



Das Programm wurde auf Richtigkeit hin überprüft. Für eventuelle Fehler im Programm, im Text, in den Beispielen oder entstehende Schäden durch eine Benutzung des Programms kann keine Haftung übernommen werden.

Das Programm und die Beschreibung können jederzeit ohne Bekanntgabe geändert werden.

Herstellung und Vertrieb: sds computer Service
Mainzer Straße 47
5568 Daun - Eifel

06592 / 1646

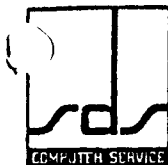
Alle Rechte vorbehalten.

Jede Art der Vervielfältigung, auch des Programms, oder Speicherung auf Datenträgern ist untersagt.

Copyright (c) 1986 by :

Frank Lauter
Gustav von Veit Str. 12
5300 Bonn 1

Seite 5



Start des FLDOS's

Starten des FLDOS's: Legen Sie die FLDOS Master - Diskette in Laufwerk eins ein. Im IPL - Monitor - Menue 'F' drücken. Jetzt besteht die Möglichkeit, schon beim Laden des Betriebssystems einige Sonderfunktionen aufzurufen.

1. **Retten der PCG:** Durch Festhalten der Taste 'A' wird die 700er PCG nach 2000H verschoben.
2. **Start Diskedit:** Durch Festhalten der Taste 'E' wird direkt Diskedit gestartet.
3. **Start Bootloader:** Durch Festhalten der Taste 'F' wird zuerst der Bootloader gestartet und nicht das FLDOS. D.h. das Directory (nur 9. Sector) wird ausgegeben und durch die Tasten 1-8 können die acht Programme, falls vorhanden, direkt gestartet werden. Diese Funktion kann umgekehrt werden. Anstatt COH muss C&H im absoluten Sector 29H im Byte 83H eingetragen werden. (Diskedit)

Nach dem Booten des FLDOS's meldet sich das Betriebssystem mit dem blinkendem Wort 'EINGABE' und einer Auswahl von Unterprogrammen oder Funktionen (Commands:). Der erste Buchstabe ist jeweils invertiert dargestellt und reicht bereits als Eingabe um den jeweiligen Befehl auszuführen. Spätestens an dieser Stelle wird der erste Vorteil sichtbar. Anstatt wie im Basic 'DIR'+CR einzugeben, reicht das Drücken der Taste 'D'.

Dir / L-load - S-save File / R-read - W-rite Sector(s) / Q-uite

Load: Durch Drücken von 'L' wird die EINGABE - Zeile gelöscht und das Betriebssystem wartet mit 'Filename:' und Unterstrichcursor auf den Namen des zu ladenden Programms. Wird dies mit CR beantwortet, erscheint das Directory der Diskette. Sollte kein Programm mit dem eingegebenen Namen auf der Diskette sein, geschieht dies ebenfalls. Nur bei Vorhandensein des Programms oder bei Abbruch durch 'ESC' ist ein Verlassen dieses Eingabemodus möglich. Ist das Programm vorhanden, erscheint die nächste Kommando - Ebene und die File - Daten (Name / Ladeadresse / Länge / Autostart).



CR = Accept Data / W-rong file / M-odify Data / Q-uit

Durch Betätigen der Taste 'CR' wird das Programm geladen und gestartet. 'W' erlaubt die Eingabe eines anderen Programmnamens. Durch 'M' läßt sich die Ladeadresse verändern (Innerhalb gewisser Schranken siehe Anhang) und das Starten wird unterdrückt. 'Q' springt in die oberste Kommando - Ebene.

Save: Durch Drücken von 'S' wird die EINGABE Zeile gelöscht und das Betriebssystem wartet mit 'Filename:' und Unterstrichcursor auf den Namen des zu sichernden Programms. Wird dies mit CR beantwortet, erscheint das Directory der Diskette. Sollte schon ein Programm mit dem eingegebenen Namen auf der Diskette sein, wird der Fehler ausgegeben und man befindet sich in der obersten Kommando - Ebene. Nur wenn ein Programmname eingegeben wird der noch nicht auf Diskette gespeichert ist oder bei Abbruch durch 'ESC' ist ein Verlassen dieses Eingabemodus möglich. Ist der Programmname nicht vorhanden, erscheint eine Folge von Abfragen.

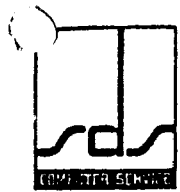
Achtung: Alle Eingaben werden Hexadezimal interpretiert.

1. RAM Adresse: Geben Sie die Anfangsadresse ein.
2. Länge: Geben Sie die Länge des Programms ein.
3. Autostart: Geben Sie die Startadresse ihres Programms ein.

Nachdem das Programm auf Diskette geschrieben ist und keine Fehler aufgetreten sind, erscheint die Meldung 'File saved' und man befindet sich wieder in der oberen Kommando - Ebene

Erase: Durch drücken von 'E' wird die EINGABE Zeile gelöscht und das Betriebssystem wartet mit 'Filename:' und Unterstrichcursor auf den Namen des zu löschenden Programms. Wird dies mit CR beantwortet, erscheint das Directory der Diskette. Sollte kein Programm mit dem eingegebenen Namen auf der Diskette sein, geschieht dies ebenfalls. Nur bei Vorhandensein des Programms oder bei Abbruch durch 'ESC' ist ein Verlassen dieses Eingabemodus möglich. Ist das Programm vorhanden wird es aus dem Directory gelöscht und es erscheint die Meldung 'File Erased'.

TIP: Sollten Sie ein falsches Programm gelöscht haben und sonst noch keinen schreibenden Zugriff auf die Diskette vorgenommen haben, können Sie durch folgende Schritte das Programm wieder 'retten'.



1. Diskedit aufrufen.
2. Im Sector 10H ff. steht das Directory in dem das erste Byte vor jedem Programmnamen angibt, ob jenes Programm noch auf Diskette ist. Nachdem Sie vor dem Namen 01 eingetragen haben, können Sie es mit File I/O wieder laden.
3. Mit Load das Programm jetzt in den Speicher ab 1200H laden (Modify Data 1200).
4. Jetzt mit Save Programm wieder auf Diskette schreiben. (RAM Adresse = 1200)
5. Mit Diskedit 01 vor dem alten Namen wieder in 00 ändern. Eventuell Ladeadresse (Byte 24,23) auf Originalwert ändern.

Nur das exakte Ausführen dieser Schritte garantiert die fehlerfreie Funktion der Diskette.

Read / Write Mit diesen Unterprogrammen lassen sich Bereiche (Sektoren) von der Diskette in den Speicher laden bzw. aus dem Speicher auf die Diskette schreiben. (kein Eintrag ins Directory)

Quit Verlassen dieser Kommando - Ebene zur Obersten.

Oberste Kommando - Ebene:

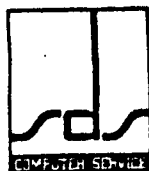
F-file IO File input und output Sprung in oben beschriebene Kommando - Ebene.

D-rive Setzt Zugriff auf Laufwerk 1-4. Die Umschaltung von File IO nach Edit und umgekehrt erfolgt immer von Laufwerk eins.

E-dit Schaltet um auf den Disk-edit Teil des FLDOS's. (Master - Diskette oder Diskette mit FLDOS muß in Laufwerk eins sein).

B-oot Springt in den Monitor und bootet Diskette.

M-onitor Startet den Monitor. (700 Modus wird initialisiert).



Diskedit:

Read Nach dem Umschalten von File IO nach Edit befindet man sich in der Read Kommando - Ebene.

Absolute Sector / F-file begin / Q-uit

Durch Drücken von 'Q' wird die oberste Kommando - Ebene angesprungen. Diese ist gleich der obersten in File IO mit dem einzigen Unterschied, daß von hier aus File IO nachgeladen wird und nicht Edit.

Absoluter Sector Ein Sector mit dem eingegebenen Hexwert wird geladen und als Hex - Ascii - Dump dargestellt.

File begin Durch Drücken von 'F' wird die EINGABE Zeile gelöscht und das Betriebssystem wartet mit 'Filename:' und Unterstrichcursor auf den Namen des zu bearbeitenden Programms. Wird dies mit CR beantwortet, erscheint das Directory der Diskette. Sollte kein Programm mit dem eingegebenen Namen auf der Diskette sein, geschieht dies ebenfalls. Nur bei Vorhandensein des Programms oder bei Abbruch durch 'ESC' ist ein Verlassen dieses Eingabemodus möglich. Ist das Programm vorhanden, erscheint die nächste Kommando - Ebene und der erste Sector des Programms wird geladen und angezeigt im Hex - Ascii Format. Mit dem Zusatz, daß die Adresse im File angezeigt wird, d.h. falls das Programm sich nicht mehr verschiebt, läßt sich so die Stelle im RAM errechnen an der der betreffende Sector später stehen wird. Der Offset im Sector läßt sich leicht durch die Bezifferung vor jeder Zeile ablesen.

Nach diesen beiden Möglichkeiten der Eingabe erscheint das eigentliche Edit - Menue.

N-ext / L-ast / M-odify / U-pdate / R-ead / Z-eichensatz 2 / Q-uit

Next / Last Zeigen den nächsten bzw. den Sector vor dem Momentanen.

Modify Verändern des Sectors. Eingabemöglichkeit in Hex und ASCII Format. (wird mit INST und DEL umgeschaltet). Siehe Kommando - Zeile.



Update

Vor dem Drücken von 'U' und somit Verändern des Sectors auf der Diskette, sollten Sie sich im klaren sein, daß Sie bei einer falschen Veränderung z.B. ein oder mehrere Programme zerstören können !!

Zeichensatz

Schaltet zwischen 1. und 2. Zeichensatz um.

Fehlermöglichkeiten:

Um das Betriebssystem möglichst kurz zu halten, wurde auf einige programmtechnische Mittel verzichtet.

1. Es gibt keinen Pseudo - Bildspeicher.
2. Da das FLDOS ohne Scrollen auskommt, wurde auf eine Scrollroutine und eine Routine die testet ob der Cursor außerhalb des Bildschirms steht, verzichtet.
3. Um den Basicinterpreter in das Directory eintragen zu können, benutzt das FLDOS 9 Sektoren als Directory gegenüber dem Basic das nur 8 benutzt. Daher kann es passieren das der Basicinterpreter ein Programm mit einem Namen zuläßt das schon im 9. Sector steht. (Dieser läßt sich im Zweifelsfalle mit Diskedit ändern).
4. Die Lade - Run - Routine ruft zum verschieben die im IPL befindliche Routine auf. Daher ist ein verschieben auf eine Adresse kleiner als 1200H nur möglich, wenn Start und Ladeadresse gleich null sind. Sollten Sie die Ladeadresse mit Modify verändert wollen ist darauf zu achten, das das FLDOS dabei nicht zerstört wird.



Installation FLDOS:

Um das FLDOS von der gelieferten Master - Diskette auf eine Basic Diskette zu überspielen sind folgende Schritte notwendig.

1. Die Filedaten des Basicinterpreters aus dem Absoluten Sector 0H in den Absoluten Sector 18H Übertragen.

Z. B.

Absoluter Sector 0:

```
00: 03 49 50 4C 50 52 4F 4D 5A 38 30 30 44 2D 42 42
10: 53 49 43 0D 00 B0 00 00 00 00 00 00 00 FF 30 00
```

Die unterstrichenen Bytes müssen an folgende Stelle im Directory stehen.

Absoluter Sector 18H

```
00: 01 42 41 53 49 43 0D 00 00 00 00 00 00 00 00 00
10: 00 00 00 00 00 B0 00 00 00 00 00 00 00 00 30 00
```

Hierbei bedeutet 00 B0 Die Länge des Basicinterpreters B000H oder 450% Dezimal.

FF bedeutet Master - Diskette.

30 00 Bedeutet Absoluter Sector 0030H.
(Startsector des Basicinterpreters)

2. Ab Sector 0000H F00H Bytes Übertragen.
Ab Sector 0019H ~~200H~~ Bytes von der Master - Diskette auf die Basic Diskette Übertragen. *16FF H*

Weiterhin besteht die Möglichkeit außer dem Basic weitere sieben Programme in den Absoluten Sector 18H einzutragen. Diese Programme können dann vom Bootloader des FLDOS's geladen und gestartet werden.

Sollte Ihnen die ausgereifte Farbgebung nicht zusagen oder auf Ihrem Bildschirm nich zu lesen sein, kann Ihnen geholfen werden.

Die Farbe für den Diskedit - Teil steht im Absoluten Sector 1 Byte 01AH ist die Schreibfarbe (hier 00) Byte 01BH ist die Hintergrund-Farbe (hier 07).

Im File IO - Teil des FLDOS's steht die Farbe im Absoluten Sector 01BH Byte 00H, C1H (siehe oben).



Anhang:

Zusätzlich zu den bisherigen Möglichkeiten bietet das FLDOS noch einige leistungsstarke Routinen für den 'geübten' Maschinen - Sprache Programmierer.

Der Hauptspeicher steht Ihnen von 1200H bis FE00H zur Verfügung.

Folgende Speicherbereiche werden weiterhin benötigt.

Beim umschalten von File I/O nach Edit C000H - C020H (Ladeprogramm). Beim umschalten von Edit nach File I/O oder Booten von FLDOS 1200H - 2300H und von CE00H - CFFFH

Sollten Sie nur auf die direkten Disketten - Routinen Wert legen, ist die erste für Sie nutzbare Adresse 2ADH. (siehe Disk I/O)

Alle Routinen die nicht durch EXT (External) gekennzeichnet sind springen bei Auftreten eines Fehlers nach 053FH (Fehlertext steht ab DE) oder nach 053FH zu FEHLER. (siehe FEHLER - Routine)

053FH FEHLER Bereitet nach Fehlernummer im Accu Text aus dem Monitor auf. Der Fehlertext steht nach dem Aufruf an der durch die Speicherstelle 0543H adressierten Stelle.

```
Beispiel: LD A, Fehlernummer
          LD HL, BUFF
          LD (0543H), HL
          LD HL, EXIT
          LD (0561H), HL
          JP 053FH
EXIT:     LD DE, BUFF
          CALL 03DDH ; MSG
          HALT
BUFF:     DEFS 40, 0
```

Ausgaberroutinen:

EXIT **031FH DISP** Displaymatrix ab Cursorposition
Ausgabe von beliebigen Matrizen
adressiert ab dem DE - Register

Beispiel:

```
ORG 1200H
; Bildschirm initialisieren
; und löschen, Cursor setzen
LD DE, TABELLE ; z.B. für '/'
CALL 031FH
HALT
TABELLE: DEFS 1, 2, 4, 8, 16, 32, 64, 128
```



EXT

0336H PRINT

Ausgabe des Accus als ASCII Zeichen. Ausführung von Kontrollcodes (ASCII < 20H)

1 = SWAP schaltet auf 2. Zeichensatz. bzw. bei erneutem Aufruf auf 1. Zeichensatz.

2 = REVERS schaltet auf Negativschrift bzw. bei erneutem Aufruf auf Normalmodus.

14H= CSRL setzt Cursor ein Zeichen nach links.

16H= CLS löscht Bildschirm im Modus 640 * 200 in einer Farbe

Beispiel: LD A,65 ; 'A'
CALL 0336H
HALT

Gibt ein 'A' auf dem Bildschirm an der momentanen Cursor - Position aus.

Die Register AF, BC, DE, HL bleiben erhalten.

EXT

03DDH MSG Ausgabe von Texten im ASCII Format (wiederholtes Aufrufen von PRINT; Ende bei 0) Ausführung von Kontrollcodes.
Beispiel: ORG 1200H

; Bildschirm initialisieren
; und löschen, Cursor setzen
LD DE, TEXT
CALL 0336H
HALT

TEXT: DEFM 'TEST',1,'TEST',2,'TEST'
DEFM 1,'TEST',2,'TEST'
DEFB 0 ; Textende

Es erscheint das Wort TEST fünfmal auf dem Bildschirm. Zuerst normal dann im 2. Zeichensatz, Revers 2. Zeichensatz, Revers 1. Zeichensatz und wieder normal.

EXT

061CH PHLHEX Schreib das HL - Register als Hexzahl mit führenden Nullen.



EXT **0621H PANEX** Schreibt Accu als Hexzahl mit führenden Nullen.

EXT **0C7EH DEZHL** Wandelt den Wert des HL - Registers in eine Dezimalzahl (DE) in ASCII - Format um
 Beispiel: LD HL, 0D000H
 LD DE, BUFF
 LD B, 0 ; Unterdrückt führende Nullen
 CALL 0C7EH
 LD DE, BUFF
 CALL MSG
 HALT
 DEFS 6

Gib ab der Cursor-Position '53248' aus. BUFF ist ein Speicherbereich der für die Aufbereitung benötigt wird.

EXT **0574H NIBBEL** Wandelt eine im Accu befindliche ASCII - Zahl in den unteren Nibbel um. Das Carry - Flag ist bei Fehler gesetzt.

Bildschirmsteuer - Routinen:

Die Cursor - Position befindet sich in der Speicherstelle (02AFH) dies entspricht 1171H im Monitor. Die Farben in der die Texte ausgegeben werden befinden sich in den Speicherstellen (02C0H) Vordergrund und (02C1H) Hintergrund.

EXT **03F4H CR** Setzt den Cursor eine Zeile tiefer. (Kein SCROLLEN des Bildschirms), d.h. es ist selbst darauf zu achten, daß der Cursor innerhalb des sichtbaren Bereiches bleibt.

EXT **0402H INIT80** Screen initialisieren in 80 Zeichen. Initialisiert den Bildschirmmodus 640*200 in einer Farbe, Paletten, Write-Readformat auf Normalwerte. (Kein CLS) Farbe nach (02C0H) und (02C1H). Löscht Cursor Anzeige Flag.



EXT 0381H REVERS wie PRINT mit Accu = 2

EXT 0431H CURSOR Setzt Cursorunterstrich bzw. löscht ihn je nach Anzeige Flag. (siehe SINIT80).

EXT 0490H INIT Schaltet alles auf RAM und SINIT80

Disketten - Routinen:

EXT DISK I/O Direktes lesen oder schreiben auf Diskette. Es wird nur der Speicher von 0000H bis 02ADH benötigt.

HL - Register = Ladeadresse

DE - Register = Länge

BC - Register = Absoluter Startsector

A - Register = Drivenummer 0-3

0165H WRITE Schreiben auf Diskette.

0162H READ Lesen von Diskette.

Carry - Flag bei Fehler gesetzt.

00E0H MOTORAUS Schaltet den Motor aus und setzt

Restore - Flag zurück.

IDBLOCK

Ist ein Speicherbereich der von den FLDOS - Routinen benutzt wird.

06CCH	1. Länge des Files	z. B. DEFW 0E00H
06CEH	2. Ladeadresse	z. B. DEFW 1200H
06D0H	3. Autostart	z. B. DEFW 1300H
02BEH	4. Drivenummer	z. B. DEFB 0
02B7H	5. Absolute Sectornummer	z. B. DEFW 0030H
02B9H	6. Erster Sector	z. B. DEFW 0123H

080FH SAYE Schreibt File auf Diskette.

Beispiel: Speicherbereich von 1200H-2000H mit Autostart 1300H

LD HL, NAME

LD DE, 1044H ; Speicher für Name

LD B, 0

LOOP1: LD A, (HL)

LD (DE), A

INC HL



```

INC DE
INC B
CP ODH
JR NZ, LOOP1
LD HL, FEHLER ; Eintrag für
                ; eigene Fehler
                ; Routine. Sonst
                ; 053FH bei Fehler
LD (082BH), HL ; Rücksprung bei
                ; Fehler
LD A, 0C9H ; 280 RET
LD (083BH), A
LD A, B ; Länge Filename
CALL 080FH
JR OK
FEHLER: ; Accu = Fehlernummer
        ; Eigene Fehlerroutine

OK: LD HL, FERTIG ; Oder Error
        ; (DE= 'File Saved')
LD (0870H), HL ; Exit eigen
LD HL, 1200H ; Ladeadresse
PUSH HL ; Auf Stack
LD HL, 0E00H ; Länge
PUSH HL ; Auf Stack
LD HL, 1300H ; Autostart
PUSH HL ; Auf Stack
IOR A ; Drive 0
LD (02BEH), A
CALL 0850H ; Kein Return
FERTIG: HALT
NAME: DEFN 'TESTPROG', ODH
    
```

Programm ist auf Diskette im Directory und in der
Sectorbelegungstabelle eingetragen.

OABIN RUNPROG Startet ein Maschinen Programm

```

Beispiel: LD HL, NAME
          LD DE, 1044H ; Speicher für Name
          LD B, 0
LOOP2: LD A, (HL)
        LD (DE), A
        INC HL
        INC DE
        INC B
        CP ODH
        JR NZ, LOOP2
        LD A, B
        LD (02BCH), A ; Länge Name
        CALL 0BF1H ; EXT ENTRY
                    ; Liest Eintrag
                    ; Werte nach
                    ; IDBLOCK
    
```



```

; Drive Nr.
; ABSSEC
LD HL, (02B7H)
PUSH HL ; ABSSEC
LD HL, 1200H ; Ladeadresse
PUSH HL
LD HL, (06CCH); Länge
PUSH HL
PUSH HL ; Autostart
JP 0AB1H

```

Programm wird nach 1200H geladen. Nach dem Laden verschoben (IPL) und für MZ - 700er Programme initialisiert.

0872H CALCLEN Errechnet aus der Anzahl der Bytes die benötigten Sektoren auf Diskette.
 Beispiel: LD HL, Anzahlbytes
 LD (06CCH), HL
 CALL 0872H ; Fehler = Error
 LD HL, (02B4H)
 z. B. . CALL 061CH ; PHLHEX
 HALT

0884H SFLEN 'Search for len' Sucht Lücke auf Diskette.
 Beispiel: LD HL, Anzahlsektoren
 LD (02B4H), HL; siehe CALCLEN
 CALL 0884H ; Fehler = 053FH
 LD HL, (02B9H); Erster zu
 ; belegender
 ; Sector
 HALT

08EDH WFOB 'Write file on disc' Schreibt File nach Daten aus IDBLOCK auf Diskette.

0902H SAYEDIR Trägt Filename und Parameter in das Directory ein. Daten aus IDBLOCK.
 Filename aus 1058H

092AH DELFILE Löscht File aus Directory. Filename ab 1044H.

0944H LOCK Sperrt Sektoren auf Diskette.
0952H ERASE Gibt Sektoren auf Diskette frei.
 Start bei 02B9H
 Anzahl aus 02B4H
 Beispiel: LD HL, 0
 LD (02B9H), HL
 LD A, 30H
 LD (02B4H), A



CALL 0952H

HALT

Gibt ab dem ersten Sector 45 Sektoren frei.
(nicht empfehlenswert !!!)

0BF1H ENTRY Sucht Eintrag im Directory. Daten des
Files stehen nach dem Aufruf in
IDBLOCK.

Eingabe - Routinen:

Bei jeder Eingabe wird die ESC - Taste abgefragt.
Beim Betätigen wird ein Sprung nach ERROR ausge-
führt. DE - Register steht auf 'Stopped by User'.

02C2H GETKY Tastatur wird abgefragt. Accu = Taste.

0465H READKY Tastatur entprellt. Eingabe mit
Cursor Accu = Taste. Rückkehr wenn
Taste gedrückt.

049FH INPLN Eingabe einer Zeile. Abschluss mit
Carriage Return (CR). Die maximale
Eingabelänge muß nach 02BDH gelegt
werden. Die tatsächliche Länge steht
in der Speicherstelle 02B6H. Der Text
ab 1044H.

05D0H ANEX Liest Accu als Hexzahl. Eingabe ab
Cursorposition. Bei Eingabefehlern
werden so viele Stellen, wie in der
Speicherstelle (05F6H) angegeben sind
gelöscht. Ausgang aus der Routine nur
durch richtige Eingabe oder ESC.

0602H HLHEX Gleich wie ANEX nur wird das HL -
Register als Hexzahl gelesen.

EXT 0D8CH PCG Verschiebt den Zeichensatz des 700er
Bildschirms nach Inhalt von (0D97H).
Beispiel: LD HL, 2000H
LD (0D97H), HL
CALL 0D8CH
HALT

Der Wert von (0D97H) muß zwischen 2000H und 8000H



liegen.

OFCH

Tastaturbelegungstabelle