

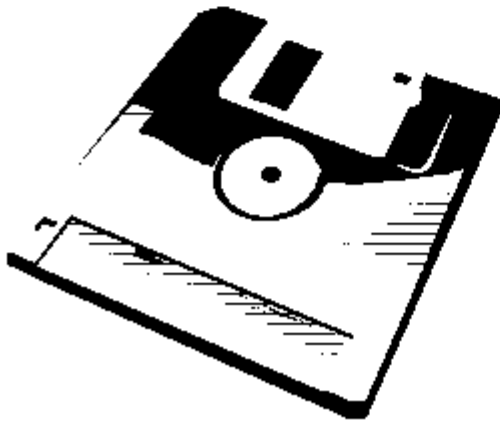
B-View 7

Das Fanzine rund um BonnyDOS/286 und GOS

Programmieren mit
dem A86-Assembler

Die BonnyDOS-API
Teil 1

BDOS trifft
MS-DOS



...und mehr!

Made with
GOS-Publisher

Editorial

Hallo und willkommen zur Ausgabe 7!

Da ist sie mal wieder – die Seite mit der Kaffeetasse. Das Vorwort zur aktuellen Ausgabe. Das Sommerloch haelt zwar weiterhin an – doch gibt es trotzdem wieder einmal einiges zu berichten. Da waere zum Beispiel der erste Assembler fuer BonnyDOS/286, der auch noch direkt unter diesem Betriebssystem einsetzbar ist. Zum Anderen ist da beispielsweise LEDIT, der Texteditor von WoBo. Er liegt in einer neuen Version vor.

Ansonsten beschraenkt sich diese Ausgabe auf pures Wissen. Passend zum A86-Assembler beginnt in der aktuellen B-View die Uebersicht ueber die wichtigsten BonnyDOS/286-API-Funktionen. Auch Turbo Pascal-Programmierer koennen davon profitieren. Reinschauen lohnt sich auf jeden Fall.

Da auf bisher keiner Ausgabe ein Feedback erfolgte (auch nicht auf die Frage in Ausgabe 6, wie die Zukunft von GOS / BDOS aussehen soll), bleiben nur zwei Schlussfolgerungen. Die erste, dass alles so wie es ist in Ordnung ist, und die B-View keine Kritikpunkte offen laesst. Die zweite, dass es keine Leser gibt.

Es bereitet – trotz des geringen Umfangs – recht viel Arbeit, die Artikel zu schreiben und anschliessend zu Layouten. Eine Arbeit, die zwar gerne verrichtet wird, aber nicht fuer die sprichwoertliche Katz sein soll. Es bleibt daher zu hoffen, dass es einmal einige Rueckmeldungen in Form von "Leserbriefen" oder gar Artikeln gibt. Ansonsten duerfte dies eine der letzten Ausgaben sein (zumindest fuer dieses Jahr).

Also ran an die Tastaturen – und los gehts! Doch jetzt heisst es wieder einmal mehr: Viel Spass beim Lesen dieser Ausgabe!

T. Brueggemann



A86-Assembler im Test

Programmieren direkt unter BonnyDOS/286 – bisher ein Traum. Doch jetzt wird dieser Traum zur Realitaet! Dank dem A86, einem 8086-Assembler kann jetzt jeder, zumindest in Maschinensprache, unter BonnyDOS/286 kreativ taetig werden. Lesen Sie, wo die Staerken und Schwaechen des Assemblers liegen.

Die Systemvoraussetzungen sind gewohnt niedrig. Eine 80286 CPU genuegt, um das Programm nutzen zu koennen. Disketten-Benutzer ohne Festplatte sollten von der BonnyDOS-eigenen RAM-Disk aus das Programmierhandwerk ausueben. Zwar reicht auch ein Diskettenlaufwerk, dies kann jedoch schnell zur Geduldsprobe werden. Wer weniger als 1 MB RAM besitzt, und daher DEVB nicht nutzen kann, der kann mittels RESIDENT den Editor, sowie den Assembler in den Speicher laden.

Als Editor kann entweder ED.APL (gehört zu BonnyDOS), oder LEDIT.APL (von Wobo) verwendet werden. LEDIT kann hier jedoch voll ueberzeugen und seine Staerken ausspielen. Lediglich Internet-Faule werden wohl auf ED zurueck greifen, da er direkt mit BonnyDOS ausgeliefert wird. Die Editoren sollen uns aber nicht weiter kuemmen – A86 heisst der Testkandidat.



Der Assembler beherrscht den 8086-Befehlssatz, das heisst, das alles, was ueber diese Instruktionen hinaus reicht, nicht angewendet werden kann. Zwar ist ein Umweg ueber den Pseudo-Opcode DB (dazu spaeter mehr) moeglich – empfohlen wird dies aber nicht. Ebenso unbekannt sind alle 8087-Befehle. Somit lautet das Motto: FPU ist tabu! Zwar duerfte das allen Einsteigern nicht viel ausmachen, Profis hingegen duerfte es sauer aufstossen.

Der Assembler fuehrt ebenfalls keine Berechnungen durch. Komplexe Ausdruecke wie z. B. MOV label+100,AX funktionieren daher nicht. Hier ist man also voll und ganz auf das Wohlwollen des 8086-Befehlssatzes angewiesen. Nun, vielleicht kommen dann auch einmal so selten genutzte Kommandos wie XLAT zum Einsatz? Kommen wir zum wichtigsten – dem Endergebnis der Arbeit von A86. Der Assembler selbst erzeugt reine Binaries – also Code und Daten, der auf kein Betriebssystem zugeschnitten ist. Dies heisst, man kann APL-Dateien, ebenso wie COM-Dateien kreieren.

Ebenfalls ist die Ausgabedatei auf ein Segment, und somit auf maximal 64K Groesse beschraenkt. Unter BonnyDOS/286 ist es jedoch kein Problem, mehr Speicher zu verwenden (z. B. fuer Daten) – dazu bedarf es am Anfang des Programmes lediglich der Auswertung von BX, das bekanntlich das Segment beinhaltet, an dem die residenten Programme/Treiber beginnen. Rein

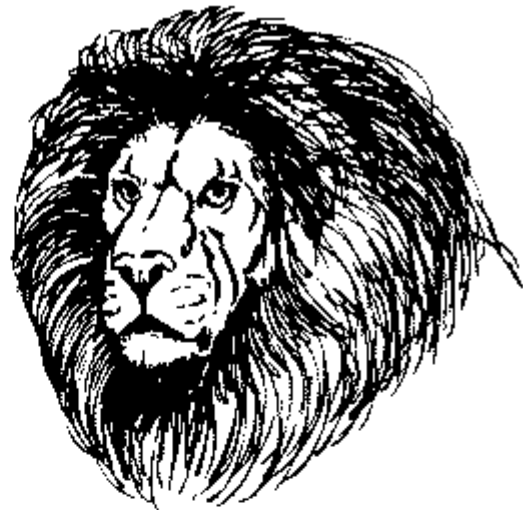
theoretisch spielt die Groesse der Quelldatei keine Rolle - allerdings gibt es unter BonnyDOS/286 bisher keine Moeglichkeit, Texte zu erstellen, die laenger als 64K (ED, LEDIT) bzw. 287K (WritePad) sind.

Die Syntax des Assemblers lehnt sich weitgehend an den 8086-Standard und den gebraeuchlichen Pseudo-Opcodes an. Es gibt jedoch einige Abweichungen, die dem Profi vielleicht hier und da Kopfzerbrechen bereiten. Statt z. B. MOV WORD schreibt man MOVW. Byteweisen Zugriff erhaelt man durch Anhaengen eines "B". In der Yahoo-Group zu BonnyDOS gibt es gleich ein paar Beispiel-Sourcecodes zum Download. Das Reinschauen lohnt sich. Als weiterfuehrende Literatur wird das BonnyDOS Kernel Reference Manual ans Herz gelegt. Dies gibt es fuer den Entwickler auf Nachfrage in der Y!-Gruppe.

Das Fazit faellt gemischt aus. Zum Einen steht jetzt endlich eine Programmiersprache unter BonnyDOS zur Verfuegung (abgesehen vom minimalistischen TBASIC), zum Anderen ist dies ausgerechnet Assembler - eine Sprache, die nicht jedermanns Sache ist, wohl aber als Ausgangspunkt fuer weitere Sprachen (und Compiler) dienen kann. Und wer weiss schon, was die Zukunft bringt. Einsteiger sollten auf jeden Fall einen Blick auf A86 werfen. Das damit gewonnene Wissen laesst sich ueberall nutzen.

A86 facts

- 500 Symbole
- 8086 Op codes
- 8087 nicht unterstuetzt.



LEDIT der loewenstarke Editor

Vor einiger Zeit brachte Wobo einen Texteditor Namens LEDIT (Lousy Editor) heraus. Gedacht war er als Alternative zum BonnyDOS-eigenen ED. Nach einer Pause erscheint nun die Version 10a. Ist das Programm zurecht als V10 deklariert? Sie erfahren es hier!

Wer mir ED schon einmal gearbeitet hat, der weiss, dass es unter Umstaenden recht schwierig ist, laengere Texte zu bearbeiten. Gerade das Einfuegen von Textpassagen artet in Arbeit aus. Da koennen auch der Rahmen-Modus und die Ueberschriften (die nur in ED als solche angezeigt werden) nichts daran aendern. Sicher - als Alternative koennte man jetzt GOS/286 und WritePad heranziehen. Doch das dies garnicht notwendig ist - das beweist LEDIT.

Der puristische BonnyDOS-User bekommt hier einiges geboten, was er garnicht gewohnt ist. Eine deutsche Tastaturbelegung. Sie haben richtig gelesen. LEDIT erlaubt das Umschalten zwischen englischer und deutscher Tastatur. Ganz ohne Treiber. Bis denn einmal ein deutsches Tastaturlayout fuer BonnyDOS erhaeltlich ist (SETMAP), erweist sich gerade dieses Leistungsmerkmal als sehr nuetzlich.

Auch kann LEDIT im Text enthaltene Umlaute durch 'Ae Oe Ue' usw. ersetzen. Daneben ist die korrekte Dekodieren von Windows-Umlauten in beide Richtungen (Win-BDOS und BDOS-Win) moeglich. Einzig die Funktion, alle Pascal-Direktiven in Gross-Buchstaben zu wandeln blieb bisher unklar. Die Groesse eines Textes unter LEDIT

ist, wie unter ED, auf 64K, also 800 Zeilen zu je 80 Zeichen, beschränkt. Als Hindernis ist dies jedoch nicht zu sehen. Textblöcke lassen sich markieren und in eine Zwischenablage kopieren. Diese Ablage kann man beliebig oft in beliebige Dateien einfügen. LEDIT ist als ausgereifter Editor anzusehen - auch das Erscheinungsbild präsentiert sich ordentlich. Zu keiner Zeit ist der Anwender überfordert oder von Funktionen erschlagen. Per Funktionstasten oder sogenannten Hotkeys (Tastenkürzel) können alle Menüpunkte aufgerufen werden. Sehr gut gefallen hat die Load-Box. Dateinamen können bequem per Pfeiltasten ausgewählt werden.

Vor jedem wichtigen Schritt fragt LEDIT zur Sicherheit nach, ob man sich der gewählten Funktion sicher ist (z. B. vor dem Überschreiben von Dateien). Dies hätte man ruhig auch in ED einbauen können. Überhaupt lies sich LEDIT kein einziges mal 'aus der Ruhe' bringen oder verursachte einen Absturz, geschweige denn einen Datenverlust. Die Version 1.0 trägt LEDIT somit zurecht, wobei meiner Meinung nach die 1.0a-Phase nicht mehr zeitgemäß ist, denn dazu ist das Programm einfach zu ausgereift.

Gibt es etwas zu verbessern? Diese Frage ist schwer zu beantworten. Im Großen und Ganzen ist LEDIT so wie es ist fertig. Das Arbeiten damit geht leicht von der Hand (gerade in Verbindung mit dem 86-Assembler). Eventuell könnte man das Programm optisch etwas überarbeiten, und vielleicht eine Maussteuerung erlauben. Doch ist dies nicht unbedingt notwendig und zeugt eher vom persönlichen Geschmack.

Für alle, die LEDIT noch nicht auf Diskette/Festplatte haben kann das Fazit nur lauten: Zugreifen, downloaden! Das Programm darf auf keinen Fall im Verzeichnis o:SYSTEM fehlen! Legt man unbedingt Wert auf Texte mit einem größeren Umfang, kann man zur Not auf WritePad zurück greifen, sofern man GOS benutzt. Wobei... LEDIT für GOS wäre auch eine prima Sache...

...damit sollte der Test eigentlich abgeschlossen sein. Doch flatterte aus heiterem Himmel die Version 1.02 ins Haus! Diese verwendet jetzt den kompletten freien Speicher (am Testrechner blieben gute 26K übrig), um darin Texte abzulegen. Sage und schreibe 3000 Zeilen sind ohne weiteres möglich! Damit hebt sich LEDIT entgeltlich an die Spitze der BonnyDOS-Editoren!

LEDIT Facts

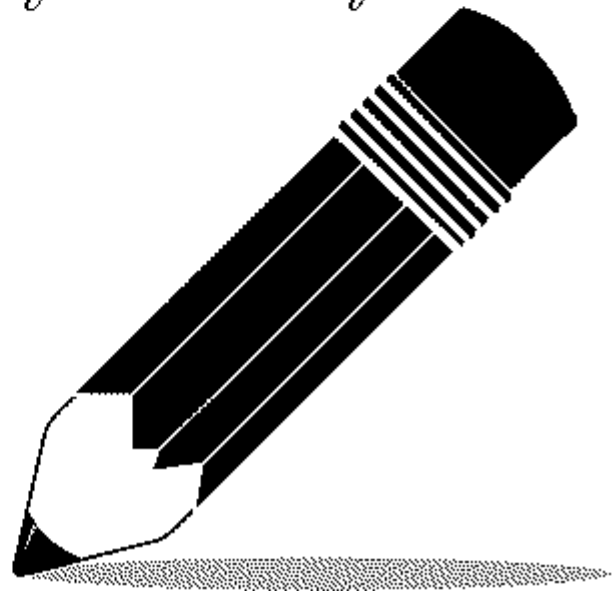
- Unter TurboPascal entwickelt
- Deutsche Tastatur
- Texte mit 3000 Zeilen möglich!
- Freeware
- Viele Optionen, ausgereiftes Programm

Leserecke

...die Seite von Lesern, fuer Leser

Die Leserecke sollte ja eigentlich schon lange ein fester Bestandteil dieses Fanzines sein. Sicher - viele BonnyDOS user gibt es nicht - doch koennte die Resonanz eine groessere sein. Wie dem auch sei. Haben Sie Fragen, Probleme oder gar Ratschlaege und Loesungen auf bestehende Probleme? Egal ob diese BonnyDOS betreffen oder es einfach nur darum geht, wie man bestimmte Hardware z. B. unter PASCAL programmiert.

Schreiben Sie einfach eine kurze Mail samt Ihrem Text. Sobald als moeglich erscheint dieser in der naechsten B-view! versprochen. Jetzt bleibt also nur noch das Zuruecklehnen und warten. Eins ist auf jeden Fall sicher. Der Winter kommt auch wieder - und dann werden die Tage draussen kuerzer, aber am PC da fuer laenger.



Kurzmeldungen

Und ewig lockt der SX

Um die Testbedingungen - aber auch die Abstimmung von Entwicklungen - zu vereinheitlichen, wurde der BonnyDOS- und GOS-Rechner zurueck geruestet. Der 486DX/2-50 sagt 'Auf Wiedersehen' und gibt die Leitung der Maschine wieder an den eigentlichen Besitzer, den 486SX-25 ab. Grund hierfuer ist die staendige Weiterentwicklung von GOS und zugehoeriger Software.

Schliesslich darf das absolute Minimum, ein 80286, nicht aus den Augen verloren werden. Ebenfalls sollen Fehler nicht mehr passieren, wie sie bei der Entwicklung zu A86 auftraten; Da wird ein Programm angeboten, das einen 8087 voraussetzt, obwohl es diesen garnicht braucht. Dank 486DX/2 merkt man dies beim Kompilieren nicht - erst auf einem 386er ist es letztlich aufgefallen.

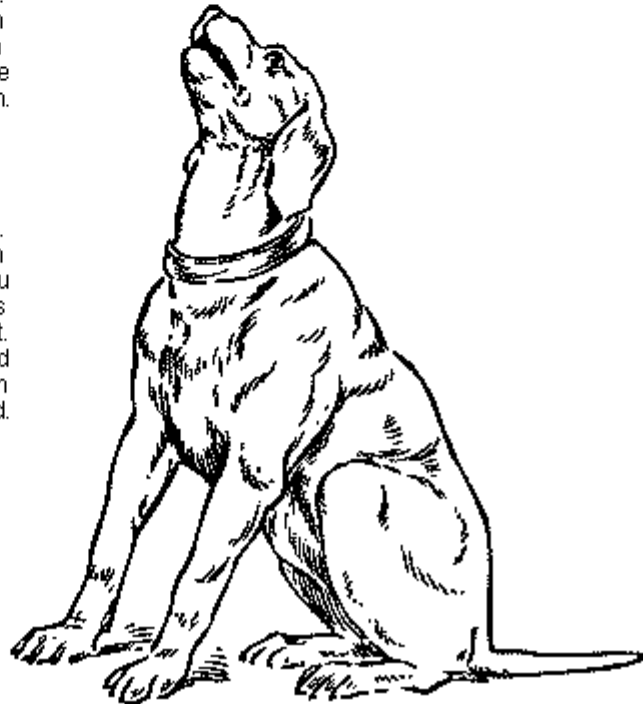
Der naechste, logische Schritt, wird die Anschaffung eines 80286er sein. Erst, wenn dort alle Tests erfolgreich verlaufen sind, kann man sich an neue Projekte wagen. Vorher heisst es eben Optimieren, Optimieren und nochmals Optimieren!

Der GOS/286 Dateimanager

Das neue GOS-Release steht in den Startloechern. Die Arbeiten am neuen Dateimanager fuer GOS kommen gut voran. Lediglich das Kopieren von Dateien muss noch integriert werden. In Kuerze ist das Programm erhaeltlich und wird in das bestehende GOS-Paket integriert werden.

GOS-Handbuch

Ob man es brauchen wird, steht in den Sternen. Fakt ist aber, dass das GOS/286-Handbuch ebenfalls fast fertig gestellt ist. Es wird, genau wie die BonnyDOS-Dokumentation, als PDF-Datei kostenlos zur Verfuegung gestellt. Einige Kapitel muessen noch ueberarbeitet und aktualisiert werden. Dann gibt es die ersten Exemplare zum Download.



Die BonnyDOS-API

Teil 1

Erstmals mit dem A86-Assembler ist es moeglich, direkt unter BonnyDOS eigene Programme zu schreiben. Daher wird es Zeit, den Programmierer mit den wichtigsten BonnyDOS-Funktionen vertraut zu machen. Natuerlich profitieren auch Pascal-Programmierer davon – denn auch die BDOS286.TPU ist noch lange nicht das Ende der Fahnenstange.

Bei der Entwicklung von BonnyDOS wurde dafuer gesorgt, dass sich Funktionen, die einen bestimmten Bereich zugehoeren, auch einen gemeinsamen Interrupt bekommen. Somit unterscheidet sich BonnyDOS in diesem Punkt von MS-DOS, dass in der Regel nur den Interrupt \$20 und \$21 benutzt. Die BonnyDOS- Interrupts reichen von \$F1 bis \$F5. Der Interrupt \$F1 beinhaltet alles, was zur Textausgabe am Bildschirm gehoert. Interrupt \$F2 kuummert sich um die Tastatur. Bei Interrupt \$F3 sind Kernel-Spezifische Dinge erreichbar. Der Extra-Interrupt \$F4 ist die einzige Ausnahme; hier sind alle Routinen aufrufbar, die sich sonst nirgends einordnen lassen (z. B. Kommandozeilen-Parameter auswerten). Letztendlich bleibt noch der Interrupt \$F5, der wohl den groessten Umfang hat. Hier tummelt sich alles, was zum Thema Laufwerk und Dateien gehoert.

Beim Aufruf ist darauf zu achten, dass in den Bits 15-8 des Akkumulators (AH) die Funktionsnummer zu finden ist. Sind alle weiteren notwendigen Parameter festgelegt worden, kann der Aufruf durch den INT-Befehl erfolgen. Das gelieferte Ergebnis haengt natuerlich von der jeweiligen Routine ab. Nicht immer (aber meistens) kann das Resultat am niederwertigen Teil des Akkus (AL) abgeleitet werden. Programmieren Sie irgendwann einmal Software, die sich in die Interrupt-Vektoren einklinkt, so ist darauf zu achten, dass auch BonnyDOS selbst seine eigenen Interrupts verwendet – Fehlermeldungen also beispielsweise per API ausgegeben werden.

Merken Sie sich also auf jeden Fall den Original-Vektor, bevor Sie ihn 'verbiegen' – sonst koennte dies einen handfesten Absturz zur Folge haben. Wir beginnen in diesem Teil eins mit dem Kernel-Pack (\$F3). Warum nicht mit dem Text-Pack \$F1? Nun – die (im Normalfall) Tastatureingabe und Bildschirmausgabe sind auch ueber \$F3 zu erreichen. Allerdings koennte man ja ein abweichendes Ein-/Ausgabegeraet verwenden. Was dann? Der Interrupt \$F1 arbeitet ausschliesslich mit dem Bildschirm zusammen. \$F2 nur mit der Tastatur. Ueber die Kernel-Funktionen verwendet man jedoch automatisch das aktive Ein-/Ausgabegeraet. Der Programmierer muss sich also um nichts kuummern.

Gewoehnen Sie sich also an, wann immer es moeglich ist, \$F3 fuer die Textein-/ausgabe zu verwenden. Erst, wenn Ihr Programm keine andere Wahl hat (z. B. ein schnell scroller Texteditor), dann koennen Sie entweder direkt auf den Textspeicher zugreifen, oder ueber INT \$F1 den Bildschirm verwenden. Bei normalen Kommandozeilen-Programmen ist dies aber bestimmt nicht notwendig. Beginnen wir also unseren Streifzug durch das Kernel-Pack.

Funktion 0 ist, mit Ausnahme vom Extra-Interrupt \$F4, immer Tabu und wird nur von BonnyDOS selbst zur Initialisierung des jeweiligen Packs aufgerufen. Vermeiden Sie es, diese Funktionsnummer zu verwenden. Im schlimmsten Fall gehen Ihnen einige Daten von offenen Files verloren! Die Funktion 1 gibt das Zeichen in AL auf das aktuelle Ausgabegeraet aus. Analog dazu liest Funktion 2 ein Zeichen vom aktuellen Eingabegeraet und gibt dieses in AL zurueck. Als Vorgabe entspricht dies dem Warten auf eine Taste.

Funktion 9 gibt einen String, auf den ES:SI zeigt, und der mit einem Nullbyte abgeschlossen sein muss, auf das aktuelle Ausgabegeraet aus. Sie sollten beachten, dass grundsaeztlich ALLE Strings unter BonnyDOS vom Typ 0-Terminated String sein muessen. Mittels Funktion 3 koennen Sie bestimmen, ob eine Ausgabe nicht nur auf das aktuelle Ausgabegeraet, sondern auch auf dem Bildschirm erfolgen soll (SET SCREEN ECHO). AL mit 0 geladen schaltet es ab, AL mit einem Wert groesser 0 geladen, schaltet es an.

Die Funktion 4 gibt den Status des SCREEN ECHOS (0=AUS, <>0 AN) in AL zurueck. Sie moechten die Ein-/Ausgabe auf ein anderes Geraet umleiten? Dann uebergeben Sie der Funktion 5 einfach in AL die Device-Nummer (0=DEV0, 1=DEV1,...) und in ES:SI den Zeiger auf den entsprechenden Treiber. Nach Aufruf zeigt Ihnen eine Null in AL, dass Ihr Geraet nun verwendet wird. Um ein Geraet zu finden sollte die Funktion 15 (siehe dort) Anwendung finden.

Moechten Sie die Ein-/Ausgabegeraete wieder auf Tastatur bzw. Bildschirm setzen, so koennen Sie AH mit 7 (DefIn) bzw. 8 (DefOut) laden. Die aktuelle BonnyDOS-Kernel-Version erhalten Sie mit Funktion 10 (AH=Version, AL=Revision). Nutzen Sie also z. B. die Version 1.24, so beinhaltet AH eine 1, und AL eine 24. Eine APL-Datei kann mit 11 geladen werden. Dazu muss in ES:SI der Zeiger auf einen Dateinamen uebergeben werden. Als Ergebnis erhalten Sie in BX das (neue HIMEM-Segment) und in AL den Fehlercode (0=ok, >=\$80 Dateifehler). Beachten Sie, das Programme vom Typ 0 immer nach \$0800-\$0000 geladen werden. Das Ausfuehren des Programms (Typ 0) erfolgt nicht automatisch von Kernel.

Vielleicht fuer Diskettenuser sind die beiden Funktionen 12 (GetDiskRetry) und 13 (SetDiskRetry) wichtig. Sie geben bzw. uebernehmen die Wiederholungen bei Diskfehler in bzw. von AL. Die Vorgabe direkt nach dem Start von BonnyDOS ist 5. Macht Ihr Disklaufwerk ab und zu Zicken, erhoehen Sie ersteinmal die Anzahl der Versuche. Gerade auf sehr schnellen Rechnern (z. B. Core Duo) koennen Sie so

unter Umstaenden Lesefehler vermeiden. Wie bereits besprochen, koennen Sie mit Funktion 15 ein residentes Programm suchen lassen. In ES:SI wird dessen 9-Byte lange Signatur erwartet. Als Ergebnis erhalten Sie in BX:SI die Adresse - oder \$0000-\$0000, falls kein Programm mit der gesuchten Signatur im Speicher zu finden ist.

Manchmal ist es ganz nuetzlich, eigene Strukturen oder Code im Speicher unterzubringen, ohne eine APL-Datei zu laden. Sie koennen mit SetHimem (Fkt. 18) den freien Speicher einfach nach unten verlegen, und sich so Speicher fuer eigene Zwecke freihalten. Beachten Sie, dass der Speicher nur dann wieder (durch Hochsetzen) freigegeben wird, wenn kein weiteres Residentes Programm geladen wurde. RESIDENT.APL zum Beispiel macht rege von Funktion 18 gebrauch, um APL-Dateien im Speicher zu halten. In BX erwartet die Routine uebrigens das neue Himem-Segment.

Sie moechten wissen, wo Himem aktuell liegt? Rufen Sie einfach Funktion 14 auf, und Sie erhalten in AX das aktuelle Himem-Segment. Noch etwas: Sorgen Sie dafuer, dass HIMEM nie unter \$1800 faellt. Dies ist das absolute Minimum an freien Speicher unter BonnyDOS. Grund hierfuer ist, das Programme der Einfachheit immer(!) mit einem bestimmten Speicherbereich rechnen koennen. Dies erspart auch dem Programmierer Arbeit - denn so muss nicht jedes keine Tool vor Ablauf die Speicher-Konfiguration pruefen.

Zwar besitzt das Kernel-Pack noch einige weitere, z. T. interessante, Funktionen - diese sollten aber eher Stoff fuer fortgeschrittene Programmierer sein und haben daher im Rahmen dieses Kurses nichts verloren. Der zweite Teil in der naechsten B-View befasst sich mit dem ExtraPack \$F4, das sich u. A. mit ENV-Variablen und Kommandozeilen-Parameter beschaeftigt. Bis dahin - viel Spass beim Experimentieren!

Besser als ein Topf voll Gold

Die besten Tricks zu GOS und BonnyDOS

Gezieltes Setup

Es gibt eine neue Version von BonnyDOS. Sie moechten an dem neuen Paket aber nicht alles, sondern nur bestimmte Dateien neu aufspielen. Kein Problem. Deaktivieren Sie als erstes die BAKup-Funktion (falls Sie diese verwenden). Dann loeschen Sie im Installations-Verzeichnis die Dateien, die Sie neu installieren moechten. Starten Sie dann wie gewohnt SETUP von der neuen Systemdiskette. Jetzt werden nur die Dateien kopiert, die sich noch nicht auf Festplatte befinden.



Von Netzwerken und wenig Speicher

Im Handbuch nicht erwaeht, kann mit MBRMGR /L das Partitionierungsprogramm von BonnyDOS/286 auf den Netzwerk-Modus umgeschaltet werden. Erkennbar ist dies am Schriftzug "LAN" am oberen Bildrand. Wenn Sie nun auf Ihrer ersten BonnyDOS-Partition einen Lader und ggf. noch KRNL286.SYS installieren, dann startet diese Installation auch ueber das Netzwerk. Sie haben kein Netzwerk? Macht nichts - hat Ihr PC weniger als 640K RAM, so ist der Systemstart durch den Netzwerklader auch hier gewaehrleistet - ist er doch speziell dafuer gedacht, mit wenig RAM auszukommen, falls der Netzwerklader der LAN-Karte nicht als ROM, sondern als Software-Treiber im RAM liegt.

Turbo fuer die Festplatte

Das SORT-Kommando mag fuer ordnungsliebende User ganz toll sein - sortiert es doch ein Verzeichnis alphabetisch und sorgt so fuer mehr Uebersicht. Was vielleicht nur wenige wissen: Festplatten-User profitieren gleich doppelt! Ab BonnyDOS 1.27, das erstmals den Master Boot-Record benutzt, wurden einige Aenderungen am Dateisystem zwingend notwendig. Die damalige Testfestplatten (CONNER 80 MB und CONNER 40 MB) sorgten fuer einige Kopfschmerzen - dauerten Zugriffe auf Verzeichnisse (z. B. MakeDir oder das Suchen neuer Verzeichniseintraege) unnoetig lange. Daher werden Verzeichnisse nun durch einen bestimmten

Directory-Eintrag abgeschlossen. Dies beschleunigt den Zugriff auf Verzeichnisse - selbst auf den beiden 'Alteisen' mehr als deutlich. Löschen/Kopieren Sie viel Dateien, so verlängern sich ganz unbemerkt die Zugriffszeiten wieder - denn die Abschluss-Einträge im Directory wandern immer weiter nach hinten. Nur durch SORT wird das Festplattenverzeichnis wieder aufgeräumt und korrekt abgeschlossen. Es lohnt sich also, das Kommando oeffters zu verwenden.



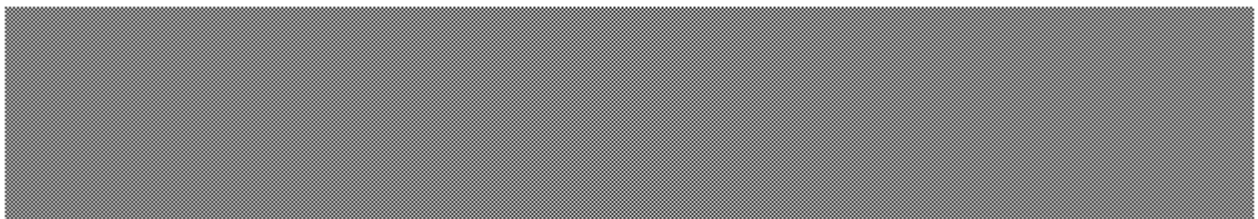
Nachts, wenn es dunkel wird vorm PC...

Es gibt Sie noch - auch, oder gerade, in diesen modernen Zeiten. Die Rede ist von den guten alten Schauergeschichten. Wer nicht weiss, wovon gesprochen wird, der moege sich jetzt einmal 'Alien Trilogy' auf seinen DOS-Rechner installieren. Wem es da nicht gleich die Fussnaegel hochrollt, der kann sich wahrlich als tapfer bezeichnen. Ein weiterer Fall fuer die (Grusel-)Grufft ist 'Spy vs Spy' unter DOS. Waren die C64-, Amiga- und Konsolen-Versionen immerhin noch bunt, so "besticht" die PC-Version durch unglaubliche CGA-Optik in vier Farben, die irgendwie nicht zum Spiel, und erst recht nicht zusammen passen. Wenn wir schon bei dem Thema sind: Street Sports Soccer von Codemasters ist ebenfalls so ein Kandidat fuer das Schreckenskabinett. Lilafarbene Spieler auf schwarzem Rasen mit tuerkischem Himmel... Oder war es ein Lila-Himmel und ein Cyan-farbener Rasen? Wie auch immer. Der Erfinder von CGA sollte sich schaemen!

Wer jetzt noch triumphierend aufschaut und sich denkt 'Was denn? Ist doch alles noch im gruenen (tuerkisen?) Bereich...', dem sei die Originalversion von Bubble Bubble (MS-DOS) ans Herz gelegt! Daraus ergibt sich ein guter Wettstreit: Wer schafft es am weitesten, ohne das das Trommelfell durch diesen extrem nervigen Bieper-Sound (ganz zu schweigen die Musik...) platzt. Falls vorher nicht die Polizei samt Sondereinsatzkommando die Wohnung stuermt, um dem endlich ein Ende zu setzen. Doch auch zu Zeiten von EGA setzt sich der Farbmonotonismus fort. Von den 64 moeglichen Paletteneintraegen bleiben 48 eigentlich

eigentlich ungenutzt. Immer die selbe (CGA-) Farbpalette, die bereits im Textmodus fuer so viel Freude sorgt. Ein Beispiel hierfuer ist 'Spirit of Adventures'. Hallo?!? Ich habe EGA! Diese Farben - das ist... alles, aber kein EGA. Das es auch anders geht zeigt das Titelbild von 'Shadowlands'. Da sind sie also - die verschollenen EGA-Farben. Die man so oft herbei gesehnt hat. Meistens vergebens... (an dieser Stelle sei nochmals die Erfindung des Farbmonitors verflucht!).

Sie haben immer noch nicht genug? Na gut! Einen gibts noch zum Abschluss! Aber sagen Sie nicht, man haette Sie nicht gewarnt. Bereit? Wirklich? Ok! Spielen Sie doch einmal eine Runde Pac Man fuer MS-DOS (Atari, 1984). Noch Fragen? Sie wissen also, warum frueher C64&Co. so beliebt waren? Jetzt sicher schon.



MS-DOS meets BonnyDOS/286

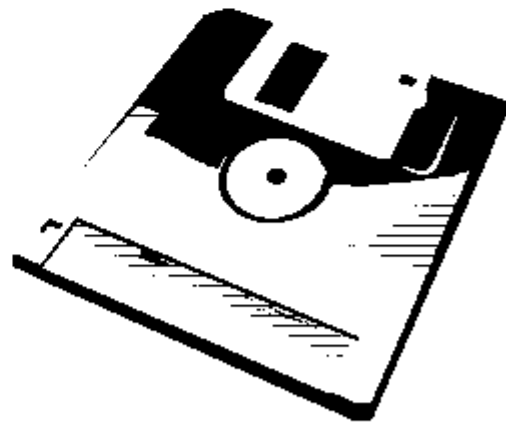
Wer bisher Dateien zwischen BonnyDOS und MS-DOS austauschen wollte, der war auf das gute, alte Diskettenlaufwerk angewiesen. Doch bald koennte dieser Technikdinosaurier in diesem Bereich zumindest ausgedient haben.

Betreibt man BonnyDOS und MS-DOS gemeinsam auf der selben Festplatte besteht Hoffnung, dass in Kuerze der Umweg ueber die Floppydisk entfaellt. Nicht nur das - eventuell ist auch das Ausfuehren von BonnyDOS- und GOS-Programmen unter MS-DOS moeglich. Gerade fuer den Entwickler ergeben sich hier voellig neue Moeglichkeiten.

Auf der Suche nach einem schnelleren Umbooten von MS- nach BonnyDOS kam mir der Gedanke, dass man doch das BonnyDOS-Kernel auch unter Microsoft's DOS laden und initialisieren koennte. Nun, das Laden klappt - selbst die Initialisierung. Allerdings nur bis zum ersten Laufwerks-Zugriff. Dann verabschiedet sich der Computer mit einem Reset.

Mit dem normalen Kernel funktioniert es also nicht. Wohl aber mit einer abgespeckten Version, die sich auf die reine API beschraenkt und ggf. unterstuetzend die MS-DOS-API verwendet. Und genau hier setzt eine neue Loesung an, unter MS-DOS auch auf die BFS16-Partition(en) zugreifen zu koennen. Warum eine Pascal-Unit hierfuer entwickeln, wenn die selben Funktionen schon lange vorhanden sind? Programme mit Einbindung der BDOS286.TPU, die sich dann um das eigentliche Kopieren von Files kuemmern, sind schnell entwickelt.

Das Aufuehren von BonnyDOS- oder gar GOS-Anwendungen ist zwar dann noch nicht moeglich (dafuer ist die RAM-Aufteilung einfach zu verschieden) - doch waere der schnelle Datenaustausch ein Anfang. Einzige Voraussetzung ist die Installation von BonnyDOS/286 auf eine Festplatte. Genau dieser Punkt ist es, der aktuell die Entwicklung von "grossen Programmpaketen" verhindert. Denn nur die wenigsten koennen oder wollen eine Festplatten-Installation. Trotzdem haben die Tests zum Datenaustausch begonnen. Sobald erste Ergebnisse vorliegen, landen diese sofort in der Y!-Group.



Das war's mit Ausgabe 7

Das war sie also, die 7. Ausgabe der B-View. Nicht ganz so umfangreich wie geplant, aber sie ist erschienen. Trotz Sommerloch, Urlaubszeit und schoenem Wetter. Sie haben Beitraege/Artikel? Ideen? Anregung oder Kritik? Vielleicht aber auch Fragen, die andere Interessieren koennten? Immer her damit! Ihre Texte schreiben Sie mit einem beliebigen ASCII-Texteditor. Speichern. Mailen. Fertig.

Benutzen Sie GOS/286, koennen Sie die Texte auch im WritePad-Format schicken. Immerhin entsteht die B-View auch komplett unter GOS/286. An dieser Stelle wuerden Sie jetzt in jeder anderen Zeitschrift den Herausgabe-Termin der naechsten Ausgabe finden. Nur hier nicht. Denn Sie wissen ja: Die B-View erscheint in eher unregelmæssigen Abstaenden. Das soll Sie jedoch nicht davon abhalten, den ein oder anderen Artikel einzureichen.

In diesem Sinne: Geniessen Sie das tolle Wetter, oder den Urlaub. Bis zur naechsten Ausgabe der B-View!

