fummelarbeit brauchen Sie allerdings nicht selbst vorzunehmen, da das Programm (Databox) bereits eine fertige Kopie des so manipulierten Inhaltsverzeichnisses enthält, die nur noch auf eine leere Diskette geschrieben werden muß.

### Organisation der Startdiskette

Natürlich sind wir daran interessiert, die Speicherplatzausbeute auf unserer präparierten Diskette so groß wie möglich zu halten. Dies ist deshalb nicht ganz ohne Schwierigkeiten, weil die Cluster des kleinen und großen Formats einander nicht glatt überlappen, sondern meistens einen oder mehrere Sektoren weit 'überstehen'. Da im

CP/M-Inhaltsverzeichnis nur mit ganzen Clustern gearbeitet wird, werden die 'überstehenden' Bereiche beim Eintragen der 'Schutzdateien' mit geschützt und sind damit für die normale Verwendung nicht mehr zu erreichen, auch wenn sie in Wirklichkeit keine Daten enthalten. Um den kritischen Überlappungsbereich möglichst klein zu halten, definieren wir daher einen Bootbereich (im CF-2-Format), der nur das absolute Minimum an Speicherplatz enthält, das für die Bootsoftware notwendig ist. Da der PCW-Bootlader, wie schon erwähnt, auch Bootdisketten mit einer von 40 abweichenden Spurzahl zuläßt, ist dies problemlos; es muß lediglich die Anzahl der eingetragenen Spuren im DPH geändert und die Prüfsumme im letzten Byte des Bootsektors entsprechend korrigiert werden. Das 'Prüfsummen'-Byte für den Bootsektor lautet bei 11 angegebenen Spuren (0Bhex) nicht mehr CF, sondern EC; wenn Sie hier etwas ändern wollen, ist für jede zusätzliche Spur von der 'Prüfsumme' 1 (hex!) abzuziehen. (Dann aber bitte das Schützen des Bereichs nicht vergessen!)

Da zum Booten das kleine Format verlangt wird, ist es notwendig, alle Da-

LISC	HE SPUR 2	PHYSIKALISCHE SPUR 10	PHYSIKALISCHE SPUR 11
PUR für 7 4	<p>4 Partition A ----- BLOCK 8 1 2 3 4</p> <p>XX XX XX XX ## ## ## ##</p> <p>1 2 1 2 BLOCK 7 BLOCK 8 A: (EMS) -&gt; BIS: 2</p>	<p>SPUR 20 ----- reserviert für Partition A ----- BL.2Ah BLOCK 2Bh BLOCK 2Ch 4 1 2 3 4 1 2 3 4</p> <p>XX XX XX XX XX XX XX XX XX XX ## ## ## ## ## ## ## ## ## ##</p> <p>2 1 2 1 2 1 2 1 2 BL.40 BL.41 BL.42 BL.43 BL.44 -- Dateien Partition A: -- ... -- ENDE SPUR 10</p>	<p>SPUR 22 -- Datenbereich der Partition B ---&gt; BLOCK 2F BLOCK 30 BL. 3 2 3 4 1 2 3 4 1 2 3</p> <p>## ## ## ## ## ## ## ## ## ##</p> <p>nicht verwendet</p> <p>SPUR 11 ( &gt; letzte angegebene Spur</p>
vorhanden"	<p>## ## ## ## XX</p> <p>2 3 4 1 BLOCK 10 BL.11 (eingeschränkt) -</p>	<p>"nicht vorhanden"</p> <p>## ## ## ## ## ## ## ## ## ##</p> <p>1 2 3 4 1 2 3 4 1 BLOCK 2Dh BLOCK 2Eh BL.2Fh  --- Daten Partition B unbeschränkt --- SPUR 21</p>	<p>"nicht vorhanden"</p> <p>## ## ## ## ## ## ## ## ## ##</p> <p>3 4 1 2 3 4 1 2 ... UND SO WEITER BIS DISK-ENDE .J --- Daten Partition B unbeschränkt ---&gt; SPUR 23</p>

bt i nsgesamt 2 KByte Inhaltsverzeichnis.  
bt i nsgesamt 8 KByte Inhaltsverzeichnis.  
Dat enbereich des CF2-Formats zu erkennen.

SEITE 0	R E S	02 03 04	06 07 08	0B 0C 0D	0F 10 11	14 15 16	18 19 1A	1D 1E 1F	21 22 23	26 27
SEITE 1	00 01 02	04 05 06	09 0A 0B	0D 0E 0F	12 13 14	16 17 18	1B 1C 1D	1F 20 21	24 25 26	28 29

Unterstrichen sind die Nummern der Cluster, deren Sektoren ausschließlich auf der Rückseite der Diskette liegen. Mit der Periode 2 physikalische Spuren. Mit derselben Periodizität fangen übrigens die DD-Cluster genau auf einem Hier sieht man, wie durch die Überlagerung der Formate der Partition B mehr Sektoren entzogen werden müssen, als Jeder Cluster auf der Diskettenrückseite, der auch noch Sektoren auf der Vorderseite belegt, also auf beide Seiten Im Überlagerungsbereich können daher im Mittel nur 33 % der Diskettenkapazität mit Daten im DD-Format belegt werden. Vorderseite der Diskette mit Daten im CF2-Format ergibt das über alles gerechnet immerhin noch 25 % Verschnitt.

Abbildung 1b: Verteilung der DD-Cluster auf die Spuren und davon eingeschlossener Überlagerungsbereich (Übersicht)

ten, die unmittelbar (!) zum Booten benötigt werden, im CF-2-Format auf der Diskette zu halten. Dies ist im Prinzip "nur" die EMS-Datei! Nach dem eigentlichen Booten soll baldmöglichst in das 'große' Format umgeschaltet werden, da es ja unser Ziel ist, in diesem zu arbeiten und von der 'kleinformatigen' Bootprozedur möglichst gar nichts zu merken.

An Daten und Dateien für den Systemstart benötigen wir also:

1. Einen ordnungsgemäß aufgebauten Bootsektor. Dieser kann, da er von DISCKIT bei 'großen' Disketten nicht erzeugt wird, mit Hilfe eines Diskmonitors auf die Diskette kopiert werden. Noch komfortabler erledigt dies das oben erwähnte Programm.

2. Die Datei J14GCPM3.EMS mit dem Betriebssystem.

Die Datei wird an einer Stelle modifiziert, um ohne Zuhilfenahme von SUBMIT das folgende Programm zu starten.

3. Ein kleines Programm, das direkt aus dem Bootvorgang heraus die Laufwerksverwaltung des CP/M auf das 'große' Diskettenformat umschaltet und anschließend SUBMIT aufruft. Dieser besondere Clou bewirkt, daß Sie Ihre gesamte persönliche Startkonfiguration auf der Diskette in dem 'großen' Format einrichten können, ohne dabei zwischen den beiden Formaten hin- und herwechseln zu müssen.

4. Einen Eintrag im Inhaltsverzeichnis des CF-2-Bereichs, der das Inhaltsverzeichnis des 'großformatigen' Bereichs schützt.

5. Einen Eintrag im Inhaltsverzeichnis des 'großformatigen' Bereichs, der den CF-2-formatierten Bereich schützt.

Um die Punkte 1, 3, 4 und 5 brauchen Sie sich hier nicht zu kümmern, wenn Sie die unten beschriebenen Programme XBOOTGEN und XSTART verwenden. Die Modifikation der EMS-Datei müssen Sie mit Hilfe eines Debuggers (zum Beispiel SID) selbst vornehmen, was nicht weiter schwierig ist.

Was XBOOTGEN mit der formatierten Diskette anstellt, sehen Sie in Bild 2. Der Ausdruck zeigt Ausschnitte aus den Inhaltsverzeichnissen mit den Schutzeinträgen (samt dem Anfang des bereits auf die Diskette kopierten Betriebssystems) und die Disk-Statusanzeige (mit DU) nach dem 'Einloggen' des jeweiligen Diskettenbereichs. (Der eine Schutzeintrag umfaßt so viele Einträge (Extents), um die 28 Cluster für 56 kByte geschützten Bereich unterzubringen.) Die Inhaltsverzeichnis-Sektoren wurden mit DU automatisch aufgesucht (g0 = Cluster 0 lesen), was unter anderem beweist, daß die Diskette tatsächlich voll kompatibel zu beiden (!) PCW-Formaten ist, ausgenommen den Disk-Parameter-Header, der natürlich nur ein Format gleichzeitig anzeigen kann. Bei der Aufnahme dieser

'Schnappschüsse' waren die beiden 3,25-Zoll-Laufwerke wie folgt in Betrieb: A: mit automatischer Formatwahl, B: mit eingefrorenem Format JOYCE-DD. Die Diskette wurde nur ein einziges Mal gewechselt (in der Mitte des Listings, nämlich von A: nach B:). Ein voller Erfolg!

### Die Erstellung der 720-kByte-Bootdisk

Für die programmunterstützte Erstellung Ihrer 720-kByte-Bootdisk erhalten Sie im folgenden ausführliche Anleitung. Als 'Abkürzung' wird dabei der Begriff 'Partition' verwendet.

Der Lohn Ihrer Mühe sind folgende Ergebnisse:

- Sie erhalten eine bootfähige 3,25-Zoll-Diskette, die nach dem unmittelbaren Bootvorgang selbsttätig unter dem 'großen' PCW-Diskettenformat zu arbeiten beginnt und auf der echte 650 kByte frei sind.

- Beim Systemstart erhalten Sie jetzt nach der Startmeldung des CP/M selbst die Startmeldung des Programms XSTART, und das System wechselt auf die DD-Partition der Startdiskette.

SEITE 0	R E S	0 1 2 3 4	4 5 6 7 8	9 ... 13	13... 17	18...
SEITE 1						

Um den Bereich der Überlagerung der Formate klein zu halten, wird für die Man sieht hier, wie im CF2-Format immer zwei Spuren eine "Gruppe" bilden, Jede 2-Spur-Gruppe enthält 9 vollständige Cluster oder 9 KB.

Abbildung 1c: Verteilung der CF-2-Cluster auf die Spuren und davon eingeschlossener Überlagerungsbereich (Übersicht)

26	27	28	2A	2B	2C	2F	30	31	.....
28	29	2A	2D	2E	2F	31	32	33	.....

entdeckt ein regelmäßiges Muster (geraden) physikalischen Spuranfang an. der Partition A verbraucht werden: verteilt ist, muß gesperrt werden. Bei lückenloser Belegung der

Wenn Sie dort eine Datei PROFILE.SUB haben, so wird diese anschließend ausgeführt, wie Sie es von 'normalen' Startdisketten schon gewöhnt sind. SUBMIT.COM muß dazu natürlich auf der Partition vorhanden sein. Die Erstellung der Bootdisk geschieht in folgenden Schritten:

1. Kopieren Sie 'auf Vorrat' auf Ihre RAM-Disk:
  - das Programm XBOOTGEN.COM
  - das Programm LOGIN.COM
  - die Datei J14GCPM3.EMS (Ihr Betriebssystem)
  - das Programm XSTART.COM
  - das Programm NSWEEP.COM (siehe unten)
  - einen Debugger, mit dem Sie gewöhnlich arbeiten, zum Beispiel SID.

Hinweis: Als Service finden Sie auf der Databox eine Version von LOGIN.COM, die nicht bei jedem Aufruf penetrant den Bildschirm löscht, und eine Version von NSWEEP.COM, die weniger Platz mit der Startmeldung und Kopfzeile des Inhaltsverzeichnisses verschwendet.

2. Modifizieren Sie die Kopie der EMS-Datei auf der RAM-Disk wie folgt:

Änderung der Datei J14GCPM3.EMS, so daß statt PROFILE.SUB die Datei BSTART.COM gestartet wird: Ab der Adresse \$59DC steht der 10-Byte-String:

```
PROFILE.S<00>
```

Dieser String ist zu ersetzen durch:

```
BSTART.CO<00>
```

Das ist alles. Speichern Sie die modifizierte Datei wieder auf der RAM-Disk ab, um sie später auf die Bootdisk kopieren zu können.

3. Formatieren Sie eine 3,25-Zoll-Diskette mit DISCKIT in Laufwerk B: als CF2DD-Diskette.

4. Stellen Sie Ihre Hardware wie folgt ein: Laufwerk B: = 3,25 Zoll, Laufwerk A: = PCW-Originallaufwerk. ACHTUNG: Stellen Sie sicher, daß alle Systemparameter (Steprate) mit dem Betrieb Ihres PCW-Originallaufwerks harmonisieren!

5. Rufen Sie XBOOTGEN auf, und folgen Sie den Anweisungen. XBOOTGEN überträgt den Bootsektor von einer 'gewöhnlichen' JOYCE-Startdiskette auf die künftige 720-kByte-Bootdisk, ändert dabei die Anzahl der eingetragenen Spuren, korrigiert die 'Prüfsumme' und erzeugt die Directory-Schutzeinträge. Anschließend erhalten Sie die notwendigen Anweisungen für das weitere Vorgehen.

Wenn Sie also keine Fehlermeldung bekommen, sollte alles in Ordnung sein.

6. Sehen Sie sich die Diskette im Laufwerk A: mit NSWEEP an. Sie sollten feststellen, daß auf der Diskette gerade 43 kByte frei sind. Kopieren Sie nun die EMS-Datei und XSTART.COM auf die Diskette (Userbereich 0), und benennen Sie XSTART.COM in BSTART.COM um. (Letzteres geht nicht mit jedem Rename-Programm, aber zum Beispiel mit der R-Funktion

von NSWEEP.) Das Inhaltsverzeichnis Ihrer Bootdisk sollte nun mit NSWEEP so aussehen:

A*:	?????????.??? :	43 K in 3 files.
		0 K free.
1.	A0:	J14GCPM3.EMS 40 K
2.	A0:	BSTART.COM 1 K
3.	A15:	Bparti*b.res 2 K

Wenn Sie die Diskette in Laufwerk B: einlegen, sollten Sie dagegen sehen:

B*:	?????????.??? :	100 K in 4 files.
		650 K free.
1.	B0:	J14GCPM3.EMS 40 K
2.	B0:	BSTART.COM 2 K
3.	B15:	Bparti*a.res 56 K
4.	B15:	Bparti*b.res 2 K

Das Startverhalten Ihrer neuen 720-kByte-Bootdisk können Sie bereits testen. Zum Test legen Sie die Diskette in Laufwerk A:, und booten Sie neu. Es sollte ungefähr erscheinen:

```
Amstrad CP/M+ [Reset resistant
RAMDISK, #01]
v 1.4, 61 K TPA, 2 Laufwerke, ... K
Laufwerk M:
XSTART V.1.27
(c) C.Frederking + DMV 1989
SUBMIT ?
```

Am Ende steht, daß SUBMIT.COM natürlich noch nicht gefunden wird, weil es noch nicht auf die Bootdisk kopiert wurde. Sie können sich aber durch Aufruf von NSWEEP davon überzeugen, daß das Laufwerk A: jetzt bereits mit der DD-Partition der Diskette arbeitet! (Nach dem Bootvorgang mit XSTART ist Laufwerk A: auf das 'große' Format 'eingefroren'. Der FREEZE-Status von Laufwerk B: wird nicht verändert.)

7. Nun können Sie Ihre 'normalen' 3,25-Zoll-Programmdisketten in das Laufwerk A: einlegen und sich durch Kopieren der entsprechenden Dateien auf die Bootdiskette in Laufwerk B: Ihre persönliche Startkonfiguration zusammenstellen. Dabei funktioniert alles genauso wie bei einer 'normalen' Startdiskette für das 'kleine' Laufwerk A: auch, nur daß Sie die DD-Partition der Diskette nicht in Laufwerk A: be-

23	23...26	27...32	32...35	36...41	41...44		

die p artition A nur ein Minimum von 11 Spuren (1 reserviert) definiert. den, d a der letzte Cluster jeder ungeraden Spur auf die nächste Spur überläuft.

schreiben können. Vor allem aber: Sie haben viel mehr Platz auf der Bootdisk!

### Wichtige Hinweise

Sollten Sie den Systemstart ausprobiert haben und anschließend Software von CF-2-Disketten auf die Bootdisk kopieren wollen, vergessen Sie bitte nicht, mit UNSPEED vorher die Steprate wieder herunterzusetzen! XSTART verkürzt sämtliche Zeitintervalle der Laufwerkssteuerung drastisch auf die Werte, die einem modernen Laufwerk (3,25 Zoll) zuzumuten sind. Das PCW-Originallaufwerk reagiert darauf aber so sauer, daß es zu einem Systemabsturz kommt.

Das Programm XSTART.COM darf "niemals" von der DD-Partition aus aufgerufen werden! Das Ergebnis wäre ein Systemabsturz, der nur durch Abschalten des Rechners zu beheben ist. Es empfiehlt sich aber auch nicht unbedingt, XSTART.COM von der CF-2-Partition aus von Hand zu starten. Ausnahme: Wenn Sie mit einer nicht präparierten EMS-Datei booten, ist der einmalige Aufruf von Hand notwendig. Wenn Sie wissen, was Sie tun, können Sie XSTART.COM auch benutzen, um die Formate und FREEZE-Flags der beiden Laufwerke einzustellen (jeweils: JOYCE DD-Format, FREEZE).

Aufgrund der inneren Auslegung von JOYCE, bei der zwei gleich große Laufwerke eigentlich nicht vorgesehen waren, ergeben sich einige geringe Einschränkungen in der Benutzung, die man kennen sollte.

Zunächst **MERKE**:

- Nur Laufwerk A: kann im CF-2-Format schreiben (CF-2-Partition).
- Nur Laufwerk B: kann im 720-kByte-Format schreiben (DD-Partition).

Beim Kopieren von Dateien auf eine 'große' Diskette beziehungsweise auf die DD-Partition der Bootdisk muß die Zieldiskette also immer in Laufwerk B: liegen. Umgekehrt muß beim Kopieren auf eine 'kleine' Diskette oder auf die CF-2-Partition der Bootdisk das Ziellaufwerk immer A: sein. Falls Sie trotzdem Fehlermeldungen bekommen, sollten Sie sich vergewissern, ob das FREEZE-Flag richtig gesetzt ist, da der DPH auf der Bootdisk das System auf CF-2 einstellt, wenn er kann.

Hinweise dazu:

Auf "A:“-Laufwerk oder -Partition immer mit LOGIN NOFREEZE schreiben (Lesen großer Formate von A: jedoch vorzugsweise mit FREEZE).

Auf "B:“-Laufwerk oder -Partition immer mit LOGIN FREEZE schreiben (Format eingefroren auf JOYCE DD).

Bei Irrtum wehrt sich das System heftig. (Hier zum Glück !)

Wenn Sie auf der 720-kByte-Bootdisk mit NOFREEZE arbeiten, egal in welchem Laufwerk, haben Sie immer die CF-2-Partition vor sich, ob Sie wollen oder nicht; denn die Wahl der aktuellen Partition hängt nur von der aktuellen Einstellung des XDPB ab. Im Zweifelsfall informieren Sie sich mit Hilfe von NSWEEP darüber, auf welcher Partition Sie sich befinden.

Im Inhaltsverzeichnis von NSWEEP kann man die Schutzdatei  $\beta$ parti\*b.res von beiden Partitionen aus sehen. Hier ist kein Druckfehler vorhanden, das scharfe S gehört zu den Dateinamen dazu, um sie auffällig zu machen und in der Sortierfolge von NSWEEP 'am Rande' zu halten! Der Stern bewirkt, daß einige CP/M-Routinen mit der Datei Schwierigkeiten haben – sollen sie auch, schließlich geht sie das nichts an.  $\beta$ parti\*a.res erscheint dagegen nur im Inhaltsverzeichnis der DD-Partition, weil es außerhalb des Überlappungsbereichs der Inhaltsverzeichnisse liegt.

Tip zum Einstellen eines Laufwerks auf die DD-Partition (eingefroren): LOGIN NOFREEZE; normale 720-kByte-Disk in Laufwerk B: einloggen; LOGIN FREEZE. Jetzt kann die DD-Partition der Bootdiskette bearbeitet werden. Nach dem Systemstart mit

```

:la                                     :lb
:~                                       :~
Statistics for drive A:                 Statistics for drive B:
Tracks:          11          00B       Tracks:          160         00
Sys tracks:      1          01         Sys tracks:      1          01
Sec/track:       36          24         Sec/track:       36          24
Groups:          44          2C         Groups:          356         0164
Dir groups:      2          02         Dir groups:      4          04
Sec/group:       8          08         Sec/group:       16          10
Dir entries:     64          40         Dir entries:     256         0100

:g0:d                                   :g0:d
G=00:00, T=1, S=1, PS=0                G=0000:00, T=1, S=1, PS=0
00 0F7E7061 7274692A 62F2E5F3 00000010 *..parti*bres....*
10 02030000 00000000 00000000 00000000 *.....*
20 004A3134 31435040 3345CD53 00000080 *.J141CPM3EMS....*
30 04050607 08090A0B 0C000E0F 10111213 *.....*
40 004A3134 31435040 3345CD53 01000080 *.J141CPM3EMS....*
50 14151617 18191A1B 1C1D1E1F 20212223 *.....!*#.f.&.'*
60 004A3134 31435040 3345CD53 02000040 *.J141CPM3EMS...$*
70 24252627 28292A2B 00000000 00000000 *$%&'()*+.....*
00 0F7E7061 7274692A 61F2E5F3 00000080 *..parti*ares....*
10 04000600 07000800 0B000C00 0D000F00 *.....*
20 0F7E7061 7274692A 61F2E5F3 01000080 *..parti*ares....*
30 10001100 14001500 16001800 19001A00 *.....*
40 0F7E7061 7274692A 61F2E5F3 02000080 *..parti*ares....*
50 1D001E00 1F002100 22002300 26002700 *.....!*#.f.&.'*
60 0F7E7061 7274692A 61F2E5F3 03000040 *..parti*ares...$*
70 28002A00 2B002C00 00000000 00000000 *{.*+.....*

```

Bild 2: Status und Inhaltsverzeichnis nach dem Präparieren

XSTART ist die Einstellung automatisch so.

Wegen des CF-2-DPH können bootfähige 720-kByte-Disks leider nicht mit DISCKIT kopiert werden, sondern müssen jedesmal neu erstellt werden. Abhilfe könnte eine auf 160 Spuren aufgebohrte Version eines der Formatier- und Kopierprogramme schaffen, die zum Beispiel im Joyce Sonderheft 3/88 vorgestellt wurden.

Die Inhaltsverzeichnisse der Partitionen überlappen sich gegenseitig. Dies macht jedoch nichts, solange man nicht von einer Partition aus versucht, auf etwas zuzugreifen, was zu der anderen Partition gehört. NSWEEP zeigt auf der DD-Partition der 720-kByte-Bootdisk nicht die korrekte Menge belegten Speicherplatz an (den freien dagegen schon), weil infolge der Überlagerung der Partitionen doppelt reservierte, also geschützte Cluster doppelt gezählt werden.

Das Programm XBOOTGEN wurde in Turbo Pascal 3.0 geschrieben und ist im Ablauf selbsterklärend. Seine Funktion beruht im Kern auf der Verwen-

dung von XBIOS-Diskettenroutinen. Die erzeugten Inhaltsverzeichnisseinträge sind im Programm 'wörtlich' gespeichert; die gespeicherten Daten haben ihren Ursprung in einem Prototyp der 720-kByte-Bootdisk, einer per Diskmonitor 'zu Fuß' erzeugten Diskette, die bereits genau dieselben Qualitäten aufzuweisen hatte. Für Programmier-Freaks dürften die Bibliotheksroutinen interessant sein.

### Die Programme von der Databox

Das Programm XSTART wurde in 8080-Assembler geschrieben und mit MAC und LINK übersetzt und gebunden. Das eigentliche Programm ist kurz, da extensiv mit Makros (weitgehend selbsterklärend) gearbeitet wurde. Benutzer anderer Laufwerke als moderner 3,25-Zoll-Laufwerke werden möglicherweise die Zeitintervalle für die Laufwerksteuerung am Anfang des Quellcodes verlängern wollen; der Aufruf des Makros *dspeed* kann aber auch einfach aus dem Programm hin-

ausgeworfen werden. Der Aufruf von SUBMIT PROFILE wird mit der BDOS-Funktion 47 erreicht.

### Abschließend

In diesem Artikel haben wir gezeigt, wie sich unter Ausnutzung der Flexibilität von CP/M Plus mit dem PCW so einiges machen läßt. Das Konzept ist noch weiter ausbaufähig: Denkbar wäre etwa ein Disk-Kopierprogramm von A: (720 kByte) nach B:, ein Formatierprogramm, das von vornherein bootfähige 720-kByte-Disketten erzeugt, ein Programm zum Lesen und Schreiben von 3,25-Zoll-(720-kByte-) MS-DOS-Disketten... Uns bleibt vorerst, Ihnen mit Ihrem bootenden Super-Joyce viel Spaß zu wünschen.

(Christian Frederking/rs)

Aufgrund des Umfangs der Listings wurde bewußt darauf verzichtet, diese im Heft abzdrukken, damit auch Leser, die nicht an diesem Artikel interessiert sind, die Möglichkeit haben, aus der PC Amstrad für Sie interessante Artikel zu entnehmen.

## Ordnung und Übersicht schaffen die beliebten DMV Sammelmappen



Bitte Bestellkarte benutzen

DMV Verlag · Postfach 250 · 3440 Eschwege



## Diskettenstationen

CPC, Joyce 8256 ,8512 u. 9512, Euro-PC, Amstrad 2086  
Externe Laufwerke - Profidesign - Internes Netzteil  
bedienungsfreundlich - sehr leiser Lauf



\* Testbericht 11/89 "Komfortabler Mitläufer" \*

CPC 5.25" 820 KB vollkompatibel zu Basic, CPM 2.2  
u. CPM Plus, umschaltbar auf 180 KB, komplett incl. Software DM: 309.00  
Joyce 5.25" 720 KB, problemlos u. schnell anschließbar mit Anleitung,  
wie ein eingebautes 3" B - Laufwerk zu betreiben DM: 299.00  
Umschalter f. Teandrivecopy 80/40 Spuren DM: 20.00  
Speicheraufrüstung 256 KB auf 512 KB origin. 257 Chip DM: 120.00  
Spezialumschaltkabel f. 8512 B-Intern zu B-Extern DM: 30.00  
problemloser Betrieb von 2 B-Laufwerken  
Teandrivecopy CPM-MS-Dos Disk.-Kopierprg. f. 360/720 KB DM: 59.00  
beide Richtungen, incl. DOS-Formatierer 360/720 KB u. Umlautkonverter

Preise zzgl. Porto u. Verpackung, Versand p. Nachnahme, Liste kostenlos  
Krebs electronic Datentechnik-Hard- u. Software 6751 Weilerbach  
Tel. 06374-6878 BTX 063744432

## BASIC-Programme gesucht!

Für unsere ständige PC-Rubrik suchen wir BASIC2-Programme sowie Tips & Tricks.

Alles, was Sie tun müssen, ist, Ihr selbstgeschriebenes Programm mit einer Bedienungsanleitung als Textdatei auf Diskette zu speichern und uns diese zuzusenden.

Als Lohn für Ihre Mühe winkt bei Veröffentlichung ein interessantes Honorar.

Übrigens liegen die besten Programme meist in den Schubladen (wo sie absolut nichts zu suchen haben) und werden aus fehlender Überzeugung nicht eingesandt.

Da wir grundsätzlich jedes Programm ausführlich begutachten, könnte Ihre Einsendung, versehen mit unseren Verbesserungsvorschlägen, vielleicht der Hit des nächsten Monats werden.

Also, auf bald...

Einsendungen bitte an den

DMV-Verlag · PC-Redaktion  
Postfach 250 · 3440 Eschwege