

BFZ-MINI-IDS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1	
		2	
		3	;
		4	;*****
		5	;
		6	;
		7	;
		8	;
		9	;
		10	;
		11	;
		12	;
		13	;
		14	;
		15	;
		16	;
		17	;
		18	;
		19	;
		20	;
		21	;
		22	;
		23	;
		24	;
		25	;
		26	;
		27	;
		28	;
		29	;
		30	;
		31	;
		32	;
		33	;
		34	;*****
		35	;
		36	;* VERSION: 1.4
		37	;* LETZTE AENDERUNG: 09.10.85
		38	;* COPYRIGHT: BFZ ESSEN, ALTENESENER STR. 80/84, 4300 ESSEN 12
		39	;
		40	;*****
		41	;
		42	

```

LOC  OBJ          LINE          SOURCE STATEMENT
                                43 ;***** ADRESSEN DER FLOPPY-CONTROLLER-KARTE *****
                                44 ;
00C0  45 BAS          EQU    0C0H          ;BASIS-ADRESSE DER FIC-KARTE
00C0  46 CMD          EQU    BAS          ;FIC KOMMANDO-REGISTER
00C0  47 STAT         EQU    BAS          ;FIC STATUS-REGISTER
00C1  48 TRK          EQU    BAS+1        ;FIC TRACK-REGISTER
00C2  49 SEC          EQU    BAS+2        ;FIC SEKTOR-REGISTER
00C3  50 DAT          EQU    BAS+3        ;FIC DATEN-REGISTER
00C4  51 PORT         EQU    BAS+4        ;STEUER-PORT: I0 - SELECT0 (LAUFWERK A)
                                ;
                                ;           I1 - SELECT1 (LAUFWERK B)
                                ;           I2 - DENSITY SELECT
                                ;           0 = DOUBLE DENSITY
                                ;           1 = SINGLE DENSITY
                                ;           I3 - SIDE SELECT
                                ;           0 = SEITE 0
                                ;           1 = SEITE 1
00C8  59 STOP         EQU    BAS+8        ;STOP-PORT. EINE AUSGABE AN DIESEN
                                ;PORT HAELT DIE CPU AN
                                60 ;
                                61 ;
                                62 ;*****
0000  63 ; STEP-RATE (GUELTIG FUER FIC-CLOCK=1MHZ, FIC-TEST-ANSCHLUSS AUF H-PEGEL)
                                64 ;
                                65 ; SR-WERT          STEP-RATE IN MS
                                66 ;   0                6
                                67 ;   1                12
                                68 ;   2                20
                                69 ;   3                30
                                70 ;
0000  71 SR           EQU    0            ;FESTLEGUNG DER STEP-RATE
                                72 ;
                                73 ;*****
                                74 ; STEUER-WORTE
                                75 ;
0000  76 CREST        EQU    0000000B+SR   ;RESTORE - STEP-RATE ENTSPRICHT "SR"
                                77 ;
0004  78 CRESTV       EQU    00000100B+SR   ;RESTORE - VERIFY TRACK NUMBER
                                79 ;           - STEP-RATE ENTSPRICHT "SR"
                                80 ;
0050  81 CSTPIN       EQU    01010000B+SR   ;STEP IN - UPDATE TRACK-REGISTER
                                82 ;           - STEP-RATE ENTSPRICHT "SR"
                                83 ;
0014  84 CSEEK        EQU    00010100B+SR   ;SEEK - VERIFY
                                85 ;           - DELAY
                                86 ;           - STEPRATE ENTSPRICHT "SR"
                                87 ;
00F0  88 CWRTRK       EQU    11110000B      ;WRITE TRACK - NO DELAY
                                89 ;
00A0  90 CWSEC        EQU    10100000B      ;WRITE SECTOR - SINGLE RECORD
                                91 ;           - NO SIDE COMPARE
                                92 ;           - NO DELAY
                                93 ;           - IAM NOT DELETED
                                94 ;
0080  95 CRSEC        EQU    10000000B      ;LESE SEKTOR - NO DELAY
                                96 ;           - NO SIDE COMPARE
                                97 ;

```

LOC	OBJ	LINE	SOURCE STATEMENT
		98	;***** FEHLER-MASKEN *****
		99	;
00DC		100	MREST EQU 11011100B ;RESTORE: B7=1 --> NOT READY
		101	; B6=1 --> WRITE PROTECT
		102	; B5
		103	; B4=1 --> SEEK-ERROR
		104	; B3=1 --> CRC-ERROR
		105	; B2=1 --> TRACK 0
		106	; B1
		107	; B0
		108	;
00B4		109	MWRTRK EQU 10000100B ;WRITE TRACK: B7=1 --> NOT READY
		110	; B6
		111	; B5
		112	; B4
		113	; B3
		114	; B2=1 --> LOST DATA
		115	; B1
		116	; B0
		117	;
00DC		118	MWSEC EQU 11011100B ;WRITE SECTOR: B7=1 --> NOT READY
		119	; B6=1 --> WRITE PROTECT
		120	; B5
		121	; B4=1 --> RECORD NOT FOUND
		122	; B3=1 --> CRC ERROR
		123	; B2=1 --> LOST DATA
		124	; B1
		125	; B0
		126	;
009C		127	MRSEC EQU 10011100B ;READ SECTOR: B7=1 --> NOT READY
		128	; B6
		129	; B5
		130	; B4=1 --> RECORD NOT FOUND
		131	; B3=1 --> CRC ERROR
		132	; B2=1 --> LOST DATA
		133	; B1
		134	; B0
		135	;
001B		136	MVERI EQU 00011000B ;VERIFY SECTOR: B7
		137	; B6
		138	; B5
		139	; B4=1 --> RECORD NOT FOUND
		140	; B3=1 --> CRC ERROR
		141	; B2
		142	; B1
		143	; B0
		144	;
00DB		145	MSEEK EQU 11011000B ;SEEK TRACK: B7=1 --> NOT READY
		146	; B6=1 --> WRITE PROTECT
		147	; B5
		148	; B4=1 --> RECORD NOT FOUND
		149	; B3=1 --> CRC ERROR
		150	; B2
		151	; B1
		152	; B0

LOC	OBJ	LINE	SOURCE STATEMENT
		153	***** DEFINITION VON KONSTANTEN *****
		154	;
0003		155	MAXTRY EQU 03H ;MAXIMALE ANZAHL DER VERIFY-VERSUCHE
0007		156	BELL EQU 07H ;ASCII BELL
0008		157	BS EQU 08H ;ASCII BACK-SPACE
000A		158	LF EQU 0AH ;ASCII ZEILEN-VORSCHUB
000D		159	CR EQU 0DH ;ASCII WAGEN-RUECKLAUF
000E		160	IMASK EQU 00001110B ;INTERRUPT-MASKE (RST 5.5 ENABLE)
0020		161	SPACE EQU 20H ;ASCII LEERZEICHEN
007F		162	DEL EQU 7FH ;ASCII DELETE
00C3		163	JUMP EQU 0C3H ;JUMP-OPCODE
00C9		164	RETURN EQU 0C9H ;RETURN-OPCODE
089F		165	PRTOFF EQU 089FH ;PRINTER OFF-FLAG
272E		166	SPS EQU 272EH ;SPS-EINSPRUNG
27E9		167	SRET1 EQU 27E9H ;MARKE IM SPS-PROGRAMM
27F0		168	CMDINP EQU 27F0H ;MARKE IM SPS-PROGRAMM
2841		169	CHROK EQU 2841H ;MARKE IM SPS-PROGRAMM
28BE		170	ZULKMD EQU 28BEH ;MARKE IM SPS-PROGRAMM
3091		171	BASIC EQU 3091H ;BASIC NEU-START
6013		172	BASBUF EQU 6013H ;BASIC-INPUT-BUFFER
6064		173	TXTUNF EQU 6064H ;ZEIGER AUF BASIC-PGM-ENDE + EINS
606F		174	TXTBGN EQU 606FH ;BASIC-PROGRAMM-ANFANG
E000		175	BUFFER EQU 0E000H ;TRACK-BUFFER FUER FORMAT
E003		176	PGMEND EQU 0E003H ;ZEIGER AUF SPS-PROGRAMMENDE
E0E0		177	PGMANF EQU 0E0E0H ;SPS-PROGRAMMANFANG
FC84		178	PRTST EQU 0FC84H ;PRINTER STATUS (EIN/AUS)
FC95		179	RSTVEK EQU 0FC95H ;RST 5.5 - VEKTOR
FCC7		180	BCKFLG EQU 0FCC7H ;FLAG FUER MAT85
FCC9		181	GROFLG EQU 0FCC9H ;FLAG FUER MAT85
FCF2		182	M85BE EQU 0FCF2H ;ENDE MAT85-INPUTBUFFER
FD16		183	DIRKOM EQU 0FD16H ;BASIC-DIREKTKOMMANDO-FLAG
FD6F		184	STARTA EQU 0FD6FH ;START-ADR
FD71		185	STOPA EQU 0FD71H ;STOP-ADR
		186	;
		187	***** UNTERPROGRAMME AUS MAT85 *****
		188	;
0040		189	KMD EQU 0040H ;MAT85-"KMD"-ROUTINE
0043		190	RCHAR EQU 0043H ;LESE ZEICHEN IN AKKU
0052		191	WCHAR EQU 0052H ;PRINT ZEICHEN IN AKKU
005B		192	PHL EQU 005BH ;PRINT HL-INHALT ALS HEX-ZAHL
005E		193	WBIN EQU 005EH ;PRINT AKKU-INHALT BINAEER
0061		194	WDEZ EQU 0061H ;PRINT AKKU-INHALT DEZIMAL
006D		195	PTXT EQU 006DH ;DRUCKE TEXT
0073		196	PTXTCR EQU 0073H ;DRUCKE CR,LF. DANN WIE PTXT
01EB		197	EXEC EQU 01EBH ;KOMMANDO-AUSFUEHRUNG
0228		198	FSTAR EQU 0228H ;PRINT "***"+TEXT+"**"
03B8		199	FKLIST EQU 03B8H ;PRINT KOMMANDO-LISTE
0A74		200	HSTART EQU 0A74H ;HOLE START-ADR
0B93		201	WBLNKI EQU 0B93H ;GEBE X MAL " " AUS (X FOLGT DEM CALL)
0CAA		202	BUFCLR EQU 0CAAH ;CLEAR MAT85-INPUT-BUFFER
0D0A		203	BREAD EQU 0D0AH ;LESE TEXT AUS BUFFER
0EB0		204	TEST EQU 0EB0H ;PRUEFE OB ZEICHEN IN TABELLE
0EE9		205	GROSS EQU 0EE9H ;WANDLE KLEIN- IN GROSS-BUCHSTABEN
1039		206	SUB2 EQU 1039H ;HL=HL-IE
		207	;

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		208	;***** UNTERPROGRAMME AUS SP1 *****
		209	;
215B		210	HSTSPA EQU 215BH ;HOLE START/STOP-ADR
31F4		211	R4 EQU 31F4H ;VERGLEICHE HL-DE
31FA		212	CMPDH EQU 31FAH ;VERGLEICHE HL-DE
320B		213	R5 EQU 320BH ;SUCHE NAECHSTES ZEICHEN UNGLEICH " "
348C		214	CLEAR EQU 348CH ;LOESCHE BASIC-PROGRAMM UND VARIABLE
3FD6		215	BCLEAR EQU 3FD6H ;BILDSCHIRM LOESCHEN
		216	;
		217	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

```

LOC OBJ          LINE          SOURCE STATEMENT
4000              218
                219              ORG      4000H          ;*****
                220              ;* <--- PROGRAMM-ANFANG *
                221              ;*****
                222 ;DAS BFZ-MINI-DOS KANN VON MAT85, SPS UND BASIC AUS AUFGERUFEN WERDEN.
                223 ;BEI UNBEKANNTEN BEFEHLEN (WIE Z.B. "F" FUER FLOPPY) PRUEFEN DIESE
                224 ;PROGRAMME, OB EINE ERWEITERUNG VORLIEGT.
                225 ;
                226 ;EIN BEISPIEL: GIBT MAN BEI SPS DEN BUCHSTABEN "F" ALS KOMMANDO EIN,
                227 ;          SO PRUEFT DAS SPS-PROGRAMM, OB IN DER SPEICHERZEILE
                228 ;          4003H DER WERT C3H STEHT (DIE ADRESSE 4003H IST IM
                229 ;          SPS-PROGRAMM FEST VORGEGEREN). C3H IST DER CODE FUER
                230 ;          DEN SPRUNG-BEFEHL "JMP". FINDET SPS DIESEN CODE, SO
                231 ;          VERZWEIGT ES ZU DER ADRESSE 4003H UND FUERT DEN DORT
                232 ;          STEHENDEN SPRUNG-BEFEHL AUS. DIE PROGRAMME MAT85 UND
                233 ;          BASIC PRUEFEN DIE SPEICHERZEILEN 4000H BZW. 4006H.
                234 ;
4000 C36A40      235 VEKT:      JMP      FMAT          ;WIRD FUER MAT85/MAT85+ BENDETIGT
4003 C3B940      236              JMP      FSPS          ;WIRD FUER SPS BENDETIGT
4006 C34F41      237              JMP      FBAS          ;WIRD FUER BASIC BENDETIGT
                238 ;
                239 ;=====
                240 ;DAS BFZ-MINI-DOS ENTHAELT ZWEI UNTERPROGRAMME, DIE RELATIVE SPRUNGE UND
                241 ;RELATIVE UNTERPROGRAMM-AUFRUFE ERMOEGLICHEN. PROGRAMME, DIE NUR RELATIVE
                242 ;VERZWEIGUNGEN ENTHALTEN, SIND IN JEDEM SPEICHERBEREICH LAUFFAEHIG.
                243 ;DIESE UNTERPROGRAMME "RELJMP" UND "RELCAL" VERAENDERN DEN INHALT DES
                244 ;HL-REGISTERPAARES.
                245 ;
                246 ;ANWENDUNGS-BEISPIELE:
                247 ;
                248 ;RELATIVER UNTERPROGRAMM-AUFRUF      |      RELATIVER SPRUNG
                249 ;          "          "                |      "          "
                250 ;          "          "                |      "          "
                251 ;          LXI B,NACH-VON              |      LXI B,NACH-VON
                252 ;          CALL RELCAL                  |      CALL RELJMP
                253 ;VON:          "          "              |      VON:          "          "
                254 ;          "          "                |      "          "
                255 ;NACH: ;UNTERPROGRAMM-ANFANG          |      NACH: ;SPRUNG-ZIEL
                256 ;
                257 ;SIND DEM ASSEMBLER DIE WERTE "VON" UND "NACH" NICHT BEKANNT, MUSS
                258 ;DIE SUBTRAKTION VOM PROGRAMMIERER DURCHGEFUEHRT WERDEN.
                259 ;
4009 C36440      260 RELCAL:      JMP      RCAL          ;RELATIVER UNTERPROGRAMM-AUFRUF
400C C36740      261 RELJMP:      JMP      RJMP          ;RELATIVER SPRUNG
                262 ;
                263 ;

```

LOC	OBJ	LINE	SOURCE STATEMENT
		264	;=====
		265	;ZUM AUFRUF VON UNTERPROGRAMMEN NUTZT MAN IM ALLGEMEINEN DEN CALL-BEFEHL
		266	;IN DER FORM "CALL 1234". UNTERPROGRAMME IM BFZ-MINI-DOS KOENNEN EBENFALLS
		267	;AUF DIESE ART AUFGERUFEN WERDEN. IN EVENTL. SPAETEREN BFZ-MINI-DOS-VERSIONEN
		268	;KOENNEN DIESE UNTERPROGRAMME ABER EVENTUELL IN ANDEREN ADRESSBEREICHEN LIEGEN.
		269	;DIE CALL-BEFEHLE VERZWEIGEN DANN NICHT MEHR ZUM RICHTIGEN UNTERPROGRAMM.
		270	;
		271	;ABHILFE KANN MAN DURCH DIE VERWENDUNG VON "FUNKTIONS-CODES" SCHAFFEN:
		272	;HIERBEI WIRD NUR NOCH EINE EINZIGE UEBERGEORDNETE ROUTINE AUFGERUFEN.
		273	;UEBER EINEN CODE (DEN FUNKTIONS-CODE) IM C-REGISTER DER CPU GIBT MAN
		274	;DABEI AN, WELCHES UNTERPROGRAMM ABGEARBEITET WERDEN SOLL. DIE EINSPRUNG-
		275	;ADRESSE DER UEBERGEORDNETEN ROUTINE MUSS DABEI NATUERLICH IN SAEMTLICHEN
		276	;VERSIONEN DES BFZ-MINI-DOS BEIBEHALTEN WERDEN. DIE EINSPRUNG-ADRESSEN DER
		277	;ANDEREN UNTERPROGRAMME KOENNEN SICH ABER AENDERN.
		278	;
		279	;EIN ANWENDUNGS-REISPIEL:
		280	;
		281	; MVI C,FUNKTIONS-CODE ; FUNKTIONSCODE IN DAS C-REGISTER LADEN
		282	; CALL ENTRY ; UEBERGEORDNETES PROGRAMM AUFRUFEN
		283	;
		284	;DIE HEXADEZIMALEN FUNKTIONS-CODES FINDEN SIE IN DEM KOMMENTAR ZUR VEKTOR-
		285	;TABELLE "VTAB" (S. U.). DIE FUNKTION DER EINZELNEN UNTERPROGRAMME ENTNEHMEN
		286	;SIE BITTE DEREN BESCHREIBUNG IM PROGRAMM-LISTING.
		287	;
400F	E5	288	ENTRY: PUSH H ;RETTE HL-REGISTERPAAR
4010	D5	289	PUSH D ;RETTE DE-REGISTERPAAR
4011	F5	290	PUSH PSW ;RETTE AKKU UND FLAGS
		291	;
		292	;DAS BFZ-MINI-DOS KANN ERWEITERT WERDEN. DIESE ERWEITERUNGEN
		293	;KOENNEN EIGENE VEKTOR-TABELLEN "VTAB" BESITZEN. DAMIT DAS BFZ-
		294	;MINI-DOS DIESE NEUEN TABELLEN VERWENDET, MUSS DIE ERWEITERUNG
		295	;EINEN BESTIMMTEN CODE ENTHALTEN.
		296	;SIE MUSS ENTWEIDER IN DER SPEICHERZEILE 500CH (ERWEITERUNGS-
		297	;STUFE 1) ODER IN DER SPEICHERZEILE 580CH (ERWEITERUNGSSTUFE 2)
		298	;DEN WERT EDH (ERWEITERTES DOS) ENTHALTEN. FINDET DAS BFZ-MINI-
		299	;DOS DIESEN WERT, SO VERWENDET ES DIE NEUE TABELLE. DIE TABELLE
		300	;DER STUFE 2 IST DER STUFE 1 UEBERGEORDNET. DIE ERWEITERUNG MUSS
		301	;DIE ANFANGS-ADRESSEN DER TABELLEN IN BESTIMMTEN SPEICHERZEILEN
		302	;BEREITHALTEN:
		303	;
		304	; ERW.-STUFE EDH-CODE IN TABELLEN-ADR IN
		305	; 1 500C 500H , 500E
		306	; 2 580C 580H , 580E
		307	;
		308	;DIE NEUEN TABELLEN MUESSEN DEN GLEICHEN AUFBAU BESITZEN, WIE
		309	;DIE TABELLE "VTAB".
		310	

LOC	OBJ	LINE	SOURCE STATEMENT	
4012	2A0D58	311	LHLI 580IH	;LADE ADR DER VEKTOR-TABELLE (ERWEIT. 2)
4015	3A0C58	312	LDA 580CH	;* LIEGT ERWEITERUNG VOR ?
4018	FEED	313	CPI 0EIH	;*
401A	CA2B40	314	JZ TABOK	;JA --> TABOK
401D	2A0D50	315	LHLI 500IH	;LADE ADR DER VEKTOR-TABELLE (ERWEIT. 1)
4020	3A0C50	316	LDA 500CH	;* LIEGT ERWEITERUNG VOR ?
4023	FEED	317	CPI 0EIH	;*
4025	CA2B40	318	JZ TABOK	;JA --> TABOK
4028	213F40	319	LXI H,VTAB	;LADE ADR DER VEKTOR-TABELLE (GRUNDVERS.)
		320	;	
402B	7E	321	TABOK: MOV A,M	;A=MAX. FUNKTIONS-CODE
402C	B9	322	CMF C	;VERGLEICHE MIT CODE IN C-REGISTER
402D	DA3140	323	JC VRETO	;SPRINGE, WENN MAX. CODE UEBERSCHRITTEN
4030	79	324	MOV A,C	;FUNKTIONS-CODE NACH A
4031	87	325	ADD A	;FUNKTIONS-CODE MAL ZWEI
4032	1600	326	MVI D,00	;MSB (DE)=00
4034	5F	327	MOV E,A	; (FUNKTIONS-CODE * 2) NACH E
4035	23	328	INX H	;ZEIGER AUF 1. VEKTOR DER TABELLE
4036	19	329	DAD D	;ZEIGER AUF RICHTIGEN VEKTOR
		330	;	
4037	5E	331	MOV E,M	;* VEKTOR NACH HL
4038	23	332	INX H	;*
4039	56	333	MOV D,M	;*
403A	EB	334	XCHG	;*
		335	;	
403B	F1	336	POP PSW	;RESTORE AKKU UND FLAGS
403C	D1	337	POP D	;RESTORE DE-REGISTERPAAR
403D	E3	338	XTHL	;HL=HL.ALT, STACK=VEKTOR
403E	C9	339	VRET: RET	;SPRUNG ZUR AUFGERUFENEN FUNKTION
		340	;	
403F	11	341	VTAB: DB 17	;MAXIMALER FUNKTIONS-CODE PLUS EINS
4040	D648	342	DW RESTORE	;* VEKTOR-TABELLE (CODE 00)
4042	DB48	343	DW STEPIN	;* (CODE 01)
4044	E048	344	DW SEEK	;* C O D I E S (CODE 02)
4046	E548	345	DW WSEC	;* S I N D (CODE 03)
4048	F248	346	DW RSEC	;* H E X A - (CODE 04)
404A	FF48	347	DW WTRK	;* D E Z I M A L (CODE 05)
404C	7549	348	DW VERIX	;* A N G E G E - (CODE 06)
404E	8B49	349	DW SELECT	;* B E N . (CODE 07)
4050	9E49	350	DW DESEL	;* (CODE 08)
4052	AA49	351	DW DREADY	;* (CODE 09)
4054	B449	352	DW DELAY	;* (CODE 0A)
4056	E84D	353	DW REPCHR	;* (CODE 0B)
4058	EF4D	354	DW WAITSP	;* (CODE 0C)
405A	2E4E	355	DW INTINT	;* (CODE 0D)
405C	7E4D	356	DW TSTCHR	;* (CODE 0E)
405E	874D	357	DW LETTER	;* (CODE 0F)
4060	F84D	358	DW TSTRS	;* (CODE 10)
4062	3E40	359	DW VRET	;* (CODE 11)
		360	;	
		361		



LOC	OBJ	LINE	SOURCE STATEMENT
		362	;=====
		363	;DIE ROUTINEN "RCAL" UND "RJMP" WERDEN BEI ANWENDUNG DER RELATIVEN
		364	;VERZWEIGUNGEN (UNTERPROGRAMME "RELCAL" UND "RELJMP") AUFGERUFEN.
		365	;
4064	E1	366	RCAL: POP H ;"VON" NACH HL
4065	E5	367	PUSH H ;RETTE RUECKSPRUNG-ADR
4066	E5	368	PUSH H ;ZUM AUSGLEICH DES FOLGENDEN "POP'S"
4067	E1	369	RJMP: POP H ;"VON" NACH HL
4068	O9	370	DAD B ;ADDIERE OFFSET
4069	E9	371	FCHL ;SPRINGE NACH "NACH"
		372	;
		373	

LOC	OBJ	LINE	SOURCE STATEMENT
		374	;;;;; DAS PROGRAMM VERZWEIGT ZU DIESER STELLE, WENN EIN UNBEKANNTES
		375	;;;;; MAT85-KOMMANDO EINGEGEBEN WURDE.
		376	;;;;; .....
		377	;DAS BFZ-MINI-DOS KANN ERWEITERT WERDEN. DIESE ERWEITERUNG KANN IN DER
		378	;SPEICHERZEILE 5000H (ERWEITERUNGSSTUFE 1) ODER 5800H (ERWEITERUNGS-
		379	;STUFE 2) DEN WERT EIH ENTHALTEN. DAS PROGRAMM PRUEFT IM UNTERPROGRAMM
		380	;"UMG", OB DIESER WERT VORLIEGT. DABEI IST DIE STUFE 2 DER STUFE 1 UEBER-
		381	;GEORDNET. WIRD EIH GEFUNDEN, SO VERZWEIGT DAS UNTERPROGRAMM "UMG". DIE
		382	;RUECKSPRUNG-ADRESSE BLEIBT DABEI IM STACK!
		383	;
		384	; ERW.-STUFE EDH GEFUNDEN BEI "UMG" VERZWEIGT NACH
		385	; 1 5000 5001
		386	; 2 5800 5801
		387	;
		388	;BEI 5001 BZW. 5801 MUSS EIN SPRUNGBEFEHL STEHEN!
		389	;
406A	210058	390	FMAT: LXI H,5800H ;ADRESSE DES 2. UMGEHUNGS-FLAGS
406D	CD1E4E	391	CALL UMG ;SPRUNGE, WENN UMGEHUNGS-FLAG GESETZT
		392	;
4070	79	393	FX: MOV A,C ;EINGABE-ZEICHEN NACH A
4071	FE46	394	CPI 'F' ;FLOPPY ?
4073	CA7F40	395	JZ FM ;JA --> FM
		396	=====
		397	;HIER, WENN NICHT "F"
		398	;
4076	3A0150	399	LDA 5001H ;* MAT85-ERWEITERUNG AB 5001H
4079	FEC3	400	CPI JUMP ;*
407B	CA0150	401	JZ 5001H ;JA --> 5001H
407E	C9	402	RET ;NEIN --> RETURN (FEHLER)
		403	=====
		404	;HIER, WENN "F"
		405	;
407F	210000	406	FM: LXI H,0000 ;* VORSCHLAGS-ADRESSE FUER
4082	226FFD	407	SHLD STARTA ;* START/STOP AUF
4085	2271FD	408	SHLD STOPA ;* 0000H SETZEN
		409	=====
		410	;RAM-VEKTOREN VERAENDERN
		411	;
4088	3EC3	412	MVI A,JUMP ;* JMP-OPCODE
408A	326BFA	413	STA XSTSP ;* EINSETZEN
408D	327AFA	414	STA XLAD1 ;* EINSETZEN
4090	327DFA	415	STA XLAD2 ;*
4093	217F4A	416	LXI H,GSTSP ;: ADRESSE
4096	226CFA	417	SHLD XSTSP+1 ;: EINSETZEN
4099	21634A	418	LXI H,GSTART ;:
409C	227BFA	419	SHLD XLAD1+1 ;:
409F	21B240	420	LXI H,MATLAD ;:
40A2	227EFA	421	SHLD XLAD2+1 ;:
40A5	3EC9	422	MVI A,RETURN ;* RET-OPCODE
40A7	326BFA	423	STA XSAV1 ;*
		424	;
40AA	214000	425	LXI H,KMD ;SPRUNGZIEL FUER DOS-"QUIT": KMD
40AD	3E01	426	MVI A,1 ;FLAG: DOS-AUFRUF VON MAT85
		427	;
40AF	C3DF41	428	JMP DOS1

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		429	;
		430	;DIESES UNTERPROGRAMM WIRD VOM DOS BEIM LADEN VON PROGRAMMEN AUFGERUFEN,
		431	;WENN DAS DOS VON MAT85 AUS AUFGERUFEN WURDE.
		432	;
40B2	2A6FFD	433	MATLAD= LHLI STARTA ;* START-ADRESSE
40B5	22D7FC	434	SHLD OFCD7H ;* ALS GO-ADRESSE EINSETZEN
40B8	C9	435	RET
		436	

BFZ-MINI-10S, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		437	;;;;;; DAS PROGRAMM VERZWEIGT ZU DIESER STELLE, WENN EIN UNBEKANNTER
		438	;;;;;; MAT85+ -, SPS- ODER EPROMMER-BEFEHL EINGEGEBEN WURDE.
		439	;;;;;; .....
		440	;DAS BFZ-MINI-10S KANN ERWEITERT WERDEN. DIESE ERWEITERUNG KANN IN DER
		441	;SPEICHERZEILE 5004H (ERWEITERUNGSSTUFE 1) ODER IN DER SPEICHERZEILE
		442	;5804H (ERWEITERUNGSSTUFE 2) DEN WERT EDH ENTHALTEN.
		443	;DAS PROGRAMM PRUEFT IM UNTERPROGRAMM "UMG", OB DIESER WERT VORLIEGT.
		444	;WIRD EDH GEFUNDEN, SO VERZWEIGT DAS UNTERPROGRAMM "UMG". DIE RUECK-
		445	;SPRUNGADRESSE BLEIBT DABEI IM STACK!
		446	;
		447	; ERW.-STUFE EDH GEFUNDEN BEI "UMG" VERZWEIGT NACH
		448	; 1 5004 5005
		449	; 2 5804 5805
		450	;
		451	;BEI 5005 BZW. 5805 MUSS EIN SPRUNGBEFEHL STEHEN!
		452	;
40B9	210458	453	FSPS: LXI H,5804H ;ADR DES 2. UMGEHUNGS-FLAGS
40BC	CD1E4E	454	CALL UMG ;SPRINGE, WENN UMGEHUNGS-FLAG GESETZT
		455	=====
		456	; "FSPS" KANN VON MAT85+, SPS UND DER EPROMMER-SOFTWARE AUS AUFGERUFEN
		457	; WERDEN. ES MUSS FESTGESTELLT WERDEN, OB DER AUFRUF VON SPS AUS ERFOLGTE.
		458	; DIES IST Z.B. MOEGLICH, INDEM MAN PRUEFT, OB "ZULKMD" IM STACK STEHT.
		459	;
40BF	E1	460	POP H ;WERT AUS STACK
40C0	D5	461	PUSH D ;RETTE EINGABE-ZEICHEN (D)
40C1	11BE28	462	LXI D,ZULKMD ;ZEIGER AUF TABELLE DER ZULAESSIGEN
		463	;SPS-EINGABE-ZEICHEN
40C4	CDFA31	464	CALL R4 ;HL=DE ?
40C7	D1	465	POP D ;RESTORE EINGABE-ZEICHEN (D)
40C8	E5	466	PUSH H ;WERT ZURUECK IN DEN STACK
40C9	C2D240	467	JNZ SP1EXP ;HL (<) DE --> SP1EXP
		468	=====
		469	;HIER, WENN DER AUFRUF VON SPS AUS ERFOLGTE
		470	;
40CC	7A	471	MOV A,D ;EINGABEZEICHEN NACH A
40CD	FE46	472	CPI 'F' ;FLOPPY ?
40CF	CADD40	473	JZ SP5DOS ;JA --> SP5DOS
		474	=====
		475	;HIER, WENN DER AUFRUF NICHT VON SPS AUS ERFOLGTE
		476	; ODER WENN NICHT "F" EINGEGEBEN WURDE.
		477	;
40D2	3A0550	478	SP1EXP: LDA 5005H ;* SPS ERWEITERUNG AB 5005H ?
40D5	FEC3	479	CPI JUMP ;*
40D7	C24128	480	JNZ CHROK ;NEIN --> FEHLER (Z-FLAC = 0 !)
40DA	C30550	481	JMP 5005H ;JA --> 5005H
		482	;

```

LOC  OBJ          LINE      SOURCE STATEMENT
483 ;=====
484 ;HIER, WENN DER AUFRUF VON SPS AUS ERFOLGTE
485 ;UND "F" EINGEGEBEN WURDE
486 ;
40DD  CD6100      487 SPSDOS:      CALL  PTXT          ;* PRINT "F"
40E0  46          488          DB      'F',00      ;*
40E1  00
40E2  CD4300      489 SIOS0:      CALL  RCHAR        ;LESE ZEICHEN VON TASTATUR
40E5  FE0D        490          CPI      CR          ;CR ?
40E7  CAF040      491          JZ      SIOS        ;JA --> SIOS
40EA  CDF84D      492          CALL  TSTBS       ;PRINT BS,SPACE,BS WENN BS ODER DEL
40ED  C2E240      493          JNZ   SIOS0       ;WEDER BS NOCH DEL --> SIOS0
494 ;=====
495 ;HIER, WENN BS ODER DEL EINGEGEBEN WURDE
496 ;
40F0  F3          497 SD1:        DI          ;DISABLE INTERRUPT
40F1  E1          498          POP    H          ;HL=ZEIGER AUF ZUL. EINGABE-ZEICHEN (SPS)
40F2  3100FC      499 SD2:        LXI    SP,0FC00H   ;RE-INIT SP
40F5  01E927      500 SD3:        LXI    B,SRET1    ;* RUECKSPRUNG-ADR
40F8  C5          501 SD4:        PUSH   B          ;* IN STACK
40F9  C3F027      502          JMP    CMDINF     ;GEBE NEUES KOMMANDO EIN
503 ;=====
504 ;HIER, WENN DER AUFRUF VON SPS AUS ERFOLGTE
505 ;UND DIE EINGABE "F" MIT <CR> ABGESCHLOSSEN WURDE
506 ;
507 ;DAS PROGRAMM AENDERT NUN RAM-VEKTOREN
508 ;
40FC  3EC9          509 SIOS:        MVI    A,RETURN   ;* RET-OPCODE EINSETZEN
40FE  326BFA      510          STA   XSTSP      ;*
4101  3EC3          511          MVI    A,JUMP     ; : JMP-OPCODE
4103  3268FA      512          STA   XSAV1      ; : EINSETZEN
4106  327AFA      513          STA   XLAD1      ; :
4109  327DFA      514          STA   XLAD2      ; :
410C  213241      515          LXI    H,LENSPS  ;* ADRESSE EINSETZEN
410F  2269FA      516          SHLD  XSAV1+1    ;*
4112  213F41      517          LXI    H,SLAD1   ;*
4115  227BFA      518          SHLD  XLAD1+1    ;*
4118  214841      519          LXI    H,SLAD2   ;*
411B  227EFA      520          SHLD  XLAD2+1    ;*
521 ;
411E  21EDE0      522          LXI    H,PGMANF  ; : SPS-PROGRAMM-ANFANG
4121  226FFD      523          SHLD  STARTA    ; : EINSETZEN
4124  2A03E0      524          LHLD  PGMEND    ;* SPS-PROGRAMM-ENDE
4127  2271FD      525          SHLD  STOPA     ;* EINSETZEN
526 ;
412A  212E27      527          LXI    H,SPS     ;SPRUNGZIEL FUER DOS-"QUIT": SPS
412D  3E02        528          MVI    A,2       ;FLAG: DOS-AUFRUF VON SPS
412F  C3DF41      529          JMP    DOS1
530 ;
531 ;

```

LOC	OBJ	LINE	SOURCE STATEMENT
		532	=====
		533	;DIE FOLGENDEN DREI UNTERPROGRAMME WERDEN VOM BFZ-MINI-IDS AUS AUFGERUFEN
		534	-----
		535	;TEST, OB SPS-PROGRAMMSPEICHER LEER. (BEI "SAVE")
		536	;
4132	2A03E0	537	LENSPS:           LHLI   PGMEND           ;* PROGRAMM-SPEICHER LEER ?
4135	11E1E0	538	LXI    D,PGMANF           ;*
4138	C1F431	539	CALL   R4                ;*
413B	CC7F48	540	CZ     CODE16            ;*JA --> CODE16
413E	C9	541	RET
		542	;
		543	-----
		544	;LOESCHE ALTES SPS-PROGRAMM IM SPEICHER. (BEI "LOAD")
		545	;
413F	21E1E0	546	SLAD1:           LXI    H,PGMANF           ;* LOESCHE ALTES PROGRAMM
4142	36FF	547	MVI   M,OFFH           ;*
4144	2203E0	548	SHLD   PGMEND           ;*
4147	C9	549	RET
		550	;
		551	-----
		552	;UEBERNEHME NEUE SPS-PROGRAMM-STOPADRESSE. (BEI "LOAD")
		553	;
414B	2A71FD	554	SLAD2:           LHLI   STOPA            ;* UEBERTRAGE NEUE STOP-ADR
414B	2203E0	555	SHLD   PGMEND           ;*
414E	C9	556	RET
		557	;
		558	

LOC	OBJ	LINE	SOURCE STATEMENT
		559	;;;;; DAS PROGRAMM VERZWEIGT ZU DIESER STELLE, WENN EIN UNBEKANNTES
		560	;;;;; BASIC-KOMMANDO EINGEGEBEN WURDE.
		561	;;;;; .....
		562	;DAS BFZ-MINI-DOS KANN ERWEITERT WERDEN. DIESE ERWEITERUNG KANN IN DER
		563	;SPEICHERZEILE 5008H (ERWEITERUNGSSTUFE 1) ODER IN DER SPEICHERZEILE
		564	;5808H (ERWEITERUNGSSTUFE 2) DEN WERT EDH ENHALTEN.
		565	;DAS PROGRAMM PRUEFT IM UNTERPROGRAMM "UMG", OB DIESER WERT VORLIEGT.
		566	;DABEI IST DIE STUFE 2 DER STUFE 1 UEBERGEORDNET. WIRD EDH GEFUNDEN,
		567	;SO VERZWEIGT DAS UNTERPROGRAMM "UMG". DIE RUECKSPRUNGADRESSE BLEIBT
		568	;DABEI IM STACK!
		569	;
		570	; ERW.-STUFE        EDH GEFUNDEN BEI        "UMG" VERZWEIGT NACH
		571	;        1                5008                                5009
		572	;        2                5808                                5809
		573	;
		574	;BEI 5009 BZW. 5809 MUSS EIN SPRUNGBEFEHL STEHEN!
		575	;
414F	210858	576	FBAS:                LXI    H,5808H                ;ADRESSE DES 2. UMGEHUNGS-FLAGS
4152	CD1E4E	577	CALL    UMG                        ;SPRINGE, WENN UMGEHUNGS-FLAG GESETZT
		578	=====
		579	;DAS BASIC-KOMMANDO "FLOPPY" IST NUR IM DIREKT-MODUS (NICHT IN EINEM
		580	;PROGRAMM) ZULAESSIG. ES MUSS GEPRUEFT WERDEN, OB DER DIREKT-MODUS
		581	;VORLIEGT.
		582	;
4155	3A16FD	583	LDA    DIRKOM                ;* DIREKT-KOMMANDO ?
4158	B7	584	ORA    A                        ;*
4159	C28041	585	JNZ    BASEXP                ;NEIN --> BASEXP
		586	=====
		587	;DER DIREKT-MODUS LIEGT VOR. WURDE ABER AUCH "FLOPPY" EINGEGEBEN ?
		588	;
415C	211360	589	LXI    H,BASBUF                ;: ZEIGT DE AUF BASIC-INPUT-BUFFER ?
4168	0606	590	CALL    R4                        ;:
4162	C28041	591	JNZ    BASEXP                ;NEIN --> BASEXP
		592	;
4165	218941	593	LXI    H,FLOPPY                ;ZEIGER AUF "FLOPPY" (VERGLEICHSTEXT)
4168	0606	594	MVI    B,6                        ;6 ZEICHEN PRUEFEN
416A	1A	595	FBAS:                LDAX    I                        ;ZEICHEN AUS BUFFER NACH A
416B	BE	596	CMP    M                        ;VERGLEICHE MIT ZEICHEN AUS "FLOPPY"
416C	C27D41	597	JNZ    BASEXO                ;UNGLEICH --> BASEXO
416F	13	598	INX    I                        ;* STELLE ZEIGER WEITER
4170	23	599	INX    H                        ;*
4171	05	600	DCR    B                        ;6 ZEICHEN GEPRUEFT ?
4172	C26A41	601	JNZ    PBAS                        ;NEIN --> PBAS
		602	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		603	=====
		604	;HIER, WENN "FLOPPY" EINGEGEBEN WURDE.
		605	;FOLGEN WEITERE ZEICHEN ?
		606	; (JA --> GILT IN DER JETZIGEN VERSION NICHT ALS BFZ-MINI-DOS-AUFRUF)
		607	; IST DIE EINGABE MIT CR ABGESCHLOSSEN ?
		608	;
4175	CD0K32	609	CALL R5 ; WELCHES ZEICHEN (AUSSER " ") FOLGT ?
4178	FE0D	610	CPI 0DH ; CR ?
417A	CABF41	611	JZ BDOS ; JA --> BDOS
		612	;
417D	111360	613	BASEX0: LXI D,BASBUF ; ZEIGER AUF ANFANG DES INPUT-PUFFERS
4180	3A0950	614	BASEXP: LDA 5009H ; * BASIC ERWEITERUNG AB 5009H ?
4183	FEC3	615	CPI JUMP ; *
4185	C0	616	RNZ ; NEIN --> RETURN (FEHLER)
4186	C30950	617	JMP 5009H ; JA --> 5009H
		618	=====
		619	;VERGLEICHSTEXT:
		620	;
4189	464C4F50	621	FLOPPY: IB 'FLOPPY'
418D	5059		
		622	=====
		623	;HIER, WENN "FLOPPY" OHNE WEITERE ZEICHEN IM DIREKTMODUS EINGEGEBEN WURDE
		624	;
		625	;DAS PROGRAMM VERAENDERT RAM-VEKTOREN
		626	;
418F	3EC9	627	BDOS: MVI A,RETURN ; * RET-OPCODE EINSETZEN
4191	326BFA	628	STA XSTSP ; *
4194	3EC3	629	MVI A,JUMP ; * JMP-OPCODE
4196	326BFA	630	STA XSAV1 ; * EINSETZEN
4199	327AFA	631	STA XLAD1 ; *
419C	327DFA	632	STA XLAD2 ; *
419F	21C641	633	LXI H,CHKLEN ; * ADRESSE EINSETZEN
41A2	2269FA	634	SHLD XSAV1+1 ; *
41A5	21D341	635	LXI H,BLAD1 ; *
41A8	227BFA	636	SHLD XLAD1+1 ; *
41AB	21D741	637	LXI H,BLAD2 ; *
41AE	227EFA	638	SHLD XLAD2+1 ; *
		639	;
41B1	216F60	640	LXI H,XTBGN ; * PROGRAMM-ANFANG
41B4	226FFD	641	SHLD STARTA ; * EINSETZEN
41B7	2A6460	642	LHLD TXTUNF ; * PROGRAMM-ENDE
41BA	2B	643	DCX H ; * EINSETZEN
41BB	2271FD	644	SHLD STOPA ; *
		645	;
41BE	219130	646	LXI H,BASIC ; SPRUNGZIEL FUER DOS-"QUIT": BASIC
41C1	3E03	647	MVI A,3 ; FLAG: DOS-AUFRUF VON BASIC
41C3	C3DF41	648	JMP DOS1 ; RUFE DOS AUF
		649	;
		650	



LOC	OBJ	LINE	SOURCE STATEMENT
		651	=====
		652	;DIE FOLGENDEN DREI UNTERPROGRAMME WERDEN VOM BFZ-MINI-IOS AUFGERUFEN
		653	-----
		654	;TEST, OB BASIC-PROGRAMMSPEICHER LEER (BEI "SAVE")
		655	;
41C6	2A6460	656	CHKLEN: LHLI TXTUNF ;* PROGRAMM-SPEICHER LEER ?
41C9	116F60	657	LXI D,TXTBGN ;*
41CC	CDF431	658	CALL R4 ;*
41CF	CC7F48	659	CZ CODE16 ;JA --> CODE16
41D2	C9	660	RET
		661	;
		662	-----
		663	;LOESCHE ALTES BASIC-PROGRAMM IM SPEICHER (BEI "LOAD")
		664	;
41D3	CD8C34	665	BLAD1: CALL CLEAR ;LOESCHE ALTES PROGRAMM
41D6	C9	666	RET
		667	;
		668	-----
		669	;UEBERNEHME NEUE BASIC-PROGRAMM-STOPADRESSE (BEI "LOAD")
		670	;
41D7	2A71FD	671	BLAD2: LHLI STOPA ;* UEBERTRAGE NEUE STOP-ADR
41DA	23	672	INX H ;*
41DE	226460	673	SHLD TXTUNF ;*
41DE	C9	674	RET
		675	;
		676	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		677 ;	*****
		678 ;	* * *
		679 ;	* DOS-INITIALISIERUNGS-TEIL *
		680 ;	* * *
		681 ;	*****
		682 ;	
41DF	3132FC	683 DOS1:	LXI SP,0FC32H ;(RE)-INIT SP
41E2	CDD63F	684 DOS2:	CALL BCLEAR ;LOESCHE BILDSCHIRM
41E5	3283FA	685 DOS3:	STA EFROM ;RETTE FLAG (ENTERED) FROM)
41E8	225CFA	686 DOS4:	SHLD DOSRET ;RETTE SPRUNGZIEL FUER "QUIT"
		687 ;	
		688	;DAS BFZ-MINI-DOS KANN ERWEITERT WERDEN. DIESE ERWEITERUNG
		689	;KANN EINE EIGENE INFO-TABELLE HABEN (S.U.). DAMIT DAS BFZ-
		690	;MINI-DOS DIE NEUE INFO-TABELLE VERWENDET, MUSS DIE ERWEITERUNG
		691	;ENTWEDER IN DER SPEICHERZEILE 500FH (ERWEITERUNGSSTUFE 1) ODER
		692	;IN DER SPEICHERZEILE 580FH (ERWEITERUNGSSTUFE 2) DEN WERT EDH
		693	;ENTHALTEN. DAS PROGRAMM SUCHT NACH DIESEM WERT. BEI DER SUCHE
		694	;IST STUFE 2 DER STUFE 1 UEBERGEORDNET. WIRD EDH GEFUNDEN, SO
		695	;WIRD DIE INFO-TABELLE AUS DER ERWEITERUNG VERWENDET. DIE INFO-
		696	;TABELLE SELBST MUSS DEN GLEICHEN AUFBAU HABEN WIE DIE TABELLE
		697	;IN DER BFZ-MINI-DOS-GRUNDVERSION. DIE ZWEI SPEICHERZEILEN,
		698	;DIE DEM ED-CODE FOLGEN, MUESSEN DIE ADRESSE DER INFO-TABELLE
		699	;ENTHALTEN:
		700	;
		701	; ERW.-STUFE EDH GEFUNDEN BEI TABELLEN-ADRESSE BEI
		702	; 1 500F 5010 , 5011
		703	; 2 580F 5810 , 5811
		704	;
41EB	2A1058	705	LHLD 5810H ;ADR II. INFO-TABELLE (STUFE 2)
41EE	3A0F58	706	LDA 580FH ;* DOS-ERWEITERUNG AB 580FH ?
41F1	FEED	707	CPI 0E1H ;* (STUFE 2)
41F3	CA0442	708	JZ DOS ;JA --> DOS (TABELLE STUFE 2)
41F6	2A1050	709	LHLD 5010H ;ADR II. INFO-TABELLE (STUFE 1)
41F9	3A0F50	710	LDA 500FH ;* DOS-ERWEITERUNG AB 500FH ?
41FC	FEED	711	CPI 0E1H ;* (STUFE 1)
41FE	CA0442	712	JZ DOS ;JA --> DOS (TABELLE STUFE 1)
4201	21R142	713	LXI H,INFOT ;ADR II. INFO-TABELLE (GRUNDV.)
		714 ;	
		715	

LOC	OBJ	LINE	SOURCE STATEMENT
		716	;=====
		717	;DAS PROGRAMM ENTNIMMT DER INFO-TABELLE NUN MEHRERE ANGABEN
		718	;-----
		719	;1. ZEIGER AUF TYF-TABELLE (TABELLE DER FILE-TYPEN WIE: MAT, SPS, BAS)
		720	;
4204	5E	721	DOS:           MOV    E,M           ;* LADE ZEIGER AUF TYF-TABELLE
4205	23	722	INX    H           ;* NACH HL
4206	56	723	MOV    D,M       ;*
4207	23	724	INX    H           ;*
4208	EB	725	XCHG   H           ;*
4209	225EFA	726	SHLD   PTYPT      ;RETTE ZEIGER AUF TYF-TABELLE
		727	;-----
		728	;2. ZEIGER AUF ERROR-VEKTOR-TABELLE. (WELCHE MELDUNG BEI WELCHEM FEHLER)
		729	;
420C	EB	730	XCHG               ;* LADE ZEIGER AUF ERROR-VEKTOR-TABELLE
420D	5E	731	MOV    E,M       ;* NACH HL
420E	23	732	INX    H           ;*
420F	56	733	MOV    D,M       ;*
4210	23	734	INX    H           ;*
4211	EB	735	XCHG   H           ;*
4212	2260FA	736	SHLD   PEVT       ;RETTE ZEIGER AUF ERROR-VEKTOR-TABELLE
		737	;-----
		738	;3. VERSIONS-NUMMER
		739	;
4215	EB	740	XCHG               ;RETTE ZEIGER AUF VERSIONS-NUMMER
4216	2262FA	741	SHLD   PVNR       ;RETTE ZEIGER AUF VERSIONS-NUMMER
		742	;-----
		743	;4. TABELLE DER ZULAESSIGEN EINGABE-ZEICHEN
		744	;
4219	23	745	INX    H           ;* STELLE ZEIGER AUF TABELLE DER
421A	23	746	INX    H           ;* ZULAESSIGEN EINGABE-ZEICHEN
421B	2264FA	747	SHLD   PTZZ       ;RETTE ZEIGER
		748	;-----
		749	;5. ZEIGER AUF KOMMANDO-TABELLE (WELCHE ROUTINE BEI WELCHER EINGABE)
		750	;
421E	7E	751	DOS5:           MOV    A,M       ;* SUCHE ENDE DER TABELLE
421F	23	752	INX    H           ;* DER ZULAESSIGEN EINGABE-ZEICHEN
4220	B7	753	ORA    A           ;*
4221	C21E42	754	JNZ   DOS5       ;*
4224	2266FA	755	SHLD   PMENUE      ;RETTE ZEIGER AUF KOMMANDO-TABELLE
		756	;=====
		757	;TRITT BEI DER AUSFUEHRUNG EINES KOMMANDOS EIN FEHLER AUF, SO VERZWEIGT
		758	;DAS PROGRAMM NACH "XERR" IM RAM. DORT WIRD AN DIESER STELLE EIN SPRUNG
		759	;NACH "ERROR" EINGESETZT.
		760	;
4227	3EC3	761	MVI    A,JUMP      ;* SETZE "JMP ERROR" EIN
4229	3280FA	762	STA    XERR       ;*
422C	218448	763	LXI    H,ERROR    ;*
422F	2281FA	764	SHLD   XERR+1     ;*
		765	

LOC	OBJ	LINE	SOURCE STATEMENT
		766	;=====
		767	;DAS BFZ-MINI-DOS MELDET SICH NUN
		768	;
4232	CD7300	769	CALL PTXTCR ;PRINT TEXT
4235	0A	770	DB LF,'BFZ-MINI-DOS V',00
4236	42465A2D		
423A	4D494E49		
423E	2D444F53		
4242	2056		
4244	00		
4245	2A62FA	771	LHLD PVNR ;ZEIGER AUF VERSIONS-NUMMER
4248	7E	772	MOV A,M ;* GEBE 1. ZIFFER DER
4249	CD5200	773	CALL WCHAR ;* VERSIONS-NUMMER AUS
424C	3E2E	774	MVI A,'.'
424E	CD5200	775	CALL WCHAR ;* GEBE '.' AUS
4251	23	776	INX H ;*
4252	7E	777	MOV A,M ;* GEBE 2. ZIFFER DER
4253	CD5200	778	CALL WCHAR ;* VERSIONS-NUMMER AUS
4256	CD7300	779	CALL PTXTCR ;PRINT TEXT
4259	0A	780	DB LF,'(C) 1985 BY BFZ, ESSEN, W. GERMANY',00
425A	28432920		
425E	31393835		
4262	20425920		
4266	42465A2C		
426A	20455353		
426E	454E2C20		
4272	572E2047		
4276	45524D41		
427A	4E59		
427C	00		
427D	00	781	NOP
427E	00	782	NOP
427F	00	783	NOP
4280	00	784	NOP
4281	00	785	NOP
4282	00	786	NOP
4283	00	787	NOP
		788	;
		789	

LOC	OBJ	LINE	SOURCE STATEMENT
		790	;=====
		791	;TRITT EIN FEHLER AUF, SO ARBEITET DAS DOS DIE "ERROR"-ROUTINE AB.
		792	;DIESE ENDET MIT EINEM SPRUNG ZU DER ADRESSE, DIE UNTER "ERRRET" IM
		793	;RAM STEHT. DAS PROGRAMM TRAEGT HIER "MENUE" EIN.
		794	;
4284	F3	795	MENUE: DI
4285	218442	796	LXI H,MENUE ;* RETURN-ADR VON ERROR
4288	2258FA	797	SHLD ERRRET ;* ABSPEICHERN
		798	;=====
		799	;AUSGABE DES MENUES
		800	;
428B	CD7300	801	CALL PTXTCR ;PRINT TEXT
428E	0A	802	DB LF,'MENUE:',LF,CR,00
428F	4D454E55		
4293	453A		
4295	0A		
4296	0D		
4297	00		
4298	2A66FA	803	LHLD PMENUE ;ZEIGER AUF KOMMANDO-TABELLE
429B	CDB803	804	CALL PKLIST ;GEBE TABELLE AUS
		805	;=====
		806	;LESE EINGABE
		807	;
429E	CDA14D	808	CALL GETCHR ;LESE ZEICHEN
42A1	2A64FA	809	LHLD PTZZ ;ZEIGER AUF TAB. D. ZULAESSIGEN ZEICHEN
42A4	CDB00E	810	CALL TEST ;ZEICHEN GUELTIG ?
42A7	DC4F48	811	CC CODE0 ;NEIN --> CODE0
		812	;=====
		813	;HIER, WENN EINGABEZEICHEN GUELTIG
		814	;
42AA	4F	815	MOV C,A ;ZEICHEN NACH C
42AB	2A66FA	816	LHLD PMENUE ;ZEIGER AUF KOMMANDO-TABELLE
42AE	C3EB01	817	JMP EXEC ;FUEHRE KOMMANDO AUS
		818	;
		819	

LOC	OBJ	LINE	SOURCE STATEMENT
		820	;=====
		821	;INFO-TABELLE DER GRUNDVERSION
		822	;
42B1	D54F	823	INFOT: DW TYPT ;ZEIGER AUF TYP-TABELLE
42B3	4C4E	824	DW ERRVEK ;ZEIGER AUF ERROR-VEKTOR-TABELLE
42B5	3134	825	DB '14' ;VERSION 1.4
42B7	4445464C	826	DB 'DEFLSQ' ;TABELLE DER ZULAESSIGEN ZEICHEN
42BB	5351		
42BD	00	827	DB 00 ; (ABGESCHLOSSEN MIT OOH)
		828	;
		829	;KOMMANDO-LISTE:
		830	;AUFBAU:
		831	;1. KOMMANDO-NAME
		832	;2. OOH
		833	;3. ADR DER ROUTINE, DIE BEI EINGABE DES KOMMANDS ABGEARBEITET
		834	; WERDEN SOLL
		835	;
		836	;PUNKTE 1,2,3 FUER WEITERE KOMMANDOS WIEDERHOLEN.
		837	;
42BE	44495245	838	DB 'DIRECTORY',00
42C2	43544F52		
42C6	59		
42C7	00		
42C8	F542	839	DW DIR
42CA	45524153	840	DB 'ERASE',00
42CE	45		
42CF	00		
42D0	7C43	841	DW ERASE
42D2	464F524D	842	DB 'FORMAT',00
42D6	4154		
42D8	00		
42D9	C743	843	DW FORMAT
42DB	4C4F4144	844	DB 'LOAD',00
42DF	00		
42E0	5946	845	DW LOAD
42E2	53415645	846	DB 'SAVE',00
42E6	00		
42E7	1847	847	DW SAVE
42E9	51554954	848	DB 'QUIT',00
42ED	00		
42EE	F142	849	DW QUIT
42F0	00	850	DB 00 ;KOMMANDO-LISTE MUSS MIT OOH BEENDET WERDEN.
		851	;
		852	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		853	***
		854	****
		855	*****
		856	***** HIER, WENN 'Q' (QUIT) EINGEGEBEN WIRD
		857	*****
		858	****
		859	***
		860	;
42F1	2A5CFA	861	QUIT:           LHLD    DOSRET           ;LADE RUECKSPRUNG-ADR
42F4	E9	862	PCHL               ;BEENDE BFZ-MINI-DOS
		863	;
		864	

LOC	OBJ	LINE	SOURCE STATEMENT
		865	;***
		866	;****
		867	;*****
		868	;***** HIER, WENN "D" (DIRECTORY) EINGEGEBEN WIRD
		869	;*****
		870	;****
		871	;***
		872	;
		873	;TRITT EIN FEHLER AUF, SO ARBEITET DAS DOS DIE ROUTINE "ERROR" AB.
		874	;DIESE ENDET MIT EINEM SPRUNG ZU DER ADRESSE, DIE UNTER "ERRRET" IM
		875	;RAM STEHT. DAS PROGRAMM TRAEGT HIER "DIRE" EIN.
		876	;
42F5	21FB42	877	DIRE: LXI H,DIRE ;* RETURN-ADR VON ERROR
42F8	2258FA	878	SHLD ERRRET ;* ABSPEICHERN
		879	;=====
		880	;VON WELCHEM LAUFWERK SOLL DAS DIRECTORY ANGEZEIGT WERDEN ?
		881	;
42FB	CD4C4D	882	DIRE: CALL LAUFW ;* LESE LAUFWERK-NAME
		883	;* PRUEFE IHN
		884	;* BERECHNE SELECT-MASKE
		885	;=====
		886	;BEI FEHLERN IM WEITEREN PROGRAMM-ABLAUF RUECKKEHR NACH "MENUE"
		887	;
42FE	218442	888	LXI H,MENUE ;: RETURN-ADR VON ERROR
4301	2258FA	889	SHLD ERRRET ;: ABSPEICHERN
		890	;
4304	CD6D00	891	CALL PTXT ;PRINT 2X LINE-FEED
4307	0A	892	DB LF,LF,00
4308	0A		
4309	00		
		893	;
430A	3E01	894	MVI A,01 ;* SETZE FLAG: "DISPLAY DIRECTORY"
430C	3254FA	895	STA ISPLD ;* (DIRECTORY ANZEIGEN)
430F	CDFF4C	896	CALL LINE1 ;PRINT UEBERSCHRIFT, INIT ZEILENZAEHLER
4312	CDB74B	897	CALL SEINTR ;LESE DIRECTORY UND ZEIGE EINTRAEGE AN
		898	;
4315	FE4F	899	CPI 79 ;79 FREIE EINTRAEGE ?
4317	C25743	900	JNZ NLEER ;NEIN --> NLEER (NICHT LEER)
		901	;
		902	



LOC	OBJ	LINE	SOURCE STATEMENT
		903	=====
		904	;HIER, WENN DIRECTORY LEER
		905	;
431A	2A84FC	906	LHLD PRTST ;DRUCKER "FLAG"
431D	E5	907	PUSH H ;RETTE ES
431E	219F08	908	LXI H,PRTOFF ;DRUCKER-AUS-"FLAG"
4321	2284FC	909	SHLD PRTST ;SETZE "FLAG" EIN (DRUCKER AUS)
4324	CD6D00	910	CALL PTXT ;GEBE STEUERZEICHEN AUS
4327	0B	911	DB 'OBH,' ',0DH,00 ;(LOESCHE KOPFZEILE)
4328	20		
4329	0D		
432A	00		
432B	E1	912	POP H ;ALTES DRUCKER-"FLAG"
432C	2284FC	913	SHLD PRTST ;SETZE ES EIN
432F	CD6D00	914	CALL PTXT ;PRINT TEXT
4332	2A2A2A20	915	DB '*** KEIN EINTRAG IM DIRECTORY ***',LF,LF,CR,00
4336	4B45494E		
433A	2045494E		
433E	54524147		
4342	20494D20		
4346	44495245		
434A	43544F52		
434E	59202A2A		
4352	2A		
4353	0A		
4354	0A		
4355	0D		
4356	00		
		916	=====
		917	;HIER, WENN DIRECTORY NICHT LEER
		918	;BZW. WENN "KEIN EINTRAG IM DIRECTORY" AUSGEGEBEN WURDE
		919	;ANZEIGE DER FREIEN 4K-BYTE-BLOECKE
		920	;
4357	CD194E	921	NLEER: CALL CRLF ;PRINT CR,LF
435A	CD6100	922	CALL WDEZ ;PRINT ANZAHL
435D	CD6D00	923	CALL PTXT ;PRINT TEXT
4360	20465245	924	DB 'FREIE 4K-BYTE-BLOECKE',LF,CR,00
4364	49452034		
4368	4B2D4259		
436C	54452D42		
4370	4C4F4543		
4374	4B45		
4376	0A		
4377	0D		
4378	00		
		925	=====
		926	;BEENDE ROUTINE
		927	;DIE ROUTINE WIRD DURCH "JMP ERROR2" BEENDET, OBWOHL KEIN FEHLER
		928	;AUFGETRETEN IST. ES WERDEN BEFEHLE IN DER "ERROR"-ROUTINE GENUTZT
		929	;UND DADURCH SPEICHERPLATZ GESPART.
		930	;
4379	C3B948	931	JMP ERROR2 ;DESELECT LAUFWERK
		932	;WARTE BIS "SPACE" EINGEGEBEN WIRD
		933	

LOC	OBJ	LINE	SOURCE STATEMENT
		934	***
		935	****
		936	*****
		937	***** HIER, WENN "E" (ERASE) EINGEGEBEN WURDE
		938	*****
		939	*****
		940	***
		941	;
		942	;TRITT EIN FEHLER AUF, SO VERZWEIGT DAS BFZ-MINI-DOS ZU DER ROUTINE
		943	; "ERROR". DIESE WIRD MIT EINEM SPRUNG ZU DER ADRESSE BEENDET, DIE
		944	; UNTER "ERRRET" IM RAM STEHT. DAS PROGRAMM TRAEGT HIER "ERERR"
		945	; (ERASE-ERROR) EIN.
		946	;
437C	218243	947	ERASE: LXI H,ERERR ;RETURN-ADR VON ERROR
437F	2258FA	948	SHLD ERRRET ;ABSPEICHERN
		949	;
		950	;AUF WELCHEM LAUFWERK SOLL GELOESCHT WERDEN ?
		951	;
4382	CD4C4D	952	ERERR: CALL LAUFW ;LESE LAUFWERK-NAME
		953	;
		954	;BEI FEHLERN IM WEITEREN PROGRAMMABLAUF RUECKKEHR NACH "MENUE"
		955	;
4385	218442	956	LXI H,MENUE ;RETURN-ADR VON ERROR
4388	2258FA	957	SHLD ERRRET ;ABSPEICHERN
		958	;
		959	;START DES LOESCHVORGANGS
		960	;
		961	;FILE-EINTRAG IM DIRECTORY SUCHEN
		962	;
438B	AF	963	XRA A ;* LOESCHE FLAG
438C	3254FA	964	STA DSPLI ;* --> DIRECTORY NICHT ANZEIGEN
438F	CD594B	965	CALL GETNAM ;LESE FILE-NAME
4392	CD874B	966	CALL SEINTR ;SUCHE EINTRAG
		967	;
		968	;WIEVIELE DIRECTORY-EINTRAEGE BESITZT DER FILE ?
		969	; (NULL EINTRAEGE = FILE NICHT IM VERZEICHNIS)
		970	;
4395	3A27FA	971	LDA ZAHLV ;ANZAHL DER ZU LOESCHENDEN EINTRAEGE
4398	B7	972	ORA A ;NULL ?
4399	CC704B	973	CZ CODE11 ;JA --> FILE NICHT IM VERZEICHNIS
		974	;
		975	;BEIM LOESCHEN (ERASE) WIRD NICHT DER EIGENTLICHE FILE GELOESCHT
		976	; ES WIRD IM DIRECTORY-EINTRAG NUR EIN BYTE VERAENDERT
		977	; (AUFBAU DES DIRECTORY: SIEHE UNTERPROGRAMM "SEINTR")
		978	; DAZU MUSS AUF DIE DISKETTE GESCHRIEBEN WERDEN
		979	;
439C	CD064B	980	CALL RESTORE ;* KOPF IST SCHON AUF SPUR NULL
		981	;* DIENT NUR ZUR AKTUALISIERUNG
		982	;* DES FDC-STATUS-REGISTERS
		983	ANI 10011100B ;: ALLES OK ?
439F	E69C	984	CPI 00000100B ;:
43A1	FE04	985	CMZ CODE1 ;NEIN --> CODE1
43A3	C4524B	986	MOV A,E ;STATUS ERNEUT NACH A
43A4	7B	987	ANI 01000000B ;WRITE PROTECT ?
43A7	E640	988	CMZ CODE4 ;JA --> CODE4
43A9	C45B4B		

LOC	OBJ	LINE	SOURCE STATEMENT
		989	-----
		990	;DER EIGENTLICHE LOESCHVORGANG WIRD VOM UNTERPROGRAMM "DELETE" AUSGEFUEHRT
		991	;
43AC	3A27FA	992	LIA ZAEHLV ;ANZAHL DER ZU LOESCHENDEN EINTRAEGE
43AF	CDB94A	993	CALL DELETE ;LOESCHE EINTRAEGE
		994	=====
		995	;LOESCHVORGANG BEENDET
		996	;
43B2	CD2802	997	CALL P*STAR ;PRINT TEXT
43B5	46494C45	998	DB 'FILE GELOESCHT',00
43B9	2047454C		
43BD	4F455343		
43C1	4854		
43C3	00		
		999	=====
		1000	;ROUTINE BEENDEN
		1001	;DIE ROUTINE WIRD DURCH EINEN SPRUNG NACH "ERET" BEENDET OBWOHL KEIN
		1002	;FEHLER AUFGETRETEN IST. DURCH DIE NUTZUNG EINIGER BEFEHLE DER "ERROR"-
		1003	;ROUTINE WIRD SPEICHERPLATZ GESPART
		1004	;
43C4	C3CC48	1005	JMP ERET ;FERTIG
		1006	;
		1007	

LOC	OBJ	LINE	SOURCE STATEMENT
		1008	;***
		1009	;****
		1010	;*****
		1011	;***** HIER, WENN "F" (FORMAT) EINGEGEBEN WIRD
		1012	;*****
		1013	;****
		1014	;***
		1015	;
43C7	3132FC	1016	FORMAT: LXI SP,0FC32H ;(RE)-INIT SP
		1017	;
		1018	;WENN EIN FEHLER AUFTRITT, WIRD DIE ROUTINE "ERROR" ABGEARBEITET. DIESE
		1019	;ENDET MIT EINEM SPRUNG ZU DER ADRESSE, DIE UNTER "ERRRET" IM RAM STEHT.
		1020	;DAS PROGRAMM SETZT HIER "FORM1" EIN.
		1021	;
43CA	214244	1022	LXI H,FORM1 ;RETURN-ADR VON ERROR-ROUTINE
43CD	2258FA	1023	SHLD ERRRET ;ABSPEICHERN
		1024	;
		1025	;WARNUNG AUSGEBEN
		1026	;
43D0	CD7300	1027	CALL PTXTCR ;PRINT TEXT
43D3	07	1028	DB 07,LF
43D4	0A		
43D5	41204320	1029	DB 'A C H T U N G !',LF,CR
43D9	48205420		
43DD	55204E20		
43E1	472021		
43E4	0A		
43E5	0D		
43E6	50524F47	1030	DB 'PROGRAMME IM BEREICH E000 - FFFF (Z. B. SPS)',LF,CR
43EA	52414D4D		
43EE	4520494D		
43F2	20424552		
43F6	45494348		
43FA	20453030		
43FE	30202D20		
4402	46464646		
4406	20285A2E		
440A	20422E20		
440E	53505329		
4412	0A		
4413	0D		
4414	554E4420	1031	DB 'UND DATEN AUF DER DISKETTE WERDEN ZERSTOERT !',00
4418	44415445		
441C	4E204155		
4420	46204445		
4424	52204449		
4428	534B4554		
442C	54452057		
4430	45524445		
4434	4E205A45		
4438	5253544F		
443C	45525420		
4440	21		
4441	00		
		1032	;

LOC	OBJ	LINE	SOURCE STATEMENT
		1033	=====
		1034	;IN WELCHEM LAUFWERK SOLL FORMATIERT WERDEN ?
		1035	=====
4442	CI4C4D	1036	FORM1: CALL LAUFW ;LAUFWERK-NAME EINLESEN, MASKE ERSTELLEN
		1037	=====
		1038	;BITTE DISKETTE EINLEGEN, DANN <SPACE>
		1039	;
4445	CD7300	1040	CALL PTXTCR ;PRINT TEXT
4448	4449534B	1041	DB 'DISKETTE IN LAUFWERK ',00
444C	45545445		
4450	20494E20		
4454	4C415546		
4458	5745524B		
445C	2022		
445E	00		
445F	3A53FA	1042	LDA CHAR ;LAUFWERK-BUCHSTABE NACH A
4462	CI5200	1043	CALL WCHAR ;PRINT BUCHSTABE
4465	CI6D00	1044	CALL PTXT ;PRINT TEXT
4468	222C2044	1045	DB ' ", DANN <SPACE>',00
446C	414E4E20		
4470	3C535041		
4474	43453E		
4477	00		
4478	CI6F4D	1046	CALL WAITSP ;WARTEN, BIS <SP> EINGEGEBEN WIRD
447B	CD194E	1047	CALL CRLF ;PRINT CR,LF
447E	1628	1048	MVI I,40 ;* 40 STRICHE
4480	CD0F4E	1049	CALL PLINE ;* AUSGEBEN
		1050	=====
		1051	; LAUFWERK SELEKTIEREN
		1052	;
4483	AF	1053	XRA A ;SEITE = 0
4484	3284FA	1054	STA SEITE
4487	CD8B49	1055	CALL SELECT ;SELECT LAUFWERK
		1056	=====
		1057	; "ERRRET" MUSS ERNEUT MIT "FORM1" GELADEN WERDEN,
		1058	; DA DER WERT DURCH EIN UNTERPROGRAMM VERAENDERT WURDE.
		1059	;
448A	214244	1060	LXI H,FORM1 ;RETURN-ADR VON ERROR-ROUTINE
448D	2258FA	1061	SHLD ERRRET ;ABSPEICHERN
		1062	=====
		1063	;LAUFWERK BEREIT ?
		1064	;
4490	CDAA49	1065	CALL DREADY ;LAUFWERK BEREIT ?
4493	DC5848	1066	CC CODE3 ;FEHLER, WENN LAUFWERK NICHT BEREIT
		1067	=====
		1068	;INITIALISIERUNG DER INTERRUPT-ROUTINE
		1069	; (DAS BFZ-MINI-DOS NUTZT DIE FDC-SIGNALE "DRQ" UND "INTRQ" ZUR
		1070	;SYNCHRONISIERUNG DER DATENUEBERTRAGUNG.)
		1071	;
4496	CD2E4E	1072	CALL INTINT ;INTERRUPT-ROUTINE INITIALISIEREN
		1073	

LOC	OB.I	LINE	SOURCE STATEMENT
		1074	;=====
		1075	; KOPF AUF SPUR 0
		1076	;
4499	CDD648	1077	CALL RESTORE ;RESTORE
449C	E640	1078	ANI 0100000B ;WRITE PROTECT ?
449E	C45B48	1079	CNZ CODE4 ;JA --> FEHLER-MELDUNG
44A1	7B	1080	MOV A,E ;STATUS NACH A
44A2	E6C4	1081	ANI 11000100B ;* ALLES OK ?
44A4	FE04	1082	CFI 00000100B ;*
44A6	C45248	1083	CNZ CODE1 ;NEIN --> RESTORE ERROR
44A9	210F00	1084	LXI H,15 ;* 15 MS VERZOEGERUNG, DAMIT DER KOPF
44AC	CDB449	1085	CALL DELAY ;* NACH RESTORE RUHIG STEHT
		1086	;=====
		1087	; TRACK IM RAM AUFBAUEN
		1088	;
44AF	21E24F	1089	LXI H,VERTAB ;INIT ZEIGER AUF VERSATZTABELLE
44B2	5E	1090	MOV E,M ;LESE ERSTEN EINTRAG
44B3	2256FA	1091	SHLD VERPTR ;RETTE POINTER
44B6	2100E0	1092	LXI H,BUFFER ;ZEIGER AUF TRACK BUFFER
		1093	-----
		1094	;SPUR-VORSPANN
		1095	; "VOR-GAP"
		1096	;
44B9	3E4E	1097	MVI A,4EH ;80*4E
44BB	0650	1098	MVI B,80
44BD	CDE84D	1099	CALL REPCHR
		1100	;
44C0	3E00	1101	MVI A,0 ;12*00
44C2	060C	1102	MVI B,12
44C4	CDE84D	1103	CALL REPCHR
		1104	;
44C7	3EF6	1105	MVI A,0F6H ;3*F6 (SCHREIBT C2, FEHLENDER
44C9	0603	1106	MVI B,3 ; CLOCK ZWISCHEN BIT 3 UND 4)
44CB	CDE84D	1107	CALL REPCHR
		1108	;
44CE	36FC	1109	MVI M,0FCH ;1*FC (INDEX MARK)
44D0	23	1110	INX H
		1111	;
44D1	3E4E	1112	MVI A,4EH ;50*4E
44D3	0632	1113	MVI B,50
44D5	CDE84D	1114	CALL REPCHR
		1115	-----
		1116	;DIE SEKTOREN
		1117	-----
		1118	; "GAP"
		1119	;
44DB	3E00	1120	SECTOR: MVI A,0 ;12*00
44DA	060C	1121	MVI B,12
44DC	CDE84D	1122	CALL REPCHR
		1123	;
44DF	3EF5	1124	MVI A,0F5H ;3*F5 (SCHREIBT A1, FEHLENDER CLOCK
44E1	0603	1125	MVI B,3 ; ZWISCHEN BIT 4 UND 5)
44E3	CDE84D	1126	CALL REPCHR
		1127	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1128	-----
		1129	; "MARKE"
		1130	;
44E6	36FE	1131	MVI M,0FEH ;1*FE (ID ADDRESS MARK)
44E8	23	1132	INX H
		1133	-----
		1134	; "SEKTOR-KENNUNGSFELD"
		1135	;
44E9	23	1136	INX H ;BLEIBT FREI FUER TRACK-NUMMER
		1137	;
44EA	23	1138	INX H ;BLEIBT FREI FUER SEITEN-NUMMER
		1139	;
44EB	73	1140	MOV M,E ;SEKTOR-NUMMER
44EC	23	1141	INX H
		1142	;
44ED	3602	1143	MVI M,02 ;CODE: 512 BYTES / SEKTOR
44EF	23	1144	INX H
		1145	;
44F0	36F7	1146	MVI M,0F7H ;1*F7 (ERZEUGT ZWEI CRC BYTES)
44F2	23	1147	INX H
		1148	-----
		1149	; "GAP"
		1150	;
44F3	3E4E	1151	MVI A,4EH ;22*4E
44F5	0616	1152	MVI B,22
44F7	CDE84D	1153	CALL REPCHR
		1154	;
44FA	3E00	1155	MVI A,0 ;12*00
44FC	060C	1156	MVI B,12
44FE	CDE84D	1157	CALL REPCHR
		1158	;
4501	3EF5	1159	MVI A,0F5H ;3*F5 (SCHREIBT A1, FEHLENDER CLOCK
4503	0603	1160	MVI B,3 ; ZWISCHEN BIT 4 UND 5)
4505	CDE84D	1161	CALL REPCHR
		1162	-----
		1163	; "MARKE"
		1164	;
4508	36FB	1165	MVI M,0FBH ;1*FB (DATA ADDRESS MARK)
450A	23	1166	INX H
		1167	-----
		1168	; DATENBLOCK EINES SEKTORS
		1169	;
450B	3EE5	1170	MVI A,0E5H ;512 * E5 (512 DATEN-BYTES)
450D	0E02	1171	MVI C,2
450F	0600	1172	MVI B,0
4511	CDE84D	1173	FDATA: CALL REPCHR
4514	0D	1174	DCR C
4515	C21145	1175	JNZ FDATA
		1176	-----
		1177	; "PRUEFBYTES"
		1178	;
4518	36F7	1179	MVI M,0F7H ;1*F7 (ERZEUGT 2 CRC-BYTES)
451A	23	1180	INX H
		1181	

LOC	OBJ	LINE	SOURCE STATEMENT
		1182	;- - - - -
		1183	; "GAP"
		1184	;
451B	3E4E	1185	MVI A,4EH ;54*4E
451D	0636	1186	MVI B,54
451F	CDE84D	1187	CALL REPCHR
		1188	;
		1189	;8 SEKTOREN IN SPUR-PUFFER EINGETRAGEN ?
		1190	;NEIN --> LESE NAECHSTE SEKTOR-NUMMER AUS DER VERSATZ-TABELLE
		1191	; UND TRAGE DEN SEKTOR IM SPUR-PUFFER EIN
		1192	;
		1193	;JA --> WEITER BEI "TRKEND"
		1194	;
4522	7B	1195	MOV A,E ;SEKTOR-NUMMER NACH A
4523	FE08	1196	CPI 8 ;SEKTOR 8 (TRACK-ENDE) ?
4525	CA3545	1197	JZ TRKEND ;JA --> TRKEND
		1198	;
4528	E5	1199	PUSH H ;RETTE BUFFER-ZEIGER
4529	2A56FA	1200	LHLD VERPTR ;HL=POINTER AUF VERSATZ TABELLE
452C	23	1201	INX H ;STELLE ZEIGER WEITER
452D	5E	1202	MOV E,M ;LESE TABELLEN-EINTRAG
452E	2256FA	1203	SHLD VERPTR ;RETTE POINTER AUF VERSATZ-TABELLE
4531	E1	1204	POP H ;RESTORE BUFFER-ZEIGER
4532	C3D844	1205	JMP SECTOR
		1206	;
		1207	; "NACH-GAP"
		1208	;
4535	3E4E	1209	TRKEND: MVI A,4EH ;FUELLE BUFFER MIT 4E
4537	0600	1210	MVI B,0
4539	1606	1211	MVI D,6
453B	01EB4D	1212	TRKEO: CALL REPCHR
453E	15	1213	DCR D
453F	C23B45	1214	JNZ TRKEO
		1215	;
		1216	;GEBE <CR> AUS (CURSOR AUF ANFANG DER "-----"-ZEILE)
		1217	;
4542	3E0D	1218	MVI A,CR ;* WAGENRUECKLAUF
4544	015200	1219	CALL WCHAR ;*
		1220	;
		1221	



BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1222	=====
		1223	;SCHREIBE TRACK FUER TRACK AUF DIE DISKETTE
		1224	;
		1225	-----
		1226	;VORBEREITUNG
		1227	;
4547	1600	1228	LOOP0: MVI D,0 ;INIT TRACK-ZAEHLER
4549	AF	1229	LOOP: XRA A ;* SEITE NULL
454A	3284FA	1230	STA SEITE ;*
454D	67	1231	MOV H,A ;*
454E	6A	1232	MOV L,D ;TRACK-NUMMER
454F	22A2E0	1233	SHLD BUFFER+162 ;* SPUR- UND
4552	2214E3	1234	SHLD BUFFER+788 ;* TRACK-NUMMER
4555	2286E5	1235	SHLD BUFFER+1414 ;* IM SPUR-PUFFER
4558	22F8E7	1236	SHLD BUFFER+2040 ;* EINTRAGEN
455B	226AEA	1237	SHLD BUFFER+2666 ;*
455E	22DCEC	1238	SHLD BUFFER+3292 ;*
4561	224EEF	1239	SHLD BUFFER+3918 ;*
4564	22C0F1	1240	SHLD BUFFER+4544 ;*
		1241	-----
		1242	;AKTUELLEN TRACK AUF SEITE 0 SCHREIBEN
		1243	;
4567	3E03	1244	MVI A,MAXTRY ;* INIT VERSUCHSZAehler FUER VERIFY
4569	3288FA	1245	STA RETRY ;*
456C	CDFF48	1246	LOOP10: CALL WTRK ;SCHREIBE TRACK (SEITE 0)
456F	E684	1247	ANI MWRTRK ;ALLES OK ?
4571	C45548	1248	CNZ CODE2 ;NEIN --> WRITE ERROR
		1249	-----
		1250	;PRUEFE TRACK. BEI FEHLERN WIRD TRACK NEU GESCHRIEBEN (BIS MAXTRY MAL)
		1251	;
4574	CD6149	1252	CALL VERIFY ;PRUEFE TRACK
4577	CA8745	1253	JZ LOOP11 ;ALLES OK --> LOOP11
457A	3A88FA	1254	LDA RETRY ;ANZAHL DER VERSUCHE
457D	3D	1255	ICR A ;-1
457E	CC5E48	1256	CZ CODE5 ;0 --> VERIFY-ERROR
4581	3288FA	1257	STA RETRY ;NICHT 0 --> SPEICHERE NEUEN ZAEHLER
4584	C36C45	1258	JMP LOOP10 ;NEUER SCHREIB-VERSUCH
		1259	-----
		1260	;VORBEREITUNG ZUM SCHREIBEN DES TRACKS AUF SEITE 1
		1261	;
4587	3E01	1262	LOOP11: MVI A,1 ;* SEITE 1
4589	3284FA	1263	STA SEITE ;*
458C	32A3E0	1264	STA BUFFER+163 ;*
458F	3215E3	1265	STA BUFFER+789 ;*
4592	3287E5	1266	STA BUFFER+1415 ;*
4595	32F9E7	1267	STA BUFFER+2041 ;*
4598	326BEA	1268	STA BUFFER+2667 ;*
459B	32DDEC	1269	STA BUFFER+3293 ;*
459E	324FEF	1270	STA BUFFER+3919 ;*
45A1	32C1F1	1271	STA BUFFER+4545 ;*
		1272	

LOC	OBJ	LINE	SOURCE STATEMENT
		1273	;
		1274	;AKTUELLEN TRACK AUF SEITE 1 SCHREIBEN
		1275	;
45A4	3E03	1276	MVI A,MAXTRY ;* INIT VERSUCHSZAehler FUER VERIFY
45A6	3288FA	1277	STA RETRY ;*
45A9	CDFF48	1278 LOOP13:	CALL WTRK ;WRITE TRACK (SEITE 1)
45AC	E684	1279	ANI MWRTRK ;ALLES OK ?
45AE	C45548	1280	CNZ CODE2 ;NEIN --> WRITE ERROR
		1281	;
		1282	;TRACK PRUEFEN. BEI FEHLERN TRACK NEU SCHREIBEN (BIS MAXTRY MAL)
		1283	;
45B1	CD6149	1284	CALL VERIFY ;PRUEFE TRACK
45B4	CAC445	1285	JZ LOOP12 ;ALLES OK --> LOOP12
45B7	3A88FA	1286	LDA RETRY ;ANZAHL DER VERSUCHE
45BA	3D	1287	DCR A ;-1
45BB	CC5E48	1288	CZ CODE5 ;0 --> VERIFY-ERROR
45BE	3288FA	1289	STA RETRY ;NICHT 0 --> SPEICHERE NEUEN ZAEHLER
45C1	C3A945	1290	JMP LOOP13 ;NEUER SCHREIB-VERSUCH
		1291	;
		1292	;EINEN TRACK PRO SEITE GESCHRIEBEN. GEBE "+" AUS
		1293	;
45C4	CD094E	1294 LOOP12:	CALL PPLUS ;* PRINT "+"
		1295	;
		1296	;40 TRACKS PRO SEITE GESCHRIEBEN ?
		1297	;
45C7	14	1298	INR I ;NAECHSTE TRACK-NUMMER
45C8	7A	1299	MOV A,I ;TRACK-NUMMER NACH A
45C9	FE28	1300	CPI 40 ;TRACK 40 ?
45CB	CA0A45	1301	JZ FERTIG ;JA --> FERTIG
		1302	;
		1303	;ES MUESSEN NOCH TRACKS GESCHRIEBEN WERDEN
		1304	;
45CE	CDD848	1305	CALL STEPIN ;STEP IN (KOPF UM EINE SPUR NACH INNEN)
45D1	210F00	1306	LXI H,15 ;* 15 MS VERZOEGERUNG, DAMIT DER KOPF
45D4	CDB449	1307	CALL DELAY ;* NACH STEPIN RUHIG STEHT
45D7	C34945	1308	JMP LOOP ;SCHREIBE NAECHSTEN TRACK
		1309	;
		1310	;
		1311	; DISKETTE IST FORMATIERT, DIRECTORY ANLEGEN.
		1312	;
45DA	AF	1313 FERTIG:	XRA A ;A=0 (DIRECTORY WIRD AUF SEITE 0 GESCHR.)
45DB	3284FA	1314	STA SEITE ;SEITE 0
45DE	47	1315	MOV B,A ;) SEKTOR-PUFFER (512 BYTES)
45DF	2100FB	1316	LXI H,SECB ;) MIT 00H FUELLEN.
45E2	CDE84D	1317	CALL REPCHR ;)
45E5	CDE84D	1318	CALL REPCHR ;)
		1319	;
		1320	;DIRECTORY WIRD AUF SPUR 0 GESSCHRIEBEN. DAHER "RESTORE" (KOPF AUF SPUR 0)
		1321	;
45E8	CDD648	1322	CALL RESTORE ;TRACK 0
45EB	E6DC	1323	ANI MREST ;* ALLES OK ?
45ED	FE04	1324	CPI 00000100B ;*
45EF	C45248	1325	CNZ CODE1 ;NEIN --> CODE1
		1326	;

LOC	OBJ	LINE	SOURCE STATEMENT
		1327	;
		1328	;DAS DIRECTORY BELEGT AUF DER SPUR DIE SEKTOREN 1 ... 5
		1329	;SCHREIBE DEN INHALT DES SEKTOR-PUFFERS (ALLES 00H) IN DIE 5 SEKTOREN
		1330	; (00H ---) DIRECTORY LEER)
		1331	;
45F2	3E01	1332	MVI A,1 ;SEKTOR-NUMMER
45F4	D3C2	1333	INITD: OUT SEC ;AN FDC
45F6	47	1334	MOV B,A ;RETTE SEKTOR-NUMMER IN B
45F7	1603	1335	MVI D,MAXTRY ;MAXIMALE ANZAHL DER VERSUCHE
45F9	CD1249	1336	INITD1: CALL WVSEC ;SCHREIBE UND PRUEFE SEKTOR
45FC	CA0646	1337	JZ INITD0 ;KEIN FEHLER --> INITD0
45FF	15	1338	DCR D ;NOCH WEITERE VERSUCHE ERLAUBT ?
4600	C2F945	1339	JNZ INITD1 ;JA --> INITD1
4603	C38448	1340	JMP ERROR ;SPRINGE, WENN MEHR ALS "MAXTRY" VERSUCHE
		1341	;
4606	78	1342	INITD0: MOV A,B ;SEKTOR-NUMMER NACH A
4607	3C	1343	INR A ;NAECHSTER SEKTOR
4608	FE06	1344	CPI 6 ;BEREITS 5 SEKTOREN GESCHRIEBEN ?
460A	C2F445	1345	JNZ INITD ;NEIN --> INITD
		1346	;
		1347	;FERTIG.
		1348	;
460D	F3	1349	DI ;DISABLE INTERRUPT
460E	CD9E49	1350	CALL DESEL ;DESELECT LAUFWERK
4611	CD7300	1351	CALL PTXTCR ;PRINT TEXT
4614	0A	1352	DB LF
4615	2A2A2A20	1353	DB '*** DISKETTE FORMATIERT UND GEPRUEFT',LF,CR
4619	4449534E		
461D	45545445		
4621	20464F52		
4625	4D415449		
4629	45525420		
462D	554E4420		
4631	47455052		
4635	55454654		
4639	0A		
463A	0D		
463B	2A2A2A20	1354	DB '*** VERZEICHNIS ANGELEGT',LF,CR,00
463F	5645525A		
4643	45494348		
4647	4E495320		
464B	414E4745		
464F	4C454754		
4653	0A		
4654	0D		
4655	00		
4656	C34244	1355	JMP FORM1
		1356	

LOC	OBJ	LINE	SOURCE STATEMENT
		1357	;***
		1358	;****
		1359	;*****
		1360	;***** HIER, WENN "L" (LOAD) EINGEGEBEN WURDE
		1361	;*****
		1362	;****
		1363	;***
		1364	;
		1365	;TRITT EIN FEHLER AUF, SO ARBEITET DAS BFZ-MINI-DOS DIE ROUTINE "ERROR" AB.
		1366	;DIESE ENDET MIT EINEM SPRUNG ZU DER ADRESSE, DIE IM RAM UNTER "ERRRET" STEHT.
		1367	;DAS PROGRAMM TRAEGT HIER "LER" (LOAD-ERROR) EIN
		1368	;
4659	215F46	1369	LOAD: LXI H,LER ;RETURN VON ERROR
465C	2258FA	1370	SHLD ERRRET ;ABSPEICHERN
		1371	;
		1372	;VON WELCHEM LAUFWERK SOLL GELADEN WERDEN ?
		1373	;
465F	CD4C4D	1374	LER: CALL LAUFW ;LESE LAUFWERK-NAME
4662	CD194E	1375	CALL CRLF ;PRINT CR,LF
		1376	;
		1377	;BEI FEHLERN IM WEITEREN PROGRAMMABLAUF RUECKKEHR NACH "MENUE"
		1378	;
4665	218442	1379	LXI H,MENUE ;RETURN-ADR
4668	2258FA	1380	SHLD ERRRET ;ABSPEICHERN
		1381	;
		1382	;WENN DAS FILE IN DEM DIRECTORY GESUCHT WIRD, SOLL DAS DIRECTORY
		1383	;NICHT ANGEZEIGT WERDEN
		1384	;
466B	AF	1385	XRA A ;* DIRECTORY NICHT ANZEIGEN
466C	3254FA	1386	STA DSPLD ;*
		1387	;
		1388	;LESE FILE-NAME UND SUCHE IHN IN DEM DIRECTORY
		1389	;
466F	CD594B	1390	CALL GETNAM ;LESE FILE-NAME
4672	CD874B	1391	CALL SEINTR ;SUCHE EINTRAG
		1392	;
		1393	;WENN FILE VORHANDEN, MUSS MINDESTENS EIN EINTRAG IM DIRECTORY
		1394	;VORHANDEN SEIN
		1395	;("ZAEHLV"=ANZAHL DER VORHANDENEN EINTRAEGE FUER DEN GESUCHTEN FILE)
		1396	;
4675	3A27FA	1397	LDA ZAEHLV ;* FILE VORHANDEN ?
4678	B7	1398	ORA A ;*
4679	CC7048	1399	CZ CODE11 ;NEIN --> CODE11
		1400	;
		1401	;HIER, WENN FILE VORHANDEN
		1402	;
467C	2A28FA	1403	LHLD TVORH ;LESE POSITION DES 1. EINTRAGS
467F	7C	1404	MOV A,H ;SEKTOR-NUMMER NACH A
4680	D3C2	1405	OUT SEC ;SEKTOR-NUMMER AN FDC
4682	4D	1406	MOV C,L ;EINTRAGS-NUMMER NACH C
4683	CD4C49	1407	CALL RRS ;LESE SEKTOR (MAXTRY VERSUCHE)
4686	79	1408	MOV A,C ;EINTRAGS-NUMMER NACH A
4687	CD4D4B	1409	CALL ANFEIN ;HL ALS ZEIGER AUF EINTRAGS-ANFANG
		1410	

```

LOC  OBJ          LINE      SOURCE STATEMENT
1411 ;=====
1412 ;STIMMEN FILE-TYP (MAT,SPS,BAS) UND AKTUELLER "EFROM"-CODE UEBEREIN ?
1413 ;
1414 ; DOS-AUFRUF VON      ZUM LADEN ERFORDERLICHER FILE-TYP      "EFROM"-CODE
1415 ;     MAT85                MAT ODER SPS ODER BAS                1
1416 ;     SPS                  SPS                                2
1417 ;     BASIC                BAS                                3
1418 ;
468A  7E          1419 LKENN:      MOV      A,M          ;LESE KENN-BYTE
468E  E67F        1420          ANI      01111111B ;LOESCHE BIT 7
468D  5F          1421          MOV      E,A          ;KENNUNG NACH E
468E  3A83FA      1422          LDA      EFROM        ;LESE ENTERED-FROM-CODE
4691  FE01        1423          CFI      01           ;DOS VON MAT AUFGERUFEN ?
4693  CA9A46      1424          JZ       KENNOK       ;JA --> KENNOK
4696  BB          1425          CMP      E            ;VERGLEICHE MIT KENNUNG
4697  C47348      1426          CNZ     CODE12        ;(<) --> FALSCHER FILE-TYP
1427 ;=====
1428 ;HIER, WENN KENNUNG OK
1429 ;LESE START-ADRESSE UND LAENGE AUS DEM DIRECTORY-EINTRAG
1430 ;
469A  110D00      1431 KENNOK:    LXI      D,13         ;OFFSET ZUR START-ADR (IM DIR-EINTRAG)
469D  19          1432          DAD     D            ;ADDIERE OFFSET --> ZEIGER AUF START-ADR
469E  5E          1433          MOV     E,M          ;* LESE START-ADR
469F  23          1434          INX    H            ;*
46A0  56          1435          MOV     D,M          ;*
46A1  EB          1436          XCHG                    ;*
46A2  226FFD      1437          SHLD   STARTA        ;RETTE START-ADR
46A5  ED          1438          XCHG                    ;* LESE LAENGE
46A6  23          1439          INX    H            ;*
46A7  5E          1440          MOV     E,M          ;*
46A8  23          1441          INX    H            ;*
46A9  56          1442          MOV     D,M          ;*
46AA  EB          1443          XCHG                    ;*
46AB  2200FA      1444          SHLD   LAENGE        ;RETTE LAENGE
1445 ;=====
1446 ;STIMMT DIE ANZAHL DER DIRECTORY-EINTRAEGE MIT DER ANZAHL UEBEREIN, DIE SICH
1447 ;AUS DER FILE-LAENGE BERECHNEN LAESST ?
1448 ;
46AE  CD044B      1449          CALL   ANZAHL        ;BERECHNE ANZAHL DER BENOETIGTEN EINTRAEGE
46B1  3A27FA      1450          LDA   ZAEHLV        ;LADE ANZAHL DER VORHANDENEN EINTRAEGE
46B4  BB          1451          CMP   B             ;VERGL. MIT ANZ. DER BENOETIGTEN EINTR.
46B5  C47648      1452          CNZ   CODE13        ;NICHT GLEICH --> DIRECTORY-FEHLER
1453 ;=====
1454 ;DOS-AUFRUF VON MAT85:          FORDERE START-ADR AN
1455 ;DOS-AUFRUF VON SPS ODER BASIC: LOESCHE ALTES SPS/BASIC-PROGRAMM IM SPEICHER
1456 ;
46B8  CD7AFA      1457          CALL   XLAD1         ;MAT      : FORDERE START-ADR AN
1458 ;                               ;SPS, BAS: LOESCHE ALTES PROGRAMM
1459 ;=====
1460 ;WIEVIELE EINTRAEGE MUESSEN EINGELESEN WERDEN ? DRUCKE DIE ENTSPRECHENDE
1461 ;ANZAHL VON STRICHEN
1462 ;
46BB  3A27FA      1463          LDA   ZAEHLV        ;ANZAHL DER ZU LESENDEN EINTRAEGE
46BE  57          1464          MOV   D,A           ;ANZAHL NACH D
46BF  CD0F4E      1465          CALL   PLINE        ;PRINT D-MAL "--"

```

LOC	OBJ	LINE	SOURCE STATEMENT
		1466	;=====
		1467	;VORBEREITUNG:
		1468	;SPEICHER VORBELEGEN
		1469	;BERECHNUNG DER STOP-ADRESSE
		1470	;RAM-VEKTOREN AENDERN
		1471	;
46C2	2A6FFD	1472	LHLD STARTA ;START-ADR
46C5	2205FA	1473	SHLD TEINT ;ZEIGER FUER MOVE-ROUTINE
46C8	EB	1474	XCHG ;START-ADR NACH DE
46C9	2A00FA	1475	LHLD LAENGE ;LAENGE NACH HL
46CC	19	1476	DAD I ;START-ADR PLUS LAENGE
46CD	2B	1477	ICX H ;HL=STOP-ADR
46CE	2271FD	1478	SHLD STOPA ;RETTE STOP-ADR
		1479	;
46D1	2128FA	1480	LXI H,TVORH ;: INIT ZEIGER AUF "TVORH"
46D4	2225FA	1481	SHLD PTVORH ;:
46D7	2125FA	1482	LXI H,PTVORH ;* ZEIGER AUF TABELLEN-POINTER
46DA	224EFA	1483	SHLD ZWSP4 ;*
		1484	;
46DD	3EC3	1485	MVI A,JUMP ;: JUMP-OPCODE
46DF	3271FA	1486	STA XGP1 ;: EINSETZEN
46E2	3274FA	1487	STA XGP2 ;:
46E5	3277FA	1488	STA XGP3 ;:
46E8	214C49	1489	LXI H,RRS ;* ADRESSEN EINSETZEN
46EB	2272FA	1490	SHLD XGP1+1 ;*
46EE	218D4A	1491	LXI H,MOVE ;*
46F1	2275FA	1492	SHLD XGP2+1 ;*
46F4	21094E	1493	LXI H,PPLUS ;*
46F7	2278FA	1494	SHLD XGP3+1 ;*
		1495	;=====
		1496	;PRINT <CR>. (CURSOR AUF DEN ANFANG DER "-----"-ZEILE)
		1497	;
46FA	CD6I00	1498	CALL PTXT ;: PRINT CR
46FD	0D	1499	DB CR,00
46FE	00		
		1500	;=====
		1501	;LESE FILE EIN
		1502	;
46FF	CDC649	1503	CALL GETPUT ;LESE FILE
		1504	;
		1505	;DOS-AUFRUF VON SPS BZW. BASIC: SETZE START-/STOP-POINTER
		1506	;DOS-AUFRUF VON MAT85 : SETZE GO-ADRESSE
		1507	;
4702	CD7DFA	1508	CALL XLAD2 ;SPS,BAS: SETZE START-/STOP-POINTER
		1509	;MAT : SETZE GO-ADRESSE
		1510	;
		1511	;VOLLZUGSMELDUNG
		1512	;
4705	CD2802	1513	CALL PSTAR ;PRINT TEXT
4708	46494C45	1514	DB 'FILE GELADEN',00
470C	2047454C		
4710	4144454E		
4714	00		
		1515	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1516	;=====
		1517	;ROUTINE BEENDEN
		1518	;DIE ROUTINE WIRD DURCH EINE SPRUNG NACH "ERET" BEENDET, OBWOHL KEIN FEHLER
		1519	;AUFGETRETEN IST. DURCH DIE VERWENDUNG EINIGER BEFEHLE DER "ERROR"-ROUTINE
		1520	;WIRD SPEICHERPLATZ GESPART
		1521	;
4715	C3CC48	1522	JMP ERET
		1523	;
		1524	

LOC	OBJ	LINE	SOURCE STATEMENT
		1525	***
		1526	****
		1527	*****
		1528	***** HIER, WENN "S" (SAVE) EINGEGEBEN WIRD
		1529	*****
		1530	*****
		1531	***
		1532	;
		1533	;DOS-AUFRUF VON SFS ODER BASIC: IST DER PROGRAMM-SPEICHER LEER (FEHLER) ?
		1534	;DOS-AUFRUF VON MAT85 : -----
		1535	;
4718	CD68FA	1536	SAVE: CALL XSAV1 ;BAS, SFS: FEHLER, WENN PGM-SPEICHER LEER
		1537	;MAT : NOP
		1538	;
		1539	;TRITT BEI DER PROGRAMMAUSFUEHRUNG EIN FEHLER AUF, SO VERZWEIGT DAS PROGRAMM
		1540	;ZUR "ERROR"-ROUTINE. DIESE ENDET MIT EINEM SPRUNG ZU DER ADRESSE, DIE IM
		1541	;RAM UNTER "ERRRET" GESPEICHERT IST. DAS PROGRAMM TRAEGT HIER "SAVER" EIN
		1542	;
471B	212147	1543	LXI H,SAVER ;RETURN VON ERROR
471E	2258FA	1544	SHLD ERRRET ;ABSPEICHERN
		1545	;
		1546	;MIT WELCHEM LAUFWERK SOLL GESPEICHERT WERDEN ?
		1547	;
4721	CD4C4D	1548	SAVER: CALL LAUFW ;LESE LAUFWERK-NAME
4724	CD194E	1549	CALL CRLF ;PRINT CR,LF
		1550	;
		1551	;BEI WEITEREN FEHLERN IM PROGRAMMABLAUF RUECKKEHR NACH "MENUE"
		1552	;
4727	218442	1553	LXI H,MENUE ;RETURN-ADR
472A	2258FA	1554	SHLD ERRRET ;ABSPEICHERN
		1555	;
		1556	;DOS-AUFRUF VON MAT85: HOLE START/STOP-ADR
		1557	;DOS-AUFRUF VON SFS ODER BASIC: -----
472D	CD6BFA	1558	CALL XSTSP ;MAT: HOLE START/STOP-ADRESSE
		1559	;BAS, SFS: NOP
		1560	;
		1561	;BERECHNE LAENGE
		1562	;
4730	2A6FFD	1563	LHLD STARTA ;HL=START-ADR
4733	EB	1564	XCHG ;DE=START-ADR
4734	2A71FD	1565	LHLD STOPA ;HL=STOP-ADR
4737	CD3910	1566	CALL SUB2 ;* BERECHNE LAENGE
473A	23	1567	INX H ;*
473B	7C	1568	MOV A,H ;: HL=0000 ?
473C	B5	1569	ORA L ;:
473D	CC6D48	1570	CZ CODE10 ;JA --> CODE10
4740	2200FA	1571	SHLD LAENGE ;RETTE LAENGE
		1572	;
		1573	;BEIM FOLGENDEN AUFRUF VON "SEINTR" SOLL DAS DIRECTORY NICHT ANGEZEIGT WERDEN
		1574	;
4743	AF	1575	XRA A ;* DIRECTORY NICHT ANZEIGEN
4744	3254FA	1576	STA DSPLD ;*
		1577	;
		1578	;



BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1579	=====
		1580	;UNTER WELCHEM NAMEN SOLL DER FILE GESPEICHERT WERDEN ?
		1581	;
4747	CD594B	1582	CALL GETNAM ;LESE FILE-NAME
		1583	=====
		1584	;PRUEFE, OB FILE BEREITS VORHANDEN UND OB DER FREIE SPEICHERPLATZ AUF DER
		1585	;DISKETTE ZUM SPEICHERN AUSREICHT
		1586	;
474A	CIB74B	1587	CALL SEINTR ;SUCHE EINTRAG
		1588	=====
		1589	;WIEVIELE EINTRAEGE WERDEN ZUM ABSPEICHERN BENOETIGT ?
		1590	;
474D	CD044B	1591	CALL ANZAHL ;BERECHNE ANZ. DER BENOET. EINTRAEGE
4750	C5	1592	PUSH B ;RETTE ANZAHL DER BENOETIGTEN EINTRAEGE
		1593	=====
		1594	;GIBT ES AUF DER DISKETTE NUTZBARE EINTRAEGE ?
		1595	; (FREIE EINTRAEGE BZW. EINTRAEGE, DIE EINEM FILE GLEICHEN NAMENS
		1596	; ZUGEORDNET SIND)
		1597	;
4751	3A04FA	1598	LDA ZAEHL ;* NUTZBARE EINTRAEGE ?
4754	B7	1599	ORA A ;*
4755	CC674B	1600	CZ CODES ;NEIN --> DISK VOLL
		1601	=====
		1602	;HIER, WENN MINDESTENS EIN NUTZBARER EINTRAG.
		1603	;DIESER EINTRAG IST: - ENTWEDER FREI
		1604	; - ODER ER IST EINEM FILE MIT GLEICHEM NAMEN ZUGEORDNET
		1605	;
4758	B8	1606	CMP B ;VERGLEICHE MIT DEN BENOETIGTEN PLATZ
4759	IC7C4B	1607	CC CODE15 ;--> SPEICHERPLATZ REICHT NICHT AUS
		1608	=====
		1609	;HIER, WENN SPEICHERPLATZ AUSREICHT
		1610	;DER NUTZBARE SPEICHERPLATZ IST: - ENTWEDER FREI
		1611	; - ODER ER IST EINEM FILE MIT GLEICHEM
		1612	; NAMEN ZUGEORDNET
		1613	;
		1614	;..... ;IST DER FILE BEREITS VORHANDEN ?
475C	3A27FA	1615	LDA ZAEHLV ;* FILE BEREITS VORHANDEN ?
475F	B7	1616	ORA A ;*
4760	CAI547	1617	JZ NEW ;NEIN --> NEW
		1618	;
		1619	

LOC	OBJ	LINE	SOURCE STATEMENT
		1620	=====
		1621	;HIER, WENN FILE BEREITS VORHANDEN
		1622	-----
		1623	;GEBE WARNUNG AUS
		1624	;
4763	CD7300	1625	SAVE2: CALL PTXTCR ;PRINT TEXT
4766	0A	1626	DB LF,07
4767	07		
4768	2A2A2A20	1627	DB '*** FILE BEREITS VORHANDEN !',CR,LF
476C	46494C45		
4770	20424552		
4774	45495453		
4778	20564F52		
477C	48414E44		
4780	454E2021		
4784	0D		
4785	0A		
4786	2A2A2A20	1628	DB '*** ALTES FILE UEBERSCHREIBEN ?',CR,LF
478A	414C5445		
478E	53204649		
4792	4C452055		
4796	45424552		
479A	53434852		
479E	45494245		
47A2	4E203F		
47A5	0D		
47A6	0A		
47A7	2A2A2A20	1629	DB '*** J = JA, N = NEIN',00
47AB	4A203D20		
47AF	4A412C20		
47B3	4E203D20		
47B7	4E45494E		
47BB	00		
47BC	CD9E49	1630	CALL DESEL ;DESELECT LAUFWERK
		1631	=====
		1632	;SOLL DER VORHANDENE FILE UEBERSCHRIEBEN WERDEN ?
		1633	;
47BF	CDA14D	1634	CALL GETCHR ;JA/NEIN-ANTWORT LESEN
47C2	FE4E	1635	CPI 'N' ;NEIN ?
47C4	CACC48	1636	JZ ERET ;'N' --> ERET
47C7	FE4A	1637	CPI 'J' ;JA ?
47C9	C26347	1638	JNZ SAVE2 ;WEIER 'J' NOCH 'N' --> SAVE2
		1639	=====
		1640	;HIER, WENN DER VORHANDENE FILE UEBERSCHRIEBEN WERDEN SOLL
		1641	-----
		1642	;DER ALTE FILE WIRD GELOESCHT
		1643	;
47CC	CD194E	1644	CALL CRLF ;PRINT CR,LF
47CF	3A27FA	1645	LDA ZAHLV ;ANZAHL DER ZU LOESCHENDEN EINTRAEGE
47D2	CD894A	1646	CALL DELETE ;LOESCHE ALTE EINTRAEGE
		1647	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1648	;
		1649	;ABSPEICHERUNG DES FILES
		1650	;
		1651	;IST DIE DISKETTE SCHREIBGESCHUETZT ?
		1652	;
47D5	CD184A	1653	NEW: CALL WPTST ;WRITE-PROTECT-TEST
47D8	C45B48	1654	CNZ CODE4 ;SCHREIBGESCHUETZT --> CODE4
		1655	;
		1656	;HIER, WENN DIE DISKETTE NICHT SCHREIBGESCHUETZT IST
		1657	;
		1658	;VORBEREITUNG
		1659	;
47DB	F1	1660	POP PSW ;ANZAHL DER BENOETIGTEN EINTRAEGE
47DC	324AFA	1661	STA ZWSP2 ;ANZAHL NACH ZWSP2
		1662	;
47DF	57	1663	MOV I,A ;ANZAHL NACH I
47E0	CD0F4E	1664	CALL PLINE ;DRUCKE I-MAL "-"
		1665	;
47E3	2105FA	1666	LXI H,TEINT ;: INIT ZEIGER AUF
47E6	2202FA	1667	SHLD PTEINT ;: "TEINT"-TABELLE
47E9	2A6FFD	1668	LHLD STARTA ;START-ADR FUER FILE
47EC	2248FA	1669	SHLD ZWSP1 ;START-ADR FUER AKTUELLEN EINTRAG
47EF	CD8B49	1670	CALL SELECT ;RE-SELECT LAUFWERK
47F2	CDAA49	1671	CALL DREADY ;LAUFWERK BEREIT ?
47F5	DC5848	1672	CC CODE3 ;NEIN --> CODE3
		1673	;
		1674	;PRINT (CR). (CURSOR AN DEN ANFANG DER "-----"-ZEILE)
		1675	;
47F8	3E0D	1676	MVI A,CR ;* WAGENRUECKLAUF
47FA	CD5200	1677	CALL WCHAR ;*
		1678	;
		1679	;VORBEREITUNG
		1680	;
47FD	2102FA	1681	LXI H,PTEINT ;: ZEIGER INITIALISIEREN
4800	224EFA	1682	SHLD ZWSP4 ;:
4803	3EC3	1683	MVI A,JUMP ;LADE JMP-OPCODE
4805	3271FA	1684	STA XGP1 ;* SETZE OP-CODE EIN
4808	3277FA	1685	STA XGP3 ;*
480B	3EC9	1686	MVI A,RETURN ;LADE RET-OPCODE
480D	3274FA	1687	STA XGP2 ;SETZE OP-CODE EIN
4810	212649	1688	LXI H,RWVSEC ;* SETZE ADRESSEN EIN
4813	2272FA	1689	SHLD XGP1+1 ;*
4816	21094E	1690	LXI H,PPLUS ;*
4819	2278FA	1691	SHLD XGP3+1 ;*
		1692	;

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1693	=====
		1694	;EIGENTLICHER SAVE-VORGANG
		1695	;
481C	CDC649	1696	CALL GETPUT ;SCHREIBE FILE AUF DISKETTE
		1697	=====
		1698	;EINTRAEGE FUER FILE IM DIRECTORY VERMERKEN
		1699	;
481F	2105FA	1700	SAVE0: LXI H,TEINT ;: INIT ZEIGER AUF "TEINT"
4822	2248FA	1701	SHLD ZWSP1 ;:
4825	3EC3	1702	MVI A,JUMP ;: JMP-OPCODE
4827	326EFA	1703	STA XDIR ;: EINSETZEN
482A	211C4B	1704	LXI H,VEINTR ;* ADRESSE EINSETZEN
482D	226FFA	1705	SHLD XDIR+1 ;*
4830	CD04A	1706	CALL MDIR ;VERMERKE EINTRAEGE IN DIRECTORY
		1707	=====
		1708	;FERTIG
		1709	;
4833	CD194E	1710	DEB0: CALL CRLF ;PRINT CR,LF
4836	CD2802	1711	CALL PSTAR ;PRINT TEXT
4839	46494C45	1712	DB 'FILE ABGESPEICHERT',00
483D	20414247		
4841	45535045		
4845	49434845		
4849	5254		
484B	00		
		1713	=====
		1714	;ROUTINE BEENDEN
		1715	;DIE ROUTINE WIRD DURCH EINEN SPRUNG NACH "ERET" BEENDET, OBWOHL KEIN
		1716	;FEHLER AUFGETRETEN IST. DURCH DIE VERWENDUNG EINIGER BEFEHLE DER
		1717	; "ERROR"-ROUTINE WIRD SPEICHERPLATZ GESPART
		1718	;
484C	C3CC48	1719	JMP ERET
		1720	;
		1721	

```

LOC  OBJ          LINE          SOURCE STATEMENT
1722 ;             *****
1723 ;             *
1724 ;             *                CODE-TEIL                *
1725 ;             *
1726 ;             * WIRD BEI FEHLERN WIE EIN UNTERPROGRAMM AUFGERUFEN. *
1727 ;             * HIER WIRD DEM C-REGISTER EIN CODE ZUGEWIESEN, DER *
1728 ;             *             DIE ART DES FEHLERS ANGIBT.             *
1729 ;             *
1730 ;             *****
1731 ;
1732 ;LADE C-REGISTER MIT FEHLER-CODE. EINSPRUNG JE NACH FEHLER.
1733 ;DURCH DIE ANWEISUNG "DB 21H" SIEHT DER PROZESSOR NACH DER ERSTEN
1734 ;"MVI C,.."-ANWEISUNG NUR "LXI H,...."-ANWEISUNGEN (21H = LXI H,....)
1735 ;DER WERT DES C-REGISTERS WIRD DADURCH NICHT VERAEENDERT.
1736 ;VORTEIL: BYTE-ERSPARNIS, DA SONST JEDEM "MVI C,.. " EIN SPRUNG ZUR
1737 ;"XERR"-ROUTINE FOLGEN MUESSTE.
1738 ;
1739 ; BEISPIEL: A) OE   EINSPRUNG BEI A) : DIE CPU "SIEHT" DAS BEFEHLSBYTE
1740 ;             00           OE "MVI C,.. " UND DEN OPERANDEN 00.
1741 ;             21           DIES ERGIBT: "MVI C,00". DAS FOLGENDE
1742 ;             B) OE           BYTE (21) NIMMT DIE CPU ALS BEFEHLS-
1743 ;             01           BYTE FUER "LXI H,....". DIE FOLGENDEN
1744 ;             21           BYTES (0E UND 01) ERGEREN "LXI H,010E"
1745 ;             .             FUER DIE CPU FOLGT NUN WIEDER EIN BE-
1746 ;             .             FEHLSBYTE (21). DAS BEDEUTET: ERNEUT
1747 ;             .             DER BEFEHL "LXI H,...." USW.
1748 ;             .   EINSPRUNG BEI B) : DIE CPU SIEHT DAS BEFHLSBYTE 0E
1749 ;             .             "MVI C,..". WEITERER ABLAUF WIE OBEN.
1750 ;
1751 ;WERTE BEI "JMP XERR":  C = FEHLER-CODE
1752 ;                       H = MAXIMAL MOEGLICHER FEHLER-CODE
1753 ;                       L = 0EH (CODE FUER "MVI C,..")
1754 ;
484F  OE00         1755 CODE0:      MVI    C,0          ;FALSCHE EINGABE
4851  21           1756             DB    21H
4852  OE01         1757 CODE1:      MVI    C,1          ;RUECKSTELL-FEHLER
4854  21           1758             DB    21H
4855  OE02         1759 CODE2:      MVI    C,2          ;SCHREIB-FEHLER
4857  21           1760             DB    21H
4858  OE03         1761 CODE3:      MVI    C,3          ;LAUFWERK NICHT BEREIT
485A  21           1762             DB    21H
485B  OE04         1763 CODE4:      MVI    C,4          ;DISKETTE SCHREIBGESCHUETZT
485D  21           1764             DB    21H
485E  OE05         1765 CODE5:      MVI    C,5          ;PRUEF-FEHLER
4860  21           1766             DB    21H
4861  OE06         1767 CODE6:      MVI    C,6          ;SUCH-FEHLER
4863  21           1768             DB    21H
4864  OE07         1769 CODE7:      MVI    C,7          ;LESE-FEHLER
4866  21           1770             DB    21H
4867  OE08         1771 CODE8:      MVI    C,8          ;DISKETTE VOLL
4869  21           1772             DB    21H
486A  OE09         1773 CODE9:      MVI    C,9          ;UNERLAUBTER NAME
486C  21           1774             DB    21H
1775 ;

```

LOC	OBJ	LINE	SOURCE STATEMENT
486D	0E0A	1776 CODE10:	MVI C,10 ;FILE > 65535 (DEZ.) BYTES
486F	21	1777	DB 21H
4870	0E0B	1778 CODE11:	MVI C,11 ;FILE NICHT IM VERZEICHNIS
4872	21	1779	DB 21H
4873	0E0C	1780 CODE12:	MVI C,12 ;FALSCHER FILE-TYP
4875	21	1781	DB 21H
4876	0E0D	1782 CODE13:	MVI C,13 ;DIRECTORY-FEHLER
4878	21	1783	DB 21H
4879	0E0E	1784 CODE14:	MVI C,14 ;SPEICHER-FEHLER
487B	21	1785	DB 21H
487C	0E0F	1786 CODE15:	MVI C,15 ;SPEICHERPLATZ AUF DER D. ZU KLEIN
487E	21	1787	DB 21H
487F	0E10	1788 CODE16:	MVI C,16 ;PROGRAMM-SPEICHER LEER
		1789 ;	
4881	C380FA	1790	JMP XERR ;SPRUNG UEBER RAM-VEKTOR
		1791	; (WIRD VOM BFZ-MINI-DOS ALS
		1792	;"JMP ERROR" INITIALISIERT.)
		1793	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1794	;*****
		1795	;PROGRAMM-SEGMENT "ERROR"
		1796	;WIRD BEI DOS-FEHLERN VOM CODE-TEIL AUFGERUFEN
		1797	;DIESELECTIERT LAUFWERK UND GIBT FEHLERMELDUNG AUS
		1798	;
		1799	;WERTE BEIM AUFRUF: C = FEHLER-CODE
		1800	;=====
		1801	;RETTE FEHLER-STELLE IM RAM UNTER "ERRP". WIRD VOM DOS NICHT GENUTZT.
		1802	;NUTZUNG DES "ERRP"-WERTES DURCH ANDERE PROGRAMME ABER MOEGLICH
		1803	;
4884	F3	1804	ERROR: DI
4885	E1	1805	POP H ;RETURN-ADR (FEHLER-STELLE)
4886	225AFA	1806	SHLD ERRP ;RETTE ADR
		1807	;=====
		1808	;GEBE "*** FEHLER:" AUS
		1809	;
4889	CD7300	1810	CALL PTXTCR ;PRINT TEXT
488C	0A	1811	DB LF,BELL,'*** FEHLER: ',00
488D	07		
488E	2A2A2A20		
4892	4645484C		
4896	45523A20		
489A	00		
		1812	;=====
		1813	;PRUEFE, OB FEHLER-CODE OK
		1814	;NEIN --> LADE C-REGISTER MIT SPEZIELLEN CODE FUER "UNGUELTIGER FEHLERCODE"
		1815	;
489E	2A60FA	1816	LHLD PEVT ;LADE POINTER AUF ERROR-VEKTOR-TABELLE
489E	7E	1817	MOV A,M ;LESE MAXIMALEN FEHLER-CODE
489F	E9	1818	CMP C ;VERGLEICHE MIT TATSAEHLICHEM CODE
48A0	I2A548	1819	JNC ECOK ;SPRINGE, WENN ERROR-CODE OK
48A3	4F	1820	MOV C,A ;* LADE CODE FUER:
48A4	0C	1821	INR C ;* "UNGUELTIGER FEHLER-CODE"
		1822	;=====
		1823	;LADE ZEIGER AUF FEHLERMELDUNG
		1824	;
48A5	23	1825	ECOK: INX H ;ZEIGER AUF 1. VEKTOR
48A6	0600	1826	MVI B,0 ;HIGH (BC) = 0
48A8	09	1827	DAI B ;* STELLE ZEIGER AUF
48A9	09	1828	DAD B ;* RICHTIGEN EINTRAG
48AA	4E	1829	MOV C,M ;* EINTRAG NACH BC
48AB	23	1830	INX H ;*
48AC	46	1831	MOV B,M ;*
		1832	;=====
		1833	;GEBE FEHLERMELDUNG AUS
		1834	;
48AD	0A	1835	ERROR1: LDAX B ;TEXT-ZEICHEN NACH A
48AE	E7	1836	ORA A ;TEXT ENDE ?
48AF	CAB948	1837	JZ ERROR2 ;JA --> ERROR2
48E2	CD5200	1838	CALL WCHAR ;PRINT ZEICHEN
48E5	03	1839	INX B ;STELLE ZEIGER WEITER
48E6	C3AD48	1840	JMP ERROR1 ;NAECHSTES ZEICHEN
		1841	

LOC	OBJ	LINE	SOURCE STATEMENT
		1842	;=====
		1843	;DESELEKT LAUFWERK. WARTEN AUF <SPACE> (FEHLER-QUITTIERUNG)
		1844	;
48B9	CD9E49	1845	ERROR2: CALL DESEL ;DESELECT LAUFWERK
48BC	CD7300	1846	CALL PTXTCR ;PRINT TEXT
48BF	3D3D3E20	1847	DB '==> SPACE',00
48C3	53504143		
48C7	45		
48C8	00		
48C9	CDEF4D	1848	CALL WAITSP ;WARTE AUF SPACE
		1849	;=====
		1850	;DESELEKT LAUFWERK. RE-INITIALISIERE SP. RETURN (ERRRET)
		1851	;
		1852	;
48CC	CD9E49	1853	ERET: CALL DESEL ;DESELECT LAUFWERK
48CF	3132FC	1854	LXI SP,0FC32H ;RE-INIT SP
		1855	;
48D2	2A58FA	1856	LHLD ERRRET ;"RETURN"-ADRESSE (WURDE VOM PROGRAMM
48D5	E9	1857	PCHL ;BEI "ERRRET" EINGESETZT)
		1858	;
		1859	;



BFZ-MINI-IDS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		1860	;*****
		1861	;*
		1862	;*                                    U N T E R P R O G R A M M E                                   *
		1863	;*
		1864	;*****
		1865	;
		1866	;UNTERPROGRAMM "RESTORE"
		1867	;GIBT DAS KOMMANDO "RESTORE" (FAHRE KOPF NACH TRACK 0) AN DEN FDC.
		1868	;SACKGASSEN-UNTERPROGRAMM. WIRD DURCH DEN INTERRUPT VERLASSEN, DEN
		1869	;DER FDC NACH DER AUSFUEHRUNG DES KOMMANDOS ERZEUGT.
		1870	;
		1871	;WERTE BEIM AUFRUF: ---
		1872	;WERTE NACH RETURN: SIEHE INTERRUPT-SERVICE-ROUTINE ISR
		1873	;
48D6	3E00	1874	RESTORE:           MVI     A,CREST                           ;RESTORE
48D8	D3C0	1875	OUT     CMD                            ;KOMMANDO AN FDC
48DA	76	1876	HLT                                    ;WARTE AUF INTERRUPT
		1877	;
		1878	;*****
		1879	;UNTERPROGRAMM "STEPIN"
		1880	;GIBT AN DEN FDC DEN BEFEHL "STEP IN" AUS.
		1881	;SACKGASSEN-UNTERPROGRAMM. WIRD PER INTERRUPT VERLASSEN
		1882	;
		1883	;WERTE BEIM AUFRUF: ---
		1884	;WERTE NACH INTERRUPT: SIEHE INTERRUPT-SERVICE-ROUTINE ISR
		1885	;
48DB	3E50	1886	STEPIN:           MVI     A,CSTPIN                   ;STEP IN
48DD	D3C0	1887	OUT     CMD
48DF	76	1888	HLT                                   ;WARTE AUF INTERRUPT
		1889	;
		1890	;*****
		1891	;UNTERPROGRAMM "SEEK"
		1892	;GIBT AN DEN FDC DEN BEFEHL "SEEK" AUS.
		1893	;SACKGASSEN-UNTERPROGRAMM. WIRD PER INTERRUPT VERLASSEN
		1894	;
		1895	;WERTE BEIM AUFRUF: SOLL-TRACK IM FDC-DATENREGISTER
		1896	;WERTE NACH INTERRUPT: SIEHE INTERRUPT-SERVICE-ROUTINE ISR
		1897	;
48E0	3E14	1898	SEEK:             MVI     A,CSEEK                   ;SEEK
48E2	D3C0	1899	OUT     CMD
48E4	76	1900	HLT                                   ;WARTE AUF INTERRUPT
		1901	;
		1902	;

LOC	OBJ	LINE	SOURCE STATEMENT
		1903	;*****
		1904	;UNTERPROGRAMM "WSEC"
		1905	;SCHREIBT EINEN SEKTOR AUF DISKETTE. NACH DER AUSGABE DES
		1906	; "WRITE SECTOR"-KOMMANDOS AN DEN FDC HAELT SICH DIE CPU DURCH
		1907	; "OUT STOP" AN (READY GEHT AUF L-PEGEL). WENN DER FDC DAS ERSTE
		1908	; DATEN-BYTE ANFORDERT, GIBT ER AM ANSCHLUSS DRQ EIN SIGNAL AUS, DAS
		1909	; DIE CPU WIEDER STARTET (READY GEHT AUF H). DIE CPU GIBT DAS
		1910	; DATEN-BYTE AN DEN FDC UND HAELT SICH ERNEUT AN.
		1911	; DURCH DIE STEUERUNG UEBER DAS READY-SIGNAL ENTFAELT DAS ABFRAGEN
		1912	; DES FDC-STATUS. DIE CPU KANN SCHNELLER AUF EINE DATEN-ANFORDERUNG
		1913	; (DRQ, DATA REQUEST) DES FDC REAGIEREN. SO IST TROTZ DER RELATIV
		1914	; NIEDRIGEN CPU-TAKTFREQUENZ VON 2 MHZ DOUBLE-DENSITY-BETRIEB MOEGlich.
		1915	;
		1916	;SACKGASSEN-UNTERPROGRAMM. WIRD PER INTERRUPT VERLASSEN
		1917	;
		1918	;WERTE BEI AUFRUF: HL=ZEIGT AUF DAS ERSTE BYTE DER ZU SCHREIBENDEN DATEN
		1919	; (TRACK BEREITS ANGEFAHREN, SEKTOR-NUMMER IM SEKTOR-REG.)
		1920	;
		1921	;WERTE NACH INTERRUPT: HL=VERAENDERT
		1922	; (SIEHE AUCH: INTERRUPT-SERVICE-ROUTINE ISR)
		1923	;
48E5	3EA0	1924	WSEC: MVI A,CWSEC ;WRITE SECTOR
48E7	D3C0	1925	OUT CMD ;BEFEHL AN FDC
48E9	D3C8	1926	WSECO: OUT STOP ;WARTE AUF DRQ
48EB	7E	1927	MOV A,M ;ZEICHEN IN AKKU
48EC	D3C3	1928	OUT DAT ;ZEICHEN AN FDC
48EE	23	1929	INX H ;ZEIGER WEITERSTELLEN
48EF	C3E948	1930	JMP WSECO ;NAECHSTES ZEICHEN
		1931	;
		1932	;

LOC	OBJ	LINE	SOURCE STATEMENT
		1933	;*****
		1934	;UNTERPROGRAMM "RSEC"
		1935	;Liest einen Sektor von Diskette. nach der Ausgabe des "READ SECTOR"-
		1936	;Kommandos an den FIC haelt sich die CPU durch "OUT STOP" an (READY
		1937	;geht auf L-Pegel). Wenn der FIC das erste Daten-Byte von der Diskette
		1938	;gelesen hat, gibt er am Anschluss DRQ ein Signal aus, das die CPU wieder
		1939	;startet (READY geht auf H). Die CPU liest das Daten-Byte aus dem FIC-
		1940	;Datenregister und haelt sich erneut an.
		1941	;durch die Steuerung ueber das Ready-Signal entfaellt das Abfragen
		1942	;des FIC-Status. Die CPU kann schneller auf eine Daten-Anforderung
		1943	; (IRQ, DATA REQUEST) des FIC reagieren. So ist trotz der relativ
		1944	;niedrigen CPU-Taktfrequenz von 2 MHz Double-Density-Betrieb moeglich.
		1945	;
		1946	;Sackgassen-Unterprogramm. Wird per Interrupt verlassen
		1947	;
		1948	;Werte bei Aufruf: HL=zeigt auf Anfang des Speicherbereichs, in dem die
		1949	;gelesenen Bytes gespeichert werden sollen.
		1950	; (Track bereits angefahren, Sektor-Nummer im
		1951	;Sektor-Register)
		1952	;
		1953	;Werte nach Interrupt: HL=veraendert
		1954	; (siehe auch: Interrupt-Service-Routine ISR)
		1955	;
48F2	3E80	1956	RSEC: MVI A,CRSEC ;READ SECTOR
48F4	D3C0	1957	OUT CMD ;BEFEHL AN FIC
48F6	D3C8	1958	RSEC0: OUT STOP ;WARTE AUF DRQ
48F8	DBC3	1959	IN DAT ;ZEICHEN IN AKKU
48FA	77	1960	MOV M,A ;ZEICHEN IN BUFFER
48FB	23	1961	INX H ;ZEIGER WEITERSTELLEN
48FC	C3F648	1962	JMP RSEC0 ;NAECHSTES ZEICHEN
		1963	;
		1964	;

LOC	OBJ	LINE	SOURCE STATEMENT
		1965	*****
		1966	;UNTERPROGRAMM "WTRK"
		1967	;SCHREIBT EINEN TRACK AUF DIE DISKETTE. NACH DER AUSGABE DES
		1968	; "WRITE TRACK"-KOMMANDOS AN DEN FDC HAELT SICH DIE CPU DURCH
		1969	; "OUT STOP" AN (READY GEHT AUF L-PEGEL). WENN DER FDC DAS ERSTE
		1970	; DATEN-BYTE ANFORDERT, GIBT ER AM ANSCHLUSS DRQ EIN SIGNAL AUS, DAS
		1971	; DIE CPU WIEDER STARTET (READY GEHT AUF H). DIE CPU GIBT DAS
		1972	; DATEN-BYTE AN DEN FDC UND HAELT SICH ERNEUT AN.
		1973	; DURCH DIE STEUERUNG UEBER DAS READY-SIGNAL ENTFAELT DAS ABFRAGEN
		1974	; DES FIC-STATUS. DIE CPU KANN SCHNELLER AUF EINE DATEN-ANFORDERUNG
		1975	; (DRQ, DATA REQUEST) DES FDC REAGIEREN. SO IST TROTZ DER RELATIV
		1976	; NIEDRIGEN CPU-TAKTFREQUENZ VON 2 MHZ DOUBLE-DENSITY-BETRIEB MOEGLICH.
		1977	;
		1978	; DIES IST EIN SACKGASSEN-UNTERPROGRAMM. WENN DER FDC DEN BEFEHL
		1979	; "WRITE TRACK" ABGEBEARBEITET HAT, ERZEUGT ER EIN INTERRUPT-SIGNAL
		1980	; DAS UNTERPROGRAMM WIRD DURCH DIESEN INTERRUPT VERLASSEN
		1981	;
		1982	; WERTE BEIM AUFRUF: TRACK BEREITS ANGEFAHREN
		1983	; WERTE NACH INTERRUPT: HL=VERAENDERT
		1984	; (SIEHE AUCH: INTERRUPT-SERVICE-ROUTINE ISR)
		1985	;
48FF	2100E0	1986	WTRK: LXI H,BUFFER ;ZEIGER AUF TRACK BUFFER
4902	CD8B49	1987	CALL SELECT ;SELECT LAUFWERK
4905	3EF0	1988	MVI A,CWRTRK ;WRITE TRACK
4907	D3C0	1989	OUT CMD
4909	D3C8	1990	OUT STOP ;WARTE BIS DRQ
490B	7E	1991	MOV A,M ;LESE DATE AUS BUFFER
490C	D3C3	1992	OUT DAT ;GEBE DATE AN FDC
490E	23	1993	INX H ;STELLE BUFFER-ZEIGER WEITER
490F	C30949	1994	JMP WARTE
		1995	;
		1996	*****
		1997	;UNTERPROGRAMM "WVSEC"
		1998	;SCHREIBT DEN INHALT DES SEKTOR-BUFFERS (SECB) AUF DISKETTE UND
		1999	;PRUEFT DEN GESCHRIEBENEN SEKTOR.
		2000	;BEIM AUFRUF MUESSEN TRACK- UND SEKTOR-NUMMER BEREITS AN DEN FDC
		2001	;UEBERGEBEN WORDEN SEIN. DER KOPF MUSS SICH UEBER DER RICHTIGEN SPUR
		2002	;BEFINDEN.
		2003	;
		2004	;KEIN FEHLER: Z-FLAG = 1, C = 5
		2005	; FEHLER: Z-FLAG = 0, C = 5 (VERIFY-ERROR)
		2006	; C = 2 (WRITE-ERROR)
		2007	;
4912	2100F8	2008	WVSEC: LXI H,SECB ;ZEIGER AUF SEKTOR-BUFFER
4915	CD8B49	2009	WVSECO: CALL SELECT ;RE-SELECT DRIVE
4918	CDE548	2010	CALL WSEC ;SCHREIBE SEKTOR
491B	0E02	2011	MVI C,2 ;CODE FUER "SCHREIB-FEHLER"
491D	E6DC	2012	ANI WVSEC ;ALLES OK ?
491F	C0	2013	RNZ ;NEIN --> RETURN
4920	CD7549	2014	CALL VERIX ;VERIFY SEKTOR
4923	0E05	2015	MVI C,5 ;CODE FUER "PRUEF-FEHLER"
4925	C9	2016	RET
		2017	

LOC	OBJ	LINE	SOURCE STATEMENT
		2018	;*****
		2019	;UNTERPROGRAMM "RWVSEC"
		2020	;SCHREIBT EINEN SEKTOR AUF DISKETTE UND PRUEFT DEN GESCHRIEBENEN SEKTOR.
		2021	;BEI FEHLERN WIRD DER SCHREIBVORGANG "MAXTRY" MAL WIEDERHOLT.
		2022	;
		2023	;WERTE BEIM AUFRUF: BEIM AUFRUF MUESSEN TRACK- UND SEKTOR-NUMMER BEREITS
		2024	; AN DEN FDC UEBERGEHEN WORDEN SEIN. DER KOPF MUSS SICH
		2025	; UEBER DER RICHTIGEN SPUR BEFINDEN.
		2026	;
		2027	; HL= SEKTOR-PUFFER-ZEIGER
4926	1603	2028	RWVSEC: MVI D,MAXTRY ;MAX ANZAHL DER SCHREIB-VERSUCHE
4928	224CFA	2029	SHLD ZWSP3 ;RETTE ZEIGER
492B	CD8B49	2030	RWVSO: CALL SELECT ;RE-SELECT DRIVE
492E	2A4CFA	2031	LHLD ZWSP3 ;LESE ZEIGER
4931	CD1549	2032	CALL WVSECO ;SCHREIBE SEKTOR
4934	C8	2033	RZ ;RETURN, WENN OK
4935	15	2034	DCR D ;WEITERE VERSUCHE ERLAUBT ?
4936	C22B49	2035	JNZ RWVSO ;JA --> RWVSO
4939	CD5548	2036	CALL CODE2 ;NEIN --> CODE2
		2037	;*****
		2038	;UNTERPROGRAMM "SAVSEC"
		2039	;SCHREIBT DEN INHALT DES SEKTOR-PUFFERS (SECB) AUF DISKETTE UND PRUEFT
		2040	;DEN GESCHRIEBENEN SEKTOR.
		2041	;BEI FEHLERN WIRD DER SCHREIBVORGANG "MAXTRY" MAL WIEDERHOLT.
		2042	;
		2043	;WERTE BEIM AUFRUF: BEIM AUFRUF MUESSEN TRACK- UND SEKTOR-NUMMER BEREITS
		2044	; AN DEN FDC UEBERGEHEN WORDEN SEIN. DER KOPF MUSS SICH
		2045	; UEBER DER RICHTIGEN SPUR BEFINDEN.
		2046	;
493C	1603	2047	SAVSEC: MVI D,MAXTRY ;ANZAHL DER VERSUCHE
493E	CD8B49	2048	NTRY: CALL SELECT ;RE-SELECT DRIVE
4941	CD1249	2049	CALL WVSEC ;SPEICHERE SEKTOR AUF DISK
4944	C8	2050	RZ ;RETURN, WENN ALLES OK
4945	15	2051	DCR D ;WEITERE VERSUCHE ERLAUBT ?
4946	C23E49	2052	JNZ NTRY ;JA --> NTRY
4949	CD5548	2053	CALL CODE2 ;NEIN --> CODE2
		2054	;*****
		2055	;UNTERPROGRAMM "RRS"
		2056	;LIEST EINEN SEKTOR IN DEN SEKTOR-PUFFER (SECB).
		2057	;BEI FEHLERN WIRD DER LESEVORGANG "MAXTRY" MAL WIEDERHOLT.
		2058	;
		2059	;WERTE BEIM AUFRUF: BEIM AUFRUF MUESSEN TRACK- UND SEKTOR-NUMMER BEREITS
		2060	; AN DEN FDC UEBERGEHEN WORDEN SEIN. DER KOPF MUSS SICH
		2061	; UEBER DER RICHTIGEN SPUR BEFINDEN.
		2062	;
494C	1603	2063	RRS: MVI D,MAXTRY ;MAXIMALE ANZAHL DER LESE-VERSUCHE
494E	CD8B49	2064	RRSO: CALL SELECT ;RE-SELECT DRIVE
4951	2100F8	2065	LXI H,SECB ;BUFFER-ADR
4954	CDF248	2066	CALL RSEC ;LESE SEKTOR
4957	E69C	2067	ANI MRSEC ;ALLES OK ?
4959	C8	2068	RZ ;JA --> RETURN
495A	15	2069	DCR D ;NOCH EIN VERSUCH ERLAUBT ?
495B	C24E49	2070	JNZ RRSO ;JA --> RRSO
495E	CD6448	2071	CALL CODE7 ;NEIN --> LESE-FEHLER
		2072	;



LOC	OBJ	LINE	SOURCE STATEMENT
		2118	;*****
		2119	;UNTERPROGRAMM "VERI1"
		2120	;LIEST EINEN SEKTOR EIN OHNE IHN IM RAM ZU SPEICHERN. WIRD ZUM PRUEFEN
		2121	;DER SEKTOREN BENUTZT. WICHTIG SIND DIE STATUS-FLAGS "RECORD NOT FOUND"
		2122	;UND "CRC-ERROR" DES FIC NACH DEM EINLESEN.
		2123	;SACKGASSEN-UNTERPROGRAMM. WIRD DURCH DEN INTERRUPT VERLASSEN, DEM DER FIC
		2124	;NACH DEM EINLESEN DES SEKTORS ERZEUGT.
		2125	;ZUM VERSTAENDNIS VON "OUT STOP": SIEHE WTRK (WRITE TRACK)
		2126	;
		2127	;WERTE BEIM AUFRUF: SEKTOR- UND TRACK-NUMMER BEREITS AN FIC UEBERGEHEN
		2128	; TRACK ANGEFAHREN
		2129	;WERTE NACH RETURN: SIEHE INTERRUPT-SERVICE-ROUTINE ISR
		2130	;
497B	3E80	2131	VERI1: MVI A,CRSEC ;READ SECTOR
497D	I3C0	2132	OUT CMD ;GEBE KOMMANDO AUS
497F	I3C8	2133	VERI2: OUT STOP ;WARTE BIS DRQ
4981	I8C3	2134	IN DAT ;LESE DATE (RESET DRQ)
4983	C37F49	2135	JMP VERI2 ;LOOP
		2136	;
		2137	;*****
		2138	;UNTERPROGRAMM "COMPSM"
		2139	;BERECHNET DIE SELECT-MASKE UND SPEICHERT SIE IM SELMSK AB
		2140	;
		2141	;WERTE BEIM AUFRUF: A = 0 (LAUFWERK A) BZW. 1 (LAUFWERK B)
		2142	;WERTE NACH RETURN: A = 1 (LAUFWERK A) BZW. 2 (LAUFWERK B)
		2143	;
4986	3C	2144	COMPSM: INR A ;ERGIBT SELECT-MASKE
4987	3285FA	2145	STA SELMSK ;RETTE SELECT-MASKE
498A	C9	2146	RET
		2147	;
		2148	;

```

LOC  OBJ      LINE      SOURCE STATEMENT
2149 ;*****
2150 ;UNTERPROGRAMM "SELECT"
2151 ;WAHLT LAUFWERK UND SEITE AUS. DAS MOTOR-ON-OFF AUF DER FDC-KARTE
2152 ;WIRD DURCH "OUT PORT" FUER EWTWA 3 SEC. GESETZT.
2153 ;
2154 ;WERTE BEIM AUFRUF: (SEITE) = SEITENNUMMER 0 ODER 1 (BINAER)
2155 ;                      (SELMSK) = SELECT-MASKE - LAUFWERK A = 1
2156 ;                      - LAUFWERK B = 2
2157 ;                      SDBA *)
2158 ; WERTE NACH RETURN: A = 0000001B, 01H (LAUFWERK A, SEITE 0, DOUBLE DENSITY)
2159 ;                      A = 00001001B, 09H (LAUFWERK A, SEITE 1, DOUBLE DENSITY)
2160 ;                      A = 00000010B, 02H (LAUFWERK B, SEITE 0, DOUBLE DENSITY)
2161 ;                      A = 00001010B, 0AH (LAUFWERK B, SEITE 1, DOUBLE DENSITY)
2162 ;
2163 ; *) S =      SEITE, 0 = SEITE 0
2164 ;           1 = SEITE 1
2165 ;       D =      DENSITY, 0 = DOUBLE DENSITY
2166 ;           1 = SINGLE DENSITY
2167 ;       B = LAUFWERK B, (WENN 1)
2168 ;       A = LAUFWERK A, (WENN 1)
2169 ;
2170 ; DER AUSGEBEBENE WERT WIRD IM RAM UNTER "LSEL" GESPEICHERT.
2171 ;
498B C5      2172 SELECT:      PUSH      B          ;RETTE B-REGISTER
498C 3A84FA  2173             LDA      SEITE      ;LESE SEITEN-NUMMER
498F 87      2174             ADD      A          ;* SCHIEBE BIT IN POSITION D1
4990 87      2175             ADD      A          ;*                      D2
4991 87      2176             ADD      A          ;*                      D3
4992 47      2177             MOV      B,A        ;RETTE BIT IN B
4993 3A85FA  2178             LDA      SELMSK     ;LESE SELECT-MASKE
4996 B0      2179             ORA      B          ;SETZE SEITEN-BIT
4997 D3C4    2180             OUT      PORT        ;SELECT LAUFWERK
4999 3286FA  2181             STA      LSEL       ;RETTE AUSGABEWERT
499C C1      2182             POP      B          ;RESTORE B-REGISTER
499D C9      2183             RET
2184 ;
2185 ;*****
2186 ;UNTERPROGRAMM "DESEL"
2187 ;DESELEKTIERT LAUFWERK.
2188 ;IST DER LETZTE AUSGABEWERT (LSEL) = 00H ERFOLGT SOFORT EIN RETURN
2189 ;(IN DIESEM FALL IST KEIN LAUFWERK AKTIV). DIES VERHINDERT DAS ER-
2190 ;NEUTE ANLAUFEN DER MOTOREN.
2191 ;
2192 ;IST "LSEL" UNGLEICH 00H, SO WIRD 00H AUSGEBEBEN. DIESER WERT WIRD UNTER
2193 ;"LSEL" GESPEICHERT.
2194 ;
2195 ;NACH RETURN: A=00
2196 ;
499E 3A86FA  2197 DESEL:      LDA      LSEL       ;LETZTER AUSGABEWERT
49A1 B7      2198             ORA      A          ;NULL ?
49A2 C8      2199             RZ          ;JA --> RETURN
49A3 AF      2200             XRA      A          ;* DESELECT LAUFWERK
49A4 D3C4    2201             OUT      PORT        ;*
49A6 3286FA  2202             STA      LSEL       ;RETTE AUSGABEWERT
49A9 C9      2203

```



LOC	OBJ	LINE	SOURCE STATEMENT
		2204	;*****
		2205	;UNTERPROGRAMM "DREADY"
		2206	;DAS UNTERPROGRAMM GIBT DEM MOTOR DES DISKETTEN-LAUFWERKS 1 SEKUNDE
		2207	;ZEIT UM SEINE NENNDREHZAHLE VON 300 UPM ZU ERREICHEN. DANACH WIRD DAS
		2208	;NOT-READY-BIT IM FIC-STATUS-REGISTER ABGEFRAGT. DAS UNTERPROGRAMM
		2209	;WIRD UEBER DEN "RET"-BEFEHL VERLASSEN. DAS CARRY-FLAG GIBT DEN
		2210	;LAUFWERK-ZUSTAND AN:
		2211	;
		2212	;
		2213	NICHT BEREIT --- CARRY = 1 (GESETZT)
		2214	BEREIT --- CARRY = 0 (NICHT GESETZT)
		2215	; WERTE BEIM AUFRUF: ---
		2216	;
		2217	; WERTE NACH RETURN: A = UNBESTIMMT
		2218	HL = 0000
		2219	CARRY-FLAG: SIEHE OBEN
		2220	;
49AA	21E803	2221	DREADY:      LXI   H,1000          ;* EINE SEKUNDE VERZOEGERUNG
49AD	C0B449	2222	CALL  DELAY          ;* (MOTOR HOCHLAUF-ZEIT)
49B0	0B0C0	2223	IN    STAT          ;LESE STATUS
49B2	07	2224	RLC                  ;READY ?
49B3	C9	2225	RET                  ;RETURN (CARRY = 0 --)      BEREIT)
		2226	;          (CARRY = 1 --) NICHT BEREIT)
		2227	;
		2228	;*****
		2229	;UNTERPROGRAMM "DELAY"
		2230	;VERZOEGERUNGSSCHLEIFE. LAUFZEIT ETWA HL*1 MS (BEI 2 MHZ TAKT)
		2231	;
		2232	;WERTE BEIM AUFRUF: HL = LAUFZEIT IN MILLISEKUNDEN
		2233	;WERTE NACH RETURN: A = 0
		2234	HL = 0000
		2235	;
49B4	E5	2236	DELAY:      PUSH  H              ;RETTE ZAEHLER
49B5	215300	2237	LXI   H,83          ;
49B8	2B	2238	DELAY0:      DCX   H              ;* LAUFZEIT ETWA 1 MS
49B9	7C	2239	MOV   A,H          ;* BEI 2 MHZ TAKT
49BA	B5	2240	ORA   L              ;*
49BB	C2B849	2241	JNZ   DELAY0      ;*
49BE	E1	2242	POP   H              ;RESTORE ZAEHLER
49BF	2B	2243	DCX   H              ;VERMINDERE ZAEHLER UM EINS
49C0	7C	2244	MOV   A,H          ;* HL=0000 ?
49C1	B5	2245	ORA   L              ;*
49C2	C2B449	2246	JNZ   DELAY      ;NEIN -->DELAY
49C5	C9	2247	RET
		2248	;
		2249	;

LOC	OBJ	LINE	SOURCE STATEMENT
		2250	*****
		2251	;UNTERPROGRAMM "GETPUT"
		2252	;SPEICHERT / LIEST FILES
		2253	;FUNKTION WIRD DURCH RAM-VEKTOREN FESTGELEGT. DAS DOS INITIALISIERT
		2254	;DIE VEKTOREN WIE FOLGT:
		2255	;
		2256	; VEKTOR LOAD SAVE
		2257	; XGP1 CALL RRS CALL RWVSEC
		2258	; XGP2 CALL MOVE RET
		2259	; XGP3 CALL PPLUS CALL PPLUS
		2260	;
		2261	;WERTE BEIM AUFRUF: LAENGE = FILE-LAENGE
		2262	; ZWSP4 = ZEIGER AUF TABELLE DER NUTZBAREN DIR-EINTR
		2263	; (SIEHE AUCH UNTERPROGRAMM "SEINTR".)
		2264	; ZWSP1 = START-ADR (NUR FUER WRITE)
		2265	; SELMSK = SELECT-MASKE DES AUSGEW. LAUFWERKS
		2266	;
49C6	2A00FA	2267	GETPUT: LHLI LAENGE ;* REST = LAENGE
49C9	2250FA	2268	SHLD REST ;*
		2269	;
49CC	2A4EFA	2270	GETPO: LHLI ZWSP4 ;ZEIGER AUF TABELLEN-POINTER
49CF	CD564A	2271	CALL GETPOS ;LESE POSITION AUS TABELLE
49D2	CD294A	2272	CALL POS ;POSITIONIERE KOFF
		2273	;
49D5	0600	2274	MVI B,0 ;SEKTOR 1 MINUS EINS
49D7	2A48FA	2275	GNPXT: LHLI ZWSP1 ;ANFANG DIESES TEILS (NUR FUER WRITE)
49DA	04	2276	INR B ;NAECHSTER SEKTOR
49DB	78	2277	MOV A,B ;* SEKTOR-NUMMER AN FIC
49DC	D3C2	2278	OUT SEC ;*
49DE	CD71FA	2279	CALL XGP1 ;LOAD: LESE SEKTOR
		2280	;SAVE: SCHREIBE SEKTOR
		2281	;
49E1	2248FA	2282	GPOK: SHLD ZWSP1 ;RETTE ZEIGER (NUR FUER WRITE)
49E4	CD74FA	2283	CALL XGP2 ;LOAD: MOVE SEKTOR-BUFFER-INHALT
		2284	; ZUM ZIEL-SPEICHER
		2285	;SAVE: NOP
49E7	2A50FA	2286	LHLI REST ;REST-LAENGE
49EA	110002	2287	LXI D,512 ;SEKTOR-LAENGE
49ED	CD3910	2288	CALL SUB2 ;REST-LAENGE MINUS SEKTOR-LAENGE
49F0	DA074A	2289	JC GPO ;REST < 0000 --> GPO
49F3	7D	2290	MOV A,L ;: REST=0000 ?
49F4	B4	2291	ORA H ;:
49F5	CA074A	2292	JZ GPO ;REST = 0000 --> GPO
49F8	2250FA	2293	SHLD REST ;RETTE NEUE REST-LAENGE
49FB	78	2294	MOV A,B ;AKTUELLE SEKTOR-NUMMER NACH A
49FC	FE08	2295	CPI B ;SEKTOR-NUMMER = 8
49FE	C2D749	2296	JNZ GPNXT ;NEIN --> NAECHSTER SEKTOR,
		2297	; GLEICHE SPUR
4A01	CD77FA	2298	CALL XGP3 ;LOAD, SAVE: PRINT "+"
4A04	C3CC49	2299	JMP GETPO ;NAECHSTER EINTRAG
		2300	;
4A07	CD77FA	2301	GPO: CALL XGP3 ;LOAD, SAVE: PRINT "+"
4A0A	C9	2302	RET
		2303	

LOC	OBJ	LINE	SOURCE STATEMENT
		2304	;*****
		2305	;UNTERPROGRAMM "POSDIR"
		2306	;UEBERGIBT DEM FDC DIE DATEN ZUR KOPF-POSITIONIERUNG UEBER DER
		2307	;DIRECTORY. IN "SEMSK" MUSS DIE SELECT-MASKE DES GEWUENSCHTEN LAUFWERKS
		2308	;STEHEN. IST DAS LAUFWERK NACH 1 SEKUNDE NICHT BEREIT, VERZWEIGT DAS
		2309	;PROGRAMM NACH "CODE3" (LAUFWERK NICHT BEREIT). SONST GEHT DAS PROGRAMM
		2310	;OHNE RETURN IN DAS UNTERPROGRAMM "WPTST" UEBER. (SIEHE AUCH DORT).
		2311	;
4A0B	AF	2312	POSDIR: XRA A ;A=0
4A0C	3284FA	2313	STA SEITE ;SEITE 0
4A0F	CD8B49	2314	CALL SELECT ;SELECT LAUFWERK
4A12	CDAA49	2315	CALL DREADY ;LAUFWERK BEREIT ?
4A15	DC5848	2316	CC CODE3 ;NEIN --> CODE3
		2317	;*****
		2318	;UNTERPROGRAMM "WPTST"
		2319	;DAS UNTERPROGRAMM STELLT DEN KOPF DES SELEKTIERTEN LAUFWERKS UEBER
		2320	;SPUR 0 UND PRUEFT, OB DIE DARIN ENTHALTENE DISKETTE SCHREIBGESCHUETZT IST.
		2321	;TRETEN BEIM RESTORE-BEFEHL FEHLER AUF, VERZWEIGT DAS PROGRAMM NACH "CODE1"
		2322	;
		2323	;WERTE NACH RETURN: Z-FLAG = 0 (NICHT GESETZT): DISKETTE SCHREIBGESCHUETZT
		2324	; Z-FLAG = 1 ( GESETZT ): DISKETTE NICHT SCHR.-GESCHUETZT
		2325	;
4A18	CD2E4E	2326	WPTST: CALL INTINT ;INTERRUPT-ROUTINE INITIALISIEREN
4A1B	CD1648	2327	CALL RESTORE ;KOPF AUF SPUR NULL
4A1E	E69C	2328	ANI 10011100B ;* OK ?
4A20	FE04	2329	CPI 00000100B ;*
4A22	C45248	2330	CMZ CODE1 ;NEIN --> CODE1
4A25	7B	2331	MOV A,E ;RESTORE STATUS
4A26	E640	2332	ANI 01000000B ;WRITE PROTECTED ?
4A2B	C9	2333	RET ;Z-FLAG = 0 --> JA
		2334	;Z-FLAG = 1 --> NEIN
		2335	

LOC	OBJ	LINE	SOURCE STATEMENT
		2336	*****
		2337	;UNTERPROGRAMM "POS"
		2338	;DAS UNTERPROGRAMM POSITIONIERT DEN KOPF ENTSPRECHEND DEN WERTEN IM
		2339	;B- UND C-REGISTER UEBER EINEN FILE.
		2340	;
		2341	;WERTE BEIM AUFRUF: B=NUMMER DES DIRECTORY-SEKTORS IN DEM DER FILE-
		2342	EINTRAG STEHT
		2343	; C=RELATIVE NUMMER DES EINTRAGS INNERHALB DES DIR.-
		2344	SEKTORS
		2345	;
		2346	; (B- UND C-WERTE: SIEHE AUCH UNTERPROG. "SEINTR")
		2347	;
		2348	; "SELMSK" MUSS DIE SELECT-MASKE DES AUSGEWAELHTEN
		2349	LAUFWERKS ENTHALTEN
		2350	;
4A29	60	2351	POS: MOV H,B ;SEKTOR-NUMMER NACH H
4A2A	69	2352	MOV L,C ;EINTRAGS-NUMMER NACH L
4A2B	CD404A	2353	CALL CTS ;BERECHNE TRACK UND SEITE
4A2E	CD8B49	2354	CALL SELECT ;SELEKTIERE LAUFWERK UND SEITE
4A31	3A52FA	2355	LDA SPUR ;* POSITIONIERE KOPF AUF SPUR
4A34	D3C3	2356	OUT DAT ;*
4A36	CDE048	2357	CALL SEEK ;*
4A39	E698	2358	ANI 10011000B ;SEEK ERROR ?
4A3B	C46148	2359	CNZ CODE6 ;JA --> CODE6
4A3E	7B	2360	MOV A,E ;RESTORE STATUS
4A3F	C9	2361	RET
		2362	;
		2363	

LOC	OBJ	LINE	SOURCE STATEMENT
		2364	;*****
		2365	;UNTERPROGRAMM "CTS" (COMPUTE TRACK AND SIDE)
		2366	;DIESES UNTERPROGRAMM ERMITTELT DIE SPUR (TRACK) UND DIE DISKETTENSEITE,
		2367	;AUF DER SICH EIN FILE (BZW. EIN 4K-BLOCK DAVON) BEFINDET.
		2368	;BEIM AUFRUF DIESES UNTERPROGRAMMS MUESSEN DIE REGISTER B UND C FOLGENDE
		2369	;ANGABEN ENTHALTEN:
		2370	;
		2371	;B-REGISTER: NUMMER DES DIRECTORY-SEKTORS, IN DEM DER EINTRAG ZU DEM FILE
		2372	; STEHT (1...5)
		2373	;C-REGISTER: RELATIVE NUMMER DES EINTRAGS IM DIRECTORY-SEKTOR (1..16)
		2374	;
		2375	;DIE ERRECHNETEN WERTE FUER SPUR UND SEITE WERDEN UNTER "SPUR" BZW. "SEITE"
		2376	;IM RAM GESPEICHERT.
		2377	;
		2378	;NACH RETURN SIND DIE INHALTE FOLGENDER REGISTER VERAENDERT:
		2379	;
		2380	; A=(SPUR)
		2381	; B=00
		2382	;
4A40	3EF0	2383	CTS: MVI A,-16 ;* BERECHNE ABSOLUTE
4A42	C610	2384	ADDSEC: ADI 16 ;* NUMMER DES EINTRAGS
4A44	05	2385	DCR B ;* (1 ... 79)
4A45	C2424A	2386	JNZ ADDISEC ;*
4A48	81	2387	ADD C ;*
		2388	;
4A49	F5	2389	PUSH PSW ;RETTE EINTRAGS-NUMMER (1 ... 79)
4A4A	E601	2390	ANI 0000001B ;MASKIERE SEITEN-BIT
		2391	; ( GERADE EINTRAGS-NUMMER = SEITE 0,
		2392	; UNGERADE EINTRAGS-NUMMER = SEITE 1)
4A4C	3284FA	2393	STA SEITE ;RETTE SEITEN-NUMMER
4A4F	F1	2394	POP PSW ;RESTORE EINTRAGS-NUMMER (1 ... 79)
4A50	K7	2395	ORA A ;CLEAR CARRY
4A51	1F	2396	RAR ;* BERECHNE RELATIVE SPUR-NUMMER
		2397	;* (0 ... 39) AUF DER SEITE
4A52	3252FA	2398	STA SPUR ;RETTE SPUR-NUMMER
4A55	C9	2399	RET
		2400	;
		2401	;

```

LOC OBJ      LINE      SOURCE STATEMENT
2402 ;*****
2403 ;UNTERPROGRAMM "GETPOS"
2404 ;DAS BFZ-MINI-DOS TEILT FILES IN 4K-BYTE-BLOECKE EIN. FUER JEDEN DIESER
2405 ;BLOECKE WIRD EIN DIRECTORY-EINTRAG BENOETIGT. DIE POSITIONEN DER EINTRAEGE
2406 ;EINES FILES KANN MAN AUCH EIN EINER TABELLE ANGEBEN. SOLCHE TABELLEN WERDEN
2407 ;Z.B. VOM UNTERPROGRAMM "SEINTR" AUFGESTELLT.
2408 ;
2409 ;DIE TABELLEN BESITZEN FOLGENDEN AUFBAU:
2410 ;
2411 ; REL. POSITION D. EINTRAGS IM DIR.-SEKTOR ) ANGABE FUER DEN
2412 ; NUMMER DES DIRECTORY-SEKTORS          ) 1. EINTRAG
2413 ;
2414 ; REL. POSITION D. EINTRAGS IM DIR.-SEKTOR ) ANGABE FUER DEN
2415 ; NUMMER DES DIRECTORY-SEKTORS          ) 2. EINTRAG
2416 ;
2417 ;          USW.
2418 ;
2419 ;WILL MAN Z.B. EINEN FILE LESEN, SO MUESSEN DER REIHE NACH ALLE ZU DEN
2420 ;EINZELNEN EINTRAEGEN GEMOERENDEN BLOECKE EINGELESEN WERDEN.
2421 ;ES EXISTIERT EIN ZEIGER, DER ZU BEGINN AUF DEN TABELLENANFANG ZEIGT.
2422 ;Liest man die angaben fuer einen eintrag aus der tabelle, so wird der
2423 ;zeiger auf die naechsten angaben weitergestellt.
2424 ;
2425 ;MIT HILFE DIESES UNTERPROGRAMMS KOENNEN DIE EINZELNEN ANGABEN AUS DER
2426 ;TABELLE GELESEN WERDEN. DAS PROGRAMM STELLT DEN TABELLENZEIGER AUTO-
2427 ;MATISCH WEITER.
2428 ;
2429 ;DA MEHRERE TABELLEN EXISTIEREN KOENNEN, DIENT DAS HL-REGISTERPAAR BEIM
2430 ;UNTERPROGRAMMAUFRUF AUCH ALS ZEIGER. ES ZEIGT AUF DEN GERADE AKTUELLEN
2431 ;TABELLENZEIGER:
2432 ;
2433 ; HL --> TABELLENZEIGER --> TABELLE
2434 ;
2435 ;DIE ANGABEN AUS DER TABELLE WERDEN BEIM AUSLESEN IN DIE REGISTER B UND C
2436 ;GELADEN:
2437 ;
2438 ;B-REGISTER: DIRECTORY-SEKTOR
2439 ;C-REGISTER: REL. NUMMER DES EINTRAGS IM DIR.-SEKTOR
2440 ;
2441 ;NACH RETURN SIND FOLGENDE REGISTER VERAENDERT:
2442 ;
2443 ;BC = (SIEHE OBEN)
2444 ;DE = NEUER WERT DES TABELLENZEIGERS
2445 ;
2446 ;

```

LOC	OBJ	LINE	SOURCE STATEMENT
4A56	5E	2447	GETPOS: MOV E,M ;E=LOW(TABELLEN-ZEIGER)
4A57	23	2448	INX H ;ZEIGER AUF HIGH(TAB-ZEIGER)
4A58	56	2449	MOV D,M ;D=HIGH(TAB-ZEIGER)
4A59	EB	2450	XCHG ;HL=TAB-ZEIGER, DE=ZEIGER AUF H(TAB-ZEIG.)
4A5A	4E	2451	MOV C,M ;C=LOW(TAB-EINTRAG)
4A5B	23	2452	INX H ;ZEIGER AUF HIGH(TAB-EINTRAG)
4A5C	46	2453	MOV B,M ;B=HIGH(TAB-EINTRAG)
4A5D	23	2454	INX H ;ZEIGER AUF LOW(NAECHST. TAB-EINTRAG)
4A5E	EB	2455	XCHG ;DE=TAB-ZEIGER, HL=ZEIGER AUF H(TAB-ZEIG.)
4A5F	72	2456	MOV M,D ;* SPEICHERE NEUEN TAB-ZEIGER
4A60	2B	2457	ICX H ;*
4A61	73	2458	MOV M,E ;*
4A62	C9	2459	RET
		2460	;
		2461	*****
		2462	;UNTERPROGRAMM "GSTART"
		2463	;DRUCKT START-ADRESSEN-VORSCHLAG (STEHT IN STARTA) UND HOLT EVTL.
		2464	;NEUE ADRESSE. DIE (NEUE) START-ADR WIRD IN "STARTA" GESPEICHERT
		2465	;DAS AUSGEWAELHTE LAUFWERK (SELEKT-MASKE IN "GELMSK") WIRD SELEKTIERT.
		2466	;FALLS ES NICHT BEREIT IST, VERZWEIGT DAS PROGRAMM NACH "CODE3"
		2467	;
4A63	CD9E49	2468	GSTART: CALL DESEL ;DESELECT LAUFWERK
4A66	CDAA0C	2469	CALL RUFCLR ;LOESCHE MAT85-INPUT-BUFFER
4A69	2A6FFD	2470	LHLD STARTA ;LADE START-ADR
4A6C	CD740A	2471	CALL HSTART ;NEUE START-ADR ?
4A6F	226FFD	2472	SHLD STARTA ;SPEICHERE (NEUE) START-ADR
4A72	CD194E	2473	CALL CRLF ;PRINT CR,LF
4A75	CD8R49	2474	CALL SELECT ;SELECT LAUFWERK
4A78	CDAA49	2475	CALL IREADY ;LAUFWERK BEREIT ?
4A7E	DC584B	2476	CC CODE3 ;NEIN --> CODE3
4A7E	C9	2477	RET
		2478	;
		2479	*****
		2480	;UNTERPROGRAMM "GSTSP"
		2481	;DRUCKT VORSCHLAGS-ADRESSEN FUER START/STOP (ADRESSEN AUS "STARTA" BZW.
		2482	; "STOPA"), HOLT NEUE ADRESSEN UND SPEICHERT SIE IN "STARTA" BZW. "STOPA"
		2483	;AB.
		2484	;
		2485	;WERTE NACH RETURN: DE = START-ADRESSE
		2486	; HL = STOP-ADRESSE
		2487	;
4A7F	CDAA0C	2488	GSTSP: CALL RUFCLR ;LOESCHE MAT85-INPUT-BUFFER
4A82	CD5B21	2489	CALL HSTSPA ;HOLE START/STOP-ADR
4A85	226FFD	2490	SHLD STARTA ;RETTE START-ADR
4A88	EB	2491	XCHG ;HL=STOP-ADR
4A89	2271FD	2492	SHLD STOPA ;RETTE STOP-ADR
4A8C	C9	2493	RET
		2494	;
		2495	;

LOC	OBJ	LINE	SOURCE STATEMENT
		2496	*****
		2497	;UNTERPROGRAMM "MOVE"
		2498	;DIESES UNTERPROGRAMM KOPIERT DIE DATENBYTES AUS DEM SEKTOR-PUFFER "SECB"
		2499	;IN DEN ZIEL-SPEICHER. ES PRUEFT, OB DAS BYTE ERFOLGREICH IN DEN ZIEL-
		2500	;SPEICHER GESCHRIEBEN WURDE. BEI EINEM FEHLER VERZWEIGT DAS PROGRAMM NACH
		2501	;"CODE14".
		2502	;
		2503	;WERTE BEIM AUFRUF: (TEINT) = ANFANG DES ZIELSPEICHERS
		2504	; (REST) = ANGABE UEBER DIE ANZAHL DER ZU KOPIERENDEN
		2505	; BYTES:
		2506	; (REST) >= 512 --> 512 BYTES KOPIEREN
		2507	; (REST) < 512 --> (REST) BYTES KOPIEREN
		2508	;
		2509	;WERTE NACH RETURN: (TEINT) = ALTER WERT + ANZ. DER KOPIERTEN BYTES
		2510	;
		2511	;VERAENDERTE REGISTER: A = 00
		2512	;
		2513	;
4ABD	C5	2514	MOVE:           PUSH   B           ;* RETTE REGISTER
4ABE	D5	2515	PUSH   D           ;*
4ABF	E5	2516	PUSH   H           ;*
		2517	;
4A90	2A50FA	2518	LHLI   REST       ;LADE REST-LAENGE
4A93	EB	2519	XCHG               ;REST-LAENGE NACH DE
4A94	210002	2520	LXI    H,512      ;SEKTOR-LAENGE
4A97	CD431	2521	CALL   R4         ;REST > SEKTOR-LAENGE
4A9A	D29E4A	2522	JNC    MOVE0      ;NEIN --> MOVE0
4A9D	EB	2523	XCHG               ;DE=SEKTOR-LAENGE (512)
4A9E	2A05FA	2524	MOVE0:           LHLI   TEINT      ;ZIEL-ADR NACH HL
4AA1	0100F8	2525	LXI    B,SECB     ;ZEIGER AUF SEKTOR-BUFFER
4AA4	0A	2526	MOVE1:           LDIAX B         ;ZEICHEN AUS BUFFER IN AKKU
4AA5	77	2527	MOV    M,A        ;ZEICHEN IN ZIEL-SPEICHER
4AA6	BE	2528	CMP    M         ;ZEICHEN IN ZIEL-SPEICHER ?
4AA7	C47948	2529	CNZ    CODE14     ;NEIN --> CODE14
4AAA	23	2530	INX    H         ;* ZEIGER WEITERSTELLEN
4AAE	03	2531	INX    B         ;*
4AAC	1B	2532	DCX    D         ;EIN BYTE UEBERTRAGEN
4AAD	7A	2533	MOV    A,D        ;* ZAEHLER = 0000 ?
4AAE	B3	2534	ORA    E         ;*
4AAF	C2A44A	2535	JNZ    MOVE1      ;NEIN --> NAECHSTES BYTE
4AB2	2205FA	2536	SHLD  TEINT      ;RETTE NEUE ZIEL-ADR
4AB5	E1	2537	POP    H         ;* RESTORE REGISTER
4AB6	D1	2538	POP    D         ;*
4AB7	C1	2539	POP    B         ;*
4AB8	C9	2540	RET
		2541	;
		2542	;



LOC	OBJ	LINE	SOURCE STATEMENT
		2543	*****
		2544	;UNTERPROGRAMM "DELETE"
		2545	;BEREITET DAS UNTERPROGRAMM "MDIR" ZUM LOESCHEN VON DIRECTORY-EINTRAEGEN
		2546	;VOR. DAS UNTERPROGRAMM "DELETE" GEHT OHNE RETURN IN DAS UNTERPROGRAMM
		2547	; "MDIR" UEBER.
		2548	;
		2549	;WERTE BEIM AUFRUF: A=ANZAHL DER ZU LOESCHENDEN EINTRAEGE
		2550	; DIE POSITIONEN DER EINTRAEGE MUESSEN IN DER TABELLE
		2551	; "TVORH" STEHEN.
		2552	; AUFBAU DER TABELLEN-ANGABEN:
		2553	; DIE TABELLE ENTH. FUER JEDEN EINTRAG EINE 2-BYTE ANGABE.
		2554	; LOW-BYTE: POSITION DES EINTRAGS INNERH. D. DIR-SEKTORS
		2555	; HIGH-BYTE: NUMMER DES DIRECTORY-SEKTORS
		2556	;
4AB9	324AFA	2557	DELETE: STA ZWSP2 ;ANZAHL DER ZU LOESCHENDEN EINTRAEGE
4ABC	2128FA	2558	LXI H,TVORH ;* ZEIGER AUF "TVORH" INITIALISIEREN
4ABF	2248FA	2559	SHLDI ZWSP1 ;*
4AC2	213600	2560	LXI H,0036H ;: CODE: "MVI M,00"
4AC5	226EFA	2561	SHLDI XDIR ;: EINSETZEN
4AC8	3EC9	2562	MVI A,RETURN ;* RET-OPCODE
4ACA	3270FA	2563	STA XDIR+2 ;* EINSETZEN
		2564	;
		2565	

LOC	OBJ	LINE	SOURCE STATEMENT	
		2566	;*****	
		2567	;UNTERPROGRAMM "MDIR" (MODIFIZIERE DIRECTORY)	
		2568	;DIESES UNTERPROGRAMM MODIFIZIERT DAS DIRECTORY. DIE ART DER MODIFIKATION	
		2569	;(EINTRAG ERSTELLEN, EINTRAG LOESCHEN) MUSS VOR DEM UNTERPROGRAMM-AUFRUF	
		2570	;DURCH MODIFIKATION DES VEKTORS "XDIR" FESTGELEGT WERDEN.	
		2571	;	
		2572	;EINTRAG LOESCHEN : XDIR = MVI M,00	
		2573	; RET	
		2574	;EINTRAG ERSTELLEN: XDIR = CALL VEINTR	
		2575	;	
		2576	;WERTE BEIM AUFRUF: (ZWSF1)=ZEIGER AUF EINEN TABELLENZEIGER	
		2577	; (ZWSF1) ---> TAB.-ZEIGER ---> TABELLE	
		2578	; DIE TABELLE ENTHAELT ANGABEN UEBER DIE POSITION	
		2579	; DER ZU MODIFIZ. EINTRAEGE INNERHALB DES DIRECTORY.	
		2580	; PRO EINTRAG ENTH. DIE TABELLE EINE 2-BYTE-ANGABE:	
		2581	; LOW-BYTE: POS. DES EINTRAGS INNERH. DES DIR.-SEKTORS	
		2582	; HIGH-BYTE: NUMMER DES DIRECTORY-SEKTORS	
		2583	;	
		2584	; (ZWSF2)=ANZAHL DER ZU MODIFIZIERENDEN EINTRAEGE	
		2585	;	
4ACD	CD0B4A	2586	MDIR:	CALL POSDIR ;POSITIONIERE KOPF AUF DIRECTORY
		2587	;	
4AD0	2148FA	2588	LXI	H,ZWSF1 ;ZEIGER AUF TABELLEN-POINTER
4AD3	CD564A	2589	CALL	GETPOS ;LESE POSITION AUS TABELLE
		2590	;	
4AD6	78	2591	MDIR1:	MOV A,B ;SEKTOR-NUMMER NACH A
4AD7	D3C2	2592	OUT	SEC ;SEKTOR-NUMMER AN FDC
4AD9	CD4C49	2593	CALL	RRS ;LESE SEKTOR (MAXTRY VERSUCHE)
4ADC	CD8B49	2594	CALL	SELECT ;KEEP DRIVE ROTATING
4ADF	79	2595	MOV	A,C ;EINTRAGS-NUMMER NACH A
4AE0	CD4D4B	2596	CALL	ANFEIN ;ZEIGER AUF EINTRAGS-ANFANG
4AE3	CD6EFA	2597	CALL	XDIR ;MODIFIZIERE DIRECTORY-SEKTOR
4AE6	214AFA	2598	LXI	H,ZWSF2 ;ZEIGER AUF ZAEHLER
4AE9	35	2599	DCR	M ;ZAEHLER MINUS EINS
4AEA	CA004B	2600	JZ	MDEND ;SPRINGE, WENN ALLE EINTRAEGE MODIFIZIERT
4AED	78	2601	MOV	A,B ;ALTE SEKTOR-NUMMER NACH A
4AEE	2148FA	2602	LXI	H,ZWSF1 ;ZEIGER AUF TABELLEN-POINTER
4AF1	CD564A	2603	CALL	GETPOS ;LESE POSITION AUS TABELLE
4AF4	B8	2604	CMF	B ;GLEICHER SEKTOR WIE LETZTER EINTRAG ?
4AF5	CA1C4A	2605	JZ	MDIRO ;JA --> MDIR0
4AF8	C5	2606	PUSH	B ;RETTE SEKTOR-NR. (B) UND EINTRAGS-NR. (C)
4AF9	CD3C49	2607	CALL	SAVSEC ;SCHREIBE GEAENDERTEN DIRECTORY-SEKTOR
4AFC	C1	2608	POP	B ;RESTORE SEKTOR- (B) UND EINTRAGS-NR. (C)
4AFD	C3D64A	2609	JMP	MDIR1 ;LESE ZU AENDERNDEN SEKTOR EIN
		2610	;	
4B00	CD3C49	2611	MDEND:	CALL SAVSEC ;SCHREIBE GEAENDERTEN DIRECTORY-SEKTOR
4B03	C9	2612	RET	
		2613	;	
		2614	;	

```

LOC OBJ          LINE          SOURCE STATEMENT
2615 ;*****
2616 ;UNTERPROGRAMM "ANZAHL"
2617 ;DIESES UNTERPROGRAMM BERECHNET AUS DER FILE-LAENGE WIEVIELE 4K-BYTE-
2618 ;BLOCKS ZUM ABSPEICHERN BENOETIGT WERDEN.
2619 ;
2620 ;WERTE BEIM AUFRUF: HL=FILE-LAENGE
2621 ;
2622 ;WERTE NACH RETURN: HL=REST (FILELAENGE/4096)
2623 ;
2624 ;
2625 ;           WENN HL=0000           WENN HL (<) 0000
2626 ;           B=FILELAENGE/4096     B=FILELAENGE/4096+1
4B04 0600        2627 ANZAHL:      MVI     B,0           ;* BERECHNE:
4B06 2A00FA      2628             LHLI    LAENGE      ;* WIEVIELE VOLLE TRACKS
4B09 110010      2629             LXI     D,4096      ;* A 4K-BYTE WERDEN ZUM
4B0C C13910      2630 ANZ1:       CALL    SUB2         ;* ABSPEICHERN BENOETIGT
4B0F DA164B      2631             JC      ANZO        ;* RECHNUNG: HL/1000H
4B12 04          2632             INR    B           ;* ERGEBNIS: B
4B13 C30C4B      2633             JMP    ANZ1         ;* REST      : HL
4B16 19          2634 ANZO:       DAD    D           ;*
4B17 7C          2635             MOV    A,H         ;* REST = 0000 ?
4B18 B5          2636             ORA    L           ;*
4B19 C8          2637             RZ              ;*JA --> RETURN
4B1A 04          2638             INR    B           ;*PLUS EIN EINTRAG FUER DEN REST
4B1B C9          2639             RET
2640 ;*****
2641 ;UNTERPROGRAMM "VEINTRAG" (VERMERKE DIRECTORY-EINTRAG)
2642 ;WIRD VON "MDIR" UEBER DEN VEKTOR "XDIR" AUFGERUFEN, WENN NEUE DIRECTORY-
2643 ;EINTRAEGE ERSTELLT WERDEN SOLLEN.
2644 ;
2645 ;WERTE BEIM AUFRUF: (EFROM) = ENTERED-FROM-CODE (VON WO WURDE DAS DOS
2646 ;                       AUFGERUFEN?) 01=MAT85, 02=SPS, 03=BASIC
2647 ;
2648 ;
2649 ;           (ZWSP2) = ANZAHL DER BENOETIGTEN EINTRAEGE
2650 ;                       (EIN FILE BENOETIGT FUER JEDEN 4K-BYTE-BLOCK
2651 ;                       EINEN EINTRAG. DIESES UNTERPROGRAMM ERZEUGT
2652 ;                       NUR EINEN EINTRAG UND MUSS DESHALB ENTSPRECHEND
2653 ;                       OFT AUFGERUFEN WERDEN. WIRD DER LETZTE EINTRAG
2654 ;                       EINES FILES ERZEUGT, SO ERHAELT DIESER EINE BE-
2655 ;                       SONDERE MARKIERUNG. UM ERKENNEN ZU KOENNEN, DASS
2656 ;                       ES SICH UM DEN LETZTEN EINTRAG HANDELT, MUSS DER
2657 ;                       INHALT VON "ZWSP2" DANN DEN WERT "01" HABEN.)
2658 ;
2659 ;           (START) = START-ADRESSE DES FILES
2660 ;
2661 ;           (LAENGE)= LAENGE DES FILES
2662 ;
2663 ;           MAT85-INPUT-PUFFER = FILE-NAME (RUECKWAERTS)
2664 ;           (EVNTL. MIT 80H AUF 12 ZEICHEN AUFGEFUELLT)

```

LOC	OBJ	LINE	SOURCE STATEMENT
4B1C	C5	2665	VEINTR: PUSH B ;RETTE BC
4B1D	3A83FA	2666	LDA EFROM ;LESE ENTERED-FROM-CODE: 01 = MAT
		2667	;
		2668	;
		2669	;
4B20	57	2669	MOV I,A ;RETTE CODE IN I
4B21	3A4AFA	2670	LDA ZWSP2 ;ANZAHL DER BENOETIGTEN EINTRAEGE
4B24	3D	2671	DCR A ;ANZAHL MINUS EINS
4B25	CA2A4B	2672	JZ LASTE ;SPRINGE, WENN LETZTER EINTRAG
4B28	3E80	2673	MVI A,10000000B ;SETZE BIT 7, LOESCHE BIT 1 ... BIT 6
4B2A	E2	2674	LASTE: ORA I ;VERKNUEPFE MIT ENTERED-FROM-CODE
4B2E	77	2675	MOV M,A ;SETZE KENNYBYTE EIN
		2676	;
4B2C	23	2677	INX H ;ZEIGER AUF NAMENS-FELD
4B2D	11F1FC	2678	LXI I,M85BE-1 ;MAT85-INPUT-BUFFER (FILE-NAME)
4B30	060C	2679	MVI B,12 ;LAENGE (NAME)
4B32	1A	2680	NXTCHR: LDAX I ;ZEICHEN IN AKKU
4B33	77	2681	MOV M,A ;ZEICHEN IN BUFFER
4B34	23	2682	INX H ;* STELLE ZEIGER WEITER
4B35	1B	2683	DCX I ;*
4B36	05	2684	DCR B ;ALLE ZEICHEN UEBERTRAGEN ?
4B37	C2324B	2685	JNZ NXTCHR ;NEIN --> NXTCHR
		2686	;
4B3A	EB	2687	XCHG ;DE=BUFFER-ZEIGER
4B3B	2A6FFD	2688	LHLI STARTA ;HL=START-ADR
4B3E	EB	2689	XCHG ;DE=START-ADR, HL=BUFFER-ZEIGER
4B3F	73	2690	MOV M,E ;* SETZE START-ADR EIN
4B40	23	2691	INX H ;*
4B41	72	2692	MOV M,D ;*
4B42	23	2693	INX H ;*
		2694	;
4B43	EB	2695	XCHG ;DE=BUFFER-ZEIGER
4B44	2A00FA	2696	LHLI LAENGE ;HL=LAENGE
4B47	EB	2697	XCHG ;DE=LAENGE, HL=BUFFER-ZEIGER
4B48	73	2698	MOV M,E ;* SETZE LAENGE EIN
4B49	23	2699	INX H ;*
4B4A	72	2700	MOV M,D ;*
		2701	;
4B4B	C1	2702	POP B ;RESTORE BC
4B4C	C9	2703	RET
		2704	*****
		2705	;UNTERPROGRAMM "ANFEIN"
		2706	;DIESES UNTERPROGRAMM STELLT DEN ZEIGER IM HL-REGISTERPAAR UEBER DEN
		2707	;ANFANG EINES BESTIMMTEN DIRECTORY-EINTRAGS. DER DIRECTORY-SEKTOR MUSS
		2708	;DAZU IM SEKTOR-PUFFER "SECB" STEHEN UND DER AKKUMULATOR MUSS DIE NUMMER
		2709	;DES EINTRAGS (1...16) INNERHALB DES SEKTORS ENTHALTEN
		2710	;
4B4D	21E0F7	2711	ANFEIN: LXI H,SECB-32 ;BUFFER-ANFANG MINUS 32
4B50	112000	2712	LXI D,32 ;OFFSET V. EINTRAG ZU EINTRAG
4B53	19	2713	DADDFS: DAD I ;* ADD OFFSET
4B54	3D	2714	DCR A ;* SO OFT WIE NOETIG
4B55	C2534B	2715	JNZ DADDFS ;*
4B58	C9	2716	RET
		2717	

LOC	OBJ	LINE	SOURCE STATEMENT
		2718	*****
		2719	;UNTERPROGRAMM "GETNAM"
		2720	;Liest FILE-NAME UND PRUEFT IHN AUF GUELTIGKEIT
		2721	;
		2722	;GEBE ANFORDERUNG AUS
		2723	;
4B59	CI7300	2724	GETNAM: CALL PTXTCR ;PRINT TEXT
4B5C	0A	2725	DB LF,'NAME: ',00
4B5D	4E414D45		
4B61	3A20		
4B63	00		
		2726	=====
		2727	;VORBEREITUNG
		2728	;ERWARTET WERDEN MAXIMAL 12 ASCII-ZEICHEN (KEIN "+", KEIN "--")
		2729	;WANDELE ALLE EINGEGEBENEN BUCHSTABEN IN GROSSBUCHSTABEN
		2730	;
4B64	AF	2731	XRA A ;A=0
4B65	32C9FC	2732	STA GROFLG ;NUR GROSSBUCHSTABEN
4B68	32C7FC	2733	STA BCKFLG ;KEIN +/-
4B6B	4F	2734	MOV C,A ;ASCII
4B6C	060C	2735	MVI B,12 ;MAX 12 ZEICHEN
4B6E	CDAA0C	2736	CALL BUFCLR ;LOESCHE EINGABE-BUFFER
		2737	=====
		2738	;LESE FILE-NAME EIN
		2739	;
4B71	CI0A0D	2740	CALL BREAD ;LESE NAME
4B74	CI194E	2741	CALL CRLF ;PRINT CR,LF
		2742	=====
		2743	;PRUEFE NAME AUF GUELTIGKEIT (ER STEHT RUECKWAERTS IM MAT85-INPUT-PUFFER,
		2744	;EINE MAT85-EIGENART)
		2745	;
4B77	21F1FC	2746	TSTNAM: LXI H,M85BE-1 ;ZEIGER AUF 1. ZEICHEN
4B7A	7E	2747	MOV A,M ;LESE 1. ZEICHEN
4B7E	CI7E4D	2748	CALL TSTCHR ;BUCHSTABE ODER ZIFFER ?
4B7E	I46A48	2749	CNC CODE9 ;NEIN --> CODE9
		2750	;
4B81	0E07	2751	MVI C,7 ;MAX 7 WEITERE ZEICHEN BIS "."
4B83	2B	2752	TNAM0: DCX H ;ZEIGER AUF NAECHSTES ZEICHEN
4B84	7E	2753	MOV A,M ;LESE ZEICHEN
4B85	CI7E4D	2754	CALL TSTCHR ;BUCHSTABE ODER ZIFFER ?
4B88	I2914B	2755	JNC TNAM1 ;NEIN --> TNAM1 ( "." ODER ENDE)
4B8B	0D	2756	DCR C ;ZAEHLER MINUS EINS
4B8C	C2834B	2757	JNZ TNAM0
		2758	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
4B8F	2B	2759	DCX H ;ZEIGER AUF NAECHSTES ZEICHEN
4B90	7E	2760	MOV A,M ;LESE ZEICHEN
4B91	E7	2761	ORA A ;BIT 7 GESETZT (ENDE) ?
4B92	F8	2762	RM ;JA --> RETURN
4B93	FE2E	2763	CFI ',.' ;PUNKT ?
4B95	C46A4B	2764	CNZ CODE9 ;NEIN --> CODE9
4B98	2B	2765	DCX H ;ZEIGER AUF NAECHSTES ZEICHEN
4B99	7E	2766	MOV A,M ;LESE ZEICHEN
4B9A	CD7E4D	2767	CALL TSTCHR ;BUCHSTABE ODER ZIFFER ?
4B9D	D46A4B	2768	CNC CODE9 ;NEIN --> CODE9
		2769	;
4BA0	0E02	2770	MVI C,2 ;MAX 2 WEITERE ZEICHEN
4BA2	2B	2771	DCX H ;ZEIGER AUF NAECHSTES ZEICHEN
4BA3	7E	2772	MOV A,M ;LESE ZEICHEN
4BA4	E7	2773	ORA A ;BIT 7 GESETZT (ENDE) ?
4BA5	F8	2774	RM ;JA --> RETURN
4BA6	CD7E4D	2775	CALL TSTCHR ;BUCHSTABE ODER ZIFFER ?
4BA9	D46A4B	2776	CNC CODE9 ;NEIN --> CODE9
4BAC	0D	2777	DCR C ;ZAEHLER MINUS EINS
4BAD	C2A24B	2778	JNZ TNAM2
		2779	;
4BB0	2B	2780	DCX H ;ZEIGER AUF NAECHSTES ZEICHEN
4BB1	7E	2781	MOV A,M ;LESE ZEICHEN
4BB2	E7	2782	ORA A ;BIT 7 GESETZT (ENDE) ?
4BB3	F8	2783	RM ;JA --> RETURN
4BB4	CD6A4B	2784	CALL CODE9 ;NEIN --> CODE9
		2785	;
		2786	

```
LOC OBJ      LINE      SOURCE STATEMENT
2787 ;*****
2788 ; UNTERPROGRAMM "SEINTR" (SUCHE EINTRAG)
2789 ;
2790 ; FUNKTION: 1. WENN (DSPLD) = 00
2791 ; ES WERDEN ALLE DIRECTORY-EINTRAEGE GEZAEHLT, DIE ENTWEDER
2792 ; FREI SIND ODER DEREN NAMEN MIT DEM IM MAT85-INPUT-PUFFER
2793 ; UEBEREINSTIMMEN.
2794 ; DER ZAEHLER WIRD IN "ZAEHL" ABGESPEICHERT.
2795 ; DIE ERSTEN 16 EINTRAEGE WERDEN IN DER TABELLE "TEINT" VERMERKT.
2796 ; ANZAHL DER GUELTIGEN EINTRAEGE IN "TEINT":
2797 ; (ZAEHL) < 17 : (ZAEHL) GIBT DIE ANZAHL AN.
2798 ; (ZAEHL) >= 17 : 16 GUELTIGE EINTRAEGE.
2799 ;
2800 ;
2801 ; AUSSERDIEM WERDEN ALLE EINTRAEGE GEZAEHLT, DEREN NAMEN MIT DEM
2802 ; IM MAT85-INPUT-BUFFER UEBEREINSTIMMEN (IM GEGENSATZ ZUM ERSTEN
2803 ; ZAEHLER WERDEN FREIE EINTRAEGE NICHT MITGEZAEHLT). DA FUER
2804 ; JEDEN 4K-BYTE-BLOCK EIN EINTRAG BESTEHT UND DA DIE MAXIMALE
2805 ; FILE-LAENGE 65535 BYTE BETRAEGT, KANN DER ZAEHLER MAXIMAL DEN
2806 ; WERT 16 ANNEHMEN.
2807 ; DER ZAEHLER WIRD IN "ZAEHLV" ABGESPEICHERT.
2808 ; DIE EINTRAEGE WERDEN IN DER TABELLE "TVORH" VERMERKT.
2809 ; "ZAEHLV" GIBT DIE ANZAHL DER GUELTIGEN EINTRAEGE IN DIESER
2810 ; TABELLE AN.
2811 ;
2812 ; 2. WENN (DSPLD) (<) 00
2813 ; DAS INHALTSVERZEICHNIS WIRD ANGEZEIGT.
2814 ; ES WERDEN DANN NUR DIE FREIEN EINTRAEGE IN "ZAEHL" GEZAEHLT.
2815 ; DIE ERSTEN 16 DIESER EINTRAEGE WERDEN IN DER TABELLE "TEINT"
2816 ; VERMERKT.
2817 ; ANZAHL DER GUELTIGEN EINTRAEGE IN "TEINT":
2818 ; (ZAEHL) < 17 : (ZAEHL) GIBT DIE ANZAHL AN.
2819 ; (ZAEHL) >= 17 : 16 GUELTIGE EINTRAEGE.
2820 ;
2821 ; DAS INHALTSVERZEICHNIS IST WIE FOLGT AUFGEBAUT:
2822 ; ES BESTEHT AUS DEN SEKTOREN 1,2,3,4 UND 5 AUF SPUR 0, SEITE 0.
2823 ; JEDER SEKTOR BESTEHT AUS 16 EINTRAEGEN A 32 BYTE.
2824 ; DER LETZTE EINTRAG IM 5. SEKTOR WIRD NICHT GENUTZT.
2825 ; DAMIT ERGIBT SICH EINE ANZAHL VON 79 GENUTZTEN EINTRAEGEN.
2826 ;
2827 ; FUEER JEDES FILE WIRD PRO ANGEFANGENE 4K-BYTE EIN EINTRAG BELEGT.
2828 ; FILES, DIE MEHR ALS EINEN EINTRAG BELEGEN (LAENGE > 4K-BYTE), WERDEN
2829 ; ALS "MEHRFACH-EINTRAG" BEZEICHNET. FILES, DIE NUR EINEN EINTRAG BELEGEN
2830 ; (LAENGE <= 4K-BYTE), WERDEN ALS "EINZEL-EINTRAG" BEZEICHNET.
2831 ; BEI MEHRFACH-EINTRAEGEN BEFINDEN SICH DIE EINTRAEGE ZU DEN EINZELNEN
2832 ; 4K-BYTE-BLOECKEN IN DER REIHENFOLGE, IN DER DIE EINZELNEN BLOECKE BEIM
2833 ; ABSPEICHERN IM SPEICHER STANDEN.
2834 ;
2835 ; EIN EINTRAG ENTHAEHLT FOLGENDE INFORMATION:
2836 ; 1. BYTE : KENNBYTE
2837 ; 2. - 13. BYTE : FILE-NAME INCL. ".", RECHTS MIT 80H AUFGEFUELLT
2838 ; 14. - 15. BYTE : START-ADR (BEI MEHRFACH-EINTRAEGEN START-ADR DES 1. BLOCKS)
2839 ; 16. - 17. BYTE : LAENGE (BEI MEHRFACH-EINTRAEGEN LAENGE DES GESAMT-FILES)
2840 ; 18. - 32. BYTE : OHNE BEDEUTUNG
2841 ;
```

```

LOC  OBJ      LINE      SOURCE STATEMENT
2841 ; DAS KENNBYTE HAT FOLGENDE BEDEUTUNG:
2842 ;
2843 ; 0000 0000 - EINTRAG FREI
2844 ;
2845 ; SONST:
2846 ; 1XXX XXXX - TEIL 1 ... N-1 EINES N-TEILIGEN MEHRFACH-EINTRAGS
2847 ; OXXX XXXX - TEIL N EINES N-TEILIGEN MEHRFACH-EINTRAGS
2848 ; ODER EINZEL-EINTRAG
2849 ;
2850 ; BEI EINEM BELEGTEN EINTRAG GEBEN DIE BITS 0 BIS 6 AUSKUNFT
2851 ; UEBER DEN FILE-TYP:
2852 ; X000 0001 - MAT
2853 ; X000 0010 - SPS
2854 ; X000 0011 - BAS
2855 ; DER TYP GIBT AN, VON WO DAS IOS VOR DEM ABSPEICHERN EINES FILES
2856 ; AUFGERUFEN WURDE.
2857 ; BEISPIEL: TYP = "BAS": DAS IOS WURDE VON BASIC AUFGERUFEN
2858 ; DAS FILE IST EIN BASIC-PROGRAMM
2859 ;
2860 ; DIE POSITION EINES EINTRAGS INNERHALB DES VERZEICHNISDES GIBT
2861 ; AUSKUNFT DARUEBER, WO DER IM EINTRAG ANGEGEBENE FILE AUF DER
2862 ; DISKETTE GESPEICHERT IST.
2863 ; DAZU WERDEN DIE EINZELNEN DIRECTORY-EINTRAEGE UEBER ALLE DIRECTORY-
2864 ; SEKTOREN VON 1 BIS 79 DURCHNUMMERIERT.
2865 ;
2866 ; STELLT MAN DIE EINTRAGS-NUMMER (1 ... 79) BINAER DA, SO GIBT BIT 0
2867 ; DIE SEITE AN:
2868 ; XXXX XXX0 --- SEITE 0
2869 ; XXXX XXX1 --- SEITE 1
2870 ;
2871 ; DIE BITS 1 BIS 7 GEBEN DIE SPUR AN:
2872 ;
2873 ; BEISPIELE: EINTRAGS-NUMMER  BINAER-DARSTELLUNG  SEITE  SPUR
2874 ;           1                0000 0001          1    0
2875 ;           2                0000 0010          0    1
2876 ;           3                0000 0011          1    1
2877 ;           4                0000 0100          0    2
2878 ;           5                0000 0101          1    2
2879 ;           .                .                .    .
2880 ;           .                .                .    .
2881 ;           .                .                .    .
2882 ;           78                0100 1110          0    39
2883 ;           79                0100 1111          1    39
2884 ;
2885 ; SEITE 0, SPUR 0 IST DURCH DAS VERZEICHNIS BELEGT.
2886 ;
2887 ; NACH RETURN: 1. WENN (DSPLD) = 00
2888 ; (ZAEHL) = ANZAHL DER FREIEN EINTRAEGE
2889 ;           + ANZAHL DER EINTRAEGE MIT DEM FILE-NAMEN,
2890 ;           DER IM MAT85-INPUT-PUFFER STEHT.
2891 ; A           = ZAEHL
2892 ; WENN (ZAEHL) = 00 IST DAS Z-FLAG GESETZT.
2893 ;
2894 ; (ZAEHLV) = ANZAHL DER EINTRAEGE MIT DEM FILE-NAMEN,
2895 ;           DER IM MAT85-INPUT-PUFFER STEHT.

```



```

LOC  OBJ          LINE      SOURCE STATEMENT
                2896 ;
                2897 ;           DIE ERSTEN 16 EINTRAEGE, DIE ENTWEDER
                2898 ;           - FREI SIND ODER
                2899 ;           - DEREN NAME MIT DEM IM MAT85-INPUT-BUFFER UEBEREINSTIMMT
                2900 ;           WERDEN IN DER TABELLE "TEINT" VERMERKT.
                2901 ;           ANZAHL DER GUELTIGEN EINTRAEGE IN "TEINT":
                2902 ;           (ZAEHL) < 17 : (ZAEHL) GIBT DIE ANZAHL AN
                2903 ;           (ZAEHL) >= 17 : 16 GUELTIGE EINTRAEGE
                2904 ;
                2905 ;           DIE EINTRAEGE, DEREN NAME MIT DEM IM MAT85-INPUT-PUFFER
                2906 ;           UEBEREINSTIMMT, WERDEN IN DER TABELLE "TVORH" VERMERKT.
                2907 ;           (ZAEHLV) GIBT DIE ANZAHL DER GUELTIGEN EINTRAEGE IN
                2908 ;           DIESER TABELLE AN.
                2909 ;
                2910 ;           2. WENN (ISPLD) (<) 00
                2911 ;           (ZAEHL) = ANZAHL DER FREIEN EINTRAEGE
                2912 ;           A           = ZAEHL
                2913 ;           WENN (ZAEHL) = 00 IST DAS Z-FLAG GESETZT
                2914 ;
                2915 ;           DIE ERSTEN 16 EINTRAEGE, DIE FREI SIND, WERDEN IN DER
                2916 ;           TABELLE "TEINT" VERMERKT.
                2917 ;           ANZAHL DER GUELTIGEN EINTRAEGE IN "TEINT":
                2918 ;           (ZAEHL) < 17 : (ZAEHL) GIBT DIE ANZAHL AN.
                2919 ;           (ZAEHL) >= 17 : 16 GUELTIGE EINTRAEGE.
                2920 ;
                2921 ;           DIE EINTRAEGE IN DEN TABELLEN "TEINT" UND "TVORH" BESTEHEN AUS JE ZWEI BYTES:
                2922 ;           - LOW BYTE: NUMMER DES RELATIVEN DIRECTORY-EINTRAGS (1 ... 16)
                2923 ;           - HIGH BYTE: NUMMER DES DIRECTORY-SEKTORS (1 ... 5)
                2924 ; .....
                2925 ; VORBEREITUNG
                2926 ;
4BB7  AF          2927 SEINTR:      XRA      A           ;A=0
4BB8  3284FA     2928 STA      SEITE       ;SEITE 0
4BBE  3204FA     2929 STA      ZAEHL       ;* ZAEHLER AUF NULL
4BBE  3227FA     2930 STA      ZAEHLV      ;*
4BC1  2105FA     2931 LXI     H,TEINT      ;: ZEIGER AUF DIE TABELLEN
4BC4  2202FA     2932 SHLD   P,TEINT      ;: "TEINT" UND
4BC7  2128FA     2933 LXI     H,TVORH     ;: "TVORH"
4BCA  2225FA     2934 SHLD   P,TVORH     ;: INITIALISIEREN
                2935 ; =====
                2936 ; KOPF UEBER DIREKTORY POSITIONIEREN
                2937 ;
4BCD  CD0B4A     2938 CALL   POSDIR       ;POSITIONIERE KOPF AUF DIRECTORY
                2939 ; =====
                2940 ; LESE DIRECTORY-SEKTOR EIN
                2941 ;
4BD0  0601       2942 MVI     B,1         ;SEKTOR-NUMMER
4BD2  CD8B49     2943 NEXT2:  CALL   SELECT   ;RE-SELECT LAUFWERK
4BD5  78          2944 MOV     A,B         ;SEKTOR-NUMMER NACH A
4BD6  D3C2       2945 OUT    SEC         ;AN FIC
4BD8  CD4C49     2946 CALL   RRS         ;READ SEKTOR (MAXTRY VERSUCHE)
                2947 ;

```

LOC	OBJ	LINE	SOURCE STATEMENT
		2948	;
		2949	;PRUEFE DIE EINTRAEGE
		2950	;EINTRAG FREI ?
		2951	;
4BD8	2100F8	2952	LXI H,SECB ;ZEIGER AUF SECTOR-BUFFER
4BDE	0E01	2953	MVI C,1 ;1. EINTRAG DES SEKTORS
4BE0	2248FA	2954	NEXT1: SHLD ZWSP1 ;RETTE ZEIGER
4BE3	7E	2955	MOV A,M ;LESE ERSTES BYTE DES EINTRAGS
4BE4	B7	2956	ORA A ;EINTRAG FREI ?
4BE5	C2FB4B	2957	JNZ EBEL ;NEIN --> EBEL
		2958	;
		2959	;HIER, WENN EINTRAG FREI.
		2960	;ZAEHLE EINTRAG
		2961	;
4BE8	3A04FA	2962	EFREI: LDA ZAEHL ;LADE ZAEHLER FUER "TEINT"
4BEB	3C	2963	INR A ;PLUS EINS
4BEC	3204FA	2964	STA ZAEHL ;RETTE NEUEN ZAEHLER-STAND
4BEF	2A02FA	2965	LHLD PTEINT ;LADE ZEIGER AUF "TEINT"
4BF2	CD994C	2966	CALL REINTR ;REGISTRIERE EINTRAG
4BF5	2202FA	2967	SHLD PTEINT ;RETTE NEUEN ZEIGER
4BF8	C37C4C	2968	JMP NEXT
		2969	;
		2970	;HIER, WENN EINTRAG BELEGT
		2971	;SOLL DAS DIRECTORY ANGEZEIGT WERDEN ?
		2972	;
4BFB	57	2973	EBEL: MOV D,A ;RETTE KENN-BYTE IN D
4BFC	3A54FA	2974	LDA DSPLD ;* DISPLAY DIRECTORY ?
4BFF	B7	2975	ORA A ;*
4C00	CA584C	2976	JZ NODSFL ;NEIN --> NODSFL
		2977	;
		2978	;HIER, WENN DIRECTORY ANGEZEIGT WERDEN SOLL
		2979	;
4C03	7A	2980	MOV A,D ;RESTORE KENN-BYTE
4C04	B7	2981	ORA A ;BIT 7 GESETZT ?
4C05	FA7C4C	2982	JM NEXT ;JA --> NEXT
		2983	;
		2984	;(TEIL 1 ... N-1 EINES N-TEILIGEN
		2985	;MEHRFACH-EINTRAGS. BEI MEHRFACH-
		2986	;EINTRAEGEN WIRD NUR DER LETZTE
		2987	;EINTRAG ANGEZEIGT.
		2988	;
		2989	;EINTRAG ANZEIGEN
		2990	;
4C08	C5	2990	PUSH B ;RETTE SEKTOR-NR. (B) UND EINTRAGS-NR. (C)
4C09	060C	2991	MVI B,12 ;12 ZEICHEN ANZEIGEN
4C0B	23	2992	DISPL: INX H ;ZEIGER AUF NAECHSTES ZEICHEN
4C0C	7E	2993	MOV A,M ;LESE ZEICHEN
4C0D	B7	2994	ORA A ;BIT 7 GESETZT (80H) ?
4C0E	F2134C	2995	JP NOBO ;NEIN --> NOBO
4C11	3E20	2996	MVI A,' ' ;LADE SPACE
4C13	CD5200	2997	NOBO: CALL WCHAR ;PRINT ZEICHEN
4C16	05	2998	DCR B ;12 ZEICHEN GEDRUCKT ?
4C17	C20B4C	2999	JNZ DISPL ;NEIN --> DISPL
		3000	

LOC	OBJ	LINE	SOURCE STATEMENT
		3001	;
		3002	;DRUCKE ZUSATZINFORMATIONEN
		3003	;
		3004	;1. FILE-TYP (MAT,SFS,BAS)
		3005	;
4C1A	CD930B	3006	CALL WBLNKI ;* PRINT 3 BLANKS
4C1D	03	3007	DB 03 ;*
		3008	;
4C1E	CDA14C	3009	CALL TYP ;PRINT FILE-TYP
		3010	;
		3011	;2. START-ADRESSE
		3012	;
4C21	CD930B	3013	CALL WBLNKI ;* PRINT 6 BLANKS
4C24	06	3014	DB 06 ;*
		3015	;
4C25	23	3016	INX H ;* START-ADR LESEN
4C26	5E	3017	MOV E,M ;*
4C27	23	3018	INX H ;*
4C28	56	3019	MOV D,M ;*
4C29	EB	3020	XCHG ;*
4C2A	E5	3021	PUSH H ;RETTE START-ADR
4C2B	CD5B00	3022	CALL PHL ;PRINT START-ADR
		3023	;
		3024	;3. STOP-ADRESSE (WIRD AUS START-ADRESSE UND LAENGE BERECHNET)
		3025	;
4C2E	CD930B	3026	CALL WBLNKI ;* PRINT 10 BLANKS
4C31	0A	3027	DB 10 ;*
		3028	;
4C32	EB	3029	XCHG ;HL = BUFFER-ZEIGER
4C33	23	3030	INX H ;* LAENGE LADEN
4C34	5E	3031	MOV E,M ;*
4C35	23	3032	INX H ;*
4C36	56	3033	MOV D,M ;*
4C37	EB	3034	XCHG ;*
4C38	2200FA	3035	SHLD LAENGE ;RETTE LAENGE
4C3B	D1	3036	POP D ;RESTORE START-ADR
4C3C	19	3037	DAI D ;HL=START-ADR PLUS LAENGE
4C3D	2B	3038	DCX H ;HL=STOP-ADR
4C3E	CD5B00	3039	CALL PHL ;PRINT STOP-ADR
		3040	;
		3041	;4. ANZAHL DER BLOECKE
		3042	;
4C41	CD930B	3043	CALL WBLNKI ;* PRINT 7 BLANKS
4C44	07	3044	DB 7 ;*
4C45	CD044B	3045	CALL ANZAHL ;BERECHNE ANZAHL DER BLOECKE
4C48	78	3046	MOV A,B ;ANZAHL NACH A
4C49	CD6100	3047	CALL WDEZ ;PRINT ANZAHL
		3048	;
		3049	;NAECHSTE ZEILE. PRUEFE, OB BILDSCHIRM VOLL (UNTERPROG. "LINE").
		3050	;
4C4C	CD194E	3051	CALL CRLF ;PRINT CR,LF
4C4F	CD054C	3052	CALL LINE ;ZAEHLE DIE AUSGEBEBENEN ZEILEN
4C52	CD8B49	3053	CALL SELECT ;KEEP DRIVE ROTATING
		3054	;
4C55	C37B4C	3055	JMP NEXT0 ;NAECHSTER EINTRAG

```

LOC  OBJ          LINE      SOURCE STATEMENT
3056 ;=====
3057 ;HIER, WENN DIRECTORY NICHT ANGEZEIGT WERDEN SOLL
3058 ;-----
3059 ;IST DER FILE-NAME IM DIRECTORY-EINTRAG IDENTISCH MIT DEM IM EINGABE-PUFFER ?
3060 ;
4C58  C5          3061  NODSPL:      PUSH   B           ;RETTE SEKTOR-NR (B) UND EINTRAGS-NR. (C)
4C59  060C        3062                MVI   B,12        ;PRUEFE 12 ZEICHEN
4C5B  11F1FC      3063                LXI   I,M85BE-1   ;ZEIGER AUF FILE-NAMEN
4C5E  23          3064  EBEL0:      INX    H           ;STELLE ZEIGER AUF NAECHSTES ZEICHEN
4C5F  1A          3065                LDAX  D           ;ZEICHEN DES FILE-NAMENS NACH A
4C60  BE          3066                CMP   M           ;VERGLEICHE MIT ZEICHEN AUS DIRECTORY
4C61  C27B4C      3067                JNZ   NEXT0       ;SPRINGE, WENN UNGLEICH
4C64  1B          3068                DCX   D           ;STELLE MAT85-BUFFER-ZEIGER WEITER
4C65  05          3069                DCR   B           ;12 ZEICHEN GEPRUEFT ?
4C66  C25E4C      3070                JNZ   EBEL0       ;NEIN --> EBEL0
3071 ;=====
3072 ;HIER, WENN EINTRAGS-NAME = NAME (MAT85-INPUT-PUFFER)
3073 ;-----
3074 ;ZAEHLE EINTRAG UND REGISTRIERE IHN.
3075 ;
4C69  3A27FA      3076                LDA   ZAEHLV      ;LADE ZAEHLER
4C6C  3C          3077                INR   A           ;ERHOEHE ZAEHLER
4C6D  3227FA      3078                STA   ZAEHLV      ;RETTE NEUEN ZAEHLER-STAND
4C70  C1          3079                POP   B           ;RESTORE SEKTOR- (B) UND EINTRAGS-NR. (C)
4C71  2A25FA      3080                LHLD  PTVORH     ;ZEIGER AUF "TVORH"
4C74  CD994C      3081                CALL  REINTR      ;REGISTRIERE EINTRAG
4C77  2225FA      3082                SHLD  PTVORH     ;RETTE NEUEN ZEIGER
4C7A  C5          3083                PUSH  B           ;
3084 ;=====
3085 ;NAECHSTER EINTRAG
3086 ;
4C7B  C1          3087  NEXT0:      POP   B           ;RESTORE SEKTOR- (B) UND EINTRAGS-NR. (C)
4C7C  112000      3088  NEXT:      LXI   D,32        ;OFFSET ZUM NAECHSTEN EINTRAG
4C7F  2A48FA      3089                LHLD  ZWSP1       ;ALTEN ZEIGER LADEN
4C82  19          3090                DAD   D           ;OFFSET ADDIEREN
4C83  0C          3091                INR   C           ;NAECHSTER EINTRAG
4C84  3E11        3092                MVI   A,17        ;MAX EINTRAG + EINS
4C86  B9          3093                CMP   C           ;C=MAX EINTRAG + EINS ?
4C87  C2E04B      3094                JNZ   NEXT1       ;NEIN --> NEXT1
3095 ;=====
3096 ;HIER, WENN NAECHSTER EINTRAG IM NAECHSTEN SEKTOR
3097 ;
4C8A  04          3098                INR   B           ;NAECHSTER SEKTOR
4C8B  3E06        3099                MVI   A,6         ;MAX SEKTOR + EINS
4C8D  B8          3100                CMP   B           ;B=MAX SEKTOR + EINS ?
4C8E  C2D24B      3101                JNZ   NEXT2       ;NEIN --> NEXT2
3102 ;
4C91  3A04FA      3103                LDA   ZAEHL       ;* KORRIG. ZAEHLER
4C94  3D          3104                DCR   A           ;*
4C95  3204FA      3105                STA   ZAEHL       ;*
4C98  C9          3106                RET              ;
3107 ;
3108

```

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
		3109	;*****
		3110	;UNTERPROGRAMM "REINTR" (REGISTRIERE EINTRAG)
		3111	;REGISTRIERT GEFUNDENEN EINTRAG IN TABELLE. (ALLERDINGS MAX. 16 EINTRAEGE)
		3112	;
		3113	;WERTE BEI AUFRUF: A=ANZAHL DER BEREITS REGISTR. EINTRAEGE + EINS
		3114	; B=NUMMER DES DIRECTORY-SEKTORS (1..5)
		3115	; C=REL. NUMMER DES EINTRAGS IM DIR.-SEKTOR (1..16)
		3116	; HL=ZEIGER AUF DIE TABELLENPOSITION, IN DER DER EINTRAG
		3117	; REGISTRIERT WERDEN SOLL.
		3118	;
4C99	FE11	3119	REINTR: CPI 17 ;BEREITS 16 EINTRAEGE IN TABELLE ?
4C9B	D0	3120	RNC ;JA --> RETURN
4C9C	71	3121	MOV M,C ;RETTE EINTRAGS-NUMMER
4C9D	23	3122	INX H ;STELLE ZEIGER WEITER
4C9E	70	3123	MOV M,B ;RETTE SEKTOR-NUMMER
4C9F	23	3124	INX H ;STELLE ZEIGER WEITER
4CA0	C9	3125	RET
		3126	;
		3127	;*****
		3128	;UNTERPROGRAMM "TYP"
		3129	;GIBT BEI DER ANZEIGE DES DIRECTORY DEN FILE-TYP ENTSPRECHEND DEM
		3130	;CODE ALS TEXT AUS: 01 = MAT, 02 = SPS, 03 = BAS. FUER CODES >3 WIRD
		3131	; "???" AUSGEBEBEN. SIEHE AUCH TYP-TABELLE.
		3132	;
4CA1	E5	3133	TYP: PUSH H ;RETTE HL-REGISTERPAAR
4CA2	2A48FA	3134	LHLD ZWSP1 ;HL=ZEIGER AUF KENN-BYTE
4CA5	7E	3135	MOV A,M ;KENN-BYTE NACH A
4CA6	E67F	3136	ANI 01111111B ;BIT 7 AUF 0 SETZEN
4CA8	2A5EFA	3137	LHLD PTYPE ;HL=ZEIGER AUF TYP-TABELLE
4CAB	BE	3138	CMP M ;VERGLEICHE MIT MAX TYP-CODE + EINS
4CAC	DAB04C	3139	JC TYPOK ;SPRINGE, WENN TYP-CODE OK
4CAF	7E	3140	MOV A,M ;LADE MAX TYP-CODE
4CB0	2B	3141	TYPOK: ICX H ;* KORRIG. ZEIGER
4CB1	2B	3142	ICX H ;*
4CB2	5F	3143	MOV E,A ;CODE NACH LSB (DE)
4CB3	1600	3144	MVI D,00 ;MSB (DE) = 00
4CB5	19	3145	DAD D ;* STELLE ZEIGER AUF RICHTIGEN
4CB6	19	3146	DAD D ;* TABELLEN-EINTRAG
4CB7	19	3147	DAD D ;*
4CB8	1603	3148	MVI D,3 ;3 ZEICHEN SIND AUSZUGEBEN
4CBA	7E	3149	NTYPZ: MOV A,M ;LESE ZEICHEN
4CBB	CD5200	3150	CALL WCHAR ;PRINT ZEICHEN
4CBE	23	3151	INX H ;STELLE ZEIGER WEITER
4CBF	15	3152	DCR D ;3 ZEICHEN AUSGEBEBEN ?
4CC0	C2BA4C	3153	JNZ NTYPZ ;NEIN --> NTYPZ
4CC3	E1	3154	POP H ;RESTORE HL-REGISTERPAAR
4CC4	C9	3155	RET
		3156	;
		3157	;

LOC	OBJ	LINE	SOURCE STATEMENT
		3158	;*****
		3159	;UNTERPROGRAMM "LINE"
		3160	;WIRD NUR ABGEARBEITET, WENN DRUCKER AUS.
		3161	;
		3162	;UEBERWACHT BEI DEM DIRECTORY DIE ANZAHL DER AUSGEGEBENEN ZEILEN
		3163	;WENN DER BILDSCHIRM VOLL IST, WIRD DIE ANZEIGE DES DIRECTORY UNTER-
		3164	;BROCHEN. ES WIRD DANN DIE EINGABE VON <SP> FUER "WEITER" BZW. <CR>
		3165	;FUER "ABBRUCH" ERWARTET UND ENTSPRECHEND VERFAHREN.
		3166	;
4CC5	3AC8FC	3167	LINE: LDA OFCC8H ;*RETURN, WENN DRUCKER AN
4CC8	3D	3168	DCR A ;*
4CC9	CB	3169	RZ ;*
		3170	;
4CCA	2155FA	3171	LXI H,LCOUNT ;ZEIGER AUF ZEILEN-ZAEHLER
4CCD	35	3172	DCR M ;ZEILEN-ZAEHLER MINUS EINS
4CCE	CO	3173	RNZ ;RETURN, WENN NICHT NULL
		3174	;
4CCF	CD7300	3175	CALL PTXTCR ;PRINT TEXT
4CD2	53504143	3176	DB 'SPACE = WEITER, CR = ABBRUCH',00
4CD6	45203D20		
4CD8	57454954		
4CDE	45522C20		
4CE2	4352203D		
4CE6	20414242		
4CEA	52554348		
4CEE	00		
4CEF	CD9E49	3177	CALL DESEL ;DESELECT LAUFWERK
4CF2	CD4300	3178	LINE0: CALL RCHAR ;LESE ZEICHEN
4CF5	FE0D	3179	CPI CR ;CR ?
4CF7	CACC48	3180	JZ ERET ;JA --> ABBRUCH
		3181	;
4CFA	FE20	3182	CPI ' ' ;SPACE ?
4CFC	C2F24C	3183	JNZ LINE0 ;NEIN --> LINE0
		3184	;
4CFF	CD6D00	3185	LINE1: CALL PTXT ;PRINT TEXT
4D02	0D	3186	DB CR
4D03	2D2D2D20	3187	DB '--- NAME --- TYP START (HEX) STOP (HEX)'
4D07	4E414D45		
4D0B	202D2D2D		
4D0F	20202054		
4D13	59502020		
4D17	20535441		
4D18	52542028		
4D1F	48455829		
4D23	20202053		
4D27	544F5020		
4D2B	28484558		
4D2F	29		
4D30	20202042	3188	DB ' BLOECKE'
4D34	4C4F4543		
4D38	4B45		
		3189	

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

LOC	OBJ	LINE	SOURCE STATEMENT
4D3A	0D	3190	DB CR,LF,00
4D3B	0A		
4D3C	00		
4D3D	3E0C	3191	MVI A,12 ;* RE-INIT ZEILENZAEHLER
4D3F	3255FA	3192	STA LCOUNT ;*
4D42	CD8B49	3193	CALL SELECT ;RE-SELECT LAUFWERK
4D45	CDAA49	3194	CALL DREADY ;LAUFWERK BEREIT ?
4D48	DC5848	3195	CC CODE3 ;NEIN --> CODE3
4D4B	C9	3196	RET
		3197	;
		3198	;*****
		3199	;UNTERPROGRAMM "LAUFW"
		3200	;FORDERT ZUR EINGABE DER LAUFWERKSBEZEICHNUNG (BZW. "M" FUER "MENUE") AUF.
		3201	;PRUEFT EINGABE UND BERECHNET DIE SELECT-MASKE (UNTERPROG. "DRIVE")
		3202	;
4D4C	CD7300	3203	LAUFW: CALL PTXTCR ;PRINT TEXT
4D4F	0A	3204	DB LF,'A = LAUFWERK A',LF,CR
4D50	41203D20		
4D54	4C415546		
4D58	5745524B		
4D5C	2041		
4D5E	0A		
4D5F	0D		
4D60	42203D20	3205	DB 'B = LAUFWERK B',LF,CR
4D64	4C415546		
4D68	5745524B		
4D6C	2042		
4D6E	0A		
4D6F	0D		
4D70	4D203D20	3206	DB 'M = MENUE',00
4D74	4D454E55		
4D78	45		
4D79	00		
4D7A	CD8E4D	3207	CALL DRIVE ;LESE BUCHSTABE
		3208	;PRUEFE AUF GUELTIGKEIT
		3209	;BERECHNE MASKE
4D7D	C9	3210	RET
		3211	;
		3212	

LOC	OBJ	LINE	SOURCE STATEMENT
		3213	*****
		3214	; UNTERPROGRAMM "TSTCHR / LETTER"
		3215	;
		3216	; EINSPRUNG "TSTCHR": PRUEFT OB ZEICHEN IM AKKU ZIFFER ODER BUCHSTABE.
		3217	; WENN JA, DANN CARRY = 1
		3218	;
		3219	; EINSPRUNG "LETTER": PRUEFT OB ZEICHEN IM AKKU BUCHSTABE.
		3220	; WENN JA, DANN CARRY = 1
		3221	;
4D7E	FE3A	3222	TSTCHR: CPI '9'+1 ;>= '9'+1 ?
4D80	D2874D	3223	JNC LETTER ;JA --> LETTER
4D83	FE30	3224	CPI '0' ;< '0' ?
4D85	3F	3225	CMC ;KOMPL. CARRY (C=1, ZIFFER)
4D86	D8	3226	RC ;RET, WENN ZIFFER
		3227	;
4D87	FE5B	3228	LETTER: CPI 'Z'+1 ;>= 'Z'+1 ?
4D89	D0	3229	RNC ;JA --> RETURN
4D8A	FE41	3230	CPI 'A' ;< 'A' ?
4D8C	3F	3231	CMC ;KOMPL. CARRY (C=1, BUCHSTABE)
4D8D	C9	3232	RET
		3233	;
		3234	*****
		3235	; UNTERPROGRAMM "DRIVE"
		3236	;
		3237	; Liest Zeichen von der Tastatur. Bei "M" Verzweigung zum "MENUE"
		3238	; Bei Eingabe von "A" bzw. "B" wird der Select-Code des Laufwerks
		3239	; berechnet. Andere Eingaben als "A", "B" oder "M" fuehren zu der
		3240	; Fehlermeldung: "FALSCH EINGABE"
		3241	;
4D8E	CDA14D	3242	DRIVE: CALL GETCHR ;ZEICHEN LESEN
4D91	FE4D	3243	CPI 'M' ;ZURUECK ZUM MENUE ?
4D93	CA8442	3244	JZ MENUE ;JA --> HAUPT-MENUE
4D96	D641	3245	DRIVE0: SUI 'A' ;AUS 'A' WIRD 0, AUS 'B' WIRD 1
4D98	FE02	3246	CPI 02H ;MAXIMAL-WERT UEBERSCHRITTEN ?
4D9A	D44F48	3247	CNC CODE0 ;JA --> FALSCH EINGABE
4D9D	CD8649	3248	CALL COMPSM ;BERECHNE SELECT-MASKE
4DA0	C9	3249	RET
		3250	;
		3251	



LOC	OBJ	LINE	SOURCE STATEMENT
		3252	*****
		3253	;UNTERPROGRAMM "GETCHR"
		3254	;FORDERT ZUR EINGABE EINES BUCHSTABENS AUF. EINGABEZEICHEN WERDEN
		3255	;GEPRUEFT. DABEI WERDEN NUR BUCHSTABEN AKZEPTIERT. KLEINBUCHSTABEN
		3256	;WERDEN IN GROSSBUCHSTABEN UMGEWANDELT. DER EINGEGEBENE BUCHSTABE
		3257	;WIRD ANGEZEIGT. DIE EINGABE KANN UEBER DIE TASTEN <BS> UND <DEL>
		3258	;KORRIGIERT WERDEN. DIE EINGABE MUSS DURCH <CR> ABGESCHLOSSEN WERDEN.
		3259	;
4DA1	CD7300	3260	GETCHR: CALL PTXTCR ;PRINT TEXT
4DA4	0A	3261	DB LF,'BITTE BUCHSTABE EINGEBEN: ',00
4DA5	42495454		
4DA9	45204255		
4DAD	43485354		
4DB1	41424520		
4DB5	45494E47		
4DB9	4542454E		
4DBD	3A20		
4DBF	00		
4DC0	CD4300	3262	GETCO: CALL RCHAR ;LESE ZEICHEN VON DER TASTATUR
4DC3	CDE90E	3263	CALL GROSS ;WANDE IN GROSS-BUCHSTABEN
4DC6	FE41	3264	CPI 'A' ;< 'A'
4DC8	DAC04D	3265	JC GETCO ;JA --> GETCO
4DCB	FE5B	3266	CPI 'Z'+1 ;> 'Z'
4DD1	D2C04D	3267	JNC GETCO ;JA --> GETCO
4DD0	3253FA	3268	STA CHAR ;RETTE ZEICHEN
4DD3	CD5200	3269	CALL WCHAR ;DRUCKE BUCHSTABEN
4DD6	CD4300	3270	GETC1: CALL RCHAR ;LESE ZEICHEN VON DER TASTATUR
4DD9	CDFF84D	3271	CALL TSTBS ;BS ODER DEL ? --> BS,SPACE,BS
4DDC	CAC04D	3272	JZ GETCO ;BS ODER DEL ? --> GETCO
4DDF	FE0D	3273	CPI CR ;CR?
4DE1	C2D64D	3274	JNZ GETC1 ;NEIN --> GETC1
4DE4	3A53FA	3275	LDA CHAR ;ZEICHEN NACH A
4DE7	C9	3276	RET
		3277	;
		3278	*****
		3279	; UNTERPROGRAMM "REPCHR"
		3280	; SETZT DAS ZEICHEN IM AKKU B-MAL IM SPEICHER EIN. HL DIENT ALS ZEIGER
		3281	; UND WIRD NACH JEDEM EINSETZEN UM EINS ERHOEHT.
		3282	;
		3283	; WERTE BEIM AUFRUF: A = EINZUSETZENDES ZEICHEN
		3284	; B = ANZAHL (WIE OFT EINSETZEN)
		3285	; HL = ZEIGER AUF ERSTE EINSETZ-STELLE
		3286	;
		3287	; WERTE NACH RETURN: B = 0
		3288	; HL = ZEIGER AUF LETZTE EINSETZ-STELLE + EINS
		3289	;
4DE8	77	3290	REPCHR: MOV M,A ;SETZE ZEICHEN EIN
4DE9	23	3291	INX H ;STELLE ZEIGER WEITER
4DEA	05	3292	DCR B ;ALLE ZEICHEN EINGESETZT ?
4DEB	C2E84D	3293	JNZ REPCHR ;NEIN --> REPCHR
4DEE	C9	3294	RET ;JA --> RETURN
		3295	

LOC	OBJ	LINE	SOURCE STATEMENT
		3296	;*****
		3297	; UNTERPROGRAMM "WAITSP"
		3298	; WARTESCHLEIFE, BIS <SPACE> BETAETIGT WIRD.
		3299	;
		3300	; WERTE BEIM AUFRUF: ---
		3301	; WERTE NACH RETURN: A = 20H (ASCII SPACE)
		3302	;
4DEF	CD4300	3303	WAITSP: CALL RCHAR ;LESE ZEICHEN VON TASTATUR
4DF2	FE20	3304	CPI ' ' ;SPACE ?
4DF4	C2EF41	3305	JNZ WAITSP ;NEIN --> WAITSP
4DF7	C9	3306	RET ;JA --> RETURN
		3307	;
		3308	; UNTERPROGRAMM "TSTBS"
		3309	; PRUEFT, OB DAS ZEICHEN IM AKKU = 0BH (BACKSPACE) ODER 7FH (DELETE) IST
		3310	; WENN JA, WIRD DIE ZEICHENFOLGE "BACKSPACE, SPACE, BACKSPACE" AUSGEGEBEN
		3311	; UND DAS ZERO-FLAG WIRD GESETZT.
		3312	;
		3313	; WERTE BEIM AUFRUF: A = ZU PRUEFENDES ZEICHEN
		3314	;
		3315	; WERTE NACH RETURN: (WENN NICHT BS ODER DEL) ! (WENN BS ODER DEL)
		3316	;
		3317	; A = ZU PRUEFENDES ZEICHEN ! A=0
		3318	; Z-FLAG = 0 (NICHT GESETZT) ! Z-FLAG = 1 (GESETZT)
		3319	;
4DF8	FE08	3320	TSTBS: CPI BS ;BS ?
4IFA	CA004E	3321	JZ TSTBS0 ;JA --> TSTBS0
4DFD	FE7F	3322	CPI DEL ;DEL ?
4DFF	C0	3323	RNZ ;WEDER BS NOCH DEL --> RET
4E00	CD6D00	3324	TSTBS0: CALL PTXT ;PRINT TEXT
4E03	08	3325	DB BS,SPACE,BS,00
4E04	20		
4E05	08		
4E06	00		
4E07	AF	3326	XRA A ;SET ZERO-FLAG
4E08	C9	3327	RET
		3328	;
		3329	;*****
		3330	; UNTERPROGRAMM "PPLUS"
		3331	; GIBT DAS ZEICHEN "+" AUS
		3332	;
4E09	CD6D00	3333	PPLUS: CALL PTXT ;PRINT "+"
4E0C	2B	3334	DB '+',00
4E0D	00		
4E0E	C9	3335	RET
		3336	;
		3337	

LOC	OBJ	LINE	SOURCE STATEMENT
		3338	*****
		3339	;UNTERPROGRAMM "PLINE"
		3340	;GIBT DAS ZEICHEN "-" AUS. DIE ANZAHL DER STRICHE, DIE AUSGEBEBEN WERDEN
		3341	;SOLL, MUSS IM D-REGISTER STEHEN.
		3342	;
4E0F	CD6D00	3343	PLINE: CALL PTXT ;PRINT "-"
4E12	2D	3344	DB '-' ,00
4E13	00		
4E14	15	3345	DOR D ;NOCH EIN "-" ?
4E15	C20F4E	3346	JNZ PLINE ;JA --> PLINE
4E18	C9	3347	RET
		3348	;
		3349	*****
		3350	;
		3351	; UNTERPROGRAMM "CRLF"
		3352	; GIBT DIE ZEICHENFOLGE CR, LF AUS
		3353	;
		3354	; WERTE BEIM AUFRUF: ---
		3355	; WERTE NACH RETURN: ---
		3356	;
4E19	CD7300	3357	CRLF: CALL PTXTCR ;PRINT CR,LF
4E1C	00	3358	DB 00
4E1D	C9	3359	RET
		3360	;
		3361	

LOC	OBJ	LINE	SOURCE STATEMENT
		3362	*****
		3363	;UNTERPROGRAMM "UMG"
		3364	;PRUEFT, OB DAS DOS ERWEITERT IST UND DAS UMGEHUNGSFLAG "EDH" GESETZT IST.
		3365	;
		3366	;WERTE BEIM AUFRUF: HL=ZEIGER AUF POSITION DES UMGEHUNGSFLAGS
		3367	;
		3368	;REAKTION: FLAG = EDH --> SPRUNG NACH (HL)+1 (EIN BYTE HINTER DEM FLAG)
		3369	;
		3370	FLAG (<) EDH --> HL=HL-800
		3371	ERNEUTER TEST
		3372	WENN "EDH" --> SPRUNG WIE OBEN
		3373	WENN NICHT "EDH" --> RETURN
		3374	;
4E1E	3EED	3375	UMG: MVI A,0EDH ;UMGEHUNGS-FLAG
4E20	BE	3376	CMF M ;UMGEHUNGS-FLAG GESETZT ?
4E21	CA2C4E	3377	JZ UMGO ;JA --> UMGO
		3378	;
4E24	C5	3379	PUSH B ;RETTE BC
4E25	0100F8	3380	LXI B,-800H ;* BERECHNE ADRESSE DES NAECHSTEN FLAGS
4E28	09	3381	DIAD B ;*
4E29	C1	3382	POP B ;RESTORE BC
4E2A	BE	3383	CMF M ;UMGEHUNGS-FLAG GESETZT ?
4E2B	C0	3384	RNZ ;NEIN --> RETURN
		3385	;
4E2C	23	3386	UMGO: INX H ;HL ZEIGT AUF SPRUNG-ZIEL
4E2D	E9	3387	PCHL ;SPRINGE NACH (HL)
		3388	;
		3389	

LOC	OBJ	LINE	SOURCE STATEMENT
		3390	;*****
		3391	;UNTERPROGRAMM "INTINT"
		3392	;INITIALISIERT DEN RAM-VEKTOR FUER DEN 5.5-INTERRUPT AUF "JMP ISR"
		3393	;
4E2E	3EC3	3394	INTINT: MVI A,JUMP
4E30	3295FC	3395	STA RSTVEK
4E33	213E4E	3396	LXI H,ISR
4E36	2296FC	3397	SHLD RSTVEK+1
4E39	3E0E	3398	MVI A,IMASK
4E3B	30	3399	SIM
4E3C	FB	3400	EI
4E3D	C9	3401	RET
		3402	;
		3403	;*****
		3404	;* *
		3405	;* INTERRUPT-SERVICE-ROUTINE *
		3406	;* *
		3407	;*****
		3408	;
		3409	; NACH DER AUSFUEHRUNG EINES KOMMANDOS LOEST DER FIC EINEN INTERRUPT AUS
		3410	; DIE CPU VERZWEIGT DANN ZU DIESER ROUTINE.
		3411	; DA SICH DIE CPU BEIM AUFRUF DER ISR DIE ADRESSE DES UNTERBROCHENEN PRO-
		3412	; GRAMMS IM STACK MERKT, WUERDE DIESES PROGRAMM NACH DEM RETURN-BEFEHL
		3413	; (IN DER ISR) FORTGESETZT.
		3414	; DURCH DEN BEFEHL "POP PSW" WIRD DIE RUECKSPRUNG-ADR ZUM UNTERBROCHENEN
		3415	; PROGRAMM AUS DEM STACK ENTFERNT. BEDINGT DURCH DIE STRUKTUR DES BFZ-
		3416	; MINI-DOS TRITT EIN INTERRUPT NUR DANN AUF, WENN DIE CPU EIN UNTERPROGRAMM
		3417	; ABARBEITET. WIRD NUN (NACH "POP PSW") DER RETURN-BEFEHL AUSGEFUEHRT, SO
		3418	; KEHRT DIE CPU INS HAUPTPROGRAMM ZURUECK.
		3419	; DAS INTERRUPT-SIGNAL, DAS VOM FIC AUSGEGEBEN WIRD, WIRD DURCH DAS
		3420	; LESEN DES FIC-STATUS ABGESCHALTET. DA ES DABEI ABER ZU ZEIT-PROBLEMEN
		3421	; KOMMEN KANN, WIRD IN DER ISR DER STATUS SOLANGE GELESEN BIS DER FIC
		3422	; NICHT-BUSY IST.
		3423	;
		3424	; DER STATUS STEHT NACH DEM RETURN IM AKKU UND IM E-REGISTER
		3425	; ER WIRD AUSSERDEM IM RAM UNTER "LASTST" GESPEICHERT. DAS BFZ-MINI-10S
		3426	; NUTZT DIESEN GESPEICHERTEN WERT NICHT. ER KANN ABER VON ANDEREN
		3427	; PROGRAMMEN VERWENDET WERDEN
		3428	;
		3429	; WERTE BEIM AUFRUF: ---
		3430	; WERTE NACH RETURN: A = STATUS
		3431	; E = STATUS
		3432	;
4E3E	F1	3433	ISR: POP PSW ;SAEUBERE STACK
4E3F	DBC0	3434	ISR0: IN STAT ;LESE STATUS, LOESCHE INTERRUPT-SIGNAL
4E41	5F	3435	MOV E,A ;STATUS NACH E
4E42	1F	3436	RAR ;BUSY ?
4E43	DA3F4E	3437	JC ISRO ;JA --> ISRO (LESE STATUS ERNEUT)
4E46	7B	3438	MOV A,E ;A = STATUS
4E47	3287FA	3439	STA LASTST ;RETTE STATUS
4E4A	FB	3440	EI
4E4B	C9	3441	RET
		3442	

LOC	OBJ	LINE	SOURCE STATEMENT
		3443	*****
		3444	; VEKTOR-TABELLE AUF DIE FEHLER-MELDUNGEN
		3445	; DIE ZIFFER GIBT DEN FEHLER-CODE AN
		3446	;
4E4C	10	3447	ERRVEK: DB 16 ;MAXIMALER FEHLER-CODE
4E4D	714E	3448	DW ERRO
4E4F	814E	3449	DW ERR1
4E51	934E	3450	DW ERR2
4E53	A24E	3451	DW ERR3
4E55	B84E	3452	DW ERR4
4E57	D34E	3453	DW ERR5
4E59	E04E	3454	DW ERR6
4E5B	EC4E	3455	DW ERR7
4E5D	F84E	3456	DW ERR8
4E5F	064F	3457	DW ERR9
4E61	174F	3458	DW ERR10
4E63	314F	3459	DW ERR11
4E65	4B4F	3460	DW ERR12
4E67	5D4F	3461	DW ERR13
4E69	6E4F	3462	DW ERR14
4E6B	7E4F	3463	DW ERR15
4E6D	A64F	3464	DW ERR16
4E6F	BD4F	3465	DW ERR17
		3466	;
		3467	*****
		3468	; FEHLERMELDIUNGEN. DIE ZIFFER HINTER DEM LABEL GIBT DEN FEHLER-CODE AN
		3469	;
4E71	46414C53	3470	ERR0: DB 'FALSCH EINGABE',00
4E75	43484520		
4E79	45494E47		
4E7D	414245		
4E80	00		
4E81	52554543	3471	ERR1: DB 'RUECKSTELL-FEHLER',00
4E85	4E535445		
4E89	4C4C2D46		
4E8D	45484C45		
4E91	52		
4E92	00		
4E93	53434852	3472	ERR2: DB 'SCHREIB-FEHLER',00
4E97	4549422D		
4E9B	4645484C		
4E9F	4552		
4EA1	00		
4EA2	4C415546	3473	ERR3: DB 'LAUFWERK NICHT BEREIT',00
4EA6	5745524B		
4EAA	204E4943		
4EAE	48542042		
4EB2	45524549		
4EB6	54		
4EB7	00		

3474

LOC	OBJ	LINE	SOURCE STATEMENT
4EB8	4449534B	3475 ERR4:	DB 'DISKETTE SCHREIBGESCHUETZT',00
4EEC	45545445		
4EC0	20534348		
4EC4	52454942		
4EC8	47455343		
4ECC	48554554		
4ED0	5A54		
4ED2	00		
4ED3	50525545	3476 ERR5:	DB 'PRUEF-FEHLER',00
4ED7	462D4645		
4EDB	484C4552		
4EDF	00		
4EE0	53554348	3477 ERR6:	DB 'SUCH-FEHLER',00
4EE4	2D464548		
4EE8	4C4552		
4EEB	00		
4EEC	4C455345	3478 ERR7:	DB 'LESE-FEHLER',00
4EF0	2D464548		
4EF4	4C4552		
4EF7	00		
4EF8	4449534B	3479 ERR8:	DB 'DISKETTE VOLL',00
4EFC	45545445		
4F00	20564F4C		
4F04	4C		
4F05	00		
4F06	554E4552	3480 ERR9:	DB 'UNERLAUBTER NAME',00
4F0A	4C415542		
4F0E	54455220		
4F12	4E414D45		
4F16	00		
4F17	46494C45	3481 ERR10:	DB 'FILE > 65535 (DEZ.) BYTES',00
4F1B	203E2036		
4F1F	35353335		
4F23	20284445		
4F27	5A2E2920		
4F2B	42595445		
4F2F	53		
4F30	00		
4F31	46494C45	3482 ERR11:	DB 'FILE NICHT IM VERZEICHNIS',00
4F35	204E4943		
4F39	48542049		
4F3D	4D205645		
4F41	525A4549		
4F45	43484E49		
4F49	53		
4F4A	00		
4F4B	46414C53	3483 ERR12:	DB 'FALSCHER FILE-TYP',00
4F4F	43484552		
4F53	2046494C		
4F57	452D5459		
4F5B	50		
4F5C	00		
		3484	

LOC	OBJ	LINE	SOURCE STATEMENT
4F5D	44495245	3485 ERR13:	DB 'DIRECTORY-FEHLER',00
4F61	43544F52		
4F65	592D4645		
4F69	484C4552		
4F6D	00		
4F6E	53504549	3486 ERR14:	DB 'SPEICHER-FEHLER',00
4F72	43484552		
4F76	2D464548		
4F7A	4C4552		
4F7D	00		
4F7E	53504549	3487 ERR15:	DB 'SPEICHERPLATZ AUF DER DISKETTE ZU KLEIN',00
4F82	43484552		
4F86	504C4154		
4F8A	5A204155		
4F8E	46204445		
4F92	52204449		
4F96	534B4554		
4F9A	5445205A		
4F9E	55204B4C		
4FA2	45494E		
4FA5	00		
4FA6	50524F47	3488 ERR16:	DB 'PROGRAMM-SPEICHER LEER',00
4FAA	52414D4D		
4FAE	2D535045		
4FB2	49434845		
4FB6	52204C45		
4FBA	4552		
4FBC	00		
4FBD	554E4755	3489 ERR17:	DB 'UNGUELTIGER FEHLER-CODE',00
4FC1	454C5449		
4FC5	47455220		
4FC9	46454B4C		
4FCD	4552D43		
4FD1	4F4445		
4FD4	00		
		3490 ;	
		3491	



LOC	OBJ	LINE	SOURCE STATEMENT
		3492	*****
		3493	;TYP-TABELLE
4FD5	04	3494	TYFT: DB 04 ;MAXIMALER TYF-CODE PLUS EINS
4FD6	4D4154	3495	DB 'MAT' ;CODE 1
4FD9	535053	3496	DB 'SPS' ;CODE 2
4FDC	424153	3497	DB 'BAS' ;CODE 3
4FDF	3F3F3F	3498	DB '???'
		3499	;
		3500	*****
		3501	; VERSATZTABELLE
		3502	;
4FE2	01	3503	VERTAB: DB 1,5,2,6,3,7,4,8
4FE3	05		
4FE4	02		
4FE5	06		
4FE6	03		
4FE7	07		
4FE8	04		
4FE9	08		
		3504	;
		3505	

LOC	OBJ	LINE	SOURCE STATEMENT			
		3506	;*****			
		3507	;* *			
		3508	;* RAM-RESERVIERUNGEN *			
		3509	;* *			
		3510	;*****			
		3511	;			
F800		3512				
		3513	ORG	OF800H		
		3514	;			
F800		3515	SECR:	DS	512	;SEKTOR-BUFFER
		3516	;-----			
FA00	0000	3517	LAENGE:	DW	0000	;LAENGE DES EINES FILES
		3518	;-----			
FA02	0000	3519	TEINT:	DW	0000	;ZEIGER FUER TEINT
		3520	;-----			
FA04	00	3521	ZAEHL:	DB	00	;ZAEHLER
		3522	;-----			
FA05	0000	3523	TEINT:	DW	0000	;EINTRAG 1
FA07	0000	3524	DW	0000		;EINTRAG 2
FA09	0000	3525	DW	0000		;EINTRAG 3
FA0B	0000	3526	DW	0000		;EINTRAG 4
FA0D	0000	3527	DW	0000		;EINTRAG 5
FA0F	0000	3528	DW	0000		;EINTRAG 6
FA11	0000	3529	DW	0000		;EINTRAG 7
FA13	0000	3530	DW	0000		;EINTRAG 8
FA15	0000	3531	DW	0000		;EINTRAG 9
FA17	0000	3532	DW	0000		;EINTRAG 10
FA19	0000	3533	DW	0000		;EINTRAG 11
FA1B	0000	3534	DW	0000		;EINTRAG 12
FA1D	0000	3535	DW	0000		;EINTRAG 13
FA1F	0000	3536	DW	0000		;EINTRAG 14
FA21	0000	3537	DW	0000		;EINTRAG 15
FA23	0000	3538	DW	0000		;EINTRAG 16
		3539	;-----			
FA25	0000	3540	TVORH:	DW	0000	;ZEIGER FUER TVORH
		3541	;-----			
FA27	00	3542	ZAEHLV:	DB	00	;ZAEHLER
		3543	;-----			
FA28	0000	3544	TVORH:	DW	0000	;EINTRAG 1
FA2A	0000	3545	DW	0000		;EINTRAG 2
FA2C	0000	3546	DW	0000		;EINTRAG 3
FA2E	0000	3547	DW	0000		;EINTRAG 4
FA30	0000	3548	DW	0000		;EINTRAG 5
FA32	0000	3549	DW	0000		;EINTRAG 6
FA34	0000	3550	DW	0000		;EINTRAG 7
FA36	0000	3551	DW	0000		;EINTRAG 8
FA38	0000	3552	DW	0000		;EINTRAG 9
FA3A	0000	3553	DW	0000		;EINTRAG 10
FA3C	0000	3554	DW	0000		;EINTRAG 11
FA3E	0000	3555	DW	0000		;EINTRAG 12
FA40	0000	3556	DW	0000		;EINTRAG 13
FA42	0000	3557	DW	0000		;EINTRAG 14
FA44	0000	3558	DW	0000		;EINTRAG 15
FA46	0000	3559	DW	0000		;EINTRAG 16
		3560	;			

LOC	OBJ	LINE	SOURCE STATEMENT
FA4B	0000	3561	ZWSP1: DW 0000 ;ZWISCHEN-SPEICHER
		3562	;
FA4A	0000	3563	ZWSP2: DW 0000 ;ZWISCHEN-SPEICHER
		3564	;
FA4C	0000	3565	ZWSP3: DW 0000 ;ZWISCHEN-SPEICHER
		3566	;
FA4E	0000	3567	ZWSP4: DW 0000 ;ZWISCHEN-SPEICHER
		3568	;
FA50	0000	3569	REST: DW 0000 ;REST-LAENGE EINES ZU SPEICHERNDEN FILES
		3570	;
FA52	00	3571	SPUR: DB 00 ;ZWISCHENSPEICHER FUER SPUR-NUMMER (0-79)
		3572	;
FA53	00	3573	CHAR: DB 00 ;ZWISCHENSPEICHER FUER EINGABE-ZEICHEN
		3574	;
FA54	00	3575	DSPLD: DB 00 ;FLAG (00 = SUCHE DIRECTORY EINTRAG,
		3576	; SONST: DISPLAY DIRECTORY)
		3577	;
FA55	00	3578	LDCOUNT: DB 00 ;ZEILENZAEHLER FUER DIRECTORY-ANZEIGE
		3579	;
FA56	0000	3580	VERPTR: DW 0000 ;ZWISCHENSPEICHER FUER ZEIGER AUF
		3581	;VERSATZTABELLE
		3582	;
FA58	0000	3583	EKRRET: DW 0000 ;RETURN-ADR VON ERROR-ROUTINE
		3584	;
FA5A	0000	3585	ERRP: DW 0000 ;ADR DES LETZTEN FEHLERS
		3586	;
FA5C	0000	3587	DOSRET: DW 0000 ;RETURN-ADR VON DOS
		3588	;
FA5E	0000	3589	PTYPT: DW 0000 ;ZEIGER AUF TYP-TABELLE
		3590	;
FA60	0000	3591	PEVT: DW 0000 ;ZEIGER AUF ERROR-VEKTOR-TABELLE
		3592	;
FA62	0000	3593	PVNR: DW 0000 ;ZEIGER AUF VERSIONS-NUMMER
		3594	;
FA64	0000	3595	PTZZ: DW 0000 ;ZEIGER AUF ZULAESSIGE ZEICHEN
		3596	;
FA66	0000	3597	PMENUE: DW 0000 ;ZEIGER AUF KOMMANDO-LISTE
		3598	;
FA68	00	3599	XSAV1: DB 00 ;OP-CODE
FA69	0000	3600	DW 0000 ;ADRESSE
		3601	;
FA6B	00	3602	XSTSP: DB 00 ;OP-CODE
FA6C	0000	3603	DW 0000 ;ADRESSE
		3604	;
FA6E	00	3605	XDIR: DB 00 ;OP-CODE
FA6F	0000	3606	DW 0000 ;ADRESSE
		3607	;
FA71	00	3608	XGP1: DB 00 ;OP-CODE
FA72	0000	3609	DW 0000 ;ADRESSE
		3610	;
FA74	00	3611	XGP2: DB 00 ;OP-CODE
FA75	0000	3612	DW 0000 ;ADRESSE
		3613	;
FA77	00	3614	XGP3: DB 00 ;OP-CODE
FA78	0000	3615	DW 0000 ;ADRESSE

LOC	OBJ	LINE	SOURCE STATEMENT
FA7A	00	3616	XLAD1: DB 00 ;OP-CODE
FA7B	0000	3617	DW 0000 ;ADRESSE
		3618	;
FA7D	00	3619	XLAD2: DB 00 ;OP-CODE
FA7E	0000	3620	DW 0000 ;ADRESSE
		3621	;
FA80	00	3622	XERR: DB 00 ;OP-CODE
FA81	0000	3623	DW 0000 ;ADRESSE
		3624	;
FA83	00	3625	EFROM: DB 00 ;ENTERED FROM. GIBT AUSKUNFT DARUEBER,
		3626	;VON WO DAS DOS AUFGERUFEN WURDE.
		3627	;01 = MAT85
		3628	;02 = SPS
		3629	;03 = BASIC
		3630	;
FA84	00	3631	SEITE: DB 00 ;ZWISCHENSPEICHER FUER SEITEN-NUMMER
		3632	;SEITE 0 = 0H, SEITE 1 = 1H
		3633	;
FA85	00	3634	SELMSK: DB 00 ;ZWISCHENSPEICHER FUER SELECT-MASKE
		3635	;LAUFWERK A = 1H, LAUFWERK B = 2H
		3636	;
FA86	00	3637	LSEL: DB 00 ;LAST SELECT. WERT, DER ALS LETZTER AN
		3638	; "PORT" AUSGEGEBEN WURDE
		3639	;
FA87	00	3640	LASTST: DB 00 ;LETZTER STATUS
		3641	;
FA88	00	3642	RETRY: DB 00 ;ZWISCHENSPEICHER FUER VERIFY-ZAEHLER
		3643	;
		3644	;

LOC	OBJ	LINE	SOURCE STATEMENT
3645			;***** ANHANG ZUM BFZ-MINI-DOS-LISTING *****
3646			;
3647			;DOS ERWEITERUNGEN MUESSEN WIE FOLGT AUFGEBAUT SEIN:
3648			;
3649			;ADRESSE (STUFE 1)      ADRESSE (STUFE 2)      INHALT              REMERKUNG
3650			;
3651			5000                      5800              ED                  1
3652			5001                      5801              C3                  2
3653			5002                      5802              XX
3654			5003                      5803              XX
3655			5004                      5804              ED                  3
3656			5005                      5805              C3                  4
3657			5006                      5806              XX
3658			5007                      5807              XX
3659			5008                      5808              ED                  5
3660			5009                      5809              C3                  6
3661			500A                      580A              XX
3662			500B                      580B              XX
3663			500C                      580C              ED                  7
3664			500D                      580D              XX                  8
3665			500E                      580E              XX                  9
3666			500F                      580F              ED                  10
3667			5010                      5810              XX                  11
3668			5011                      5811              XX                  12
3669			;
3670			;ERLAEUTERUNG:
3671			;
3672			1. ERKENNT MAT85 EINEN BEFEHL NICHT, UND ENTHAELT DIE UNTER 1. ANGEGE-
3673			BENE SPEICHERZEILE DEN WERTE "ED", SO FUERT DAS PROGRAMM DEN UNTER
3674			2. ANGEGEBENEN SPRUNG-BEFEHL AUS. DER IN DER JETZIGEN DOS-VERSION
3675			ENTHALTENE TEST AUF "F" WIRD UMGANGEN.
3676			;
3677			2. SIEHE 1. ZUSAETZLICH: WIRD EIN UNBEKANNTES MAT85-KOMMANDO EINGEGEBEN,
3678			SO PRUEFT DAS IOS, OB ES SICH UM DAS "F"-KOMMANDO HANDELT. IST DIES
3679			NICHT DER FALL, SO PRUEFT ES, OB UNTER 2. (STUFE 1) DER WERT "C3" STEHT.
3680			WENN JA, WIRD DER SPRUNG-BEFEHL AUSGEFUERT. NEIN --> FEHLERMELDUNG
3681			;
3682			3. ERKENNT SP1 (AUSSER BASIC --> SPS, MAT85+ EPROMMER) EINEN BEFEHL NICHT,
3683			UND ENTHAELT DIE UNTER 3. ANGEGERENE SPEICHERZEILE DEN WERT "ED", SO
3684			FUEHRT DAS PROGRAMM DEN UNTER 4. ANGEGEBENEN SPRUNG-BEFEHL AUS. DER
3685			IN DER JETZIGEN DOS-VERSION VORHANDENE TEST (WURDE "F" VON SPS AUS
3686			EINGEGEBEN) WIRD UMGANGEN.
3687			;
3688			4. SIEHE 3. ZUSAETZLICH: WIRD EIN UNBEKANNTES SP1 EINGEGEBEN, SO PRUEFT
3689			DAS IOS, OB ES SICH UM DAS "F"-KOMMANDO HANDELT UND OB DIESES KOMMANDO
3690			VON SPS AUS EINGEGEBEN WURDE. IST ES NICHT DAS "F"-KOMMANDO ODER WURDE
3691			ES NICHT VON SPS AUS EINGEGEBEN, PRUEFT DAS DOS, OB DIE UNTER 4. (STUFE 1)
3692			ANGEGERENE ADRESSE "C3" ANTHAELT. WENN JA, WIRD DER SPRUNG AUSGEFUERT.
3693			WENN NEIN, ERFOLGT EINE FEHLERMELDUNG.
3694			;

LOC	OBJ	LINE	SOURCE STATEMENT
3695	;	5.	ERKENNT BASIC EINEN BEFEHL NICHT, UND ENTHAELT DIE UNTER 6. ANGEGEBENE
3696	;		SPEICHERZEILE DEN WERT "ED", SO FUEHRT DAS PROGRAMM DEN UNTER 4. ANGE-
3697	;		GERENEN SPRUNG-BEFEHL AUS. DER IN DER JETZIGEN DOS-VERSION VORHANDENE
3698	;		TEST (WURDE "FLOPPY" IM DIREKTMODUS EINGEGEBEN) WIRD UMGANGEN.
3699	;		
3700	;	6.	SIEHE 5. ZUSAETZLICH: WIRD EIN UNBEKANNTES BASIC-KOMMANDO EINGEGEBEN,
3701	;		SO PRUEFT DAS DOS, OB ES DAS "FLOPPY"-KOMMANDO IST UND OB ES IM DIREKT-
3702	;		MODUS EINGEGEBEN WURDE. TRIFFT EINE DER BEIDEN BEDINGUNGEN NICHT ZU, SO
3703	;		PRUEFT DAS DOS, OB DIE UNTER 6. (STUFE 1) ANGEGEBENE ADRESSE "C3" ENT-
3704	;		HAELT. WENN JA, WIRD DER SPRUNG AUSGEFUEHRT. WENN NEIN, ERFOLGT EINE
3705	;		FEHLERMELDUNG.
3706	;		
3707	;	7.	EIN "ED" IN DER UNTER 7. ANGEGEBENEN ADRESSE ZEIGT DEM DOS AN, DASS NICHT
3708	;		DIE VEKTOR-TABELLE DER GRUND-VERSION GENUTZT WERDEN SOLL. DIE ADRESSE DER
3709	;		NEUEN VEKTOR-TABELLE MUSS UNTER 8. UND 9. ANGEGEBEN SEIN.
3710	;		
3711	;	8.	LOW-BYTE DER ADRESSE FUER DIE NEUE VEKTOR-TABELLE
3712	;		
3713	;	9.	HIGH-BYTE DER ADRESSE FUER DIE NEUE VEKTOR-TABELLE
3714	;		
3715	;	10.	EIN "ED" IN DER UNTER 10. ANGEGEBENEN ADRESSE ZEIGT DEM DOS AN, DASS NICHT
3716	;		DIE INFO-TABELLE DER GRUND-VERSION GENUTZT WERDEN SOLL. DIE ADRESSE DER
3717	;		NEUEN INFO-TABELLE MUSS UNTER 11. UND 12. ANGEGEBEN SEIN.
3718	;		
3719	;	11.	LOW-BYTE DER ADRESSE FUER DIE NEUE INFO-TABELLE
3720	;		
3721	;	12.	HIGH-BYTE DER ADRESSE FUER DIE NEUE INFO-TABELLE
3722	;		
3723	;		DIE ERWEITERUNGS-STUFE 2 IST DER STUFE 1 UEBERGEORDNET.
3724	;		EINE ERWEITERUNG MUSS MINDESTENS EIN "ED" BZW. MINDESTENS EINEN "C3 XX XX"-
3725	;		BEFEHL ENTHALTEN.
3726	;		
3727			END

## USER SYMBOLS

ADDSEC A 4A42	ANFEIN A 4B4D	ANZO A 4B16	ANZ1 A 4B0C	ANZAHL A 4B04	BAS A 00C0	BASBUF A 6013
BASEX0 A 417D	BASEXP A 4180	BASIC A 3091	BCKFLG A FCC7	BCLEAR A 3FD6	BDOS A 418F	BELL A 0007
BLAD1 A 41D3	BLAD2 A 41D7	BREAD A 0D0A	BS A 0008	BUFCLR A 0CAA	BUFFER A E000	CHAR A FA53
CHKLEN A 41C6	CHROK A 2841	CLEAR A 348C	CMD A 00C0	CMDINP A 27F0	CMFDH A 31FA	CODE0 A 4B4F
CODE1 A 4B52	CODE10 A 4B6D	CODE11 A 4B70	CODE12 A 4B73	CODE13 A 4B76	CODE14 A 4B79	CODE15 A 4B7C
CODE16 A 4B7F	CODE2 A 4B55	CODE3 A 4B5B	CODE4 A 4B5B	CODE5 A 4B5E	CODE6 A 4B61	CODE7 A 4B64
CODE8 A 4B67	CODE9 A 4B6A	COMP5M A 4986	CR A 000D	CREST A 0000	CRESTV A 0004	CRLF A 4E19
CRSEC A 00B0	CSEEK A 0014	CSTPIN A 0050	CTS A 4A40	CWRTRK A 00F0	CWSEC A 00A0	DAD0FS A 4E53
DAT A 00C3	DEB0 A 4B33	DEL A 007F	DELAY A 49B4	DELAY0 A 49B8	DELETE A 4AB9	DESEL A 499E
DIR A 42F5	DIRE A 42FB	DIRKOM A FD16	DISPL A 4C0B	DOS A 4204	DOS1 A 41IF	DOS2 A 41E2
DOS3 A 41E5	DOS4 A 41E8	DOS5 A 421E	DOSRET A FA5C	DREADY A 49AA	DRIVE A 4D8E	DRIVE0 A 4D96
ISPLD A FA54	EBEL A 4BFB	EBELO A 4C5E	ECHK A 4BA5	EFREI A 4BEB	EFROM A FAB3	ENTRY A 400F
ERASE A 437C	ERERR A 43B2	ERET A 4BCC	ERRO A 4E71	ERR1 A 4EB1	ERR10 A 4F17	ERR11 A 4F31
ERR12 A 4F4B	EKR13 A 4F5D	ERR14 A 4F6E	ERR15 A 4F7E	ERR16 A 4FA6	ERR17 A 4FB0	ERR2 A 4E93
ERR3 A 4EA2	ERR4 A 4EB8	ERR5 A 4ED3	ERR6 A 4EE0	ERR7 A 4EEC	ERR8 A 4EF8	ERR9 A 4F06
ERROR A 4BB4	ERROR1 A 4BA0	ERROR2 A 4BB9	ERRP A FA5A	ERRRET A FA5B	ERRVEK A 4E4C	EXEC A 01EB
FBAS A 414F	FDATA A 4511	FERTIG A 45DA	FLOPPY A 41B9	FM A 407F	FNAT A 406A	FORM1 A 4442
FORMAT A 43C7	FSFS A 40B9	FX A 4070	GETCO A 4DC0	GETC1 A 4DD6	GETCHR A 4DA1	GETNAM A 4B59
GETP0 A 49CC	GETPOS A 4A56	GETPUT A 49C6	GPO A 4A07	GFNXT A 49D7	GPOK A 49E1	GROFLG A FCC9
GROSS A 0EE9	GSTART A 4A63	GSTSP A 4A7F	HSTART A 0A74	HSTSPA A 215B	IMASK A 000E	INFOT A 42B1
INITD A 45F4	INITDO A 4606	INIT1 A 45F9	INTINT A 4E2E	ISR A 4E3E	ISRO A 4E3F	JUMP A 00C3
KENNOK A 469A	KMD A 0040	LAENGE A FA00	LASTE A 4B2A	LAUFW A 4D4C	LAUFW A 4D4C	LCCOUNT A FA55
LENSFS A 4132	LER A 465F	LETTER A 4D87	LF A 000A	LINE A 4CC5	LINE0 A 4CF2	LINE1 A 4CFF
LKENN A 468A	LOAD A 4659	LOOP A 4549	LOOP0 A 4547	LOOP10 A 456C	LOOP11 A 45B7	LOOP12 A 45C4
LOOP13 A 45A9	LSEL A FA86	M85BE A FCF2	MATLAD A 40B2	MAXTRY A 0003	MDEND A 4B00	MDIR A 4ACD
MDIRO A 4ADC	MDIR1 A 4AD6	MENUE A 42B4	MOVE A 4ABD	MOVE0 A 4A9E	MOVE1 A 4AA4	MREST A 00DC
MRSEC A 009C	MSEEK A 00D8	MVERI A 0018	MWRTRK A 00B4	MWSEC A 00DC	NEW A 47D5	NEXT A 4C7C
NEXT0 A 4C7B	NEXT1 A 4BE0	NEXT2 A 4BD2	NLEER A 4357	NOB0 A 4C13	NOISPL A 4C58	NTRY A 493E
NTYFZ A 4CBA	NXTCHR A 4B32	PBAS A 416A	PEVT A FA60	PGMANF A E0ED	PGMEND A E003	PHL A 005B
PKLIST A 03B8	PLINE A 4E0F	PMENUE A FA66	PORT A 00C4	POS A 4A29	POSIR A 4A0B	PPLUS A 4E09
PRTOFF A 0B9F	PRTST A FC84	PSTAR A 0228	PTEINT A FA02	PTVORH A FA25	PTXT A 006D	PTXTCR A 0073
PTYPT A FASE	PTZZ A FA64	PUNR A FA62	QUIT A 42F1	R4 A 31F4	R5 A 320B	RCAL A 4064
RCHAR A 0043	REINTR A 4C99	RELCAL A 4009	RELJMP A 400C	REPCHR A 4DE8	REST A FA50	RESTOR A 4BD6
RETRY A FAB8	RETURN A 00C9	RJMP A 4067	RRS A 494C	RRSO A 494E	RSEC A 4B82	RSECO A 4B86
RSTVEK A FC95	RWVSO A 492B	RWVSEC A 4926	SAVE A 4718	SAVE0 A 4B1F	SAVE2 A 4763	SAVER A 4721
SAVSEC A 493C	SD1 A 40F0	SD2 A 40F2	SD3 A 40F5	SD4 A 40FB	SDOS A 40FC	SDOS0 A 40E2
SEC A 00C2	SECB A F800	SECTOR A 44D8	SEEK A 4BEO	SEINTR A 4BB7	SEITE A FA84	SELECT A 498B
SELSK A FA85	SLAD1 A 413F	SLAD2 A 4148	SP1EXP A 40D2	SPACE A 0020	SPS A 272E	SPSDOS A 40DD
SPUR A FA52	SR A 0000	SRET1 A 27E9	STARTA A FD6F	STAT A 00C0	STEPIN A 4BD8	STOP A 000C
STOPA A FD71	SUB2 A 1039	TABOK A 402B	TEINT A FA05	TEST A 0EB0	TNAM0 A 4B83	TNAM1 A 4B91
TNAM2 A 4BA2	TRK A 00C1	TRKE0 A 453B	TRKENI A 4535	TSTBS A 4DF8	TSTBS0 A 4E00	TSTCHR A 4D7E
TSTNAM A 4B77	TVORH A FA2B	TXTBGN A 606F	TXUNT A 6064	TYF A 4CA1	TYPOK A 4CB0	TYPT A 4FD5
UMG A 4E1E	UMGO A 4E2C	VEINTR A 4B1C	VEKUF A 4000	VERIO A 4966	VERI1 A 497B	VERI2 A 497F
VERIFY A 4961	VERIX A 4975	VERPTR A FA56	VERTAB A 4FE2	VRET A 403E	VRET0 A 4031	VTAB A 403F
WAITSP A 4DEF	WARTE A 4909	WBIN A 005E	WBLNKI A 0B93	WCHAR A 0052	WDEZ A 0061	WPTST A 4A18

BFZ-MINI-DOS, VERSION 1.4, STAND 09.10.85, (C) BFZ ESSEN

---

WSEC	A 48E5	WSEC0	A 48E9	WTRK	A 48FF	WVSEC	A 4912	WVSEC0	A 4915	XDIR	A FA6E	XERR	A FA80
XGP1	A FA71	XGP2	A FA74	XGP3	A FA77	XLAD1	A FA7A	XLAD2	A FA7D	XSAV1	A FA68	XSTSP	A FA6B
ZAEHL	A FA04	ZAEHLV	A FA27	ZULKMD	A 28BE	ZWSP1	A FA48	ZWSP2	A FA4A	ZWSP3	A FA4C	ZWSP4	A FA4E

ASSEMBLY COMPLETE, NO ERRORS