

Der größte ATARI 8-bit User Club der Welt.

A B B U C
ATARI BIT BYTER USER CLUB



40

SONDERMAGAZIN

2. Halbjahr 2008

ABBUC SOFTWARE WETTBEWERB 2008



DIE TEILNEHMER: Animal Party, Blackbox, Zedi, Pokey Glider, XDOS 2.4, YASH, SDG

ZKZ25161

Design & Artwork by des-OR-mad (R.E.L.AG)

IMPRESSUM

© 2008 Atari Bit Byter User Club e.V.
Wolfgang Burger, Wieschenbeck 45
D-45699 Herten ☎ ☒ +49 2366 39623
✉ wolfgang@abbuc.de

Artikel in diesem Magazin sind durch entsprechende Copyrights geschützt. Kein Teil des Magazins darf in irgendeiner Form (Druck, Fotokopie, Scan oder einem andere Verfahren) ohne schriftliche Genehmigung reproduziert oder unter Verwendung elektronischer Systeme verarbeitet, vervielfältigt oder verbreitet werden.

INHALT :

- Seite 3 ABBUC Softwarewettbewerb**
Seite 4 Animal Party
Seite 5 Blackbox
Seite 7 Zedi
Seite 8 Pokey Glider
Seite 10 XDOS 2.4
Seite 27 YASH
Seite 32 SDG

ABBUC Software Wettbewerb 2008

Nach dem überragenden letzten Jahr für den ABBUC Software Wettbewerb treten in diesem Jahr sechs Programme an, um die Nachfolge des Vorjahressiegers "Yoomp!" auszufechten.

Der Schwerpunkt liegt im Gegensatz zum vorherigen Jahr auf Anwendungsprogramme, letztes Jahr waren mehr Spiele eingegangen.

Und wieder habt ihr, die ABBUC-Mitglieder, die Wahl. Ihr bestimmt den kommenden Sieger und die weiteren Platzierungen. Nehmt euch doch gleich die beiliegende Magazindiskette und Sondermagazindiskette vor und prüft die eingereichten Programme.

Nach ausgiebiger Prüfung der Programme habt ihr wie in den Vorjahren die Möglichkeit, eure Stimme auf dem beiliegenden Stimmzettel abzugeben. Für jedes Programm müssen 1 bis 10 Punkte vergeben werden. Das Programm mit den meisten Punkten wird der neue Sieger des Software-Wettbewerbes. Bitte schreibt auf den Stimmzettel euren Namen und schickt ihn an Wolfgang, an den Software-Ressortleiter oder bringt ihn zur Jahreshauptversammlung (JHV) des ABBUC am 25. Oktober mit. Dort werden die Stimmzettel ausgewertet und die Gewinner bekannt gegeben.

Einsendemöglichkeiten: per E-Mail an wolfgang@abbuc.de,
per FAX an: + 49 1805 323266 50812
Per Post an: ABBUC, Wieschenbeck 45, 45699 Herten

Und nun viel Spaß mit den Beiträgen!

*Gunnar Kanold
ABBUC Software Ressort
Goldberg 31
25791 Linden*



ANIMAL PARTY

A game for ABBUC CONTEST 2008

Minimal requirements:
Atari 800XL (64K) NTSC or PAL

Author: Mario Caillahuz (Allas)
MSX intro: Marius Diepenhorst
Cover MSX Ingame: Michal Szpilowski

Test your hability in this game by stunning the teasing animals that show up through the holes. It is very easy to play, just point the stick to choice the hole and press the button to use the hammer to knock the animals.

Be careful of hammering an empty hole or your hammer will get stuck in it for some seconds.

Time is your enemy. You need to stun the set amount of animals before time over's. Each time you pass a level you will be rewarded with points and extra time, and if you hit some animal as soon as he show up you'll get bonus points.



In faster levels beware of Dynamites or you will lose precious time. Also pay lot of attention to catch the Time bonus for extending the playing time.

Go fun with ANIMAL PARTY !

BLACKBOX

Gfx, Code by Chris Martin
 Title Screen and Blackbox text GFX
 by Adam Powroznik

Instructions

Booting Blackbox

Blackbox does not require extra memory banks so all 48Kb Atari 8-bit machines can run Blackbox. Disable BASIC and Load the blackbox.atr via APE. Blackbox will disable the loading sounds from the disk drive. So please be patient while loading...

Alternately, disable BASIC and load the blackbox.xex file via APE PC Mirror and the APELOADER. The blackbox.xex file does not have the title screen!

The blackbox.atr and blackbox.xex can be used directly with Atari800Win Plus 4.0 (<http://www.a800win.atari-area.prv.pl>)

Sound is Stereo and sounds great with the dual-pokey up grade. But this is not needed. Blackbox also sounds great with only one pokey!!!

Playing Blackbox

In the beginning...

1. Press any key to exit the title screen.
2. Press "Option" to select the Skill Level. Skill Levels range from 1 (Easy), 2 (Medium), 3 (Difficult), 4 (Insane)
3. Press "Start" to start the Game. Also use start to start a new level.

Controls:

1. Use the Joystick to move the white blinking cursor over grid squares and lasers on the top, bottom and left of the grid.
2. Pressing the fire button on the grid will place a red crystal. Pressing the fire button again on the same grid square will toggle



the crystal color or remove the crystal. The colors change from red, to green, to blue, remove it, then back to red.

3. Pressing the fire button over one of the lasers will fire the laser into grid.

Objective:

Blackbox is fun and addictive puzzle game. You must guess where all the crystals are, and you must have all the crystals the correct color.

HOW DO YOU DO THIS?

Well, each laser is both a laser and a crystal sensor. The sensor will turn red if it senses that a crystal should be in the laser path, but is not on the grid. The sensor will turn green when all crystals are placed in the correct locations for that laser's path AND all the crystals are the correct color.

Notice what the sensors are telling you and turn them all GREEN!

To find where a crystal should go, fire select a laser whose sensor is red and press the fire button to fire a white laser into the grid. The laser will bounce off of the unseen crystals as well as gain the color of each crystal. You must notice where the laser comes out of the grid and what the color is. The big lasers on the left and the right of the screen will help you with the color.

When a laser hits a crystal, it adds its color to the output laser. The laser output can be red, green, blue, yellow, cyan, purple or white.

Output color and meanings:

- If the output laser is RED, then the laser hit only RED crystals
- If the output laser is GREEN, then the laser hit only GREEN crystals
- If the output laser is BLUE, then the laser hit only BLUE crystals
- If the output laser is YELLOW, then the laser hit RED and GREEN crystals
- If the output laser is CYAN, then the laser hit GREEN and BLUE crystals
- If the output laser is PURPLE, then the laser hit RED and BLUE crystals
- If the output laser is WHITE, then the laser hit RED, GREEN and BLUE crystals

When all the laser sensors turn green, all the crystals with all the correct colors have been correctly placed.

That is it! Enjoy! Have Fun! And find all the Crystals in the Blackbox!!!

Special Thanks and Greetings to:

Adam Powroznik, Fandal, Mario Caillahuz, Bunsen, Emkay, Carsten, Doctor Clu, a8maestro, Heaven, puppet_mark, bf2k+, Steven Tucker, Almost Rice, MetalGuy66, Gunnar Kanold, Albert, And many others!

ZEDI

Von Winfried Fiedler

Anleitung für Zeisa-Editor
(c) 2008 wf

Dieser Editor ermöglicht gleichzeitig
3 Zeichensätze zu bearbeiten.

Die Darstellung erfolgt in GR0 oder
GR12 (umschaltbar im Editor-
Modus) in 3 Reihen untereinander, wobei nur die unterste gesavt wird.



Die Bedienung erfolgt mit Tasten und dem Joystick. ESC immer zurück zum Menü.

1. Editor-Modus. Setzen der BITS mit Joy und Trigger. Anzeige in Datas und Color. ESC zum Hauptmenü. Zeichen Auswahl mit Cursor hoch, runter, links, rechts
2. Zeisa 1 laden
3. Zeisa 2 laden
4. Zeisa 3 saven
5. Zeisa 3 löschen
6. 5 Farbwerte wählbar, Cursor links, rechts Farbe, Cursor hoch runter Helligkeit. Enter next Farbe, ESC Menü
7. Programm verlassen
8. ausgewählten Zeisa in unterste Zeile kopieren
9. Directory. SELECT = Laufwerk wählen; OPTION = Start



Pokey Glider

Version 1.0

August 10, 2008

8-bit ATARI computer synth created by Bart Jaszcz in the year 2008.

Most of the settings are pretty self-explanatory, so I'll just mention the non-standard things:



Setting	Function
STEPS	Number of steps between notes (glide steps)
SPEED	How fast to step through steps (glide speed)
CH-MOD	Channel distortion mode (5 & 7 are pure tone)
FILTER	Filter (bit mask) channel 1 by channel 2
CLOCK	Channel 1 & 2 clocks (7=1.79Mhz 6=64Khz 5=15Khz)
TEMPO	Non-linear tempo variable (this is not BPM!)
SHIFT	How much to slow down tempo by random interval

Modes:

When you start the program, you'll notice the MODE display with a '*' character. This means the sequencer is running, and keyboard input will override notes in memory. This way, while you play your pattern, it will be recorded "on the fly".

If you hit TAB, the mode character will change to a '>', which means the sequencer is running (playing), but keyboard input is not recorded, instead, it will be played over the stored pattern. This way, you can "play along" with a stored pattern, playing over the stored notes, but not recording over those notes.

Now, if you hit RETURN, the sequencer will start/stop, and the mode character will change too. While the sequencer is stopped, it will be a '#', indicating the sequencer is stopped and keyboard input is blocked. This mode essentially turns the Atari into a keyboard. Notice that in this mode, notes aren't quantized, they respond to the keyboard press duration. Finally, a '!' character indicates the sequencer is stopped and keyboard input is recorded. Use this mode if you want to edit the pattern "offline", while the sequencer is not playing. Also, you can use the cursor keys in this mode to position the cursor.

Other Notes:

* The POKEY normally has four 8-bit channels, but since this is a gliding synth, I've chosen to use the POKEY

in the channel-combined mode (two 16-bit channels). It's important to realize that the POKEY registers hold frequency values, not amplitude, so this isn't the same as 16 bit PCM value.

* Since, the POKEY is in 16 bit mode, it needs a high frequency clock, so you should normally use the CPU clock or 1.79Mhz which is marked as "7" in the synth display. This will give the highest gliding resolution. To see the difference, notice that in other clock modes, the synth doesn't glide at all.. this is because the clock isn't fine enough. I suppose the same thing would apply if I had used 8-bit registers, the intervals between frequency values wouldn't be fine enough. But what do I know, some strange sounds can be gotten from using the other clocks, especially in conjunction with filtering.

* If the POKEY is in non-pure-tone modes (not 5 or 7), then the octave value should be set pretty high. This is because these modes generate the random wave, so in lower octaves sound too low. These modes are really for generating explosions, but try using higher octaves with these modes, you can sometimes get tones that way, or use them for percussion. Another tip is to change the Ch-1/Ch-2 mix balance in these modes (for example, adding quiet noise to pure tone).

* The temp and shift variables are important in two ways, first they can be used to beat match with other applications (if running on an emulator for example). Shift is actually an empty loop in the synthesizer code with the number of interactions determined by a random number. Shift would therefore be the upper limit of this random number. This corresponds to a musician "holding back" a random short interval. So, this obviously has a "humanizing" effect. Not only that, it then can be used to aid in "beat matching" as mentioned before. If for example the tempo variable can't match the beat exactly, you can change the shift variable to finely tune the tempo. It's amazing how much complexity can be generated by just this method.

--Bart



XDOS 2.4

(c) 2008 by Stefan Dorndorf

Doku vom 30.08.2008

XDOS (Extended Disk Operating System) ist ein kompaktes DOS 2.5-kompatibles DOS für die ATARI-8Bit-Rechner.

Es besteht lediglich aus der Datei DOS.SYS, das sonst übliche DUP.SYS ist mit in DOS.SYS enthalten. Trotzdem ist DOS.SYS nur 37 Single-Density-Sektoren lang und nimmt damit nicht mehr Platz weg als DOS 2.5.

Damit ist XDOS auf jedem ATARI-Rechner bequem nutzbar – auch ohne RAM-Disk, denn das DUP muss ja nicht von Diskette nachgeladen werden. Außerdem hat XDOS natürlich noch eine Menge nützlicher Zusatzfunktionen.

Auf der XDOS-Systemdiskette befinden sich folgende Dateien:

DOS.SYS	Das eigentliche XDOS in der bootfähigen Normalversion für alle ATARI-Laufwerke
XDOS24N.COM	Das XDOS als COM-Datei, kann von jedem DOS/Game-DOS gestartet werden
XDOS24F.COM	Die Fastversion des XDOS als COM-Datei (unterstützt auch Ultra Speed und Turbo Drive)
AUTOCONF.COM	Konfiguriert XDOS mit einer Standard-Konfiguration vor und richtet die RAM-Disk D8: ein
RAMDISK.COM	Richtet unter D8: automatisch eine RAM-Disk ein (passend zur Speichererweiterung)
SAVEDOS.COM	Speichert XDOS mitsamt Konfiguration als COM-File ab
CONF.CMD	Dient zum Konfigurieren von XDOS. Im DUP zum Start CONF eingeben, dann auf die richtige Seite blättern, dann ESC drücken und schließlich den zu ändernden Wert in der Zeile, die mit „=" beginnt, überschreiben und RETURN drücken
HELP.CMD	Eine kurze Befehlsübersicht des DUP sowie weitere Infos. Zum Ansehen im DUP einfach HELP eingeben
ERROR.CMD	Zeigt eine Kurzerklärung der wichtigsten Fehlercodes an
CON.CMD	Schaltet ein Switch-Cartridge (wie ACTION! oder MAC/65) ein
MAP.COM	Zeigt alle Header (d.h. Segmentadressen) sowie INIT- und RUN- Adressen einer COM-Datei an
MAP.ASM	Assembler-Sourcecode zu MAP.COM

Erste Schritte

XDOS gibt es in zwei Versionen: Die Normalversion (XDOS 2.4N) unterstützt alle Standard-ATARI-Laufwerke (810, 1050, XF551) sowie die Hyper-XF-Erweiterung für die XF551. Die Fastversion (XDOS 2.4F) unterstützt zusätzlich die hohe Geschwindigkeit des SIO2USB und



SIO2SD sowie der 1050-Laufwerke mit Speedy-, Happy- und Turbo-Erweiterung. Allerdings ist die Fastversion um 384 Byte länger, d.h. die untere Grenze des freien Speichers liegt höher (\$1F70 statt \$1DF0). Daher kann es Programme geben, die mit der Fastversion nicht funktionieren (viele sind das aber nicht).

Die Normalversion kann von der XDOS-Systemdiskette gebootet werden. Nach dem Start kann bei Bedarf die Fastversion durch Eintippen von XDOS24F (RETURN) gestartet werden. Nach dieser Versionsauswahl sollte die automatische Konfiguration gestartet werden (AUTOCONF eingeben). Nun ist XDOS einsatzbereit. Es empfiehlt sich, zunächst mit dem DUP-Befehl FS# eine Diskette zu formatieren und mit INI das konfigurierte XDOS zu speichern.

Bitte auch die XDOS-Referenz am Ende dieser Anleitung beachten – dort ist alles Wichtige aufgeführt.

Das XDOS-DUP

Das DUP (Disk Utility Package) besteht bei XDOS nicht aus einem Menü, sondern einem Kommandointerpreter.

Mittels des BASIC-Befehls „DOS“ oder durch Antippen der RESET-Taste bei gedrückter OPTION-Taste wird das DUP aufgerufen. Es erscheint der XDOS-Titel und das Kürzel (Prompt) „D1:“. Nun können beliebige DUP-Befehle eingeben und mit RETURN ausführt werden. Man kann auch mit dem Cursor auf andere Zeilen fahren – beim Drücken von RETURN wird stets die Zeile ausgeführt, auf der sich der Cursor befindet. Mit SELECT+RESET kann man jederzeit einen Kaltstart auslösen und mit START+RESET ein eingestecktes Cartridge bzw. das BASIC neu initialisieren.

Für die DUP-Befehle gelten folgende allgemeine Regeln:

- Fast jeder DUP-Befehl besteht aus drei Buchstaben, z.B. DIR (=Directory), gefolgt von den zugehörigen Parametern (Dateinamen oder Zahlen), die durch Leerzeichen oder Komma getrennt sind.
- Mit C: wird im CSAVE-Format, mit C2: wird im SAVE“C:“-Format auf Kassette gelesen bzw. geschrieben.
- In Dateiangaben darf statt : auch ; benutzt werden, z.B. „D8;NAME“. Auch darf das „D“ vor Laufwerksnummern fehlen, z.B. „;8;NAME“.
- In Dateinamen kann *.* durch – abgekürzt werden.
- In Dateinamen ist der Unterstrich erlaubt (z.B. DATEI_A.TXT).
- Bei allen DUP-Befehlen wird ein fehlender Dateiname stets durch *.* ersetzt. (Ausnahme: REN- und SAV-Befehl)
- Mit +Dateiname kann ein Name, so wie er im Directory steht, eingegeben werden, also ohne Punkt und evtl. mit Leerschritten. + erlaubt auch die Verwendung von beliebigen Zeichen in Dateinamen.

Wenn man in Dateinamen Wildcards (* ? -) benutzt, führt XDOS eine Yes/No-Abfrage für die betreffenden Dateien durch (außer beim DIR-Befehl). Jeder passende Dateiname wird mit ? angezeigt. Weiter geht's dann mit:

Y oder RETURN	Ja, Datei bearbeiten
N oder Leertaste	Nein, nächsten Dateinamen anzeigen
ESC oder BREAK	DUP-Befehl abbrechen

Auch beim Speichern (SAV-Befehl) findet diese Abfrage bei Verwendung von Wildcards statt, aber nur, wenn die betreffende Datei auf der Disk bereits existiert. Wenn man auf alle angebotenen Dateinamen mit der Taste N antwortet, werden alle Wildcards im Dateinamen durch Leerzeichen ersetzt, danach wird gespeichert. Wenn man nicht speichern will, einfach ESC oder BREAK drücken, der dabei angezeigte Error 169 hat keine Bedeutung.

Bei der Yes/No-Abfrage werden gesperrte Dateien immer als nicht gesperrt angesehen, um einen Abbruch mit Error 167 (File Locked) zu vermeiden.

Folgende Kürzel können an Dateinamen angehängt werden:

- /N Keine Yes/No-Abfrage durchführen (No Query), außerhalb des DUP ist dies der Standard
- /Q Yes/No-Abfrage durchführen (Query), im DUP ist dies der Standard
- /S Keine Yes/No-Abfrage durchführen, aber die Namen aller betroffenen Dateien anzeigen (Show)
- /D Lese-Zugriff auf das Directory anstatt einer Datei
- /A Beim Speichern einer Datei diese nicht überschreiben, sondern anhängen (Append)
Beim Anzeigen des Directory dieses zweispaltig (statt einspaltig) anzeigen

ACHTUNG: Die Yes/No-Abfrage und die neuen Dateinamenkürzel funktionieren nur beim Gerät „D:“, nicht jedoch bei anderen Geräten mit Dateinamen (wie z.B. dem „H:“-Gerät im Atari800Win-Emulator).

Alle Zahlen in DUP-Befehlen können hexadezimal oder auch durch Voranstellen eines Punktes dezimal eingegeben werden, z.B. entspricht die Hex-Zahl 600 der Dezimalzahl .1536.

Die DUP-Befehle

Es folgt die Beschreibung aller DUP-Befehle des XDOS. Angaben in Klammern sind dabei optional. Dateiangaben sind immer optional, fehlende Dateinamen werden durch *.* ersetzt (Ausnahme: REN- und SAV-Befehl).

I; bis **9;** oder **D1:** bis **D9:**

Ändert die aktuelle Standard-Laufwerksnr. Bei Eingabe von Dateinamen ohne Laufwerksangabe benutzt das DUP immer die Standard-Laufwerksnr. Diese Nummer wird zur Erinnerung nach jedem ausgeführten DUP-Befehl angezeigt. XDOS unterstützt immer die Laufwerke D1: bis D9:.

Beispiel: Mit 8; wird auf D8: umgeschaltet. TYP TEXT.DAT zeigt dann die Datei D8:TEXT.DAT an.

DIR Dateifilter / **Dateifilter** **RETURN** **SHIFT-RETURN**

Zeigt das Inhaltsverzeichnis einer Disk an. Nach jeder Seite (22 Zeilen) wird die Bildschirmausgabe angehalten. Mit RETURN oder der Leertaste geht's weiter, mit ESC oder BREAK wird die Ausgabe abgebrochen. Durch Angabe eines Dateinamens mit Wildcards kann die Anzeige entsprechend eingeschränkt werden. Wird an den Dateinamen ein /A angehängt, so wird das Directory zweispaltig ausgegeben. Zur Abkürzung kann man statt DIR auch nur / eingeben oder nur die RETURN-Taste drücken. SHIFT-RETURN liefert immer das Directory von D8: (meist die RAM-Disk).

Am Ende des Directory steht im XDOS-DUP immer „xxx FREE #yy <z>“. xxx ist die Anzahl der freien Sektoren:

999 bedeutet, dass mindestens (!) 999 Sektoren frei sind und ??? zeigt an, dass die Disk wahrscheinlich nicht XDOS-kompatibel ist (mehr als 1023 freie Sektoren). yy ist die Anzahl der gelisteten Dateien und <z> das Format der Disk:

SAV Datei Start End (Init |– (Run)) SAV Datei Start End / SAV Datei/A – (Init |– (Run))

SAV Datei Start End speichert den Bereich von „Start“ bis „End“ im COM-Format ab. Wie bei DOS 2.5 können auch INIT- und/oder RUN-Adressen angegeben werden. Um eine RUN-Adresse, aber keine INIT-Adresse zu speichern, ersetzt man die INIT-Adresse durch ein – . Steht hinter „End“ ein /, wird die Datei ohne (!) den 6 Byte langen COM-Header abgespeichert. Schließlich kann man mit SAV Datei/A – Init Run auch INIT- und/oder RUN-Adresse an eine vorhandene COM-Datei anhängen.

Beispiele: SAV SCREEN.DAT 9C40 9FFF speichert den Bereich von \$9C40 bis \$9FFF in der Datei SCREEN.DAT. Diese Datei lässt sich mit LOA SCREEN.DAT wieder laden. SAV SCREEN.DAT 9C40 9FFF / speichert denselben Bereich ohne COM-Header. Zum Laden muss dann LOA SCREEN.DAT 9C40 eingegeben werden. SAV PROG.COM .32000 .32500 – .32100 speichert ein Maschinenprogramm von 32000 bis 32500 mit der RUN-Adresse 32100 und ohne INIT-Adresse ab.

RUN (RUN-Adresse)

Startet ein Maschinenprogramm (per 6502-JSR) an der “RUN-Adresse”, falls diese nicht 0 ist. Wird RUN ohne RUN-Adresse eingegeben, nimmt XDOS die letzte RUN-Adresse, die auch im XDOS-Titel unter <Run> steht (Low-Byte, High-Byte). Die aktuelle RUN-Adresse wird auch beim Laden einer COM-Datei (EXE, LOA) neu gesetzt.

Beispiel: RUN E477 springt zu \$E477, womit ein Kaltstart ausgeführt wird.

COP Quelldatei Zieldatei

Kopiert die Quelldatei(en) unter gleichzeitiger Umbenennung in den Zieldateinamen. Ist kein Zieldateiname angegeben oder ist dieser *.* findet natürlich keine Umbenennung statt. Die Datei DOS.SYS wird vom COP-Befehl nicht kopiert – stattdessen wird das XDOS automatisch bootfähig auf die Zieldiskette geschrieben. Sind Quell- und Ziellaufwerk gleich (Kopieren von einer Disk auf eine andere mit nur einem Laufwerk) muss auf die Hinweise „Target?“ bzw. „Source?“ die Ziel- bzw. Quelldisk eingelegt werden, worauf das Kopieren mit RETURN fortgeführt werden kann. Dabei bedeutet die Meldung „OK“, dass eine Datei fertigkopiert wurde, d.h. man kann mit ESC oder BREAK abbrechen, wenn keine weiteren Dateien kopiert werden sollen.

ACHTUNG: Zum Kopieren benutzt XDOS den gesamten Hauptspeicher von LOMEM bis HIMEM, d.h. Programme im Speicher werden dabei überschrieben. Daher wird beim nächsten Sprung zum BASIC bzw. eingesteckten Cartridge dieses automatisch neu initialisiert, sowie die aktuelle RUN-Adresse gelöscht (d.h. auf 0 gesetzt). Die CAR-Adresse (siehe CAR-Befehl) wird aber nicht (!) gelöscht.

ACHTUNG: Bei anderen Geräten mit Dateinamen als „D:“ (z.B. dem Gerät „H:“ des Atari800Win-Emulators) kann pro COP-Befehl nur eine Datei kopiert werden und der Zieldateiname muss immer vollständig (!) angegeben werden.

Beispiele: Quelldatei Zieldatei (Dn: ist hier immer das Standard-Laufwerk)
 COP HELP Dn:HELP Dn:HELP kopiert mit Target/Source-Rückfrage
 COP ASM TEST Dn:ASM Dn:TEST kopiert mit Target/Source-Rückfrage
 COP 2;ASM 8; D2:ASM D8:ASM
 COP D2;TXT C: D2:TXT C: speichert auf Kassette im CSAVE-Format
 COP E: TEXT E: Dn:TEXT erstellt eine Text-Datei, Dateiende mit CTRL-3
 COP *.* 8; kopiert alle Dateien von Dn: auf D8: mit Yes/No-Abfrage.
 COP *.BAS/N 8;*.BAK kopiert alle BAS-Dateien (ohne Abfrage) auf D8: und benennt diese in *.BAK um.
 COP /D P; druckt das Directory von Dn: auf dem Drucker aus.

hat genau dasselbe Format wie beim FD#-Befehl, kann also auch von einem 1050-Laufwerk bearbeitet werden.

ACHTUNG: Mit F2# wird nur auf der Vorderseite ein Directory eingerichtet. Auf die Rückseite muss mit dem CL#-Befehl (siehe unten) noch ein Directory geschrieben werden. Zum Zugriff auf die Rückseite einer F2#-Disk muss eine Laufwerksnummer (z.B. D5:) eingerichtet werden. Dazu den DUP-Befehl =71x y eingeben: x ist die Laufwerksnummer (z.B. 2), y ist der Buchstabe, der der hinten an der XF551 eingestellte Gerätenummer entspricht: A=1, B=2, C=3, D=4. Siehe auch XDOS-Konfiguration (XDRV).

CL# (Laufwerk;)

Dieser Befehl löscht sämtliche Dateien auf einer Diskseite und legt ein neues Directory an. CL# funktioniert mit allen bereits einmal formatierten Disketten. Abhängig von der XDOS-Konfiguration (XOPT) werden Disketten in mittlerer Dichte DOS 2.5- oder MYDOS-kompatibel initialisiert.

= (Adresse) CTRL-3

Zeigt den Inhalt der ersten 8 Byte ab der angegebenen Adresse hexadezimal und als ATASCII-Zeichen an. Mit = oder durch Drücken von CTRL-3 werden jeweils die nächsten 8 Byte angezeigt. Statt = kann wie in XDOS 2.3 auch > benutzt werden (> wird ab XDOS 3.0 für Unterverzeichnisse benötigt).

= Adresse Byte1 Byte2 ...

Hiermit lässt sich der Speicherinhalt ab der „Adresse“ ändern. Die „Bytes“ können nicht nur dezimal oder hexadezimal sondern durch Voranstellen eines ' auch als ATASCII-Zeichen eingegeben werden. Statt der „Bytes“ können auch Adressen stehen – deren Bytes werden dann in der Reihenfolge Low-High-Byte abgelegt.

ACHTUNG: Die Speicheradressen 24, 25, 26 und 27 dürfen mit diesem Befehl nicht geändert werden!

Beispiel: =600 A4 .49 'A 6FF .32768 legt ab \$600 folgende Werte ab: \$A4 \$31 \$41 \$FF \$06 \$00 \$80

Batchdateien (CMD-Dateien)

Eine Batchdatei ist eine Datei, die aus lauter DUP-Befehlen besteht. Batchdateien können mit dem EXE-Befehl gestartet werden – die in der Datei stehenden DUP-Befehle werden dann nacheinander ausgeführt. Die Befehle werden dabei nicht auf dem Bildschirm angezeigt. Für Batchdateien gilt folgendes:

- Speziell für Batchdateien gibt es noch drei weitere DUP-Befehle:
 - ' Text Text anzeigen
 - “ Text Kommentarzeile
 - WAI Zeigt ein ? an und wartet auf einen Tastendruck (Wait): Mit RETURN oder der Leertaste geht's weiter, mit ESC oder BREAK wird die Batchdatei abgebrochen
- Jede Batchdatei sollte die Endung .CMD haben und muss (!) mit einer Kommentarzeile (also mit “) beginnen.
- In Batchdateien sollten zur besseren Lesbarkeit keine Abkürzungen für DUP-Befehle benutzt werden.
- Batchdateien können nur von Disk (d.h. dem Gerät „D:“) gestartet werden, nicht z.B. von Kassette.

- Batchdateien werden immer über Kanal (IOCB) #5 gestartet, dieser darf also innerhalb einer Batchdatei nicht (von gestarteten COM-Programmen bzw. mit RUN gestarteten Maschinenprogrammen) benutzt werden.
- Die Abarbeitung einer Batchdatei wird beendet, sobald sie eine andere Batchdatei startet oder ein Fehler auftritt.
- Vor Ausführung des letzten DUP-Befehls in einer Batchdatei wird diese vom XDOS geschlossen. Damit kann in der letzten Batchzeile ein COM-Programm gestartet werden, ohne einen geöffneten Kanal #5 zu hinterlassen.
- Batchdateien können genau wie COM-Dateien auch beim Booten automatisch gestartet werden, d.h. es genügt, die Datei in AUTORUN.SYS umzubenennen.

Batchdateien können auch leicht mit dem COP-Befehl erzeugt werden, wenn mal kein Editor greifbar ist. Beispiel:

COP E; MYBAT.CMD

“ Beispiel einer einfachen Batchdatei

EXE RAMDISK.COM

konfiguriert die RAM-Disk D8: neu

BON

schaltet das BASIC ein

COP HELP.CMD 8;

kopiert HELP.CMD in die RAM-Disk

CAR RUN“D:DEMO.BAS“

startet das BASIC-Programm DEMO.BAS

Danach CTRL-3 drücken, um die Eingabe zu beenden. XDOS schreibt nun die Datei unter dem Namen MYBAT.CMD auf Diskette. Mit EXE MYBAT.CMD (oder einfach nur MYBAT) wird die Batchdatei gestartet.

Die RAM-Disk

XDOS kann sechs verschiedene RAM-Disk-Größen verwalten, je nach verfügbarem RAM-Speicher. Das Programm RAMDISK.COM (oder AUTOCONF.COM) konfiguriert und initialisiert die RAM-Disk auf D8: automatisch. Statt D8: können auch andere Laufwerksnummern für die RAM-Disk benutzt werden (siehe XDOS-Konfiguration, XDRV). Welche RAM-Disk gerade aktiv ist, kann man am Kürzel unten im Directory-Listing sehen:

Kürzel	Größe	Erläuterung
<L>	14 KB	für 800XL/XE ohne Speichererweiterung
<X>	64 KB	für 130 XE ohne zusätzliche Speichererweiterung
<S>	90 KB	Single Density
<M>	128 KB	Medium Density
<D>	180 KB	Double Density
<H>	256 KB	Maximalgröße der XDOS-RAM-Disk

Die 14KB-RAM-Disk benutzt das RAM unter dem OS-ROM und schaltet daher das OS und damit den Zeichensatz beim Zugriff ab, wodurch es zu einem kleinen harmlosen "Gewitter" auf dem Bildschirm kommt. Die anderen RAM-Disks benutzen die 16KB-Speicherbänke des 130 XE bzw. einer Speichererweiterung (siehe XDOS-Konfiguration, XRAM).

ACHTUNG: Manche Programme wie z.B. Turbo-BASIC benutzen den Bereich der 14KB-RAM-Disk für eigene Zwecke und funktionieren daher nicht mit dieser RAM-Disk zusammen.

Vor Nutzung einer RAM-Disk muss diese initialisiert (gelöscht) werden. Dies ist mit dem CL#-Befehl möglich.

Beim Booten des XDOS wird die RAM-Disk D8: automatisch gelöscht. Dies kann durch Drücken der HELP-Taste beim Booten oder auch mittels der XDOS-Konfiguration (Setzen von Bit 5 in XOPT=\$70B) verhindert werden.

Neue und geänderte Fehlermeldungen

Error 160 – Drive Number Error

Es wurde versucht, zwei Dateien gleichzeitig (!) auf verschiedenen (!) Laufwerken zum Schreiben (!) zu öffnen: Da XDOS nur einen Laufwerkspuffer hat, können mehrere Dateien zum Schreiben gleichzeitig nur auf einem Laufwerk geöffnet werden. Beim Lesen gilt diese Einschränkung nicht.

Error 163 – Disk Structure damaged

Diese Fehlermeldung wird von XDOS nicht erzeugt. (Bei DOS 2.5 erscheint sie, wenn Byte 5 von Sektor 360 nicht 0 ist.)

Error 167 – File Locked

Die zu ändernde Datei ist gesperrt oder ein MYDOS-Unterverzeichnis.

Error 173 – Bad Sectors

Statt dieser Formatier-Fehlermeldung zeigt XDOS Error 144 an. Bei einer Disk mit defekten Sektoren dauert ein Formatierungsversuch sehr lange (mindestens 2 Minuten). Daran lassen sich defekte Disks erkennen.

Error 180 – Bad Load File

Die mit dem EXE- oder LOA-Befehl bzw. XIO 40 angesprochene Datei ist weder eine COM-Datei noch eine Batchdatei. COM-Dateien müssen mit \$FFFF beginnen, Batchdateien mit einer Kommentarzeile (also mit "). Batchdateien können außerdem nur mit EXE (nicht mit LOA) und auch nur von Disk gestartet werden (nicht z.B. von Kassette).

Error 181 – Bad Address Range

Beim SAV-Befehl oder bei mindestens einem Segment der zu ladenden COM-Datei war die Startadresse größer als die Endadresse. Oder es wurde versucht, eine COM-Datei zu laden, die beim Laden einen Teil von XDOS überschrieben hätte. Im letzteren Fall können weniger DOS-Puffer zum Erfolg führen (siehe XDOS-Konfiguration, XBUF). Direkt nach Auftreten dieses Fehlers im DUP kann man sich mit dem DUP-Befehl = die Adresse anzeigen lassen, die zu diesem Fehler führte.

Error 182 – Bad Parameter

Bei einem DUP-Befehl bzw. XIO 39 wurde eine fehlerhafte Zahl bzw. Adresse eingegeben, z.B. RUN G474.

Neue Dateikürzel

Die im DUP möglichen neuen Dateikürzel - + /D /N /Q /S /A funktionieren auch außerhalb des DUP. Wird dabei in Dateiangaben statt des : ein ; verwendet, ist dies gleichbedeutend mit dem Dateikürzel /Q (sofern weder /N noch /S benutzt werden). In diesem Fall wird auch ein ganz fehlender Dateiname durch *.* ersetzt. In Dateinamen ist auch der Unterstrich erlaubt (wie bei MYDOS).

Beispiele (BASIC-Befehle):

LOAD "D:P-" entspricht LOAD "D:P*:.*".
 LOAD "D;" entspricht LOAD "D:*.*/Q", es wird also für jeden Dateinamen eine Yes/No-Abfrage gemacht.
 LOAD "D:" ergibt einen Error 165 - Bad Filename.
 LIST "D2:PROG/A" fügt das BASIC-Programm im Speicher an D2:PROG an.
 LOAD "D:+MEINPROGRAM" lädt das BASIC-Programm D:MEINPROG.RAM.

Die noch folgenden Kapitel sind für fortgeschrittene User gedacht, die alles über XDOS wissen wollen:

Neue XIO-Befehle

XIO 34, #k, 0, 0, "Dn:" : X=PEEK(746)

Stellt die Dichte einer Disk fest. X>127 bedeutet mittlere, X<32 einfache, sonst doppelte Dichte. Funktioniert auch mit der XF551. n ist das gewünschte Laufwerk, k eine Kanalnummer von 1 bis 7.

XIO 39, #k, 0, 0, "Dn:DUP-Befehl"

Führt den hinter Dn: stehenden max. 40 Zeichen langen DUP-Befehl aus, n ist dabei das Standard-Laufwerk.

Mit XIO 39, #1, 0, 0, "D2:DIR" oder XIO 39, #1, 0, 0, "D:DIR 2;" wird z.B. das Directory von D2: angezeigt.

XIO 40, #k, 0, 0, "Dn:Dateiname"

Entspricht dem EXE-Befehl des DUP, es können also Batch- oder COM-Dateien gestartet werden.

Eine COM-Datei wird immer über Kanal #k geladen, eine Batchdatei über Kanal #5 und für die DUP-Befehle in der Batchdatei wird Kanal #k benutzt. Daher darf beim Ausführen einer Batchdatei k nicht 5 sein.

Nach Ausführen einer Batchdatei wird nicht zum Aufrufer zurückgekehrt, sondern immer ins DUP gesprungen. Wird eine COM-Datei gestartet und ruft diese selbst XIO 40 auf (also rekursiv), so darf die COM-Datei weder mit RTS zurückkehren noch die XDOS-Routinen für COM-Programme benutzen (siehe „COM-Programme mit Parametern“).

Beispiel: Aus BASIC kann mit XIO 40, #1, 0, 0, "D:ASMB.CMD" die Batchdatei ASMB.CMD gestartet werden. ASMB.CMD könnte mit BOF das BASIC ausschalten und dann mit EXE ASM.COM den Assembler starten.

XIO f, #k, 0, 0, "Dn:"

Formatiert bzw. löscht Disketten. f kann folgende Werte annehmen:

251 = F2# 252 = FD# 253 = FS# 254 = FM# 255 = CL#

Die XF551

Das XF551-Laufwerk kann die Dichte einer eingelegten Disk leider nicht automatisch bestimmen, daher übernimmt XDOS dies. Ebenfalls unterstützt wird das XF High Speed-Protokoll sowie das schnellere Hyper Speed-Protokoll der XF551-Erweiterung "Hyper-XF".

Da der XF551-Density-Check etwas dauert, kann er für jedes Laufwerk Dn: auch abgeschaltet werden (Bit 5 von \$71n setzen, siehe XDOS-Konfiguration, XDRV). Dann muss man aber nach einem Diskwechsel den DUP-Befehl TES aufrufen, um die XF551 auf die neue Dichte einzustellen.

len. Die XF-High-Speed bzw. XF-Hyper-Speed kann auch einzeln für jedes Laufwerk Dn: abgeschaltet werden (Bit 4 von \$71n setzen, siehe XDOS-Konfiguration, XDRV).

ACHTUNG: Bei eingeschalteter XF-High-Speed bzw. Hyper-Speed werden die Disketten etw. anders formatiert (bei normalen XF551-Laufwerken gilt das nur für FD# und F2#), damit die höhere Geschwindigkeit auch ausgenutzt wird. Normal formatierte Disketten können auch mit High- bzw. Hyper-Speed nicht schneller gelesen werden und mit High- bzw. Hyper-Speed formatierte Disketten werden mit Normal-Speed nur sehr langsam gelesen. Vor dem Formatieren sollte man sich also entscheiden, ob man High- bzw. Hyper-Speed benutzen will oder nicht.

Die XDOS-Konfiguration

Die Konfigurationstabelle des XDOS befindet sich im Bereich \$709 bis \$71F. Ändern kann man die Konfiguration mit dem DUP-Kommando = (z.B. mit =709 die Konfiguration anzeigen und durch Überschreiben und RETURN ändern).

Etwas bequemer geht es mit der Batchdatei CONF.CMD:

CONF.CMD zeigt nach dem Start (dazu im DUP einfach CONF eingeben) seitenweise die Konfiguration zusammen mit kurzen Erläuterungen an. Auf der gewünschten Seite ESC drücken, den zu ändernden Wert (in der „=-“-Zeile) überschreiben und dann auf der gleichen Zeile RETURN drücken. \$709 bis \$710 werden auch im XDOS-Titel angezeigt und können ebenfalls durch Überschreiben und RETURN drücken geändert werden.

Alle Änderungen werden durch Neuschreiben des DOS auf Diskette gespeichert (DUP-Kommando INI). Für die mit * gekennzeichneten Werte wird garantiert, dass diese sich auch in zukünftigen XDOS-Versionen nicht ändern, die anderen Werte sollten in selbstgeschriebenen Programmen nicht benutzt werden.

* **XBUF = \$709** Anzahl der DOS-Puffer (wie bei DOS 2.5)

In \$709 steht die Anzahl der DOS-Puffer (maximal 5), jede gleichzeitig geöffnete Datei braucht einen Puffer. Der Normalwert ist 2, was nahezu immer ausreicht. Das XDOS-DUP benötigt nur 1 DOS-Puffer, lediglich zum Ausführen von Diskettenbefehlen in Batchdateien sind 2 Puffer erforderlich (außer für die letzte Batchzeile).

ACHTUNG: Jeder DOS-Puffer belegt 256 Byte Speicher und infolge dessen erhöht sich LOMEM (die Untergrenze des freien Speichers). Ist LOMEM zu groß, kann es sein, dass einige Programme nicht mehr funktionieren.

Bei 2 DOS-Puffern liegt LOMEM bei \$1DF0 (bei der XDOS-Normalversion), was fast immer ausreicht. Notfalls kann man \$709 auf 1 setzen, was LOMEM auf \$1CF0 senkt. LOMEM wird auch links oben im XDOS-Titel angezeigt.

* **XBIT = \$70A** Drive-Bitmaske (wie bei DOS 2.5)

Dieses Byte ist nur zur Kompatibilität mit DOS 2.5 vorhanden, es hat unter XDOS keine Bedeutung. Unter DOS 2.5 legen die 8 Bits von \$70A fest, welche der 8 Laufwerke D1: bis D8: angemeldet sind. Unter XDOS stehen immer alle Laufwerke D1: bis D9: zur Verfügung, daher ist dieses Byte gleich \$FF.

XOPT = \$70B Optionen

Dieses Byte steuert einige spezielle Funktionen des XDOS, und zwar wie folgt (alle Werte sind hexadezimal angegeben):

+20 RAM-Disk D8: beim Booten nicht löschen

- +10 Disks in mittlere Dichte beim Formatieren/Löschen MYDOS-kompatibel einrichten (statt DOS 2.5-kompatibel)
 - +02 Titelfinfos beim Aufruf des DUP nicht anzeigen
 - +01 Im DUP das Directory immer automatisch zweispaltig anzeigen
- Standardmäßig steht dieses Byte auf 00. Zum Ändern die gewünschten Optionen addieren und in \$70B eintragen.

*** XCAR = \$70C/D** Cartridge-Adresse (Low-Byte, High-Byte)

Hier kann man eine Rücksprung-Adresse für den CAR-Befehl ablegen. Wann immer XDOS normalerweise zum Cartridge bzw. BASIC geht (etwa beim Drücken von RESET oder nach dem Booten), wird diese Adresse angesprungen, falls sie >= \$100 ist. Die CAR-Adresse wird beim Booten und den DUP-Befehlen BON, BOF sowie NEW gelöscht.

Hier ein Beispiel für eine Batchdatei AUTORUN.SYS, die beim Booten eine CAR-Adresse für Turbo-BASIC setzt und dann Turbo-BASIC (TB.COM) startet:

```
“ Turbo-BASIC starten
=70C 2080
EXE TB.COM
```

Nach dem Booten kann man so jederzeit aus Turbo-BASIC mit „DOS“ zum DUP gehen, dort DUP-Befehle ausführen (sogar mit COP Dateien kopieren) und mit CAR oder RESET zurück zum Turbo-BASIC gehen.

ACHTUNG: Wenn man mit OPTION-RESET zum DUP geht, wird das Turbo-BASIC nicht neu initialisiert, d.h. LOMEM nicht hochgesetzt, was zur Folge hat, dass der COP-Befehl das Turbo BASIC überschreibt.

*** XRUN = \$70E/F** RUN-Adresse (Low-byte, High-Byte)

Dies ist die Adresse für den RUN-Befehl. Immer wenn man im DUP „RUN“ ohne Adresse dahinter eingibt, wird diese Adresse per 6502-JSR aufgerufen, falls sie nicht 0 ist. Die RUN-Adresse wird vom LOA- und EXE-Befehl sowie vom RUN-Befehl mit Adresse dahinter gesetzt und beim Booten sowie den Befehlen BON, BOF, NEW und COP gelöscht.

XCOL = \$710 Text-Hintergrundfarbe

Ist dieses Byte nicht 0, legt es die Text-Hintergrundfarbe fest. 0 bedeutet die Standardhintergrundfarbe.

XDRV = \$711-\$719 Laufwerk-Zuordnungstabelle

Jedes Byte \$71n legt fest, auf welches physische Laufwerk bzw. RAM-Disk beim Ansprechen von Dn: zugegriffen wird.

Für Disk-Laufwerke gibt es folgende Möglichkeiten (alle Werte sind hexadezimal angegeben):

- 00 kein Zugriff (Error 138)
- 01-09 Zugriff auf D1: bis D9:
- 0A-0F Zugriff auf Disk-Rückseite für D1: bis D6: (nur für mit F2# formatierte Disks)
- +10 XDOS-Floppy-Speeder abschalten (nur für diese Laufwerksnr.)
- +20 keinen automatischen XF-Density-Check ausführen (dann muss der DUP-Befehl TES benutzt werden)

Für RAM-Disks gibt es folgende Möglichkeiten:

- E0 800XL-RAM-Disk unter dem OS-ROM (14 KB)
- A0 130XE-RAM-Disk (64 KB)
- 80 Single Density (90 KB)
- C0 Medium Density (128 KB)

90 Double Density (180 KB)
 D0 High Density (256 KB)
 8C Zweite Single Density RAM-Disk (zusätzlich zur RAM-Disk „80“)
 +01 Nach Zugriff auf die RAM-Disk den DLI (Display List Interrupt) einschalten (wie bei DOS 2.5)

Die mit + aufgeführten Werte sind Optionen und können zu den eigentlichen Werten hinzuaddiert werden.

XKEY = \$71A/B Tastaturverzögerung und -wiederholrate
 \$71A legt die Verzögerung vom Niederdrücken einer Taste bis zum Einsetzen der Wiederholung fest, \$71B die Tastenwiederholgeschwindigkeit. AUTOCONF.COM setzt die Werte auf \$10 und \$02. Wem das zu schnell ist, der sollte \$18 und \$03 benutzen. Ein Wert \$00 bedeutet, dass XDOS die Tastaturverzögerung bzw. -wiederholrate nicht setzt.

XRAM = \$71C-\$71F Blocktabelle für die RAM-Disk
 Jedes Byte dieser Tabelle repräsentiert einen 64KB-Block Zusatzspeicher. Zum Zugriff auf die RAM-Disk schreibt XDOS das entsprechende Byte in PORTB (\$D301). Dabei dienen die Bits 2 und 3 zum Zugriff auf eine 16KB-Bank innerhalb der 64KB und müssen in der Tabelle stets 0 sein. Sofern Bit 0 und 1 nicht zur Blockauswahl dienen, müssen diese Bits auf 1 gesetzt werden (d.h. alle Werte haben dann die Form x3). Je nach benutzter RAM-Diskgröße und -typ müssen in den ersten Bytes dieser Tabelle die richtigen Werte stehen.

Ausnahmen: Die 800XL-RAM-Disk braucht keine Blocks und die 130XE-RAM-Disk benutzt nicht \$71C sondern \$71F als 64KB-Block (damit kann man z.B. gleichzeitig die 180KB-RAM-Disk und die 130XE-RAM-Disk nutzen). Schließlich können in den ersten 192 KB Zusatzspeicher auch zwei 90KB-RAM-Disks eingerichtet werden (Laufwerk-bytes 80 und 8C in die XDRV-Tabelle eintragen). Hier einige Einstellungen für bekannte Speichererweiterungen:

ATARI 130XE:	E3 E3 E3 E3 (64 KB)	ATARI-Mag./Megaram:	83 A3
C3 E3 (256 KB)			
Buchholz/ABBUC:	A3 C3 E3 E3 (192 KB)	Compy-Shop:	23 63 A3
E3 (256 KB)			

COM-Programme mit Parametern

XDOS erlaubt bei COM-Programmen die Übergabe von Parametern, diese können (durch Leerzeichen oder Komma getrennt) gleich hinter dem Dateinamen des COM-Programms stehen und/oder vom COM-Programm später eingelesen werden. Zur Auswertung der Parameter stellt XDOS einige Flags sowie vier Routinen zur Verfügung – diese dürfen nur von COM-Programmen benutzt werden, die den 6502-Stack nicht löschen (d.h. LDX #255: TXS ist nicht erlaubt):

XDOS = \$700 XDOS-Kennzeichen
XPAT = \$86F XDOS-Bugfix- und Patchnummer
XVER = \$870 XDOS-Versionsnummer

In XDOS=\$700 steht das ASCII-Zeichen "X". Programme, die nur ab XDOS 2.4 laufen, sollten dieses Byte prüfen und zusätzlich XGLIN=\$871 auf \$4C prüfen (zur Unterscheidung zu XDOS 2.3).

In XVER=\$870 steht die Versionsnummer von XDOS im BCD-Format, bei XDOS 2.4 steht hier \$24. Dies sollte in Programmen abgefragt werden, die nur mit XDOS 2.4 lauffähig sind (etwa ein Konfigurationsprogramm für XDOS 2.4).

XPAT=\$86F enthält im oberen Nibble die Bugfixnummer (z.B: 1 bei XDOS 2.41) und Bit 0 ist gesetzt, falls die Fastversion im Speicher ist.

XFILE = \$87D Dateinamen-Puffer

XLINE = \$880 DUP-Eingabezeile

Siehe die folgende Beschreibung.

XGLIN = \$871 Get Line

Diese Routine holt von dem durch das X-Register vorgegebenen Kanal (IOCB) eine max. 44 Zeichen lange Eingabezeile und legt sie im Eingabepuffer des DUP (ab Adresse XLINE=\$880) ab. Vor dem ersten Aufruf dieser Routine steht ab Adresse XFILE=\$87D "Dn:" und dahinter der Dateiname (ggf. mit Wildcards) des COM-Programms mit den dahinter-stehenden (durch Komma oder Leerzeichen getrennten) Parametern.

XSKIP = \$874 Skip Parameter

Setzt das Y-Register als Index in XLINE=\$880 auf den nächsten (durch Leerzeichen oder Komma getrennten) Parameter, nach dem Aufruf von XSKIP steht also das erste Zeichen des nächsten Parameters in XLINE+Y.

Ist dieses Zeichen ein EOL=\$9B, so bedeutet dies, dass kein weiterer Parameter folgt.

Steht z.B. vor dem Aufruf von XSKIP ab XLINE=\$880 "ASM 4F", so ergibt sich:

Ist vor JSR XSKIP Y=0-3, so ist nach dem Aufruf Y=4 und in XLINE+Y steht '4=\$34,

ist vor JSR XSKIP Y=4-6, so ist nach dem Aufruf Y=6 und in XLINE+Y steht EOL=\$9B.

XMOVE = \$877 Move Filename

Verschiebt eine ab XLINE+Y=\$880+Y stehende DUP-Dateiangabe (z.B. 2;PROG) nach vorne, so dass ab Adresse XFILE=\$87D ein gültiger FMS-Dateiname steht, der mit OPEN usw. verwendet werden kann, z.B.: Steht ab XLINE "2;PROG 12" und ist Y=0, so steht nach JSR XMOVE ab XFILE "D2;PROG 12".

Steht ab XLINE "FF TEST" und ist Y=3, so steht nach JSR XMOVE ab XFILE "Dn;TEST" (n=Standard-Laufwerk).

XGNUM = \$87A Get Number

Interpretiert eine ab XLINE+Y stehende Zahl (Hex oder Dez), im Akku wird das Low-Byte, im X-Register das High-Byte zurückgeliefert. Das Y-Register zeigt nach dem Aufruf dieser Routine direkt hinter die Zahl, z.B.:

Steht ab XLINE "E474 ABC" und ist Y=0, so ist nach JSR XGNUM A=\$74, X=\$E4, Y=4.

Treten bei Verwendung von XGLIN I/O-Fehler auf oder wird bei XGNUM eine falsche Zahl entdeckt, bricht XDOS das COM-Programm mit einem Fehler ab, das DUP zeigt dann Error xxx an. Gleiches gilt, wenn das COM-Programm eine Fehlernummer >=128 in ERRNO=73 ablegt und mit RTS zum XDOS zurückkehrt.

Auch kleine COM-Programme wie z.B. Utilities belegen meist Speicher oberhalb des DOS, daher sollte sichergestellt sein, dass nach dem Ausführen des COM-Programms das BASIC bzw. ein eingestecktes Cartridge neu initialisiert wird. Unter XDOS ist dies möglich, indem das COM-Programm APPMHI+1=\$000F auf 0 setzt. Das COM-Programm sollte ggf. auch die CAR-Adresse zurück auf 0 setzen.

Als Beispiel zur Parameterübergabe siehe das Assembler-Listing MAP.ASM zum Programm MAP.COM:

MAP.COM zeigt alle COM-Header (d.h. Ladebereiche) sowie die RUN- und INIT-Adressen einer COM-Datei an.

Mit "MAP Dateiname" oder nur "MAP" starten.

ACHTUNG: MAP benutzt den gesamten Hauptspeicher von \$2400 bis HIMEM.

Speicherbelegung und DOS-Kompatibilität

24-27	DUP-Zeropage (nur temporär, wie bei DOS 2.5)		
48-51	FMS- und DUP-Zeropage (nur temporär)		
67-73	FMS-Zeropage (wie bei DOS 2.5)		
\$700-\$1BEF	XDOS (FMS und DUP)	\$700-\$1D6F	XDOS-Fastversion (FMS und DUP)
\$1BF0	Startadresse der DOS-Puffer	\$1D70	Startadresse der DOS- Puffer bei der Fastversion
\$1DF0	LOMEM bei 2 DOS-Puffern	\$1F70	LOMEM bei 2 DOS- Puffern bei der Fastversion

XDOS ist auch intern kompatibel zu DOS 2.5. Die folgenden wichtigen Adressen befinden sich bei XDOS an der gleichen Stelle wie bei DOS 2.5:

XBUF = \$709	Anzahl der DOS-Puffer (legt Anzahl der gleichzeitig offenen Dateien fest)
XBIT = \$70A	Bitmaske für die unter DOS 2.5 zur Verfügung stehenden Laufwerke (bei XDOS immer \$FF)
BSIO = 1900	Block-SIO-Routine, damit können Sektoren direkt gelesen und geschrieben werden.
WRCMD = 1913	Steht hier ein "P", wird ohne Verify geschrieben, bei "W" wird mit Verify geschrieben, das Schreiben mit Verify ist allerdings 3mal langsamer als das Schreiben ohne Verify
DINIT = \$7E0	Initialisierungsroutine des DOS, wird nach jedem RESET aufgerufen
BLOAD = \$15C8	Häufig benutzter Einsprung zum Laden/Starten einer COM-Datei aus BASIC: Zuerst mit OPEN die COM-Datei öffnen, dann \$15C8 perUSR aufrufen (mit vorigem PLA).

Grundsätzlich empfiehlt es sich, möglichst keine Spezialfunktionen eines DOS in eigenen Programmen zu benutzen, damit diese auch unter anderen DOS-Versionen lauffähig sind. Ausnahme ist lediglich XIO 40 zum Laden von COM-Dateien, da XIO 40 auch von MYDOS und SpartaDOS unterstützt wird und somit eigentlich Standard ist. XIO 40 steht allerdings bei DOS 2.5 und Turbo-DOS nicht zur Verfügung.

Auch bei COM-Dateien, die Parameter einlesen, ist es natürlich unumgänglich, die dafür vom DOS zur Verfügung gestellten Routinen zu benutzen, in diesem Fall sollte aber unbedingt XDOS=\$700, XGLIN=\$871 und ggf. XVER=\$870 abgefragt werden, um sicherzustellen, dass auch die erwartete DOS-Version im Speicher ist.

XDOS 2.4 kann die Diskettenformate der folgenden DOS-Versionen korrekt bearbeiten:

DOS 1 (außer Append), DOS 2.0, DOS 2.5, DOS 2.2/2.3 (931 Sektoren-Format), DOS XL, (Happy-)DOS II+/D (alle Versionen), BIBO-DOS (außer Quad-Format), Turbo-DOS (außer Quad-Format), sowie MYDOS (außer Unterverzeichnisse, Disks mit mehr als 1040 Sektoren sowie Append an MyDOS-Dateien).

- XDOS 2.4 unterscheidet sich von XDOS 2.3 in folgenden Punkten:
- Bei BON/BOF bleiben resetfeste Programme erhalten, dafür gibt es zusätzlich den NEW-Befehl.
- Die CAR-Adresse wird jetzt beim Booten stets gelöscht.
- Außerhalb des DUP zeigt XDOS im Directory „xxx FREE SECTORS“ an (wie bei DOS 2.5).

- Das DOS 2.5-Laufwerksbyte XBIT=\$70A wird unterstützt.
- Die DOS 2.5-Adresse \$15C8 zum Laden von COM-Dateien aus BASIC heraus wird unterstützt
- SAV Datei/A – (Init Run) zum Anhängen einer INIT-/RUN-Adresse ist jetzt möglich.
- D9: wird unterstützt.
- Auch in der Normalversion werden alle XF551-Laufwerke automatisch erkannt.
- Das Hyper Speed-Protokoll der Hyper-XF-Erweiterung für die XF551 wird unterstützt.
- Mit dem Geräte Kürzel „C2:“ wird für Kassette nun auch das SAVE"C:"-Format unterstützt.
- Das H:-Gerät des Atari800Win-Emulators wird unterstützt.
- In Dateinamen ist auch der Unterstrich erlaubt (wie bei MYDOS).
- XIO 39 und XIO 40 wurden getauscht, dies macht XIO 40 kompatibel zu Sparta-DOS.
- Nach Ausführen einer Batchdatei mit XIO 39/40 wird immer ins DUP gesprungen.
- Batchdateien werden stets über Kanal #5 geladen (vorher Kanal #2).
- Die XDOS-Konfigurations- und Jumptabellen haben sich geändert (außer XBUF, XCAR, XRUN).
- Wird eine Datei in DOS.SYS umbenannt, ist diese Datei weiterhin nicht bootfähig.
- Statt > wird jetzt = benutzt (> wird in XDOS 3.x für Unterverzeichnisse benötigt).
- 6502-BRKs werden vom XDOS nicht mehr abgefangen und gemeldet.
- Das Geräte Kürzel „M:“ zum Zugriff auf die RAM-Disk D8: ist entfallen.
- Das Ausführen eines User-DUP-Befehls (das DUP-Kommando !) ist entfallen.

XDOS 2.4 Referenz

Allgemeines

HELP beim Booten	RAM-Disk D8: nicht löschen
ESC beim Booten	AUTORUN.SYS nicht ausführen
OPTION + RESET	Zum DUP springen
SELECT + RESET	Kaltstart ausführen (Booten)
START + RESET	Modul bzw. BASIC neu starten

Dateikürzel (auch außerhalb des DUP):

/N	keine Yes/No-Abfrage (No Query)
/Q	Yes/No-Abfrage (Query)
/S	wie /N, aber Namen anzeigen (Show Names)
/A	Append bzw. Directory zweispaltig
/D	Directory-Zugriff
+	Dateinamen mit beliebigen Zeichen (z.B. "D: +x")
-	Abkürzung für * . *
;	Alternative zu : (dann ist /Q statt /N Default)

Yes/No-Abfrage: Y / Return, N / Leertaste, Esc / Break

.COM	Endung für COM-Dateien
.CMD	Endung für Batchdateien

DUP-Befehle

Zahl: hex 'char .dezimal (Byte oder Adresse)
 Datei: Dateiname Dn:Dateiname n:Dateiname
 Gerät: Datei oder Geräte Kürzel (z.B. C:)

DIR Datei	List directory
REN Datei neuer-Dateiname	Rename file(s)
DEL Datei	Delete file(s)
LOC Datei	Lock file(s)
UNL Datei	Unlock file(s)
LOA Gerät	Load COM-File
LOA Gerät Ladeadr [Anzahl]	Load any file
EXE Datei	Execute COM/CMD
SAV Gerät Start End [Init - [Run]]	Save COM file
SAV Gerät Start End /	Save any file
SAV Datei/A - [Init - [Run]]	Append Init/Run
RUN [RUN-Adresse]	Run program
COP Quellgerät Zielgerät	Copy file(s)
TYP Gerät	Type file

; Abkürzung für CAR
 / Abkürzung für DIR
 Return Abkürzung für DIR Return
 Shift-Return Abkürzung für DIR 8 ; Return
 Ctrl-3 Abkürzung für = Return
 Datei Abkürzung für EXE Datei .COM/CMD

Wichtige Fehlercodes

128: Break-Taste gedrückt
 130: Geräte Kürzel unbekannt
 136: Unerwartetes Dateieinde
 137: Eingabezeile zu lang
 138: Gerät antwortet nicht (z.B. ausgeschaltet)
 139: Gerät kennt Kommando nicht (z.B. FD# auf 1050)
 140-143: Kommunikation gestört (z.B. Wackelkontakt)
 144: Kommando nicht ausführbar (z.B. Schreibschutz)
 146: Funktion nicht möglich (z.B. Lesen von "P: ")
 161: Nicht genug DOS-Puffer (\$709 erhöhen)
 162: Diskette voll
 164: Dateistruktur defekt oder falsche Disk
 165: Dateiname falsch (z.B. "D: #NAME ")
 167: Datei gesperrt oder MyDOS-Verzeichnis
 169: Inhaltsverzeichnis voll (max. 64 Einträge)
 170: Datei nicht gefunden
 180: Datei nicht ausführbar (beim EXE-Befehl)

181: Adresse zu klein/groß (z.B. Startadr > Endadr)
 182: Parameter falsch (z.B. RUN X765)

1;	bis 9;	Standard-Laufwerk wählen
D1;	bis D9;	Standard-Laufwerk wählen
=adr		Speicher anzeigen
=adr	zahl1 zahl2 ...	Speicher ändern
BON		BASIC on
BOF		BASIC off
NEW		Remove resident progs + BOF
CAR	[BASIC-Zeile]	Go to cartridge
DOS		Restart DUP
CLS		Clear screen
AUT		Execute D1:AUTORUN.SYS
TES	[Laufwerk ;]	Test density (for XF551)
INI	[Laufwerk ;]	Init disk (Write DOS.SYS)
Fx#	[Laufwerk ;]	Format disk (x = S , M , D , 2)
CL#	[Laufwerk ;]	Clear disk (clear directory)

Für Batchdateien:

" Text Kommentarzeile (muss 1. Zeile sein)
 ' Text Text anzeigen
WAI auf Tastendruck warten

YASH 1.0 for SIO2USB

(c) 2008 by Stefan Dorndorf
 Doku vom 23.08.200

Als Zugabe zu XDOS ist auf der XDOS-Systemdiskette noch YASH (Yet Another Shell) enthalten. Dies ist ein komfortabler Dateimanager für das SIO2USB-Gerät der ABBUC-Regionalgruppe RAF. YASH bietet folgende Features:

- Schneller Start (YASH ist nur 4,6 KB lang)
- Unterstützt alle SIO2USB Shell-Funktionen (außer STA, CAT, DIR Dx:)
- Komfortable Directory-Navigation mit den Cursorstasten
- Schnelles Directory-Listing (0,5 Sekunden pro Seite)
- Kann lange Dateinamen anzeigen (wie auf dem PC)
- Intuitiv bedienbar (Menüs, Eingabehilfen, Fehlermeldungen)

Ihr könnt YASH.COM mit jedem DOS oder Game-DOS starten. Am besten packt Ihr YASH zusammen mit Eurem Lieblings-DOS auf S2USHELL.ATR. Mit dem QMEG-OS könnt Ihr alternativ S2USHELL.ATR auch auf D3: mounten und YASH mit dem File-Loader des QMEG-OS starten - dann bleibt D1: immer frei.

Nach dem Start sieht YASH so aus:

```

YASH 0.50 for SIO2USB (c) S.Dorndorf
D1=\QMFFREEZ.ATR
D2=\INT\52U5HELL.ATR
D3=\WORK2.ATR

DIR\*. * #1
\*INT <DIR> Internal Memory
SIO2USB .CFG 51B 29.06.08 01:12:46
ADVEN <DIR> 04.04.08 16:37:58
SYS <DIR> 03.04.08 01:28:18
GAME51 <DIR> 03.04.08 01:29:08
BOOT4 <DIR> 04.04.08 16:38:04
BOOTJ <DIR> 04.04.08 16:41:52
BOOT5 <DIR> 04.04.08 16:43:18
FILES <DIR> 04.04.08 16:44:26
STD <DIR> 04.04.08 16:45:04
XD0524 .ATR 184KB 15.04.08 15:35:52
WORK .ATR 184KB 01.05.08 15:00:02
WORK2 .ATR 184KB 15.06.08 21:14:06
HSC <DIR> 29.04.08 22:44:16
DM05 <DIR> 02.06.08 01:48:02
QMFFREEZ .ATR 133KB 23.09.07 21:40:38

123 Drives DIR Root Spd Refresh Song
123 Files Del Parent Rtn Subdir/Mount
ESC Anabi Disabi SSwap Dnmnt Filter
    
```

Oben sind die ATR-Pfade der 3 vom SIO2USB unterstützten Laufwerke zu sehen (grün).

Darunter steht der Pfad (rot) des aktuellen Verzeichnisses samt dessen Verzeichnislisting (blau).

Ganz unten zeigt YASH alle möglichen Tastenkombinationen an (Hauptmenü). Dabei bedeutet ^ Control, z.B. ist ^S = Control-S. Die Shift-Tastenkombinationen erscheinen, wenn Ihr die Shift-Taste drückt (Shift-Menü).

Hauptmenü (Navigationsfunktionen)

- Control-Esc Zurück zum DOS (wenn ein DOS geladen ist)
- Leertaste Aktualisieren, d.h. Pfade und aktuelles Verz. vom SIO2USB lesen
- 1 2 3 ATARI-Laufwerk auswählen
- Cursor auf/ab Datei auswählen (ohne Control-Taste)
- Cursor rechts Verzeichnis-Seite vorblättern (ohne Control-Taste)
- Cursor links Verzeichnis-Seite zurückblättern (ohne Control-Taste)
- Clear Wurzel-Verzeichnis \ bzw. \INT listen (ohne Control-Taste)
- Insert Auf erste Verzeichnis-Seite gehen (auch mit Leertaste möglich)
- Backspace Ins Stammverzeichnis gehen (Parent-Directory)
- Return Gewählte Datei auf gewähltes Laufwerk (Tasten 1 2 3) mounten bzw. das gewählte Verzeichnis listen
- Control-E Gewähltes Laufwerk aktivieren
- Control-D Gewähltes Laufwerk deaktivieren
- Control-S Gewähltes und nächstes Laufwerk tauschen (1=1+2, 2=2+3, 3=3+1)
- Control-U Gewähltes Laufwerk unmounten
- F Pfad und/oder Dateifilter für Verzeichnislisting setzen
- G Mittels direkter Pfadeingabe in anderes Verzeichnis wechseln
- L Anzeige der langen Dateinamen ein- und ausschalten
- M Mittels direkter Pfad- und Namenseingabe ein ATR mounten
- Tab Menü wechseln (Hauptmenü oder Shift-Menü)

Shift-Menü (Dateifunktionen)

- Shift-Esc Kaltstart ausführen
- Shift-Return Gewähltes ATR auf D1: mounten und Kaltstart ausführen
- Shift-N Neues Verzeichnis anlegen
- Shift-A Neues ATR anlegen (Single, Medium, Double, Quad, Octal=720KB, High=1.4MB, 1-8=1-8MB)
- Shift-R Gewählte Datei bzw. Verzeichnis umbenennen

Shift-D	Gewählte Datei bzw. leeres Verzeichnis löschen
Shift-P	Gewählte Datei bzw. Verzeichnis sperren (Schreibschutz)
Shift-U	Gewählte Datei bzw. Verzeichnis entsperren
Shift-C	Gewählte Datei kopieren
Shift-S	Freie Bytes anzeigen (rechts unten im Verzeichnislisting)
Shift-T	Datum/Zeit anzeigen und/oder ändern
Shift-F/G/L/M	wie F/G/L/M im Hauptmenü
Shift-Tab	wie Tab im Hauptmenü

YASH ist eigentlich selbsterklärend. Hier einige Details zu den einzelnen Funktionen:

Verzeichnisse: Die Nummer der angezeigten Verzeichnis-Seite steht rechts über dem Listing. Sie zählt von 1 bis 9, dann von A bis Z. Weitere Seiten haben als Seitennummer ein +. Maximal 128 Seiten pro Verzeichnis sind möglich.

Tab-Taste: Anstatt die Shift-Taste zu drücken, könnt Ihr mit der Tab-Taste zum Shift-Menü (und wieder zurück) wechseln. Solange das Shift-Menü angezeigt wird, braucht Ihr nicht die Shift-Taste zu drücken, um die Shift-Funktionen auszuführen - das wird durch den roten Hintergrund verdeutlicht. Alle Funktionen des Hauptmenüs sind immer noch über ihre Tastenkombinationen erreichbar.

Ausnahmen: Shift-Esc und Shift-Return funktionieren immer nur bei gedrückter Shift-Taste (dies wird durch das ! im Shift-Menü verdeutlicht).

Pfadanzeige: Die Pfade werden oben angezeigt. Falls ein Pfad zu lang ist, wird nur dessen letzter Teil angezeigt - dies wird durch ein führendes < verdeutlicht. Falls YASH die Pfade des SIO2USB nicht einlesen kann, wird ein ? als Pfad angezeigt. Die maximal mögliche Pfadlänge beträgt 126 Zeichen (inklusive Dateiname). Deaktivierte Laufwerke werden mit # angezeigt (wie beim SIO2USB-Gerätemenü).

Anzeige aktualisieren: Ihr könnt das SIO2USB-Gerätemenü benutzen oder den USB-Stick wechseln, während YASH läuft. Danach solltet Ihr aber die Leertaste drücken, damit YASH die ATR-Pfade und das aktuelle Verzeichnis neu einliest.

Zurück zum DOS: Ihr könnt jederzeit mit Control-Esc zurück zum DOS gehen und YASH mit RUN wieder starten (sofern das DOS sich die letzte RUN-Adresse merkt). Dabei bleiben alle YASH-Einstellungen erhalten (aktuelles Verzeichnis, L-Taste, Tab-Taste).

Fehler: Falls ein Fehler auftritt, zeigt YASH die ATARI-Fehlernummer dezimal und die SIO2USB-Fehlernummer hexadezimal zusammen mit einer kurzen Erklärung an. Falls YASH die SIO2USB-Fehlernummer nicht ermitteln kann, wird \$?? angezeigt. Ihr könnt das Fehlerfenster mit einer beliebigen Taste schließen.

Achtung: Falls eine Fehlernummer ohne Erklärung erscheint (oder die Erklärung offensichtlich falsch ist), teilt mir das bitte im ABBUC-Forum mit (das sollte nicht vorkommen)!

Filter: Mit "F" könnt Ihr einen Filter für die anzuzeigenden Dateinamen setzen - YASH hängt an die Eingabe automatisch *.* an. Z.B. setzt "POL" den Filter auf "POL*.*", d.h. nur mit POL beginnende Dateinamen werden angezeigt. Wenn Ihr nur RETURN eingibt, wird der Filter zurück auf "*.*" gesetzt. Ihr könnt vor dem Filternamen auch einen Pfad mitgeben, um direkt in das so definierte Verzeichnis zu springen. Der aktuelle Filter wird oben in der DIR-Zeile angezeigt. Unterverzeichnisse werden nicht ausgefiltert, d.h. sie werden immer gelistet.

Invers-Taste: Damit könnt Ihr das Einlesen eines Verzeichnisses anhalten. Das ist z.B. nützlich, wenn Ihr in einem langen Verzeichnis per Filter nach einer Datei sucht. Wenn die Datei angezeigt wird, drückt Ihr die Invers-Taste, um nicht abwarten zu müssen, bis YASH den Rest des Verzeichnisses durchsucht hat.

Lange Dateinamen: Mit "L" könnt Ihr zwischen der Kurznamen- und Langnamen-Anzeige hin- und herwechseln. Bei aktivierter Langnamenanzeige erscheint ein Kreis neben dem "Long" im Menü.

Falls eine Datei keinen Langnamen hat, wird immer die Kurznamen-Anzeige (mit Datum/Uhrzeit) benutzt. Alle Dateifunktionen (wie Rename, ATR make) arbeiten immer mit Kurznamen - auch wenn im Listing die Langnamen angezeigt werden. Wenn Ihr eine Datei mit einem Langnamen umbenennt oder löscht, bleibt der Langname im Verzeichnis des USB-Sticks gespeichert, er wird aber nicht mehr angezeigt. Das führt aber zu keinem Fehler (dasselbe passiert, wenn man mit MS-DOS arbeitet).

Protect, Unprotect, Delete: Diese Funktionen werden sofort ausgeführt (Delete nur nach Bestätigen mit der RETURN-Taste). Danach geht der Dateicursor automatisch auf die nächste Datei, so dass Ihr durch Festhalten der P bzw. U-Taste schnell mehrere Dateien hintereinander bearbeiten könnt. Verzeichnisse können nur gelöscht werden, wenn sie leer sind. Das Verzeichnis \INT kann nicht entsperrt, umbenannt oder gelöscht werden, da es auf den internen Speicher des SIO2USB verweist.

Eingabehilfen: Die Funktionen Rename, Copy, New dir, ATR make, Time, Filter, Goto, Mount zeigen in einem Fenster eine Eingabehilfe zusammen mit einem Mini-Editor an. [] bedeutet dabei eine optionale Eingabe. Im Editor könnt Ihr alle normalen ATARI-Editortasten benutzen - die Cursorstasten auch ohne Control. Zeichen, die in YASH keine Bedeutung haben, könnt Ihr nicht eintippen. Benutzt keine führenden Leerzeichen! Statt \ könnt Ihr auch / eintippen (es wird aber immer \ angezeigt). Pfade könnt Ihr stets relativ oder absolut (mit führendem \) eingeben.

Copy: Ihr könnt den Zielpfad oder -dateinamen weglassen, aber nicht beide. Falls Ihr den Zieldateinamen weglasst, benutzt YASH den Quellnamen als Zielnamen. Ver-

zeichnungen könnt Ihr nicht kopieren: Ist ein Verzeichnis ausgewählt, passiert nichts, wenn Ihr Shift-C drückt.

ATR make: Ihr müsst den Namen des neuen ATRs gefolgt von einem Leerzeichen und einem weiteren Zeichen eingeben, das die Größe des ATRs festlegt: S=Single=90KB, M=Medium=130KB, D=Double=180KB, Q=Quad=360KB, O=Octal=720KB, H=High=1.4MB, 1-8=1-8MB

ATR direkt mounten: Zum Mounten könnt Ihr auch die Taste "M" benutzen, um Pfad und Namen des ATRs direkt einzugeben. Wenn Ihr die Eingabe mit Shift-Return abschließt, wird auf D1: gemountet und sofort gebootet (wie Shift-Return im Shift-Menü). Wenn der eingegebene Name keinen Punkt enthält, macht YASH folgendes:

1. Hat der Name mehr als 8 Zeichen, wird er auf 8 Stellen gekürzt und falls das letzte Zeichen des Namens eine Ziffer ist, wird diese an die 8. Stelle gesetzt.
2. Es wird .ATR angehängt und dann versucht zu mounten.
3. Wurde das ATR nicht gefunden, kürzt YASH den Namen auf 6 Stellen und hängt ^1.ATR an. Falls der Name mit einer Ziffer endet, benutzt YASH die Ziffer anstatt der 1. Dann wird ein zweiter Mount probiert.

Beispiele:

Wenn Ihr "POLEPOSITION" eingibt, versucht YASH zuerst POLEPOSI.ATR zu mounten und dann POLEPO^1.ATR. Wenn Ihr "STARRAIDERS2" eingibt, versucht YASH zuerst STARRAI2.ATR zu mounten und dann STARRA^2.ATR. Auf diese Weise müsst Ihr nicht die Anzahl der Zeichen zählen oder ein ^ eingeben, um ein ATR mit langem Dateinamen direkt zu mounten.

Fragen zu YASH und XDOS könnt Ihr im ABBUC-Forum loswerden.

*Viel Spaß mit XDOS und YASH!
Stefan Dorndorf*

SDG Statistical Data Graphics & Analysis /

Claudio Fernandez Loquenz
Anwendung -
Mathematisches Statistikprogramm



USER MANUAL

Statistical Data Graphics & Analysis for ATARI 8 Bits

Written for ABBUC contest 2008

INDEX	Page
1. Introduction	32
How do I load onto your computer SDG ATARI?	33
2. Variable EDITOR Module	33
3. Statistics descriptive Module	38
4. Regression Module	39
5. Andeva Module (design of experiments).	41
a) Fixed Treatments with complete randomization	41
b) Model full block	41
c) Random Model with factorial structure	42
6.- Modules for graphics	42
7.- Printing Module	43
8.- Module for Files Edition	43
9.- Summary commands in SDG	44
A.- ANNEXES	46

1. Introduction

Graphs and statistics (SDG) is a powerful software and easy to use. Through menu you can create and edit variables, manage files, describe variables in statistical form, perform simple and multiple regression, analysis of variance for experimental designs, two-dimensional and three-dimensional graph equations, build scatter diagrams and histograms. It may also generate random variables, transform variables, variables to convert various formats to interact with various programs, create pie charts, bar, etc. All results are easy to interpret and the user requires only minimal knowledge of statistics. SDG consists of several modular programs that interact with each other through interactive menus with the user.

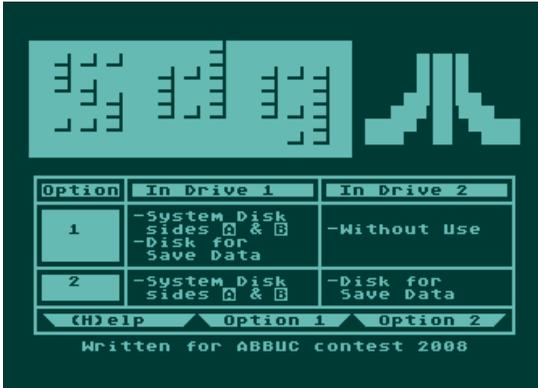
All modules are scheduled in TURBO BASIC with some subroutines into machine code that makes some processes more efficient. All programs are unprotected and can be listed and or modified.

With SDG you can graph their monthly expenses, statisticians calculate averages and other complexes, pooling their data in tables and get immediate histograms, to differentiate between products, predicting its sales or expenses, graph complex mathematical functions, and so on.

The configuration that requires you to use SDG in a physical ATARI:

1. A computer ATARI with a minimum of 48 Kb RAM.
2. A floppy drive ATARI
3. A diskette SDG (A and B sides system or two separate floppy disks) with modular programmes.

4. A blank floppy disk to store files: variables (data), graphics, tables, etc.
5. A printer EPSON RC-220 or equivalent, optional.
6. An optional printer plotter ATARI 1020 for three-dimensional Graphics



The configuration that requires you to use SDG in emulation with Atari800Win Plus:

- 1) Load Image **SDG-side-A.atr** in the Disk Drive 1
- 2) Load Image **SDG-Data.atr** in Disk Drive 2
- 3) Reset emulator and disable BASIC. When you start the program, choose Option 2

How do I load onto your computer SDG ATARI?

1. Turn on the drive 1 (and 2 drive in case of using two Drives)
2. Insert the SDG system disk by side A (or Diskette 1 if the system is recorded in two separate floppy disks).
3. Turn on your computer pressing OPTION
4. Choose the appropriate option on the screen and follow the instructions below.

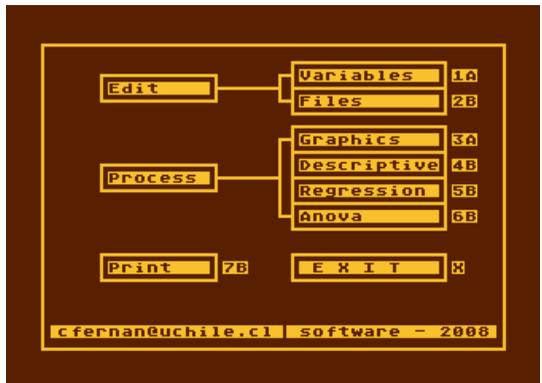
2. Variable EDITOR Module

Purpose: To create, modify or add data from a variable. The variables must be created or modified one by one, and before editing another variable, the above is recorded on the disk data.

Once released all relevant variables, the modules can be used for descriptive statistics, graphs, regression, or analysis of variance for design of experiments.

To increase the speed in the edition of variables and allow a greater degree of automation in the subsequent analysis of information, the editor lets you create variables according to their type (discrete or continuous variable), and agree to form that will be transferred to computer (or many variables with little data to be repeated). In general, a variable that is usually discreet is represented by categories or classes only because it makes no sense to establish ranks continued with their values. The variables are considered continuous if it makes sense to group their values range or ranges. However, even if the variable is discreet, can be classified as if it were ongoing when the number of different values is very high.

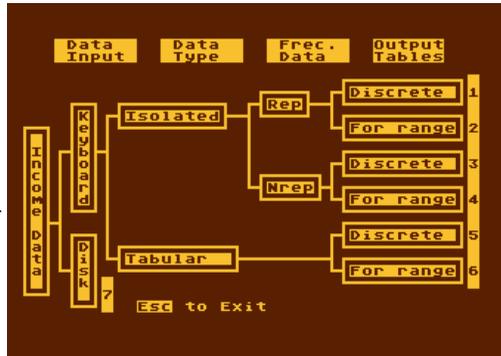
Options 1 and 3 are similar, they differ only in how they are introduced or edit data.



Options 2 and 4 are similar, they differ only in how they are introduced or edit data.
The SDG EDITOR distinction between:

a) Data isolated with values that are very repeated in the sample and to be grouped into categories with only tables (no class intervals). In this case the variable is considered Discrete. The publication of such data is done with option 1

Example: Variable daughter. The number of children from 27 families. As we enter the 27 data of this variable (daughter), it is easier to enter using the OPTION 1 program, ie variable discrete values with repeated.



2	2	2	2	3	1	1	2	3	2	1	2	3	2	1	2	3	0	2	2	0	0	0	2	1	1	1
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Note that the value 0 has a frequency of 4, the value 1 a frequency of 7, the value 2 a frequency of 12, and worth 3 a frequency of 4, ie, with many repeated values. 1 with the option of SDG facilitates the entry or editing of this type of variable.

When the descriptive process of variable with SDG (tabulation and calculation of parameters), will be something like this:

Daughter	Frequency	Frec. Acum.	Frec. %	Frec.Acum. %
0	4	4	14.81	14.81
1	7	11	25.93	40.74
2	12	23	44.44	85.19
3	4	27	14.81	100
Totales	27	-	100	-

→ Note that the categories are not intervals (Discrete Variable)

b) Data isolated with values that are not (very) repeated in the sample and to be grouped into categories with only tables (no class intervals). In this case the variable is also considered Discrete. The publication of such data is done with Option 3.

Example: Height. The height of 13 floors (m) measures with a decimal precision. Note that can be described discrete variables even if they have decimals. Although you can also use option 1, it is easier to use option 3

1.6	1.5	1.5	1.7	1.8	1.7	1.0	1.5	1.8	1.0	1.3	1.7	1.7
-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----	-----

When the descriptive process of variable with SDG (tabulation and calculation of parameters), will be something like this:

HEIGHT	Frecuency	Frec. Acum.	Frec. %	Frec.Acum. %
1.0	2	2	15.38	15.38
1.3	1	3	7.69	23.08
1.5	3	6	23.08	46.15
1.6	1	7	7.69	53.85
1.7	4	11	30.77	84.62
1.8	2	13	15.38	100
Totals	13	-	100	-

→ Note that the categories are not intervals (Discrete Variable)

Observations: The variable can be discreet when considering their classification in frequency tables does not involve the use of intervals or ranges. The classification tables in a unique class of discrete variables are possible only if the number of values "different" observed in the variable does not exceed 20.

c) Data isolated with values that are repeated in the sample and to be grouped into categories with tables intervals. In this case the variable continue is considered. The publication of such data is done with Option 2. We can also use option 4, but I would write one to one data.

Example: Variable weight. The weight (kg) of 40 persons:

45	45	45	50	50	52	40	50	48	48	38	45	50	40	50	45	45	50	37	37
45	37	37	39	50	37	42	48	45	45	45	50	52	37	37	45	45	45	48	38

When the descriptive process of variable with SDG (tabulation and calculation of parameters), will be something like this:

WEIGHT (kg)	Average	Frecuency	Frec. Acum.	Frec. %	Frec.Acum. %
Lim. Inf. - Lim. Sup.					
$36 \leq x \leq 39$	37.5	10	10	25.00	25.00
$39 < x \leq 42$	40.5	3	13	7.50	32.50
$42 < x \leq 45$	43.5	13	26	32.50	65.00
$45 < x \leq 48$	46.5	4	30	10.00	75.00
$48 < x \leq 51$	49.5	8	38	20.00	95.00
$51 < x \leq 54$	52.5	2	40	5.00	100
Totals	-	40	-	100	-

→ Note that the categories are expressed by intervals (continuously variable). SDG automatically calculates the required number of intervals and lower and upper limits for each category.

d) Data isolated with values that are not (very) repeated in the sample and to be grouped into categories with tables intervals. In this case the variable is considered Continuing. The publication of such data is done with Option 4. It also could use the option 2 but would have to write one to one data over a frequency value of one for each input.

Example: Variable height (in centimeters) of 30 persons:

146	155	145	167	178	177	160	145	178	180	173	157	177	165	171
148	175	165	176	149	150	162	163	166	147	151	154	177	157	152

When the descriptive process of variable with SDG (tabulation and calculation of parameters), will be something like this:

HEIGHT (cm)	Average	Frecuency	Frec. Acum.	Frec. %	Frec.Acum. %
Lim. Inf. - Lim. Sup.					
$145 \leq x \leq 151$	148	8	8	26.67	26.67
$151 < x \leq 157$	154	5	13	16.67	43.33
$157 < x \leq 163$	160	3	16	10.00	53.33
$163 < x \leq 169$	166	4	20	13.33	66.67
$169 < x \leq 175$	172	3	23	10.00	76.67
$175 < x \leq 181$	178	7	30	23.33	100
Totals	-	30	-	100	-

Note that the categories are expressed by intervals (continuously variable). SDG automatically calculates the required number of intervals and lower and upper limits for each category .

e) The data come from a table with discrete classes or categories. In this case use the Option 5.

Example: Classification of the number of children in 100 families:

Number of children		Number of families (frecuency)
Clase 1:	0	12
Clase 2:	1	33
Clase 3:	2	30
Clase 4:	3	15
Clase 5:	4	8
Clase 6:	5	2

f) The data come from a table with classes or categories of intervals. In this case the use OP-TION

Example: Classification of salaries of 300 employees.:

SALARIES		Number of employees (frequency)
Lim. Inf.	Lim. Sup.	
Class 1:	$0 \leq x \leq 20000$	100
Class 2:	$20000 < x \leq 40000$	80
Class 3:	$40000 < x \leq 60000$	70
Class 4:	$60000 < x \leq 80000$	30
Class 5:	$80000 < x \leq 100000$	20

Observations: The module Descriptive Statistics processed properly only when all these tables have equal intervals breadth. The upper limit of the intervals is included in each class, however the lower limit of the intervals is excluded from each class, except in Class 1.

Options 1, 2 and 3, 4 respectively, are equivalent and differ only in how you can enter data.

When the module is used descriptive statistics to analyze the data, must be chosen carefully between options 1, 2, 3, 4, 5 or 6, because it depends on which tables are incorporated into classes resulting unique or Ranges (Intervals).

The modules to build chart, making regressions, or to design experiments, they can only use variables entered with options 1, 2, 3 or 4.

Options 1 to 6 of the editor of variables are used only to enter variable (data) through the keyboard.

Option 7 is used to edit variables taped drive through any of the options described above,

Note: If you mistakenly entered the publisher, to leave him with enough pressure <Esc>.

Example: Enter through the editor the following variables (will be used later to exemplify processes). As there are few data and are not repeated, use option 3 of the publisher. There are two variables: An independent variable YEARS (YEARS) and a dependent variable production (PROD):

Var1:	YEARS	2000	2001	2002	2003	2004	2005	2006	2007
Var2:	PROD	5	10	165	25	38	52	68	84

To edit the variables above, follow these steps:

1. Turn on the drive Atari 1050
2. It floppy system by the side A
3. When the home screen of SDG, 1 choose whether their data are saved on a diskette that will drive the 1 (exchange with the floppy System), or 2 if their data is saved on a diskette that will drive the 2 (Another floppy 1050).
4. When displayed the main menu, choose Option 1 (EDITOR variable).
5. Choose option 3 Editor Variables (Keyboard - Discretos - not to repeat).
6. Enter the number of data containing the variable: 8.
7. Enter the data for one to one variable YEARS:
2000<return>

2001<return>
 2002<return>
 2003<return>
 2004<return>
 2005<return>
 2006<return>
 2007<return>

- 8) Check your data and verify they are correct.
- 9) Accept the income of the variable (Press A)
- 10) Insert the disk data and press <return>.
- 11) If you have not created variables that floppy, the directory will appear empty. Press "E" to finish watching the directory.
- 12) Enter the name of the variable: write YEARS and press <return>.

Repeat entire process prior to the variable PROD (Production).
 Finally, return to the main menu by pressing <return>.

3. Statistics descriptive Module

This module can describe any statistically variable hospitalized with options from 1 to 6 Editor variables. Handing the following information:

STATISTICS OF POSITION AND DISPERSION	
ARITHMETIC AVERAGE	1.59259259
GEOMETRIC AVERAGE	0
MEDIAN	2
QUARTILE 1	1
QUARTILE 3	2
MODE 1	2
MINIMUM VALUE	0
MAXIMUM VALUE	3
RANGE	3
MEAN DEVIATION	0.8091168096
VARIANCE	0.83401922
STANDARD DEVIATION	0.9132465275
VARIATION COEFFICIENT	57.3433867
SLOPE OF PEARSON	-0.4461089867
KURTOSIS PERCENTILIC	-0.0963333334
1TB 2HP 3HA 4PM 5PE 6SV 7PR 8EX (<=>)	

- a) Table of frequencies with complete and absolute frequency percentage, and accumulated partial . If the variable is discreet, the maximum number of categories or classes is 20. When the variable is continuous, the number of intervals is calculated according to the Rule of Sturges, who said: Number of intervals = $1 + 3.3 * \text{Logbase10}(N)$, where N is the number of isolated data. Example: If N = 200, then number intervals = $1 + 3.3 * \text{Logbase10}(200) = 9$
- b) Histograms of frequency partial and cumulative
- c) Position parameters: Arithmetic mean, Geometric mean, median, mode (one or more), Percentiles, Minimum and maximum value.
- d) Dispersion parameters: Variance, Standard deviation, rank, and coefficient of variation. The variance calculated corresponds to that of the population, and the Standard deviation is the square root of the variance earlier. The coefficient of variation is calculated as the standard deviation divided by the arithmetic mean.
- e) Bias Pearson
- f) Kurtosis percentile

At the end of this manual specifies all the formulas used by SDG.
 Examples:
 Studying the behavior of descriptive:

a) the variable PROD defined anteriormente with SDG

b) The variable X as defined below.

Use Editor variables (option 4) to enter the following values of variable X:

12.23	13.00	18.07	15.36	17.23	12.22	22.34	26.21	18.78	11.09
15.22	18.76	12.25	14.56	34.24	17.23	16.33	16.00	18.45	23.67

For both variables, once recorded, insert the floppy on the side B and choose Option Descriptive (4B). Then follow the instructions on screen.

When asked if he recorded the results and you answered yes, the original data will be recorded and ordered not ordered (as admitted). These data will be recorded as a screen and can only be read later with the printing module (7B).

When the results appear on screen Module descriptive, they will be distributed over six screens, the first two for the table of frequencies, for the following parameters, and the last three for Percentiles.

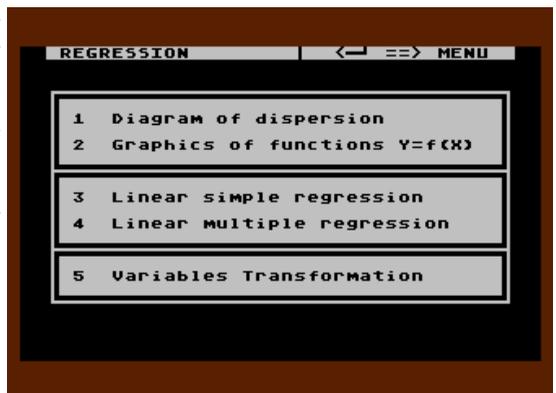
The menu commands appears on the bottom line of the screen, and allows the following:

- 1) Show from the beginning the table frequencies.
- 2) Show Histogram Frequency Partial
- 3) Show Histogram Frequency Cumulatively
- 4) Show Parameters
- 5) Show the first screen Percentiles
- 6) Record six screens described above.
- 7) Print the resultados.
- 8) Exit to Main Menu

The left and right arrows are used to move half of screens to the left or right respectively. To print or save the results, previously always press Option 1.

4. Regression Module

The module is loaded from the Main Menu B-side of the disk system. Insert the disk data and press <return>, you'll see that in diquete data are recorded by the two variables you: YEARS and PROD. Press "E" to finish watching the directory and enter as it prompted the dependent variable and the independent variable PROD YEARS. Then choose the menu screen the regression model most appropriate, or adjust all models and then pick the best of them according to the results (Coefficient of Determination and analysis of variance).



While it can be done using multiple regression models simple, you can transform any variable (add a constant, calculate logarithms, etc.). And then perform regression with the variables processed. For example, if you want to adjust the model is $Y = A + B / \text{SQR}(X)$ transforms the variable X as follows: first calculate the square root of the original variable and save this transformation with some name in the floppy data. Then transform the variable created previously by calculating the reciprocal of it.

Notación	X	à	SQR(X)	à	1/SQR(X)
Guardar como	Var1	à	Var2 (o var 1)	à	Var3 (o var2 o var 1)

Later, make a linear regression of the dependent variable "Y" on the new variable "1/SQR(X)". Virtually all models of equations are linearization through transformations. The same method is applicable to the case of multiple regression.

Observation: To record the results of regression, pre-press option 1 which rewrites the screen and press option 6.

Also you can see the chart all the sampling points and adjusted model with options 3, 4 or 5. Only you can save or print figure high resolution obtained with the option 5.

The estimated values of the equation adjusted obtained with option 2.

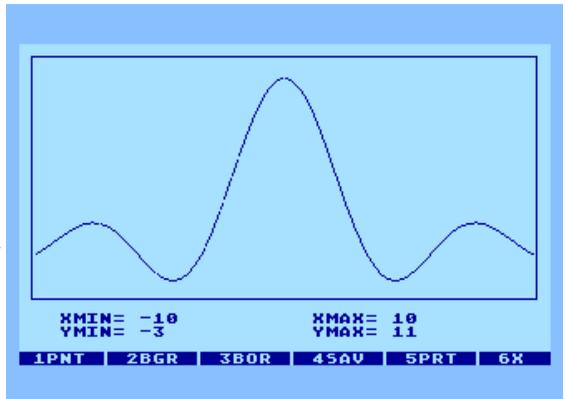
The graphics produced by the form of regression (regression simple charts and diagrams scatter graph functions) while increasing its resolution greater the time remaining on the screen.

- Graphics produced by regression simple:**
- SELECT: amending intensity of background color.**
- OPTION: changes color intensity points.**
- START + SELECT: chart records.**
- START + OPTION: graphic prints.**
- START: back to previous screen.**

Graphics functions and scatter diagrams:
When the graphic has reached the desired resolution pressures START and you'll see a menu on the bottom line of the screen that lets you:

- 1, 2 and 3: modify the intensity of background color, and border points respectively.**
- 4: graphic record.**
- 5: graphic print.**
- 6: leave the graph.**

The possibilities are many delivery this module, and only practice with the different options will allow you to take full advantage. All screens are sufficiently interactive with the user. All programs are modular unprotected, so you can review and modify some of its characteristics. Any problem that has to run the programs, you can reset with SYSTEM RESET and re-run with RUN.

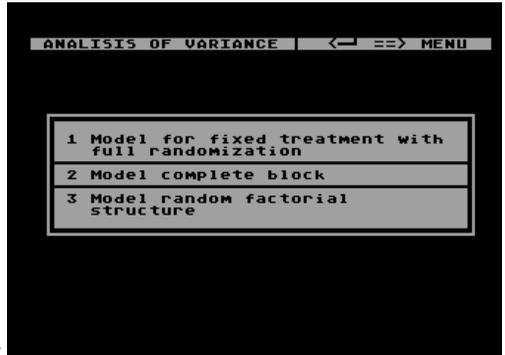


5. Andeva Module (design of experiments)

The module calculates and constructs the table Analysis of Variance (Andeva) to study the existence of significant differences between different treatments. You have the option of bringing their design of experiments to one of the following models:

a) **Fixed Treatments with complete randomization.** This model is used when the treatments were randomly assigned to the experimental units.

Example: Suppose you have four treatments (A, B, C and D) relating to four different drugs to combat disease. The drugs are tested in groups of people, repeating the process five times. The response of the drug was measured by calculating the percentage of people who Sanaa. The results were as follows:



Treatment	A	B	C	D
	40	50	80	30
	30	70	80	40
% of people who heal (5 repetitions for each drug)	20	60	90	50
	10	40	50	50
	50	30	100	30

The Analysis of Variance helps us to determine with any degree of confidence (as proof F) if possible to say that the effects of drug A, B, C or D are significantly different.

To solve this problem, you must enter each of the treatments through EDITOR variable SDG. In other words, you must create 4 variable data with 5 each. This can be used for instance option 3 Editor.

After creating the variables in the data disc, go to the Main Menu, where Choosing 6, which bear the form of analysis of variance. Then choose option 1 of the following menu that appears on screen (Model treatments fixed completely randomized). Put your floppy and enter data from one to one variable (treatments) released earlier.



b) **Model full block.** When there is no uniformity for all experimental units, can be separated into blocks, each with characteristics uniform. So for example, a soil fertility of

different stripes, each of the strips to form a bloc passes, and within each bloc, the experimental units are assigned randomly.

Blocks	Treatments				
	1	2	3	4	5
Block I	2	4	12	9	3
Block II	3	6	14	6	6
Block III	4	8	15	4	3
Block IV	3	10	11	5	4

c) Random Model with factorial structure.

When each treatment consists of the variation of 2 or more factors, the design has factorial structure. By example, if one takes 2 factors, the first with two different levels and the second with three levels, treatments will be trained $2 \times 3 = 6$.

Example:

Suppose that a trial is conducted with two fertilizers (A and B) to test its effectiveness in separately and together (interaction). Using three doses for each fertilizer (A1, A2, A3 and B1, B2, B3), the combination therapies will be formed among them: A1B1, A1B2, A1B3, A2B1, A2B2, A2B3, A3B1, A3B2 and A3B3 ($3 \times 3 = 9$ treatments). Suppose you are four replicates per treatment, and results in production are as follows:

RANDOM MODEL FACTORIAL		
ANALISIS OF VARIANCE		
F.V.	G.L.	S.C.
Trat	3	1879.2667
A	1	-4505.7333
B	1	-8871.6333
AB	1	14956.6333
Err.	36	5458.2
Tot.	39	7937.4667

F	Calculated	treatm=	3.47206047
F	Calculated	fact A=	-29.71792809
F	Calculated	fact B=	-58.51357594
F	Calc.	factor	AB= 28.64768626

1 SAVE 2 PRINT 3 EXIT

		A1B	A1B	A1B	A2B	A2B	A2B	A3B	A3B	A3B
		1	2	3	1	2	3	1	2	3
PRODUCTION		2	4	4	4	5	4	4	6	10
(ton/acre).	4	3	3	6	3	3	8	3	5	4
repetitions	per	1	2	4	2	7	4	5	8	8
treatment		2	5	2	1	5	8	4	7	10

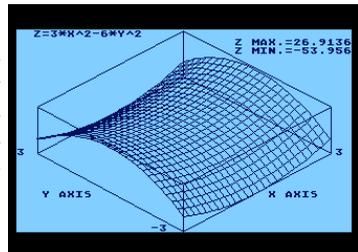
The 9 treatments are paid in advance as variables through Editor Variables, and then uses the module Andeva.

6. Modules for graphics

The graphics are the most common statistical bar (simple and grouped) and circulars (sectoral or cakes).



Any variable can be created with SDG graficada with this module, provided it is within their capabilities. The pie charts



require a maximum of one variable, however, the bar allowed up to three variables (all with the same number of data for each figure represents a bar). These graphs can be recorded and printed on paper.

This module also allows the variables are entering through the keyboard directly without passing through the publisher of variables.

Another interesting option is to graph functions of three variables in three-dimensional shape.

Observations: These graphics capability are not creción the author of SDG. The bar charts and circulars come from the software "GRAFIQUELO", and three-dimensional graphics are known as "Z_PLOTTER". These programs have been adapted to use modular SDG.

7. Printing Module

It serves to recover on the screen and / or any file printer (graphic or not) product of the process of recording them from any form of SDG.

8. Module for Files Edition

Lets see any floppy directory, rename, delete, protect or unprotect files. Also used to format disks. In other words, it avoids having to leave the DOS to conduct these operations.

It also has another option to convert the variables created with SDG in other formats of the same SDG or to make them interchangeable with other programs. This option allows the following:

- Make variables with DIF format to format SDG
- Make variables with SDG format to format DIFF
- Make variables with SDF format to format SDG
- Make variables with SDG format to format SDF
- Make variables with SDD format to format SDC
- Make variables with SDC format to format SDD

Format:

SDG: Format used by the program itself.

DIFF: Data Interchange Format.

SDF: Standard Data Format (Column pure data)

SDD: Discrete Variable created with the Editor SDG

SDC: Variable cptinua created with the Editor SDG



The transfer to DIF format makes the variables created with SDG can be used by other programs like VisiCalc for example. They can also create data with other programs and through the DIF format, can be caught by SDG.

9. Summary commands in SDG

Click 1	Click 2	Result in the Editor of Variables
1	1	Create, edit and burn discrete variables with values that are repeated.
1	2	Create, edit and burn continuous variables with values that are repeated.
1	3	Create, edit and burn discrete variables with values that are not repeated constantly.
1	4	Create, edit and burn continuous variables with values that are not repeated constantly.
1	5	Edit and recorded discrete variables that are tabulated in frequency tables to be processed by the module Descriptive Statistics.
1	6	Edit and recorded discrete variables that are tabulated in frequency tables to be processed by the module Descriptive Statistics.
1	7	Edit any variable disk previously recorded through Module Editor Variables SDG.

Click 1	Click 2	Click 3	Result in the File Editor
2	D	-	Displays directory of any disc
2	B	-	Clears files of any disc.
2	R	-	renames files from one disk
2	P	-	Protects files from one disk
2	E	-	Unprotected files from one disk
2	C	1	Converts variables with DIF format to format SDG
2	C	2	Converts variables with SDG format to format DIFF
2	C	3	Converts variables with SDF format to format SDG
2	C	4	Converts variables with SDG format to format SDF
2	C	5	Converts admitted as discrete variables to be processed by the processor as continuing Description of SDG
2	C	6	Converts admitted as a continuous variable to be processed by the processor as discrete Description of SDG

Click 1	Click 2	Click 3	Result in Gráphics
3	D	B	Create Bar graphs, capturing data variables recorded in Disk.
3	D	C	Create pie charts, capturing data variables recorded in Disk.
3	T	B	Create Bar graphs, entering data directly from the keyboard.
3	T	C	Create pie charts, entering data directly from the keyboard.
3	T	T	Create thridimensional graphics to enter with the keyboard the function of three variables.

Click 1	Result in Graphics
4	Process the variables entered with descriptive statistics.

Click 1	Click 2	Click 3	Results in Graphics, Regressions and tranformations.
5	1	-	Graf Dispersion Diagram between two variables recorded in Disk
5	2	-	Graf functions of two variables entered with the keyboard.
5	3	-	Make simple linear regression between two variables recorded on the Data Disk.
5	4	-	Make Multiple Linear regression between 2, 3, or 4 variables recorded on the Data Disk.
5	5	A	Generates variables with random data.
5	5	B	Generates another variable ordering growing data in the form of a variable saved on disk.
5	5	C	Generates a variable adding a constant to another variable of data recorded on disk.
5	5	D	Generates a variable multiplying a constant per another variable of data recorded on disk.
5	5	E	Generates another variable bringing to power a variable data recorded on disk.
5	5	F	Generates another variable adding one to one of the data of two variables recorded on disk.
5	5	G	Generates another variable by subtracting one-one data recorded on disc of two variables.
5	5	H	Generates another variable multiplying one-one data of two variables recorded on disk.
5	5	I	Generates another variable dividing one-one data of two variables recorded on disk.
5	5	J	Generates another variable by calculating the square root of the data variable saved in a disk.
5	5	K	Generates optionally another variable calculating the natural logarithm of data variable saved in a disk.

Click 1	Click 2	Click 3	Results in Graphics, Regressions and tranformations.
5	5	L	Generates optionally another variable calculating the Briggs logarithm of data variable saved in a disk.
5	5	M	Generates another variable by calculating the reciprocal of the data variable saved in a disk.
5	5	N	Generates another variable by calculating the absolute value of a data variable saved on disc.
5	5	O	Generates another variable becoming zeros the negative values of a variable saved to disk.
5	5	P	Generates another variable becoming -1 the negative values of a variable saved to disk.
5	5	Q	Generates another variable calculating Sin(x) to a variable data recorded on disk.
5	5	R	Generates another variable calculating Cos(x) to a variable data recorded on disk.
5	5	S	Generates another variable calculating ArcTan(x) to a variable data recorded on disk.
5	5	T	Generates another variable calculating the whole party to a variable data recorded on disk.
5	5	U	Generates another variable rounding to the decimal specified recorded in disk.
5	5	V	Generates another variable calculating Exp(x) to a variable data recorded on disk.

ANNEXES

Reference on some formulas used by SDG:

A) Descriptive Statistics

Arithmetic Mean	$\mu = \frac{\sum_{i=1}^n X_i}{n}$
-----------------	------------------------------------

Median	$Me = L_i + \frac{\frac{N}{2} - F_{(i-1)}}{f_i} \cdot A$
--------	--

Mode	$MO = L_i + \frac{f_i - f_{(i-1)}}{(f_i - f_{(i-1)}) + (f_i - f_{(i+1)})} \cdot A$
------	--

Geometric Mean	$Mg = \sqrt[n]{x_1 \cdot x_2 \cdot x_3 \cdot \dots \cdot x_n}$
----------------	--

Variance
$$\sigma^2 = \frac{\sum_{i=1}^n (x_i - \mu)^2}{n}$$

Standard Deviation
$$\sigma = \sqrt{\sigma^2}$$

Coefficient of Variation
$$CV = \frac{\sigma}{\mu}$$

Pearson Bias
$$S = \frac{\mu - Mo}{\sigma}$$

Percentile Kurtosis
$$K = \frac{P_{75} - P_{25}}{2(P_{90} - P_{10})} - 0.263$$

Range
$$R = (Higher\ data) - (Smaller\ data)$$

Other measures of position can be obtained from percentiles, for example:

Quartile 1
$$Q_1 = P_{25}$$

Notation:

x_i : values of a variable (*data*) : $x_1, x_2, x_3, \dots, x_n$

n : Number of data in the variable

L_i : Lower Limit in the i -interval of frequencies table.

$F_{(i-1)}$: absolute frequency accumulated in the interval $(i - 1)$

f_i : absolute frequency parcial in the interval (i)

$f_{(i-1)}$: absolute frequency parcial in the interval $(i - 1)$

$f_{(i+1)}$: absolute frequency parcial in the interval $(i + 1)$

A : Size intervals of the frequencies table

A) Regression

If the model $Y = \alpha + \beta x + \epsilon$, then a and b are estimators of α and β respectively. The coefficients a and b of the equation $Y = a + bx$ obtained using the method of least squares. Thus:

$$b = \frac{n \sum_{i=1}^n (x_i y_i) - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - (\sum_{i=1}^n x_i)^2} \quad a = \bar{Y} - b\bar{X}$$

In a similar manner calculated parameters in multiple linear regression.

In a similar manner calculated parameters in multiple linear regression. The variance analysis for regression is used to determine the extent to which the regression model used explains the phenomenon being studied across the variables used. It also measures the significance of each variable.

The value of F calculated to be compared with the value of Snedecor in the distribution of probabilities F given in statistical tables in books.

We must employ the necessary degrees of freedom (given in the analysis of variance), and a confidence level desired (90, 95 or 99%). If the value of F calculated on the analysis of variance is greater than the F to the table of probabilities, then the ratio is significantly different from zero, ie should be part of regression model.

The variance analysis to design experiments is studied in a similar manner, except that in this case is discussed if there are significant differences between treatments, or between factors if the design has factorial structure. In the latter case shows whether or not there is significant contributions of the various factors and their interactions in the response observed in the expe-

