

XDOS 2.43

(c) 2009 by Stefan Dorndorf

Doku vom 10.04.2009

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 ausgeführt 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 direkt hinter (!) Dateinamen stehen (ohne Leerzeichen dazwischen – z.B. „DATEI.COM/A“):

/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).

1; 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:

<S> = Single (90KB) <M> = Medium (128KB) <D> = Double (180KB)
 <L> = 14KB-RAM-Disk <X> = 130XE-RAM-Disk <H> = 256KB-RAM-Disk

Beispiele: DIR oder / oder DIR *.* oder DIR - listet alle Dateien.

DIR *.BAS oder /*.BAS listet alle BAS-Dateien.

/2; oder DIR 2; oder DIR 2;- oder DIR 2;*. * listet das Directory von D2:.

DIR *.* /A oder DIR -/A oder DIR /A oder /*.* /A oder -/A oder //A listet das Directory zweispaltig.

REN Datei neuer-Dateiname

Benennt Dateien um, z.B. REN 2;*.BAS *.BAK benennt alle BAS-Dateien auf D2: um. Dabei findet für jeden Dateinamen eine Yes/No-Abfrage statt. Wird DOS.SYS umbenannt, ist die Diskette nicht mehr bootfähig.

ACHTUNG: XDOS prüft nicht nach, ob der Zieldateiname auf der Disk schon existiert. Wenn man nicht aufpasst, kann es daher passieren, dass zwei oder mehr Dateien den gleichen Dateinamen bekommen. Diese doppelten Dateinamen lassen sich mit der Yes/No-Abfrage oder – wie bei DOS 2.5 – durch einen REN-Befehl ohne Wildcards wieder trennen.

DEL Datei LOC Datei UNL Datei

Mit diesen drei Befehlen werden Dateien gelöscht (DEL), gesperrt (LOC) bzw. entsperrt (UNL).

Beispiel: Mit LOC *.BAS/S werden alle BAS-Dateien gesperrt, dabei werden die Dateinamen angezeigt.

LOA Datei LOA Datei Ladeadr (Anzahl)

LOA Datei Ladeadr lädt die angegebene Datei ab der „Ladeadr“ in den Speicher (max. \$9B00 Byte). Wird „Anzahl“ mit angegeben, so werden nur die ersten „Anzahl“ Bytes der Datei geladen. Nach Ausführung des Befehls steht die Anzahl der tatsächlich geladenen Bytes in \$358/9 (und kann mit dem DUP-Befehl =358 ermittelt werden).

Wird nur LOA Datei ohne Zahlen dahinter eingegeben, so lädt XDOS die Datei nur dann, wenn sie im sogenannten COM-Format (DOS Binary Load-Format) abgelegt ist. In der Datei enthaltene INIT- oder RUN-Adressen werden dabei nicht (!) ausgeführt. Die RUN-Adresse wird im XDOS-Titel unter <Run> angezeigt und kann mit dem RUN-Befehl (ohne Parameter) aufgerufen werden.

Beispiele: LOA TEXT 9C40 lädt die Datei TEXT ab Adresse \$9C40 in den Speicher.

LOA MAP.COM lädt die COM-Datei MAP.COM in den Speicher.

LOA MAP.COM 600 6 lädt die ersten 6 Bytes von MAP.COM nach \$600-\$605.

EXE Datei Datei

EXE dient zum Laden und Starten von Batchdateien (CMD-Dateien, siehe unten) oder Dateien bzw. Maschinenprogrammen im DOS Binary Load-Format (COM-Dateien).

Wenn der Dateiname keinen DUP-Befehl bildet, genügt es, nur den Dateinamen (ohne EXE davor) einzugeben. Auch einen evtl. vorhandenen COM- oder CMD-Extender kann man dann weglassen. Außerdem wird dann die Datei an der ersten Startadresse (3. und 4. Byte der Datei) gestartet, wenn die Datei keine RUN-Adresse enthält.

Beim Booten startet XDOS automatisch AUTORUN.SYS – es sei denn, es wird beim Booten die ESC-Taste gedrückt.

Beispiel: Mit EXE MAP.COM oder MAP.COM oder einfach nur MAP wird das Programm MAP.COM gestartet.

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 Kasette 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.

TYP Datei

TYP listet die angegebene Datei seitenweise (genau wie bei DIR) auf den Bildschirm.

Beispiele: TYP HELP.CMD zeigt die Batchdatei HELP.CMD an. TYP /D/A entspricht DIR *.* /A.

BON BOF NEW

Schaltet das BASIC-ROM der XL- und XE-Rechner ein (BON) bzw. aus (BOF). Dabei wird ein eingestecktes Switch-Cartridge (wie ACTION! oder MAC/65) automatisch deaktiviert sowie die CAR-Adresse und die RUN-Adresse auf 0 gesetzt. Mit der Batchdatei CON.CMD kann ein Switch-Cartridge jederzeit wieder eingeschaltet werden. NEW entspricht BOF, zusätzlich werden aber alle resetfesten Programme beim nächsten RESET deaktiviert.

CAR (BASIC-Zeile) ;(BASIC-Zeile) RESET

Hiermit wird zum BASIC bzw. eingesteckten Cartridge gesprungen. Ist die CAR-Adresse größer oder gleich \$100, wird stattdessen die CAR-Adresse aufgerufen (siehe unter <Car> im XDOS-Titel). Die CAR-Adresse kann mit dem DUP-Befehl =70C adr gesetzt werden (siehe auch XDOS-Konfiguration, XCAR).

Wird zum BASIC oder Turbo-BASIC gegangen, kann hinter CAR auch eine BASIC-Zeile stehen, die dann vom BASIC ausgeführt wird. Statt CAR kann man auch ; eingeben oder einfach RESET drücken.

Ist das BASIC nicht eingeschaltet und kein Cartridge aktiv, wird lediglich der XDOS-Titel neu angezeigt.

Beispiel: CAR RUN"D:PROG.BAS" geht zum BASIC und startet das BASIC-Programm D:PROG.BAS.

DOS CLS AUT

DOS startet das DUP neu – genau wie der BASIC-Befehl „DOS“. CLS baut den Textbildschirm neu auf, z.B. kann man mit den DUP-Befehlen =6A 80 und CLS den Bildschirm unterhalb der Adresse \$8000 neu aufbauen.

AUT lädt und startet D1:AUTORUN.SYS – genau wie beim Booten. Vorher das BASIC ein- bzw. ausschalten! Genaugenommen ist AUT eine Abkürzung für die drei DUP-Befehle 1; EXE AUTORUN.SYS CAR

TES (Laufwerk;)

Testet die Dichte der eingelegten Disk und stellt das Laufwerk auf die richtige Dichte ein. Dieser Befehl ist nur in Zusammenhang mit XF551-Laufwerken interessant (siehe „Die XF551“ unten).

INI (Laufwerk;)

Schreibt XDOS inklusive Konfiguration bootfähig als DOS.SYS auf Diskette. Z.B. schreibt „INI 2;“ XDOS auf D2:.

FS# (Laufwerk;) FM# (Laufwerk;) FD# (Laufwerk;)

Formatiert Disketten. FS# formatiert in einfacher Dichte, FM# in mittlerer Dichte und FD# in doppelter Dichte. Für FM# kann man konfigurieren, ob DOS 2.5-kompatibel (Standard) oder MyDOS-kompatibel formatiert werden soll (siehe XDOS-Konfiguration, XOPT). Das MyDOS-Format wird übrigens auch von DOS II+/D 6.x benutzt.

ACHTUNG: Es wird ohne Rückfrage sofort formatiert. (Damit nichts versehentlich passiert, enden alle „gefährlichen“ DUP-Befehle mit # und Eingabefehler werden mit Error 182 quittiert.)

F2# (Laufwerk;)

Mit F2# kann eine Disk im XF551-Laufwerk zweiseitig formatiert werden: Jede Seite hat 707 freie Sektoren in doppelter Dichte und ein eigenes Directory. Die Vorderseite einer solchen Disk 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 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 80 wurde ein fehlerhafter Parameter eingegeben, z.B. RUN G474 oder CL# R.

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, 4, 0, "Dn:Dateiname" oder XIO 40, #k, 4, 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 39/40 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, 4, 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 80, #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 80, #1, 0, 0, "D2:DIR" wird z.B. das Directory von D2: angezeigt.

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. 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 etwas 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)
- 8C Zweite Single Density RAM-Disk (zusätzlich zur RAM-Disk „80“)
- C0 Medium Density (128 KB)
- 90 Double Density (180 KB)
- D0 High Density (256 KB)
- +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./Rambo: 83 A3 C3 E3 (256 KB)
Buchholz/ABBUC: A3 C3 E3 E3 (192 KB)	Compy-Shop/MegaRam: 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 dahinterstehenden (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.

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. Folgende Adressen befinden sich 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 per USR 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. XIO 40 steht allerdings bei DOS 2.5 und TurboDOS 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 2.43 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 39/40 laden COM-Dateien (wie bei SpartaDOS und MyDOS) und XIO 80 führt einen DUP-Befehl aus.
- EXE (und XIO 39/40) starten COM-Dateien ohne RUN-Adresse nicht mehr an der 1. Startadresse (wie bei DOS 2.5).
- Nach Ausführen einer Batchdatei mit XIO 39/40/80 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 von 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

Wichtige Fehlercodes

128: Break-Taste gedrückt
 130: Gerätekürzel 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)

DUP-Befehle

Zahl: hex 'char .dezimal (Byte oder Adresse)
 Datei: Dateiname Dn:Dateiname n:Dateiname
 Gerät: Datei oder Gerätekürzel (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

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