

# SourceCode DTopDesk64

TopDesk64de  
Version: 28.12.2021

TopDesk - Entwickelt von

Walter Knupe, H.J. Ciprina, Volker Goehrke  
(c) 1991 GEOS-USER-CLUB, GbR

MegaCom Software  
(c) 1999-2000

FORUM64.de  
(w) 2001-2024

Erweiterung / DTopDesk64

M.Kanet

(w) 2024 / R02

Dieses Dokument erhebt keinen Anspruch auf Vollständigkeit, Fehler im Quellcode oder in den Kommentaren können nicht ausgeschlossen werden!

```
1
2 ;
3 ; SourceCode: opt.DTopDesk64.s
4 ;
5
6 ;*** OPTIONEN SEITE 1: LAUFWERKE ***
7 ;
8 ; Position für die Laufwerk-Icons:
9 ;
10 ; Anzeige des Panel links oder rechts.
11 ; Rechts entspricht TopDesk, links hat
12 ; den Vorteil das es für einen Klick
13 ; neben dem Laufwerk eine Begrenzung
14 ; gibt (Config-Programme starten).
15 ;
16 ; Standard:
17 ; PANEL_RIGHT=FALSE: Laufwerke links
18 ;
19 ; Alternativ:
20 ; PANEL_RIGHT=TRUE : Laufwerke rechts
21 ;
22 .PANEL_RIGHT          = FALSE
23
24 ; Laufwerksmodus neben Icons anzeigen:
25 ;
26 ; Wird die Option aktiviert (TRUE),
27 ; dann wird neben dem Laufwerk-Icon
28 ; der Modus (41/71/81/N) angezeigt.
29 ; Die Laufwerk-Icons selbst sind dann
30 ; nicht mehr Modus-spezifisch.
31 ;
32 ; Standard:
33 ; ENABLE_DRVMODE=TRUE : Modus zeigen
34 ;
35 ; Alternativ:
36 ; ENABLE_DRVMODE=FALSE: Icons anpassen
37 ;
38 .ENABLE_DRVMODE       = TRUE
39
40 ; Laufwerk-Icon für DnD verwenden:
41 ;
42 ; Für das Ghost-Icon für Drag-N-Drop
43 ; beim Laufwerkstausch das aktuelle
44 ; Laufwerk-Icon und die Farbe für die
45 ; Statuszeile verwenden.
46 ;
47 ; Standard:
48 ; ENABLE_DRVGHOST=TRUE : Laufwerk-Icon
49 ;
50 ; Alternativ:
51 ; ENABLE_DRVGHOST=FALSE: Disk-Icon
52 ;
53 .ENABLE_DRVGHOST      = TRUE
54
55 ; Fensteranimation EIN/AUS:
```

```

; 56
; Erzeugt beim öffnen und schließen 57
; von Laufwerkfenstern eine Animation. 58
; Funktioniert nicht mit der Option 59
; "Fenster automatisch platzieren". 60
; 61
; Standard: 62
; ENABLE_ANIMATE=TRUE : Animation 63
; 64
; Alternativ: 65
; ENABLE_ANIMATE=FALSE: Keine Animation 66
; 67
.ENABLE_ANIMATE = TRUE 68
69
;*** OPTIONEN SEITE 2: FARBEN/MENÜS *** 70
; 71
; Original-Farben verwenden: 72
; 73
; Werden die Original-Farben verwendet, 74
; dann wird die Farbe für die Standard- 75
; Dialogbox in MP3/GDOS64 durch das 76
; Programm nicht verändert. 77
; 78
; Standard: 79
; ENABLE_ORIGCOL=FALSE: DTopDesk-Farben 80
; 81
; Alternativ: 82
; ENABLE_ORIGCOL=TRUE : TopDesk-Farben 83
; 84
.ENABLE_ORIGCOL = FALSE 85
86
; Verschiedene Iconfarben: 87
; 88
; Die Einstellung "Verschiedene Farben" 89
; beinhaltet verschiedene Farben für 90
; alle Icons und den Fensterhintergrund 91
; für jedes Fenster. 92
; Hier ändert dann eine PadColorPref 93
; nur die Farben des aktiven Fensters. 94
; 95
; Für verschiedene Fensterfarben mit 96
; gleichen Iconfarben gibt es hier die 97
; Option "Verschiedene Iconfarben": 98
; Ist diese Option deaktiviert, dann 99
; werden für verschiedenfarbige Fenster 100
; für Icons die Farben von Fenster#1 101
; verwendet. Eine PadColorPref wird so 102
; auf alle Fenster angewendet. 103
; 104
; Standard: 105
; ENABLE_DIFFICOL=FALSE: Fenster+Icon 106
; 107
; Alternativ: 108
; ENABLE_DIFFICOL=TRUE : Fenster/Icon 109
; 110

```

```
111 .ENABLE_DIFFICOL      = FALSE
112
113 ; Optimierte Iconfarben:
114 ;
115 ; Die Originalfarben haben je nach
116 ; Fensterfarbe andere Iconfarben, wenn
117 ; die Kombination Icon- / Fensterfarbe
118 ; nicht gut erkennbar ist.
119 ;
120 ; Geänderte Iconfarben passend für alle
121 ; Standard-Fensterfarben verwenden.
122 ;
123 ; Standard:
124 ; ENABLE_NEWICOL=FALSE: Original
125 ;
126 ; Alternativ:
127 ; ENABLE_NEWICOL=TRUE : Neue Farben
128 ;
129 .ENABLE_NEWICOL        = FALSE
130
131 ; Menüs begrenzen:
132 ;
133 ; Wenn aktiviert (TRUE), dann kann der
134 ; Mauszeiger ein Untermenü nur nach
135 ; oben verlassen.
136 ;
137 ; Standard:
138 ; ENABLE_MENUUSE=FALSE: Nicht begrenzen
139 ;
140 ; Alternativ:
141 ; ENABLE_MENUUSE=TRUE : Menüs begrenzen
142 ;
143 .ENABLE_MENUUSE        = FALSE
144
145 ;*** OPTIONEN SEITE 3: ***
146 ;
147 ; GeoHelpView im Menü anzeigen:
148 ;
149 ; GeoHelpView ist ein GEOS-Programm um
150 ; Hilfetexte anzuzeigen.
151 ; Das Programm dürfte aber nur noch
152 ; selten genutzt werden bzw. ist auf
153 ; den meisten Systemdisketten nicht
154 ; mehr enthalten.
155 ;
156 ; Standard:
157 ; ENABLE_GEOHELP=FALSE: Deaktiviert
158 ;
159 ; Alternativ:
160 ; ENABLE_GEOHELP=TRUE : Aktiviert
161 ;
162 .ENABLE_GEOHELP         = FALSE
163
164 ; Bildschirm "Computer abschalten":
165 ;
```

```

; Nach dem beenden von GEOS wird ein
; Hinweis angezeigt, das der Computer
; jetzt abgeschaltet werden kann oder
; Leertaste drücken um in den BASIC-
; Modus zu wechseln.
; Diese TopDesk-Routine verwendet eine
; Vielzahl an Kernal-Routinen und ist
; komplexer als die GEOS-Routine.
; Der Verzicht auf den BASIC-Hinweis
; spart zudem Speicherplatz.
;
; Standard:
; ENABLE_BASICINFO=FALSE: Deaktiviert
;
; Alternativ:
; ENABLE_BASICINFO=TRUE : Aktiviert
;
ENABLE_BASICINFO    = FALSE

; Fenster optimieren:
;
; * EXPERIMENTEL *
;
; TopDesk kann Fenster nur teilweise
; neu zeichnen. Die Routine FrontWindow
; testen auf sich überlappende Fenster
; und zeichnet ggf. nur einen Teil des
; Fensters neu. Die Routine wurde aber
; nie komplettiert, daher werden die
; Fenster immer komplett gezeichnet.
; Funktioniert nur im Icon-Modus!
;
; Standard:
; ENABLE_OPTWIN=FALSE: Nicht optimieren
;
; EXPERIMENTEL:
; ENABLE_OPTWIN=TRUE : Fens. optimieren
;
ENABLE_OPTWIN      = FALSE

```

```
206
207 ;
208 ; SourceCode: opt.Author.s
209 ;
210
211 ; TD/Entwickler-Angaben:
212 ;             a "TD64 - FRIENDS"
213 ;             a "GEOSDev"
214
215 ;
216 ; SourceCode: DTopDeskI18N.DE.s
217 ;
218
219 ;*** Spracheinstellungen.
220 ;Der Wert für LANG wird als WORD
221 ;definiert, da der Wert mit TRUE/FALSE
222 ;kombiniert werden muss, z.B. DEBUG.
223 .LANG_DE           = $4000
224 .LANG_EN           = $8000
225 .LANG              = LANG_DE
226
227 ;
228 ; SourceCode: DTopDeskI18N.EN.s
229 ;
230
231 ;*** Spracheinstellungen.
232 ;Der Wert für LANG wird als WORD
233 ;definiert, da der Wert mit TRUE/FALSE
234 ;kombiniert werden muss, z.B. DEBUG.
235 .LANG_DE           = $4000
236 .LANG_EN           = $8000
237 .LANG              = LANG_EN
238
```

	239
;	240
; SourceCode: opt.DT64.Date.s	241
;	242
	243
; Build-Datum für DTopDesk64.	244
b "Build: 20241028"	245
	246
;	247
; SourceCode: opt.DT64DE.Build.s	248
;	249
	250
; GEOS-Klasse für DTopDesk64.	251
b "DT64-R02-DE V1.04"	252
	253
;	254
; SourceCode: opt.DT64DE.Class.s	255
;	256
	257
; GEOS-Klasse für DTopDesk64.	258
c "DT64-R02-DE V1.04"	259
	260
;	261
; SourceCode: opt.DT64EN.Build.s	262
;	263
	264
; GEOS-Klasse für DTopDesk64.	265
b "DT64-R02-EN V1.04"	266
	267
;	268
; SourceCode: opt.DT64EN.Class.s	269
;	270
	271
; GEOS-Klasse für DTopDesk64.	272
c "DT64-R02-EN V1.04"	273
	274

```

275
276 ;
277 ; SourceCode: src.DTopDesk64.s
278 ;
279
280 ; Label-Bezeichnung:
281 ;
282 ; :?xxxx : xxxx = Speicheradresse $0000-$FFFF.
283 ; :Z_____ : Internes Sprungziel Hauptmodul.
284 ; :z_____ : Internes Sprungziel VLIR-Modul.
285 ; :S_____ : Externes Sprungziel Hauptmodul.
286 ; :R_____ : Interner Vektor Hauptmodul (Adresstabelle).
287 ; :MR_____ : Sprungziel Menüroutinen.
288 ; :IR_____ : Sprungziel Iconroutinen.
289 ; :MT_____ : Text für Hauptmenü.
290 ; :T_____ : Text Hauptmodul.
291 ; :t_____ : Text VLIR-Modul.
292 ; :D_____ : Dialogbox Hauptmodul.
293 ; :d_____ : Dialogbox VLIR-Modul.
294 ; :x_____ : Sprungziel Branchbefehle.
295 ; :W_____ : Selbstmodifizierender Code Hauptmodul.
296 ; :w_____ : Selbstmodifizierender Code VLIR-Modul.
297 ; :V_____ : Variable/Register Hauptmodul.
298 ; :v_____ : Variable/Register VLIR-Modul.
299 ; :C_____ : TopDesk-Konfiguration Hauptmodul.
300 ; :F_____ : Dateiname.
301 ; :A_____ : Sprungziel Anzeigeroutine mod.#05/#06.
302 ; :N_____ : Einsprung aus Hauptmodul in VLIR-Modul.
303 ; :GR_____ : Grafikdaten (Sprites, Icons...)
304
305 if .p
306             t "TopSym"
307             t "TopSym.MP3"
308 ;           t "TopSym.ROM"
309             t "opt.DTopDesk64"
310             t "DTopDeskI18N.ext"
311
312 ; Nicht dokumentierter Einsprung in
313 ; den MegaPatch64-Kernal:
314 :TaskMan_NewJob      = $fe64
315 endif
316
317 ;--- TopDesk-Definitionen:
318 ;
319 ; Menüs CONSTRAINED/UN_CONSTRAINED:
320 if ENABLE_MENUMSE = TRUE
321 :MENU_VMODE          = CONSTRAINED
322 endif
323 if ENABLE_MENUMSE = FALSE
324 :MENU_VMODE          = UN_CONSTRAINED
325 endif
326
327 ; Horizontal immer UN_CONSTRAINED!
328 :MENU_HMODE          = UN_CONSTRAINED
329

```



; TopDesk-Ordner / GEOS-Dateityp:	330
.TDFOLDER = 11	331
.TDFOLDER_EXT = 23 ;Ersatz-Dateityp.	332
.TDFOLDER_POS0 = \$20	333
.TDSPLIT = PLAINTEXT	334
	335
; Hintergrundmuster für das Panel:	336
:PANEL_PATTERN = \$00	337
	338
; Farbe für "GEOS beenden":	339
.END_GEOS_COL = \$70	340
	341
; Max. Anzahl Dateien:	342
; TopDesk kann max, 8Bit an Dateien	343
; verwalten, wobei 0 = keine bedeutet.	344
; Es sind Datei-Nr. von 0-254 möglich.	345
; Eine Datei-Nr. 255 gibt es nicht, da	346
; der Wert in der Liste der markierten	347
; Dateien für "Nicht ausgewählt" steht.	348
.MAXFILES = 255	349
	350
; Max. Anzahl Dateien im Fenster.	351
.MAXWINFILES = 16	352
	353
; Max. Größe Ordnerliste.	354
.MAXDIRSIZE = 64	355
	356
; Zeilenhöhe Textmodus:	357
.TEXT_YOFF = 10	358
	359
; Abstand Icons:	360
.ICON_XOFF = \$0038	361
.ICON_YOFF = \$20	362
.ICON_SKIP_Y = 5	363
.ICON_SKIP_X = 6	364
	365
; Speicher für FileCopy:	366
;COPYMEM_MAX = \$1200	367
.COPYMEM_LOW = \$6f00	368
.COPYMEM_HIGH = \$7f00	369
	370
; Speicher für TD.WIN:	371
.BASE_TDWIN = \$7a00	372
	373
; Speicherübersicht \$7A00-\$7DF2:	374
; (siehe auch src.mod#00b!)	375
;--- TD.WIN	376
; \$7A00 - \$7AFF Ordnerliste #1-#4	377
; \$7B40 - \$7B43 Fenster-Status	378
; \$7B44 - \$7B49 Größe Fenster #1	379
; \$7B4A - \$7B4F Größe Fenster #2	380
; \$7B50 - \$7B55 Größe Fenster #3	381
; \$7B56 - \$7B5B Größe Fenster #4	382
; \$7B5C - \$7B5F Fensterstapel	383
; \$7C00 - \$7D37 Fenstertitel #1-#4	384

```

385 ;                4 x 80Byte = $0140.
386 ; $7DA0 - $7DAB Aktiver Ordner und
387 ;                8Byte-Ordnerliste.
388 ; $7DB0 - $7DB3 Fensterposition.
389 ; $7DB4 - $7DBB Lwfk./Part. Fenster #1-#4.
390 ; $7DF1          Anzeige Block/Kb.
391 ; $7DF2          Icon-/Text-Modus.
392 ; ---
393 .TDWIN_SUBDIR      = BASE_TDWIN
394 .TDWIN_WINDOWS     = BASE_TDWIN +$0140
395 .TDWIN_WSIZE1      = BASE_TDWIN +$0144
396 .TDWIN_WSIZE2      = BASE_TDWIN +$014a
397 .TDWIN_WSIZE3      = BASE_TDWIN +$0150
398 .TDWIN_WSIZE4      = BASE_TDWIN +$0156
399 .TDWIN_WSTACK      = BASE_TDWIN +$015c
400 .TDWIN_TITLE       = BASE_TDWIN +$0200
401 .TDWIN_SDINFO      = BASE_TDWIN +$03a0
402 .TDWIN_WINPOS      = BASE_TDWIN +$03b0
403 .TDWIN_DRVPART     = BASE_TDWIN +$03b4
404 .TDWIN_SIZEMODE    = BASE_TDWIN +$03f1
405 .TDWIN_VIEWMODE    = BASE_TDWIN +$03f2
406
407 ; Speicherbelegung RAM-TopDesk:
408 ; $0000-$00FF = VLIR-Daten.
409 ; $0100-$02FF = Backup EnterDeskTop
410 ; $0300-$03FF = Systemkonfiguration
411 ; $0400-$07FF = TD.WIN
412 ; $0800-$0FFF = Frei
413 ; $1000-$F000 = VLIR-Module
414 ; $F000-$FA3F = Fensterdaten #1/#2 oder #3/#4
415 ; $FA40-$FFFF = Frei
416 ;
417 ; Speicher für TD-VLIR-Info:
418 .TD_MAX_VLIR       = 12 +1 ;12xSubmodul + Endadresse.
419 .DACC_VLIRDAT      = $0000
420 .SIZE_VLIRDAT      = TD_MAX_VLIR *2
421
422 ; Speicher für EnterDeskTop-Routine:
423 .DACC_DTBAK        = $0100 ;Orig. EnterDeskTop-Routine.
424 .SIZE_DTBAK        = $0200 ;R2_SIZE_ENTER_DT = $0200!
425
426 ; Speicher für Systemkonfiguration:
427 :DACC_SYSVAR        = $0300 ;TopDesk64: $0200.
428 :BASE_SYSVAR        = diskBlkBuf +$86
429 :SIZE_SYSVAR        = diskBlkBuf +256 - BASE_SYSVAR
430
431 ; Speicher für TD.WIN:
432 .DACC_TDWIN         = $0400
433 .SIZE_TDWIN         = $03f3
434
435 ; Speicher für TD-Module:
436 .DACC_CORE          = $1000
437
438 ; Speicher für Dateilisten:
439 .DACC_DIRBUF        = $f000

```

.SIZE_DIRNAM	= 16*18 ;Name, Tr/Se Infoblock.	440
.SIZE_DIRICON	= 16*64 ;\$bf + Icondaten.	441
.SIZE_DIRBUF	= SIZE_DIRNAM + SIZE_DIRICON	442
		443
; Speicherbereiche in MP3/REU:		444
:R2A_BS_GRAFX	= \$5e10	445
:R2A_BS_COLOR	= \$5a28	446
		447
; VLIR-Module:		448
:VMod00	= 0	449
:VMod01	= 1	450
:VMod02	= 2	451
:VMod03	= 3	452
:VMod04	= 4	453
:VMod05	= 5	454
:VMod06	= 6	455
:VMod07	= 7	456
:VMod08	= 8	457
:VMod09	= 9	458
:VMod10	= 10	459
:VMod11	= 11	460
:VMod12	= 12	461
		462
; Zeiger auf Sprungtabelle:		463
:VJump0	= 0	464
:VJump1	= 1	465
:VJump2	= 2	466
:VJump3	= 3	467
:VJump4	= 4	468
:VJump5	= 5	469
:VJump6	= 6	470
:VJump7	= 7	471
		472
; Größe Dateiservice:		473
.SIZE_FSERVICE	= \$5d0	474
		475
; Funktionen Fensterverwaltung:		476
:WN_REDRAW	= 1	477
:WN_CLOSE	= 4	478
:WN_ACTIVATE	= 6	479
:WN_RESTORE	= 7	480
:WN_SCROLL_U	= 10	481
:WN_SCROLL_D	= 11	482
:WN_HIDE	= 12	483
:WN_USER	= 13	484
:WN_ACTIVATE2	= 14	485
		486
; --- Bildschirmgrenzen:		487
.SCREEN_X0	= \$0000	488
.SCREEN_XW	= \$0140	489
.SCREEN_X1	= SCREEN_X0 +SCREEN_XW -1	490
.SCREEN_Y0	= \$00	491
.SCREEN_YH	= \$c8	492
.SCREEN_Y1	= SCREEN_Y0 +SCREEN_YH -1	493
		494

```

495 ;--- Panel-Koordinaten:
496 .PANEL_XW          = $30
497 .PANEL_YH          = $b8
498
499 .DESK_XW           = $0110
500 .DESK_YH           = $b8
501
502 .PANEL_Y0           = $10
503 .PANEL_Y1           = PANEL_Y0 + PANEL_YH -1
504
505 .DESK_Y0            = $10
506 .DESK_Y1            = DESK_Y0 + DESK_YH -1
507
508 if PANEL_RIGHT = TRUE
509 .PANEL_X0            = $0000 + DESK_XW
510 .PANEL_X1            = PANEL_X0 + PANEL_XW -1
511 .DESK_X0             = $0000
512 .DESK_X1             = DESK_X0 + DESK_XW -1
513 else
514 .PANEL_X0            = $0000
515 .PANEL_X1            = PANEL_X0 + PANEL_XW -1
516 .DESK_X0             = PANEL_X0 + PANEL_XW
517 .DESK_X1             = DESK_X0 + DESK_XW -1
518 endif
519
520 ; Bereich Laufwerk-Icons:
521 .DRIVE_XW            = 24
522 .DRIVE_YH            = 24
523 .DRIVE_X0            = PANEL_X0
524 :DRIVE_X1            = PANEL_X1
525 .DRIVE_Y0            = PANEL_Y0
526 :DRIVE_Y1            = PANEL_Y0 +8 +4*DRIVE_YH +8 -1
527
528 ; Anzahl gew. Dateien:
529 .SLCT_X0             = $00d0
530 :SLCT_XW             = $0010
531 .SLCT_X1             = SLCT_X0 +SLCT_XW -1
532 .SLCT_Y0             = $00
533 :SLCT_YH             = $10
534 .SLCT_Y1             = SLCT_Y0 +SLCT_YH -1
535
536 ; Datum/Uhrzeit:
537 .TIME_X0             = $00f0
538 :TIME_XW             = $0050
539 .TIME_X1             = TIME_X0 +TIME_XW -1
540 .TIME_Y0             = $00
541 :TIME_YH             = $10
542 .TIME_Y1             = TIME_Y0 +TIME_YH -1
543
544 ; GEOS/Task-Icon:
545 .TASK_X0             = $00e0
546 :TASK_XW             = $0010
547 .TASK_X1             = TASK_X0 +TASK_XW -1
548 .TASK_Y0             = $00
549 :TASK_YH             = $10

```

```

.TASK_Y1          = TASK_Y0 +TASK_YH -1          550
                                                         551
; GEOS-Menü:                                           552
.MENU_X0          = $0000                          553
.MENU_XW          = $00d0                          554
.MENU_X1          = MENU_X0 +MENU_XW -1            555
.MENU_Y0          = $00                            556
:MENU_YH          = $10                            557
.MENU_Y1          = MENU_Y0 +MENU_YH -1            558
                                                         559
; Statusanzeige:                                       560
.STATUS_X0        = MENU_X0                        561
.STATUS_XW        = MENU_XW                        562
.STATUS_X1        = STATUS_X0 +STATUS_XW -1        563
.STATUS_Y0        = MENU_Y0                        564
:STATUS_YH        = MENU_YH                        565
.STATUS_Y1        = STATUS_Y0 +STATUS_YH -1        566
                                                         567
; Trash-Icon:                                          568
.TRASH_X0         = PANEL_X0 +2*8                  569
:TRASH_XW         = $0018                          570
.TRASH_X1         = TRASH_X0 +TRASH_XW -1           571
.TRASH_Y0         = $80                            572
:TRASH_YH         = $18                            573
.TRASH_Y1         = TRASH_Y0 +TRASH_YH -1           574
                                                         575
; Drucker-Icon:                                       576
.PRINT_X0         = PANEL_X0 +2*8                  577
:PRINT_XW         = $0018                          578
.PRINT_X1         = PRINT_X0 +PRINT_XW -1           579
.PRINT_Y0         = $a0                            580
:PRINT_YH         = $18                            581
.PRINT_Y1         = PRINT_Y0 +PRINT_YH -1           582
                                                         583
; Offset für Titelzeile:                             584
; Beim ändern der Werte auch DrawWindow              585
; entsprechend anpassen!                             586
:Y0 = $00                                             587
:YU = %100000000                                    588
:XL = $00                                             589
:XR = %010000000                                    590
:NP = $00                                             591
:EL = %001000000                                    592
:SX = 8                                               593
:TY = 8                                               594
:SY = 8                                               595
                                                         596
; Bereich Titelzeile.                                597
:TOL = 14 +8                                         598
:TOR = 20                                             599
:T00 = 1                                              600
:T0U = TY -1                                         601
                                                         602
; Offset für Statuszeile.                             603
:SUP = 10                                             604

```

```

605 :SUL = SUP +8
606
607 ; Höhe Scrollpfeile.
608 :HSC = 9
609
610 ; TopDesk-Definitionen:
611 .OFF_DIR_NUM      = $75
612 :MIN_HEIGHT      = 8*5
613 :MIN_WIDTH       = 8*8
614 .DISK_SIZE       = $0d80
615 .DISK_BASE_REU   = $8300
616
617 ; TopDesk-Statuszeile:
618 .STAT_COPY       = $00
619 .STAT_MOVE       = $01
620 .STAT_DEL        = $02
621 .STAT_VERIFY     = $03
622 .STAT_UNDEL      = $04
623 .STAT_CREATE     = $05
624
625 ; TopDesk-Farben:
626 if ENABLE_ORIGCOL = TRUE
627 .NAVBAR_COL      = $12 ;Farbe Navigation in Dialogbox.
628 .DBFILE_COL      = $0f ;Farbe Dateifenster in Dialogbox.
629 .PARTNR_COL      = $0f ;Farbe Part.Eingabe in Dialogbox.
630 .TDINFO_COL      = $67 ;Farbe TopDesk-Infobox.
631 .TDEXIT_COL      = $03 ;Farbe für "GEOS beenden"-DBox.
632 endif
633
634 ; Fehlercodes.
635 :BAD_SCRN_RES     = $0f ;Nur GEOS128/80Zeichen.
636 .TDLIST_FULL      = $40
637 ;ERR_NOT_OVR      = $41 ;TD1x: DB="Nicht überschreiben".
638 .FILE_EXISTS      = $41
639 :TDSUB_EXISTS     = $42
640 :ERR_MULTIFILE    = $43
641 :ERR_TDFOLDER     = $44
642 :ERR_UNKNOWN      = $45
643 :ERR_NO_DOC       = $46
644 .ERR_NO_APPL      = $47
645 .DIR_NOT_EMPTY    = $48
646 :ERR_WR_PROT      = $49
647 :ERR_NO_WINDOW    = $4a
648 :TDLIST_RELOAD    = $50
649 .PART_ERR         = $64
650 :ERR_41DBLSIDE    = $80
651 :CMD_SUBDIR       = $c6
652
653 ; Offset für Datei öffnen/mod.#09.
654 .OFF_OPEN_APPL    = 1
655 .OFF_OPEN_DA      = 2
656 .OFF_OPEN_DOC     = 3
657 .MAX_OPEN_DOC     = 7 ;3-9
658 .OFF_OPEN_PRNT    = 10
659 .OFF_OPEN_INPT    = 11

```

```

.OFF_OPEN_OTHER      = 12                                660
.MAX_OPEN_OTHER      = 2      ;12/13                      661
                                                                662
; Hinweis:                                                663
; Alle Laufwerk-Icons haben nur eine                    664
; Höhe von 20 Pixelreihen!                              665
.DRVICON_Y           = 20      ;20 Pixel hoch.           666
.DRVICON_X           = 3      ; 3 Cards breit.            667
                                                                668
; Hinweis:                                                669
; Warum ursprünglich 80Byte/Icon?                      670
;                                                        671
; Ein Icon hat mit Kopfbyte max.                        672
; => 1 +3*20 = 61 Byte.                                  673
.DRVICON_SIZE        = 80                                674
.DRVICON_SIZE        = 64      ;Inkl. Reserve für 1+3*21=Sprite. 675
                                                                676
                                n "obj.#00"                677
                                a "GEOSDev"                678
                                f APPLICATION               679
                                                                680
if LANG = LANG_DE                                          681
                                t "opt.DT64DE.Class"        682
endif                                                        683
if LANG = LANG_EN                                          684
                                t "opt.DT64EN.Class"        685
endif                                                        686
                                                                687
                                o APP_RAM                   688
                                p MAININIT                  689
                                                                690
                                z $80      ;Nur GEOS64.      691
                                                                692
                                i                            693
<MISSING_IMAGE_DATA>                                       694
                                                                695
; Programm-Icon:                                           696
;01: "#####"                                              697
;02: "#...#...#...#...#...#"                             698
;03: "#####..#####"                                     699
;04: "#.....#.."                                          700
;05: "#.##..#####..#"                                    701
;06: "#.##..#####..#"                                    702
;07: "#.....#####..#"                                    703
;08: "#.##..#.....#...#"                                  704
;09: "#.##..#.....#..#"                                  705
;10: "#.....#.....#..#"                                  706
;11: "#.##..#####..#"                                    707
;12: "#.##.....#.....#..#"                              708
;13: "#.....###.##.##.##..#"                             709
;14: "#.##...###.##.##.##..#"                             710
;15: "#.##...###.....#..#"                                711
;16: "#.....#.#####..#"                                  712
;17: "#.##...#.##.##.##..#"                                713
;18: "#.##...#.....#...#"                                714

```

```
715 ;19: "#.##...#####..."
716 ;20: "#.....#"
717 ;21: "#####..."
718
719         t "inc.mod#00a"
720         t "inc.mod#00b"
721         t "inc.mod#00c"
722
723 :MAIN_TMP      = ENDDBUF      ;Ende Fensterspeicher.
724 .MAIN_END      = MAIN_TMP      ;Ende Hauptmodul.
725 .BASE_MOD_A     = MAIN_END      ;Ladeadresse VLIR-Module.
726 .BASE_MOD_B     = CORE_END      ;Erw. Ladeadr. VLIR-Module.
727
```



```

728
729
730 ; SourceCode: inc.FKeyApps.s
731 ;
732
733 ;--- Dateinamen für F-Tasten und
734 ;   Langer Mausklick neben Laufwerk:
735 ;
736 ; Format:          b "<FILENAME>"
737 ;                  e xxx +17
738 ;
739 ; Nur Dateiname nach dem "b" ändern,
740 ; es sind max.16 Zeichen möglich!
741 ; Ende mit $00-Byte, wird durch den
742 ; Opcode "e" automatisch gesetzt!
743 ;
744 ; Langer Mausklick neben Laufwerk-Icon
745 ; startet das zusätzliche Programm zur
746 ; Laufwerkskonfiguration.
747 ;
748 ;--- LONG ----- "1234567890123456" ;--- LONG -----
749 .V4122          b "geoHDscsi" ;<- Max.16 Zeichen!
750                  e V4122 +17 ;<- Nicht ändern!
751
752 ;--- F2 ----- "1234567890123456" ;--- F2 -----
753 ; TopDesk:      "MegaAss V4.1"
754 .V4132          b "" ;<- Max.16 Zeichen!
755                  e V4132 +17 ;<- Nicht ändern!
756
757 ;--- F4 ----- "1234567890123456" ;--- F4 -----
758 ; TopDesk:      "geoDump"
759 .V4142          b "" ;<- Max.16 Zeichen!
760                  e V4142 +17 ;<- Nicht ändern!
761
762 ;--- F6 ----- "1234567890123456" ;--- F6 -----
763 ; TopDesk:      "Progrmmrs_Calc"
764 .V4152          b "" ;<- Max.16 Zeichen!
765                  e V4152 +17 ;<- Nicht ändern!
766
767 ;--- F8 ----- "1234567890123456" ;--- F8 -----
768 ; TopDesk:      "GeoPATCH 128-MP"
769 .V4162          b "" ;<- Max.16 Zeichen!
770                  e V4162 +17 ;<- Nicht ändern!
771
772 ;---
773

```

```

774
775 ;
776 ; SourceCode: inc.mod#00a.s
777 ;
778
779 .C0400          s 17          ;Dateiname.
780 if LANG = LANG_DE
781 :C0411          t "opt.DT64DE.Build"      ;GEOS-Klasse.
782 endif
783 if LANG = LANG_EN
784 :C0411          t "opt.DT64EN.Build"      ;GEOS-Klasse.
785 endif
786              b NULL
787
788 .V0422          b $00          ;Aktuelles VLIR-Mod.
789 :V0423          b $00          ;$FF = DnD/Laufwerk.
790              ;$01 = DnD/MultiFile.
791 .C0424          b $00          ;TopDesk/Laufwerk.
792
793 ; Position der VLIR-Daten in DACC,
794 ; siehe auch Z072C = NewGetModule.
795 ;---
796 :V0425          w $0000          ;VLIR-Adressbereich.
797              ; => BASE_MOD_A/B.
798              w DACC_VLIRDAT
799              w SIZE_VLIRDAT
800
801 .C0427          b $00          ;64K-Bank RAMDesk.
802 ;---
803
804 .C0428          b $00          ;>$00=RAM-TopDesk.
805 .C0429          b $00          ;>$00=System in REU.
806
807 ;--- TD.WIN
808 ; Fensterstapel.
809 .C042A          b $00          ;Aktives Fenster.
810              b $02
811              b $03
812              b $01
813 ;---
814
815 ; MessageBuffer für Fensterfunktionen.
816 ;---
817 :WIN042E        b $00          ;Fenster-Befehl.
818 :WIN042F        b $00          ;Aktives Fenster.
819 ;---
820
821 .BASE_COLCFG
822 ;---
823 .C0438          b $01          ;TopDesk in Farbe.
824
825 if ENABLE_ORIGCOL = TRUE
826 .COL0439        b $bf          ;GEOS-Farbe.
827 .COL043A        b $0c          ;Laufwerk.
828 .COL043B        b $7c          ;akt.Laufwerk.

```

```

.COL043C      b $0d  ;Dialogbox.      829
.COL043D      b $04  ;GEOS-Icon.      830
.COL043E      b $12  ;Fehlerbox.      831
.COL043F      b $0e  ;Infobox.        832
.COL0440      b $6c  ;Trash-Icon.     833
.COL0441      b $1c  ;Drucker-Icon.   834
.COL0442      b $07  ;mark.Dateien.    835
.COL0443      b $0d  ;Datum/Uhrzeit.   836
.COL0444      b $03  ;GEOS-Menü.      837
.COL0445      b $12  ;Statusanzeige.   838
else          839
; Hinweis:    840
; Werden hier Farben geändert, dann      841
; auch Werte in src.mod#12/v74AC für      842
; "Farben ändern/Reset" anpassen!        843
.COL0439      b $bf  ;GEOS-Farbe.      844
.COL043A      b $0c  ;Laufwerk.        845
.COL043B      b $dc  ;akt.Laufwerk.     846
.COL043C      b $0d  ;Dialogbox.      847
.COL043D      b $03  ;GEOS-Icon.      848
.COL043E      b $12  ;Fehlerbox.      849
.COL043F      b $0e  ;Infobox.        850
.COL0440      b $0c  ;Trash-Icon.     851
.COL0441      b $0c  ;Drucker-Icon.   852
.COL0442      b $07  ;mark.Dateien.    853
.COL0443      b $0d  ;Datum/Uhrzeit.   854
.COL0444      b $03  ;GEOS-Menü.      855
.COL0445      b $07  ;Statusanzeige.   856
endif        857
;---        858
        859
; Hinweis:    860
; Zeiger auf Fensterdaten für das        861
; aktuelle Fenster.                      862
.WIN0446      w $0000                  ;Zeiger Fensterdaten. 863
        864
;--- TD.WIN  865
.C0448        s $04                  ;$FF = Fenster aktiv. 866
;---        867
        868
.V044C        w $0000                  ;Zeiger appMain-Rout. 869
.V044E        w $0000                  ;Zeiger DoIcons-Tab.  870
:V0450        b C04BC - C04BC          ;Offset für Fenster  871
               b C04C7 - C04BC          ;#1 bis #4 auf die   872
               b C04D2 - C04BC          ;Fenstertabelle.    873
               b C04DD - C04BC          874
:V0454        b $00                  ;1=Fens. n. verdeckt. 875
:V0455        s $06                  ;Zwischenspeicher    876
               ;GEOS-Fenstergrenzen.    877
.V045B        w $0000                  ;GEOS/mouseVector.  878
:V045D        b $00                  ;0=Iconzeile zeigen. 879
.V045E        w $0000                  ;GEOS/RecoverVector. 880
               881
:V0460        s $06                  ;Buf1/Fenstergrenzen. 882
:V0466        s $06                  ;Buf2/Fenstergrenzen. 883

```

```

884
885 :V046C          b $00          ;$00=Buf1/$FF=Buf2.
886 .V046E          b $00          ;Eine/alle D.rennen.
887 :V046F          b $00          ;$FF=TaskMan starten.
888
889 ; Name akt. Datei.
890 .V0470          b $00,$00      ;Dateiname.
891
892 ; Neuer Diskname.
893 ; Hinweis:
894 ; Wird nur in mod.#12 verwendet.
895 .V0472          s 17          ;Format: Diskname.
896
897 ; Name Source-Disk.
898 .V0483          b $00,$00      ;"x:"+Diskname.
899          s 17
900
901 ; Name Target-Disk.
902 .V0496          b $00,$00      ;Zwischenspeicher
903          s 17          ;Datei-/Diskname.
904
905 .V04A9          b $00          ;$00=Copy/$01=Move.
906 .V04AA          b $00          ;Akt. Datei-Nr.
907 .V04AB          b $00          ;$01=Abbruch.
908
909 ; Hinweis:
910 ; Anfang/Ende CopyBuffer.
911 .MHB04AC        b > COPYMEM_HIGH ;HB Ende CopyBuf.
912
913 ; Hinweis:
914 ; Alter Wert aus TopDesk V1.3. Der
915 ; untere Grenzwert wird später durch
916 ; ein Unterprogram gesetzt.
917 .MLB04AD        b > $6000      ;HB Anfang CopyBuf.
918 ;          b > COPYMEM_LOW
919
920 :V04AE          b $00          ;>0=Klick auf Gadget.
921 .V04AF          b $00          ;$01=Neue PadColPref.
922 .V04B0          b $00          ;Zähler Instanz/Copy.
923 .V04B1          b $00          ;>0=Ziel ersetzen.
924 ;--- TD.WIN
925 .C04B2          b $00          ;Icon-/Text-Modus:
926          ;$00 = Icons
927          ;$01 = Text
928          ;$02 = Sort./Name
929          ;$03 = Sort./Datum
930          ;$04 = Sort./Größe
931          ;$05 = Sort.Typ
932 ;---
933 :V04B3          b $00          ;ShortCut-Flag.
934 :V04B4          b $00          ;$FF=Kop.+Ersetzen.
935 :C04B5          b $ff          ;$FF=Kopieren-Dialog.
936
937 .V04B7          b $00          ;Anzahl mark.Dateien.
938

```

```

; Geräteinformation Laufwerk A-D: 939
; $00 = 1541/71/81 940
; $01 = RAMLink 941
; $02 = RAMDrive 942
; $04 = FD2000 943
; $08 = FD4000 944
; $10 = CMD-HD 945
; $80 = 64Net 946
.V04B8 s 4 ;Geräte-Information. 947
948
:WIN_HEIGHT = $58 949
950
; Tabelle mit Fensterdaten. 951
;--- TD.WIN 952
.C04BC b DESK_Y0 +$00,DESK_Y0 +$00 +WIN_HEIGHT -1 953
w DESK_X0,DESK_X1 954
;--- 955
b $ff 956
w C0543 ;Titel Fenster #1. 957
w Z4923 ;Window-Handler. 958
959
;--- TD.WIN 960
.C04C7 b DESK_Y0 +$60,DESK_Y0 +$60 +WIN_HEIGHT -1 961
w DESK_X0,DESK_X1 962
;--- 963
b $ff 964
w C0593 ;Titel Fenster #2. 965
w Z4923 ;Window-Handler. 966
967
;--- TD.WIN 968
.C04D2 b DESK_Y0 +$08,DESK_Y0 +$08 +WIN_HEIGHT -1 969
w DESK_X0,DESK_X1 970
;--- 971
b $ff 972
w C05E3 ;Titel Fenster #3. 973
w Z4923 ;Window-Handler. 974
975
;--- TD.WIN 976
.C04DD b DESK_Y0 +$10,DESK_Y0 +$10 +WIN_HEIGHT -1 977
w DESK_X0,DESK_X1 978
;--- 979
b $ff 980
w C0633 ;Titel Fenster #4. 981
w Z4923 ;Window-Handler. 982
983
; Speicherbereiche für max. 2 Fenster. 984
; Mit einer REU wird der Bereich für 985
; 2x2 Fenster verwendet. 986
:V04E8 w DIRBUF1 987
w DIRBUF2 988
w DIRBUF1 989
w DIRBUF2 990
991
.V04F0 s 4 ;Dir.T. für Fenster. 992
.V04F4 s 4 ;Dir.S. für Fenster. 993

```

```

994
995 ;--- TD.WIN
996 .C04F8          s 4          ;Lfwk. für Fenster.
997 .C04FC          s 4          ;Part. für Fenster.
998 ;---
999
1000 .C0508          b $00        ;1=RL installiert.
1001 .C0509          b $00        ;RAMLink-Adresse.
1002 .C050A          b $00        ;TopDesk/Partition.
1003 ;--- TD.WIN
1004 .C050B          s 4          ;Fensterposition.
1005 ;---
1006 .V050F          s 4          ;Anz.Dat. Fenster.
1007 .V0513          s 4          ;Anz.Dat. auf Disk.
1008 :V0517          s 4          ;Anz.Dat. ab Offset.
1009 ;--- TD.WIN
1010 .C051B          s 4          ;Aktive Ordner-Nr.
1011 .C051F          s 8          ;Ordner-Liste:
1012                  ;Je 1 Bit für jede
1013                  ;mögl. Ordner-Nr.
1014 ;---
1015 :V0527          s 4*2        ;Anzahl freie Blocks.
1016 :V052F          s 4*2        ;Anzahl Blocks/Disk.
1017
1018 ; Hinweis:
1019 ; Die Tabelle wird beim wechseln des
1020 ; Anzeige-Modus verwendet um Inhalte
1021 ; eines geöffneten Fensters ggf. neu
1022 ; zu laden.
1023 .V0537          s 4          ;Icon-/Text-Modus:
1024                  ;$00 = Icons
1025                  ;$01 = Text
1026                  ;$02 = Sort./Name
1027                  ;$03 = Sort./Datum
1028                  ;$04 = Sort./Größe
1029                  ;$05 = Sort.Typ
1030
1031 ; Tabelle mit Zeiger auf Diskname.
1032 :V053B          w C0543
1033                  w C0593
1034                  w C05E3
1035                  w C0633
1036
1037 ;--- TD.WIN
1038 .START_TITLE_DATA
1039 :C0543          b "x:"      ;Titel Fenster #1.
1040                  s 78
1041 :C0593          b "x:"      ;Titel Fenster #2.
1042                  s 78
1043 :C05E3          b "x:"      ;Titel Fenster #3.
1044                  s 78
1045 :C0633          b "x:"      ;Titel Fenster #4.
1046                  s 78
1047 :END_TITLE_DATA
1048 .SIZE_TITLE_DATA = (END_TITLE_DATA - START_TITLE_DATA)

```

```

;--- 1049
1050
; Zeiger auf Ordner-Liste. 1051
; TD13: SubDirListTab 1052
:V0683      b < C624F 1053
              b < C628F 1054
              b < C62CF 1055
              b < C630F 1056
:V0687      b > C624F 1057
              b > C628F 1058
              b > C62CF 1059
              b > C630F 1060
1061
; Offset Dateityp für mod.#09. 1062
.V40C2      b $00 1063
1064
; Anzahl nicht kopierte Dateien für mod.#09. 1065
.V40C3      b $00 1066
1067
; Neue Partition für Fenster setzen. 1068
; Die Werte in V0728/V21B1/V5C34 1069
; gehören zusammen und werden in 1070
; mod.#11 verwendet. 1071
.V0728      s 4 ; Neue Part.-Nr. 1072
.V21B1      s 4 ; Laufwerk für Part.wechsel. 1073
.V5C34      s 4 ; RealDrvType für Part.wechsel. 1074
1075
; CMD- oder 64Net-Partition wählen. 1076
; Wird in mod.#11 gesetzt. 1077
.V3A4B      b $00 ;$00=CMD, $FF=64Net-Part. wählen. 1078
1079
; Fenster-Nr. für Suche nach aktiver Partition. 1080
; Wird nur für mod.#11 verwendet. 1081
.V3A94      b $00 ;Fens.-Nr. 1082
1083
; r1 nach r15 kopieren. 1084
:Z0670      lda  r1H 1085
              sta  r15H 1086
              lda  r1L 1087
              sta  r15L 1088
              rts 1089
1090
; r15 nach r1 kopieren. 1091
:Z0678      lda  r15H 1092
              sta  r1H 1093
              lda  r15L 1094
              sta  r1L 1095
              rts 1096
1097
; Addiere SIZE_DIRNAM zu r0. 1098
; Kopiere r0 nach r1. 1099
:Z31E0      lda  #< SIZE_DIRNAM 1100
              clc 1101
              adc  r0L 1102
              sta  r0L 1103

```

```

1104          lda    #> SIZE_DIRNAM
1105          adc     r0H
1106          sta     r0H
1107 ;          jmp     Z31E5                ;Kopiere r0 -> r1.
1108
1109 ; r0 nach r1 kopieren.
1110 ; TD13: MoveWr0r1
1111 :Z31E5      lda     r0L
1112          sta     r1L
1113          lda     r0H
1114          sta     r1H
1115          rts
1116
1117 ; r1 nach r0 kopieren.
1118 ; TD13: MoveWr1r0
1119 :Z31EE      lda     r1L
1120          sta     r0L
1121          lda     r1H
1122          sta     r0H
1123          rts
1124
1125 ; Anfang Kopierspeicher setzen.
1126 ; TD13: SetCopyMemLow
1127 .S31F7     ldx     #> COPYMEM_LOW
1128          stx     MLB04AD                ;HB Anfang CopyBuf.
1129          rts
1130
1131 ; Auf NativeMode testen.
1132 :Z0680      lda     curType                ;Laufwerkstyp.
1133          and     #ST_DM0DES
1134          cmp     #DrvNative                ;NativeMode?
1135          rts
1136
1137 ; Nur NativeMode: Hauptverzeichnis öffnen.
1138 :Z068B      ldx     #NO_ERROR
1139          jsr     Z0680                ;Lfwk. = Native?
1140          bne     :1                ; => Nein, weiter...
1141          jsr     OpenRootDir
1142 ::1        rts
1143
1144 ; Unterverzeichnis für Fenster öffnen.
1145 ; Übergabe: X = Fenster-Nr.
1146 :Z0960      lda     r1L
1147          sta     V04F0,x                ;Dir.T. für Fenster.
1148          lda     r1H
1149          sta     V04F4,x                ;Dir.S. für Fenster.
1150 ;          jmp     Z069B                ;-> Unterverzeichnis.
1151
1152 ; Nur NativeMode: Unterverzeichnis öffnen.
1153 :Z069B      ldx     #NO_ERROR
1154          jsr     Z0680                ;Lfwk. = Native?
1155          bne     :1                ; => Nein, weiter...
1156          jsr     OpenSubDir
1157 ::1        rts
1158

```



```

; Laufwerk, Verzeichnis, Partition öffnen. 1159
; Übergabe: A = Laufwerk. 1160
; TD13: NewSetDevice 1161
.S06AB      jsr    SetDevice 1162
1163
            ldy    curDrive 1164
            lda    V04B8 -8,y      ;Geräteinfo einlesen. 1165
            beq    :std           ; => 1541/71/81 1166
            bpl    :cmd           ; => CMD 1167
;           bmi    :std           ; => 64Net 1168
1169
;--- Standard-Laufwerk / 64Net. 1170
::std      jsr    Z0680           ;Lfwk. = Native? 1171
            beq    :subdir        ; => Ja, weiter... 1172
1173
::1        jmp    OpenDisk        ;Diskette öffnen. 1174
1175
;--- CMD-Laufwerk. 1176
::cmd      ldx    C042A           ;Oberstes Fenster. 1177
            lda    C04FC,x        ;Part. für Fenster. 1178
            bne    :2             ; => Part. gesetzt. 1179
1180
            ldx    #PART_ERR      ;Partitionsfehler. 1181
            jsr    S5CF3          ;Fehler ausgeben. 1182
1183
; Hinweis: 1184
; OpenRootDir führt auch OpenDisk aus. 1185
            jmp    Z068B          ;-> Hauptverzeichnis. 1186
1187
;--- CMD-Partition vorhanden. 1188
::2        pha                   ;Partitions-Nr. 1189
1190
            jsr    Z068B          ;-> Hauptverzeichnis. 1191
1192
            pla                   1193
            tax                   ;Partitions-Nr. 1194
            jsr    S0723          ;Part./Disk öffnen. 1195
            txa                   1196
            bne    :exit          ; => Fehler, Abbruch. 1197
1198
            jsr    Z0680          ;Lfwk. = Native? 1199
            bne    :exit          ; => Nein, Ende... 1200
1201
;--- Verzeichnis öffnen. 1202
::subdir   ldy    C042A           ;Oberstes Fenster. 1203
            lda    V04F0,y        ;Dir.T. für Fenster. 1204
            sta    r1L            1205
            lda    V04F4,y        ;Dir.S. für Fenster. 1206
            sta    r1H            1207
1208
;           lda    r1H            1209
            ora    r1L            ;Verzeichnis gesetzt? 1210
            beq    :root          ; => Nein, weiter... 1211
1212
::sub      jmp    Z069B          ;-> Unterverzeichnis. 1213

```

```

1214 ::root          jmp    Z068B          ;-> Hauptverzeichnis.
1215
1216 ::exit          rts
1217
1218 ; CMD-Partition/Diskette öffnen.
1219 .S0723          stx    r3H          ;Partitions-Nr.
1220                jmp    OpenPartition ;Part./Disk öffnen.
1221
1222 ; Datei auf Diskette suchen.
1223 .S1412          lda    #< V0470
1224                ldx    #> V0470
1225 .S1414          sta    r6L
1226                stx    r6H
1227                jmp    FindFile
1228
1229 ; Neues Modul einlesen.
1230 ; TD13: NewGetModule
1231 :Z072C          cmp    V0422          ;Aktuelles VLIR-Mod.?
1232                bne    :1            ; => Nein, weiter...
1233                clc                    ;Bereits im Speicher.
1234                rts
1235
1236 ::1             ldy    C0429          ;System in REU?
1237                beq    Z077F          ; => Von Disk laden.
1238
1239 ; VLIR-Mod. aus RAM laden.
1240 ::2             sta    V0422          ;Aktuelles VLIR-Mod.
1241                pha
1242
1243 if TRUE
1244                ldx    #(r3L - r0L)    ;Angaben zur Pos.
1245 ::2a            lda    V0425,x        ;VLIR-Daten in DACC.
1246                sta    r0,x
1247                dex
1248                bpl    :2a
1249 else
1250                lda    V0425 +0        ;Ladeadresse der
1251                sta    r0L            ;Modul-Adressen.
1252                lda    V0425 +1
1253                sta    r0H
1254
1255                lda    #< DACC_VLIRDAT
1256                sta    r1L
1257                lda    #> DACC_VLIRDAT
1258                sta    r1H
1259
1260                lda    #< SIZE_VLIRDAT
1261                sta    r2L
1262                lda    #> SIZE_VLIRDAT
1263                sta    r2H
1264
1265                lda    C0427          ;64K-Bank RAMDesk.
1266                sta    r3L
1267 endif
1268

```

	jsr	FetchRAM		1269
				1270
	pla		;Zeiger auf	1271
	asl		;Modul-Adresse.	1272
	tay			1273
	dey			1274
				1275
	dey			1276
	lda	(r0),y		1277
	sta	r1L		1278
	iny			1279
	lda	(r0),y		1280
	sta	r1H		1281
	iny			1282
				1283
	lda	(r0),y	;Größe Modul	1284
	sec		;berechnen.	1285
	sbc	r1L		1286
	sta	r2L		1287
	iny			1288
	lda	(r0),y		1289
	sbc	r1H		1290
	sta	r2H		1291
				1292
	jsr	FetchRAM	;Modul einlesen.	1293
	clc			1294
	rts			1295
				1296
				1297
				1298
				1299
				1300
				1301
	jsr	S07D6	;Systemlaufwerk.	1302
	bcc	:3	; => Ok, weiter...	1303
				1304
	pla		;Modul-Nr. verwerfen.	1305
	sec			1306
	bcs	Z07B7	;Lfwk. zurücksetzen.	1307
				1308
::3	lda	#< C0400	;Dateiname TopDesk.	1309
	sta	r0L		1310
	lda	#> C0400		1311
	sta	r0H		1312
	jsr	OpenRecordFile		1313
				1314
	pla			1315
	sta	V0422	;Aktuelles VLIR-Mod.	1316
	jsr	PointRecord		1317
				1318
	lda	V0425 +0	;VLIR-Adressbereich.	1319
	sta	r7L	; => BASE_MOD_A/B.	1320
	lda	V0425 +1		1321
	sta	r7H		1322
				1323

```

1324          lda    #< OS_VARS          ;Max. Größe bis
1325          sec                      ;OS_VARS = $8000, da
1326          sbc    r7L                ;sonst GEOS-System
1327          sta    r2L                ;überschrieben wird.
1328          lda    #> OS_VARS
1329          sbc    r7H
1330          sta    r2H
1331
1332          jsr    ReadRecord
1333          clc
1334
1335 ; Laufwerk zurücksetzen.
1336 :Z07B7          php
1337          jsr    S07C0                ;Lfwk. zurücksetzen.
1338          jsr    OpenDisk
1339          plp
1340          rts
1341
1342 ; Laufwerk und Partition zurücksetzen.
1343 .S07C0          lda    V07D4          ;Laufwerk öffnen.
1344          jsr    SetDevice
1345
1346          ldx    V07D5                ;Part. für Fenster?
1347          beq    :1                  ; => Nein, weiter...
1348
1349          jsr    S0723                ;Partition öffnen.
1350
1351 ::1            rts
1352
1353 .V07D4          b $00    ;Laufwerk für Fenster.
1354 :V07D5          b $00    ;Partition für Fenster.
1355
1356 ; Systemlaufwerk öffnen.
1357 ; -> RAM-TopDesk
1358 ; -> TopDesk.win speichern
1359 ; Rückgabe: C=0: OK.
1360 ; TD13: SearchDeskTop
1361 .S07D3          lda    #$ff          ;TD nicht suchen.
1362          b $2c
1363 .S07D6          lda    #$00          ;TD suchen.
1364          sta    V087B                ;$00 = TD suchen.
1365
1366          lda    curDrive              ;Aktuelles Laufwerk
1367          sta    V07D4                ;zwischenspeichern.
1368
1369          lda    #$00                  ;Keine Partition.
1370          sta    V07D5                ;Reset Part. Fenster.
1371
1372          lda    C050A                ;TopDesk/Partition?
1373          beq    :1                  ; => Nein, weiter...
1374
1375          ldx    C042A                ;Oberstes Fenster.
1376          lda    C0448,x              ;Fenster geöffnet?
1377          beq    :1                  ; => Nein, weiter...
1378

```

	lda	C04F8,x	;Lfwk. für Fenster =	1379
	cmp	curDrive	;aktuelles Laufwerk?	1380
	bne	:1	; => Nein, weiter...	1381
	tay			1382
	lda	V04B8 -8,y	;Geräteinfo einlesen.	1383
	beq	:1	; => 1541/71/81	1384
	bmi	:1	; => 64Net	1385
				1386
	lda	C04FC,x	;Part. für Fenster	1387
	sta	V07D5	;zwichenspeichern.	1388
				1389
::1	lda	C0424	;TopDesk/Laufwerk.	1390
	sta	V087C	;Erstes Laufwerk.	1391
				1392
	; TD13: SearchDeskTop2			1393
	; Übergabe: A = Laufwerk.			1394
::loop	jsr	SetDevice		1395
				1396
	lda	#< OpenDisk		1397
	ldx	#> OpenDisk		1398
				1399
	ldy	curDrive		1400
	cpy	C0424	;Systemlaufwerk?	1401
	bne	:11	; => Nein, weiter...	1402
				1403
	ldy	C050A	;Systempartition?	1404
	beq	:11	; => Nein, weiter...	1405
				1406
	sty	r3H		1407
	lda	#< OpenPartition		1408
	ldx	#> OpenPartition		1409
				1410
::11	jsr	CallRoutine	;OpenDisk/Openpart.	1411
				1412
	bit	V087B	;TopDesk suchen?	1413
	bmi	:ok	; => Nein, Ende...	1414
				1415
	txa		;Laufwerk bereit?	1416
	bne	:next	; => Nein, weiter...	1417
				1418
	lda	#< C0400	;Dateiname TopDesk.	1419
;	sta	r6L		1420
	ldx	#> C0400		1421
;	stx	r6H		1422
	jsr	S1414	;Datei suchen.	1423
	txa		;Gefunden?	1424
	bne	:next	; => Nein, weiter...	1425
				1426
	jsr	S09C7	;Infoblock einlesen.	1427
				1428
	ldy	#0		1429
::12	lda	C0411,y	;Klasse vergleichen.	1430
	beq	:ok	; => Gefunden...	1431
	cmp	diskBlkBuf +77,y		1432
	bne	:next	; => Nächst.Lfwk.	1433

```

1434          iny
1435          bne    :12
1436
1437 ::ok          clc                      ;OK.
1438          rts
1439
1440 ; Suche auf nächstem Laufwerk fortsetzen.
1441 ::next        ldx    curDrive
1442 ::21          inx
1443          cpx    #12
1444          bcc    :22
1445          ldx    #8
1446 ::22          cpx    V087C
1447          beq    :ask
1448
1449          lda    driveType -8,x
1450          beq    :21
1451
1452          txa
1453          bne    :loop
1454
1455 ; Dialogbox: Programm nicht gefunden.
1456 ::ask          lda    #< D087D          ;Fehler: "TopDesk
1457          ldx    #> D087D          ;nicht gefunden!"
1458          jsr    S2090          ;DBox ohne Recover.
1459
1460          lda    r0L
1461          cmp    #YES          ;Nochmal suchen?
1462          bne    :err          ; => Nein, Ende...
1463
1464          lda    curDrive
1465          bne    :loop          ;Erneut suchen.
1466
1467 ::err          lda    V087C          ;Laufwerk auf
1468          jsr    SetDevice          ;Anfang setzen.
1469
1470          sec                      ;Fehler.
1471          rts
1472
1473 :V087B          b $00    ;$00 = Programmdatei suchen.
1474
1475 ; Hinweis:
1476 ; Wird auch in mod.#09 für die
1477 ; Datei-Suche verwendet.
1478 .V087C          b $08    ;Akt. Laufwerk für Datei-Suche.
1479
1480 ; Hinweis:
1481 ; Die Routine wird beim nachladen von
1482 ; VLIR-Modulen und beim speichern der
1483 ; Systemeinstellungen verwendet.
1484 ; => Dialogbox angepasst.
1485 ;
1486 ; Dialogbox: Programmdatei nicht gefunden.
1487 :D087D          b %10000001
1488

```

	b YES	, \$01, \$48	1489
	b CANCEL	, \$11, \$48	1490
			1491
	b DBTXTSTR,	\$08, \$10	1492
	w T29CF		1493
			1494
	b DBTXTSTR,	\$08, \$20	1495
	w T08A9		1496
			1497
	b DBTXTSTR,	\$08, \$2c	1498
	w T29D3		1499
	b DBTXTSTR,	\$12, \$2c	1500
	w C0400		1501
			1502
	b DBTXTSTR,	\$08, \$40	1503
	w T0899		1504
			1505
	b NULL		1506
			1507
if LANG = LANG_DE			1508
:T08A9	b "Programmdatei nicht gefunden:",	0	1509
:T0899	b BOLDON, "Nochmal suchen?",	PLAINTEXT, 0	1510
endif			1511
if LANG = LANG_EN			1512
:T08A9	b "Application file not found:",	0	1513
:T0899	b BOLDON, "Search again?",	PLAINTEXT, 0	1514
endif			1515
			1516
; TopDesk-Konfiguration einlesen.			1517
; Übergabe: dirEntryBuf = TopDesk-Eintrag.			1518
:Z0956	lda C0429	;System in REU?	1519
	bne :1	; => Ja, weiter...	1520
			1521
; TopDesk-Konfiguration von Disk laden.			1522
; Übergabe: dirEntryBuf = TopDesk-Eintrag.			1523
	jsr S09C7	;Infoblock einlesen.	1524
			1525
	lda BASE_SYSVAR +5	;Werte gespeichert?	1526
	beq :exit	; => Nein, Ende...	1527
	bne :2	; => Ja, einlesen.	1528
			1529
::1	jsr Z0A51	;Zeiger auf Konfig.	1530
	jsr FetchRAM	;Konfig einlesen.	1531
			1532
::2	lda BASE_SYSVAR +0		1533
	sta C0427	;64K-Bank RAMDesk.	1534
	lda BASE_SYSVAR +1		1535
	sta C0428	;RAM-TopDesk-Status.	1536
			1537
if ENABLE_DIFFICOL = FALSE			1538
	lda BASE_SYSVAR +2		1539
	sta MT3F58	;Versch. Farben.	1540
endif			1541
if ENABLE_DIFFICOL = TRUE			1542
	lda #" "		1543

```

1544          bit    BASE_SYSVAR +2
1545          bpl     :3
1546          lda     #"*"
1547 :::3          sta     MT3F58          ;Versch. Fens.farben.
1548
1549          lda     #" "
1550          bit     BASE_SYSVAR +2
1551          bvc     :4
1552          lda     #"*"
1553 :::4          sta     MT3F59          ;Versch. Iconfarben.
1554 endif
1555
1556 if ENABLE_ANIMATE = TRUE
1557          lda     BASE_SYSVAR +3
1558          sta     MT4048          ;Animation EIN/AUS.
1559 endif
1560
1561          lda     BASE_SYSVAR +4
1562          sta     MT3F38          ;Autom. platzieren.
1563
1564 if ENABLE_DIFFICOL = FALSE
1565          lda     BASE_SYSVAR +5          ;Fensterfarben #1-#4.
1566          sta     V621B
1567          lda     BASE_SYSVAR +6
1568          sta     V622C
1569          lda     BASE_SYSVAR +7
1570          sta     V623D
1571          lda     BASE_SYSVAR +8
1572          sta     V624E
1573 endif
1574 if ENABLE_DIFFICOL = TRUE
1575          jsr     i_MoveData
1576          w BASE_SYSVAR +5          ;Fensterfarben #1-#4.
1577          w V621B
1578          w 4
1579 endif
1580
1581          jsr     i_MoveData
1582          w BASE_SYSVAR +9          ;Farbeinstellungen.
1583          w BASE_COLCFG
1584          w 14
1585
1586          lda     BASE_SYSVAR +24
1587          sta     C04B5          ;Kopieren-Dialog.
1588
1589          lda     BASE_SYSVAR +26
1590          bne     :exit          ; => Infotext, Ende.
1591
1592          jsr     i_MoveData
1593          w BASE_SYSVAR +27          ;Quicklinks.
1594          w TD_QLINK_DATA
1595          w TD_QLINK_SIZE
1596
1597 :::exit      rts
1598

```



```

; Laufwerk oberstes Fenster, CMD-Info löschen. 1599
.S09AC      ldx    C042A                ;Oberstes Fenster. 1600
            lda    C04F8,x              ;Lfwk. für Fenster. 1601
            jsr    S06AB                ;Laufwerk öffnen. 1602
            1603
            ldx    C042A                ;Oberstes Fenster. 1604
            1605
; CMD-Info löschen. 1606
; Übergabe: X = Fenster-Nr. 1607
.S09B5      lda    #$00                1608
            sta    C04FC,x              ;Part. für Fenster. 1609
            1610
; CMD-Info löschen. 1611
; Übergabe: X = Fenster-Nr. 1612
.S09B7      lda    #$00                1613
            sta    V04F0,x              ;Dir.T. für Fenster. 1614
            sta    V04F4,x              ;Dir.S. für Fenster. 1615
            rts                        1616
            1617
; Zeiger auf Datenblock/VLIR-Header. 1618
.S09BC      lda    dirEntryBuf +1       1619
            sta    r1L                  1620
            ldx    dirEntryBuf +2       1621
            stx    r1H                  1622
            rts                        1623
            1624
; Zeiger auf InfoBlock. 1625
.S09C0      lda    dirEntryBuf +19      1626
            sta    r1L                  1627
            ldx    dirEntryBuf +20      1628
            stx    r1H                  1629
            rts                        1630
            1631
; Zeiger auf fileTrScTab. 1632
.S09C4      lda    #< fileTrScTab       1633
            sta    r4L                  1634
            lda    #> fileTrScTab       1635
            sta    r4H                  1636
            rts                        1637
            1638
; Infoblock einlesen. 1639
.S09C7      jsr    S09C0                ;Infoblock > r1L/r1H. 1640
            jmp    GetBlock_dskBuf       1641
            1642
; Konfiguration in REU+Datei sichern. 1643
.S09DD      lda    C0429                ;System in REU? 1644
            beq    Z09EB                ; => Nein, weiter... 1645
            jsr    Z09EB                ;Konfig speichern. 1646
:Z09E5      jsr    Z0A0C                ;TD-Konfig kopieren. 1647
            jmp    StashRAM              ;TD-Konfig => REU. 1648
            1649
; TopDesk-Konfiguration in TopDesk-Datei speichern. 1650
:Z09EB      jsr    S07D6                ;Systemlaufwerk. 1651
            bcs    :1                   1652
            1653

```

```

1654          jsr    Z0A0C          ;TD-Konfig kopieren.
1655
1656          jsr    S09C0          ;Infoblock > r1L/r1H.
1657          jsr    PutBlock_dskBuf
1658
1659 ::1          jsr    S07C0          ;Lfwk. zurücksetzen.
1660          jmp    OpenDisk
1661
1662 ; TopDesk-Konfiguration speichern.
1663 ; Ziel: Infoblock ab Byte 134/$86
1664 ;      => BASE_SYSVAR:
1665 ; $xx86 = V0427 64K-Bank Ram-TopDesk
1666 ; $xx87 = V0428 Status Ram-TopDesk
1667 ; $xx88 = MT3F58 Bit%7: Verschiedene Fensterfarben
1668 ;      Bit%6: Verschiedene Iconfarben (Optional)
1669 ; $xx89 = MT4048 Animation EIN/AUS
1670 ; $xx8A = MT3F38 Automatisch platzieren
1671 ; $xx8B = V621B Farbe Fenster #1
1672 ; $xx8C = V622C Farbe Fenster #2
1673 ; $xx8D = V623D Farbe Fenster #3
1674 ; $xx8E = V624E Farbe Fenster #4
1675 ; $xx8F = BASE_COLCFG
1676 ; $xx9D = MT3FFD Kopieren-Dialog
1677 ; $xx9E = Frei
1678 ; $xx9F = Frei
1679 ; $xxA0 = NULL
1680 ; $xxA1 = TD_QLINK_DATA
1681 ; $xxF6 = Frei
1682 :Z0A0C      lda    #NULL
1683            sta    BASE_SYSVAR +26
1684
1685            jsr    i_MoveData
1686            w TD_QLINK_DATA          ;Quicklinks.
1687            w BASE_SYSVAR +27
1688            w TD_QLINK_SIZE
1689
1690            lda    C04B5          ;Kopieren-Dialog.
1691            sta    BASE_SYSVAR +24
1692
1693            jsr    i_MoveData
1694            w BASE_COLCFG          ;Farbeinstellungen.
1695            w BASE_SYSVAR +9
1696            w 14
1697
1698 if ENABLE_DIFFICOL = FALSE
1699            lda    V624E          ;Fensterfarben #1-#4.
1700            sta    BASE_SYSVAR +8
1701            lda    V623D
1702            sta    BASE_SYSVAR +7
1703            lda    V622C
1704            sta    BASE_SYSVAR +6
1705            lda    V621B
1706            sta    BASE_SYSVAR +5
1707 endif
1708 if ENABLE_DIFFICOL = TRUE

```

```

        jsr    i_MoveData                                1709
        w V621B                                ;Fensterfarben #1-#4. 1710
        w BASE_SYSVAR +5                                1711
        w 4                                          1712
endif                                          1713
                                          1714
        lda    MT3F38                                ;Autom. platzieren. 1715
        sta    BASE_SYSVAR +4                                1716
                                          1717
if ENABLE_ANIMATE = TRUE                                1718
        lda    MT4048                                ;Animation EIN/AUS. 1719
        sta    BASE_SYSVAR +3                                1720
endif                                          1721
                                          1722
if ENABLE_DIFFICOL = FALSE                                1723
        lda    MT3F58                                ;Versch. Farben. 1724
        sta    BASE_SYSVAR +2                                1725
endif                                          1726
if ENABLE_DIFFICOL = TRUE                                1727
        lda    #%00000000                                1728
        ldy    #" "                                1729
        cpy    MT3F58                                ;Versch. Fens.farben. 1730
        beq    :3                                          1731
        ora    #%10000000                                1732
::3      cpy    MT3F59                                ;Versch. Iconfarben. 1733
        beq    :4                                          1734
        ora    #%01000000                                1735
::4      sta    BASE_SYSVAR +2                                1736
endif                                          1737
                                          1738
        lda    C0428                                ;RAM-TopDesk-Status. 1739
        sta    BASE_SYSVAR +1                                1740
        lda    C0427                                ;64K-Bank RAMDesk. 1741
        sta    BASE_SYSVAR +0                                1742
                                          1743
; Zeiger auf Systembereich in REU.                                1744
:Z0A51    ldx    #(r2H - r0L)                                1745
::1      lda    :data,x                                1746
        sta    r0,x                                1747
        dex                                          1748
        bpl    :1                                          1749
                                          1750
        lda    C0427                                ;64K-Bank RAMDesk. 1751
        sta    r3L                                1752
        rts                                          1753
                                          1754
::data    w BASE_SYSVAR                                1755
        w DACC_SYSVAR                                1756
        w SIZE_SYSVAR                                1757
                                          1758
.V0A71    b $00    ;Nr. Ziel-Ordner.                                1759
.V0A72    b $00    ;Laufwerk für Source-Fenster.                                1760
.V0A73    b $00    ;Laufwerk für Target-Fenster.                                1761
                                          1762
; Datei kopieren.                                1763

```

```

1764 ; Übergabe: r10 = Name Source-Disk.
1765 ;           r11 = Name Target-Disk.
1766 ; Rückgabe: X = Fehler.
1767 ; TD13: CopyFile
1768 .S0A74          lda    C042A          ;Oberstes Fenster.
1769                pha
1770
1771                lda    WIN042F        ;Aktives Fenster.
1772                sta    C042A          ;Oberstes Fenster.
1773
1774                lda    V0A73          ;Lfwk. Target-Disk.
1775                jsr    S06AB          ;Target-Disk öffnen.
1776                jsr    Z4819          ;Diskinfo/Größe.
1777
1778                pla
1779                sta    C042A          ;Oberstes Fenster.
1780
1781 ; Hinweis:
1782 ; Speicher retten und dann CopyRAM für
1783 ; das kopieren der Datei erweitern.
1784 ;---
1785                lda    #$00
1786                sta    V10CF          ;SwapRAM aktivieren.
1787
1788                jsr    Z10BC          ;Swap APP_RAM.
1789
1790                lda    MLB04AD        ;HB Anfang CopyBuf.
1791                pha
1792                lda    #> COPYMEM_MAX
1793                sta    MLB04AD
1794
1795                jsr    Z0ACB          ;Datei kopieren.
1796                stx    V0AC9          ;Status speichern.
1797
1798                pla
1799                sta    MLB04AD        ;HB Anfang CopyBuf.
1800
1801                jsr    Z10BC          ;Swap APP_RAM.
1802 ;---
1803
1804                ldx    V0AC9          ;Status einlesen.
1805                bne    :exit          ; => Fehler, Ende...
1806
1807                lda    V04A9          ;Kop.od.verschieben?
1808                beq    :exit          ; => Kopieren.
1809
1810 ; Verschieben, Quell-Datei löschen.
1811                lda    V0A72          ;Lfwk. Source-Disk.
1812                jsr    S06AB          ;Source-Disk öffnen.
1813
1814                lda    r12L          ;Name Quell-Datei.
1815                ldx    r12H
1816                jsr    Z5B3A          ;Eintrag löschen.
1817
1818 ; Hinweis:

```

```

; Fehler WR_PR_ON gilt für Disketten                                1819
; mit einem Schreibschutz.                                         1820
; Fehler ERR_WR_PROT gilt für Dateien                               1821
; mit einem Schreibschutz.                                         1822
;                                                                    1823
;          cpx    #WR_PR_ON          ;Schreibschutz.              1823
;          cpx    #ERR_WR_PROT       ;Schreibschutz.              1824
;          beq    :ok                 ; => Nicht löschen.          1825
;          txa                                     1826
;          bne    :exit                ; => Fehler, Ende.           1827
;                                                                    1828
; Ordner / Verzeichnis kopieren aktiv?                              1829
; => mod.#09                                                         1830
;          lda    V04B0                ;VLIR-Modul aktiv?          1831
;          bne    :exit                ; => Ja, weiter...           1832
;                                                                    1833
; Wenn Datei verschoben wurde, dann                                  1834
; muss die Liste der markierten Dateien                             1835
; angepasst werden, sofern weitere                                 1836
; Dateien mit höherer Markierung in der                             1837
; Liste vorhanden sind.                                            1838
;                                                                    1839
;          ldx    #$ff                                                         1840
::1          inx                                                         1841
;          lda    V634F,x              ;Tab. mark.Dateien.         1842
;          cmp    #$ff                ;Datei markiert?            1843
;          beq    :ok                 ; => Nein, Ende...            1844
;          cmp    V04AA                ;< Akt. Datei-Nr.?          1845
;          bcc    :1                  1846
;          dec    V634F,x              ;Pos. mark.Datei -1.        1847
;          bcs    :1                  ; => Weitersuchen...         1848
;                                                                    1849
::ok          ldx    V0AC9                                                         1850
::exit        rts                                                         1851
;                                                                    1852
:V0AC9        b $00    ;Fehlerstatus.                                     1853
;                                                                    1854
; Dateien kopieren/duplizieren.                                     1855
; Übergabe: V0A71 = Nr. Ziel-Ordner.                                1856
;          r10    = Zeiger Name SourecDisk.                          1857
;          r11    = Zeiger Name TargetDisk.                          1858
;          r12    = Name Source-Datei.                               1859
;          r13    = Name Target-Datei.                              1860
; Hinweis: r10=r11 für Duplicate, r12=r13 für DateiCopy.           1861
:Z0ACB        lda    V0A72                ;Lfwk. Source-Disk.      1862
;          jsr    S06AB                ;Source-Disk öffnen.        1863
;          txa                                     1864
;          bne    :exit                1865
;                                                                    1866
;          lda    r12L                                                         1867
;          sta    r6L                                                         1868
;          ldx    r12H                                                         1869
;          stx    r6H                                                         1870
;          jsr    S1414                ;Datei suchen.              1871
;          txa                                     1872
;          bne    :exit                1873

```

```

1874
1875 ; Hinweis:
1876 ; Fehler-Status wird in V0B64 gesetzt.
1877         jsr     Z0B38                ;Noch Blocks frei?
1878
1879         jsr     EnterTurbo
1880         jsr     InitForIO
1881
1882         lda     #$00                ;Erster Sektor noch
1883         sta     r9L                ;nicht definiert.
1884
1885         lda     dirEntryBuf +22
1886         beq     :2                ; => Nicht-GEOS.
1887
1888         jsr     Z0B6Y                ;Ende CopyBuf.
1889
1890         jsr     S09C0                ;Infoblock > r1L/r1H.
1891         jsr     ReadBlock            ;Infoblock einlesen.
1892         txa
1893         beq     :1
1894
1895         jmp     DoneWithIO
1896
1897 ::1      lda     #< Z0BF3            ; => Copy VLIR.
1898         ldx     #> Z0BF3
1899
1900         ldy     dirEntryBuf +21
1901         bne     :3                ; Dateityp = VLIR.
1902
1903 ::2      lda     #< Z0B65            ; => Copy SEQ.
1904         ldx     #> Z0B65
1905
1906 ::3      jsr     CallRoutine          ; => Copy VLIR/SEQ.
1907         ; txa
1908         ; bne     :exit            ; => Fehler, Abbruch.
1909
1910 ::exit   rts
1911
1912 ; Datei löschen.
1913 ; Übergabe: V0470 = Dateiname.
1914 ; Rückgabe: X = Fehler-Nr.
1915 ;         Z-Flag: <>0 = Fehler.
1916 .S0B30   lda     #< V0470
1917         ldx     #> V0470
1918 .S0B34   sta     r0L
1919         stx     r0H
1920         jsr     DeleteFile
1921         txa
1922         rts
1923
1924 ; Noch ausreichend Blocks frei?
1925 :Z0B38   lda     WIN042F            ;Aktives Fenster.
1926         asl
1927         tax
1928         lda     dirEntryBuf +29

```

	sta	V0B62 +1		1929
	cmp	V0527 +1,x	;Anzahl freie Blocks.	1930
	beq	:1		1931
	bcs	:err		1932
	bcc	:ok		1933
				1934
::1	lda	dirEntryBuf +28		1935
	sta	V0B62 +0		1936
	cmp	V0527 +0,x	;Anzahl freie Blocks.	1937
	beq	:ok		1938
	bcs	:err		1939
				1940
::ok	ldx	#NO_ERROR		1941
	b	\$2c		1942
::err	ldx	#NO_BLOCKS		1943
	stx	V0B64	;Fehler-Status.	1944
	rts			1945
				1946
:V0B62	w	\$0000	;Dateigröße.	1947
:V0B64	b	\$00	;Fehler-Status.	1948
				1949
				1950
; Zeiger auf Anfang/Ende CopyBuf.				1950
:Z0B6X	ldx	#1	;HB Anfang CopyBuf.	1951
	b	\$2c		1952
:Z0B6Y	ldx	#0	;HB Ende CopyBuf.	1953
	lda	MHB04AC,x		1954
	sta	r4H		1955
	lda	#\$00		1956
	sta	r4L		1957
	rts			1958
				1959
; Seq. Datei kopieren.				1960
; TD13: CopySEQ				1961
:Z0B65	jsr	S09BC	;1.Block > r1L/r1H.	1962
				1963
::loop	jsr	Z0B6X	;Anfang CopyBuf.	1964
	jsr	Z0E17	;Sektorkette lesen.	1965
	txa			1966
	bne	exitCopy1		1967
				1968
;	lda	r1L	;Nächsten Sektor	1969
;	sta	r15L	;zwischenspeichern.	1970
;	lda	r1H		1971
;	sta	r15H		1972
	jsr	Z0670	;Move_r1_r15.	1973
				1974
	ldx	#r11	;Name Target-Disk.	1975
	jsr	Z0F35	;Datei suchen.	1976
	txa		;Bei Fehler wurde I/O	1977
	bne	:err	;deaktiviert!	1978
				1979
	lda	r9L	;1.Sektor definiert?	1980
	bne	:1	; => Ja, weiter...	1981
				1982
	jsr	Z0E9C	;Erster freier Block.	1983

```

1984          txa
1985          bne    exitCopy1
1986
1987          lda    r9L          ;Erster Sektor für
1988          sta    dirEntryBuf +1      ;Dateieintrag.
1989          lda    r9H
1990          sta    dirEntryBuf +2
1991
1992 ::1          lda    r9H
1993          pha
1994          lda    r9L
1995          pha
1996          jsr    Z0B6X          ;Anfang CopyBuf.
1997          jsr    Z0E34          ;Sektoren schreiben.
1998          pla
1999          sta    r1L
2000          pla
2001          sta    r1H
2002          txa
2003          bne    exitCopy1
2004
2005          jsr    Z0B6X          ;Anfang CopyBuf.
2006          jsr    Z0E7D          ;Sektorkette prüfen.
2007          txa
2008          bne    exitCopy1
2009
2010          lda    r15L          ;Datei-Ende?
2011          bne    :2            ; => Nein, weiter...
2012          jmp    Z0D5C          ;Neuer Dateieintrag.
2013
2014 ::2          jsr    DoneWithIO
2015          jsr    PutDirHead
2016          jsr    InitForIO
2017
2018          ldx    #r10          ;Name Source-Disk.
2019          jsr    Z0F35          ;Datei suchen.
2020          txa
2021          bne    :err          ;Bei Fehler wurde I/O
2022                                   ;deaktiviert!
2023 ;          lda    r15L
2024 ;          sta    r1L
2025 ;          lda    r15H
2026 ;          sta    r1H
2027          jsr    Z0678          ;Move_r15_r1.
2028          jmp    :loop
2029
2030 ::err        rts
2031
2032 :exitCopy1    jmp    DoneWithIO
2033
2034 ; VLIR-Datei kopieren.
2035 ; TD13: CopyVLIR
2036 :Z0BF3        lda    #$ff
2037          sta    r2L          ;Keine Daten.
2038

```



```

        jsr    S09C4                ;fileTrScTab > r4.      2039
        jsr    S09BC                ;VLIR-Hdr > r1L/r1H.    2040
                                           2041
        jsr    ReadBlock            ;Indexblock einlesen.   2042
        txa                               2043
        bne    exitCopy1            2044
                                           2045
        lda    #$ff                 2046
        sta    r15L                 2047
                                           2048
        jsr    Z0B6X                ;Anfang CopyBuf.        2049
                                           2050
        ldy    #2                   ;Erster Datensatz.     2051
        sty    r14H                 2052
::getloop    lda    fileTrScTab,y    2053
        sta    r1L                 2054
        iny                               2055
        sty    r14L                 2056
        ldx    fileTrScTab,y        2057
        stx    r1H                 2058
        tay                               2059
        bne    :1                   ;Datensatz vorhanden.   2060
        txa                               2061
        bne    :2                   ;Leerer Datensatz.     2062
                                           2063
        lda    #$ff                 ;Keine Daten mehr.     2064
        sta    r14L                 2065
        bne    :2                   ;Buffer kopieren.      2066
                                           2067
::1          jsr    Z0E17            ;Sektorkette lesen.    2068
        txa                               2069
        bne    exitCopy1            2070
                                           2071
;          lda    r1H                 2072
;          sta    r15H                 2073
;          lda    r1L                 2074
;          sta    r15L                 2075
        jsr    Z0670                ;Move_r1_r15.          2076
        bne    :putloop             2077
                                           2078
        inc    r4H                 2079
        lda    r4H                 2080
        cmp    MHB04AC              ;HB Ende CopyBuf.      2081
        bcs    :putloop             2082
                                           2083
::2          ldy    r14L             2084
        iny                               2085
        bne    :getloop             2086
                                           2087
; r9  = Nächster freier Block.      2088
; r10 = Name Source-Disk.          2089
; r11 = Name Target-Disk.         2090
; r14L = Offset in fileTrScTab (aktuell; lesen).          2091
; r14H = Offset in fileTrScTab (erster; lesen).          2092
; r15 = Tr/Se des zuletzt gelesenen Blocks.              2093

```

```

2094 ::put loop      ldx  #r11                ;Name Target-Disk.
2095                jsr  Z0F35                ;Datei suchen.
2096                txa
2097                beq  :3
2098                rts
2099
2100 ::3              lda  r9L
2101                bne  :4
2102
2103                jsr  Z0E9C                ;Erster freier Block.
2104                txa
2105                beq  :4
2106
2107                jmp  exitCopy2
2108
2109 ::4              lda  r15L
2110                cmp  #$ff
2111                bne  :5
2112                jmp  Z0D5C                ;Neuer Dateieintrag.
2113
2114 ::5              lda  r2L                ;Daten im Puffer.
2115                beq  :8                ; => Ja, schreiben.
2116
2117 ::6              ldy  r14H
2118                lda  fileTrScTab,y
2119                bne  :7
2120                inc  r14H
2121                inc  r14H
2122                beq  :12
2123                bne  :6
2124
2125 ::7              lda  r9L
2126                sta  fileTrScTab +0,y
2127                lda  r9H
2128                sta  fileTrScTab +1,y
2129
2130 ::8              lda  r9H
2131                pha
2132                lda  r9L
2133                pha
2134                lda  r14H
2135                pha
2136
2137                jsr  Z0B6X                ;Anfang CopyBuf.
2138 ::9              jsr  Z0E34                ;Sektoren schreiben.
2139                txa
2140                bne  :12
2141
2142                ldy  #$00
2143                lda  (r4),y
2144                bne  :12
2145
2146 ::10             ldy  r14H
2147                iny
2148                iny

```

	sty	r14H	2149
	beq	:12	2150
			2151
	lda	fileTrScTab +0,y	2152
	bne	:11	2153
	lda	fileTrScTab +1,y	2154
	beq	:12	2155
	bne	:10	2156
			2157
::11	inc	r4H	2158
	lda	r4H	2159
	cmp	MHB04AC ;HB Ende CopyBuf.	2160
	bcs	:12	2161
			2162
	lda	r9L	2163
	sta	fileTrScTab +0,y	2164
	lda	r9H	2165
	sta	fileTrScTab +1,y	2166
	jmp	:9	2167
			2168
::12	pla		2169
	sta	r14H	2170
	pla		2171
	sta	r1L	2172
	pla		2173
	sta	r1H	2174
			2175
	txa		2176
	bne	exitCopy2	2177
			2178
	jsr	Z0B6X ;Anfang CopyBuf.	2179
::13	jsr	Z0E7D ;Sektorkette prüfen.	2180
	txa		2181
	bne	exitCopy2	2182
			2183
	ldy	#\$00	2184
	lda	(r4),y	2185
	bne	:16 ; => Puffer voll.	2186
			2187
::14	ldy	r14H	2188
	iny		2189
	iny		2190
	sty	r14H	2191
	beq	Z0D5C ;Neuer Dateieintrag.	2192
			2193
	ldx	fileTrScTab +1,y	2194
	stx	r1H	2195
	lda	fileTrScTab +0,y	2196
	sta	r1L	2197
	bne	:15	2198
	txa		2199
	bne	:14	2200
	beq	Z0D5C ;Neuer Dateieintrag.	2201
			2202
::15	inc	r4H	2203

```

2204          lda    r4H
2205          cmp    MHB04AC          ;HB Ende CopyBuf.
2206          bcc    :13
2207
2208          jsr    Z0D41          ;Update BAM und
2209          txa          ;Zeiger auf CopyBuf
2210          bne    exitCopy2      ;zurücksetzen.
2211
2212          lda    #$ff
2213          sta    r2L          ;Keine Daten.
2214
2215          ldz    r14L
2216          iny
2217          jmp    :getloop
2218
2219 ::16          jsr    Z0D41          ;Update BAM und
2220          txa          ;Zeiger auf CopyBuf
2221          bne    exitCopy2      ;zurücksetzen.
2222
2223 ;          ldx    #$00          ;Daten im Puffer.
2224          stx    r2L
2225
2226 ;          lda    r15H
2227 ;          sta    r1H
2228 ;          lda    r15L
2229 ;          sta    r1L
2230          jsr    Z0678          ;Move_r15_r1.
2231          jmp    :1
2232
2233 :exitCopy2      jmp    DoneWithIO
2234
2235 ; BAM aktualisieren, Zeiger auf Anfang.
2236 :Z0D41          jsr    DoneWithIO
2237          jsr    PutDirHead
2238          jsr    InitForIO
2239
2240          ldx    #r10          ;Name Source-Disk.
2241          jsr    Z0F35          ;Datei suchen.
2242
2243          jsr    Z0B6X          ;Anfang CopyBuf.
2244
2245          ldx    #NO_ERROR
2246          rts
2247
2248 ; CopyFile: Dateieintrag schreiben.
2249 ; TD13: WriteInfo
2250 :Z0D5C          jsr    DoneWithIO
2251
2252          lda    dirEntryBuf +22
2253          bne    :1          ; => GEOS-Datei.
2254
2255          lda    r9L
2256          sta    r6L
2257          lda    r9H
2258          sta    r6H

```

	jsr	FreeBlock	;Infoblock löschen.	2259
	txa			2260
	beq	:2		2261
				2262
::err	rts			2263
				2264
::1	lda	r9L	;Infoblock schreiben.	2265
	sta	r1L		2266
	sta	dirEntryBuf +19		2267
	lda	r9H		2268
	sta	r1H		2269
	sta	dirEntryBuf +20		2270
				2271
	jsr	Z0B6Y	;Ende CopyBuf.	2272
				2273
	jsr	PutBlock		2274
	txa			2275
	bne	:err		2276
				2277
	lda	dirEntryBuf +21		2278
	beq	:2	; => Kein VLIR.	2279
				2280
	lda	r9L		2281
	sta	r3L		2282
	lda	r9H		2283
	sta	r3H		2284
	jsr	SetNextFree		2285
	txa			2286
	bne	:err		2287
				2288
	lda	r3L	;VLIR-Header	2289
	sta	r1L	;speichern.	2290
	sta	dirEntryBuf +1		2291
	lda	r3H		2292
	sta	r1H		2293
	sta	dirEntryBuf +2		2294
				2295
	jsr	S09C4	;fileTrScTab > r4.	2296
	jsr	PutBlock	;Header schreiben.	2297
	txa			2298
	bne	:err		2299
				2300
::2	lda	#0		2301
	sta	r10L		2302
	jsr	GetFreeDirBlk		2303
	txa			2304
	bne	:err		2305
				2306
	tya		;Offset Eintrag.	2307
	pha			2308
				2309
	ldy	#0	;Dateiname.	2310
::3	lda	(r13),y		2311
	beq	:4		2312
	sta	dirEntryBuf +3,y		2313

```

2314          iny
2315          bne      :3
2316
2317 ::4          cpy      #16
2318          bcs      :5
2319          lda      #$a0
2320          sta      dirEntryBuf +3,y
2321          iny
2322          bne      :4
2323
2324 ::5          pla
2325          pha
2326          tay                      ;Offset Eintrag.
2327          ldx      #0
2328 ::6          lda      dirEntryBuf,x
2329          sta      diskBlkBuf,y
2330          iny
2331          inx
2332          cpx      #30
2333          bcc      :6
2334
2335 ; Ordner-Nr. in Eintrag schreiben.
2336          pla                      ;Offset Eintrag.
2337          cmp      #$02             ;Erster Eintrag?
2338          bne      :7             ; => Nein, weiter...
2339
2340          clc                      ;Offset = 2+32-1 = 33
2341          adc      #TDFOLDER_POS0 -1
2342
2343 ; Eintrag #0 : diskBlkBuf +32
2344 ; Eintrag #1-7: CBMDateityp -1
2345 ::7          tay                      ;Offset -1
2346          dey
2347
2348          lda      V0A71             ;Nr. Ziel-Ordner.
2349          sta      diskBlkBuf,y
2350
2351 ; Eintrag schreiben.
2352 ;          lda      #< diskBlkBuf      ;Zeiger auf
2353 ;          sta      r4L                ;Zwischenspeicher.
2354 ;          lda      #> diskBlkBuf
2355 ;          sta      r4H
2356 ;          jsr      PutBlock_dskBuf
2357          txa
2358          bne      :exit
2359
2360          jsr      PutDirHead         ;BAM aktualisieren.
2361
2362 ::exit      rts
2363
2364 ; Sektorkette von Disk einlesen.
2365 ; TD13: ReadChain2
2366 :Z0E15      inc      r4H
2367
2368 ; TD13: ReadChain

```

:Z0E17	jsr	ReadBlock	2369
	txa		2370
	bne	:done	2371
			2372
	jsr	Z0EA0 ;Weitere Daten?	2373
	beq	:done ; => Nein, Ende...	2374
			2375
	ldy	r4H	2376
	iny		2377
	cpy	MHB04AC ;HB Ende CopyBuf.	2378
	bcc	Z0E15	2379
			2380
			2381
			2382
			2383
			2384
			2385
			2386
			2387
			2388
			2389
:Z0E34	lda	r9L	2390
	sta	r3L	2391
	sta	r1L	2392
	lda	r9H	2393
	sta	r3H	2394
	sta	r1H	2395
			2396
	dec	r4H	2397
			2398
::1	inc	r4H	2399
	jsr	SetNextFree	2400
	txa		2401
	bne	:done	2402
			2403
	ldy	#0	2404
	lda	(r4),y ;Weitere Daten?	2405
	beq	:2 ; => Nein, Ende...	2406
			2407
	lda	r3L	2408
	sta	(r4),y	2409
	iny		2410
	lda	r3H	2411
	sta	(r4),y	2412
			2413
::2	jsr	WriteBlock ;Block schreiben.	2414
	txa		2415
	bne	:done	2416
			2417
	lda	r3L	2418
	sta	r1L	2419
	sta	r9L	2420
	lda	r3H	2421
	sta	r1H	2422
	sta	r9H	2423

```

2424
2425         ldy    #0
2426         lda    (r4),y
2427         beq    :done
2428         lda    r4H
2429         tay
2430         iny
2431         cpy    MHB04AC           ;HB Ende CopyBuf.
2432         bcc    :1                ;Nächster Sektor.
2433
2434 ; Speicher voll oder Fehler.
2435 ::done      rts
2436
2437 ; Sektorkette auf Disk vergleichen.
2438 ;
2439 ; TODO:
2440 ; Verify auf 1541/71 ausführen, da nur
2441 ; bei diesen Laufwerkstreibern eine
2442 ; Verify-Funktion enthalten ist.
2443 ;
2444 ; TD13: VerifyChain2
2445 :Z0E7B      inc    r4H
2446
2447 ; TD13: VerifyChain
2448 :Z0E7D      jsr    VerWriteBlock
2449             txa
2450             bne    :done
2451
2452             jsr    Z0EA0           ;Weitere Daten?
2453             beq    :done           ; => Nein, Ende...
2454
2455             ldy    r4H
2456             iny
2457             cpy    MHB04AC           ;HB Ende CopyBuf.
2458             bcc    Z0E7B           ;Nächster Sektor.
2459
2460 ;             ldx    #NO_ERROR       ;X bereits gesetzt!
2461 ::done      rts
2462
2463 ; Ersten freien Block suchen.
2464 ; TD13: Get1stBlock
2465 ;
2466 ; TODO:
2467 ; Bei NativeMode erst ab $01/$40.
2468 :Z0E9C      ldy    #1
2469             sty    r3L
2470             dey
2471             sty    r3H
2472             jsr    SetNextFree
2473
2474             lda    r3L
2475             sta    r9L
2476             lda    r3H
2477             sta    r9H
2478             rts

```



		2479
; Link-Sektor einlesen.		2480
; TD13: n.v.		2481
:Z0EA0	ldy #1 ;Weitere Daten?	2482
	lda (r4),y	2483
	sta r1H	2484
	dey	2485
	lda (r4),y	2486
	sta r1L	2487
	rts	2488
		2489
; Dialogbox: Datei überschreiben?		2490
; TD13: überschreibenbox		2491
.D0EAF	b %10000001	2492
		2493
	b DBVARSTR,\$28,\$2e	2494
	b r13L	2495
		2496
	b DBTXTSTR,\$08,\$0e	2497
	w T04F0	2498
	b DBTXTSTR,\$08,\$18	2499
	w :t1	2500
		2501
	b YES , \$11,\$08	2502
		2503
	b DBTXTSTR,\$14,\$23	2504
	w T04F6	2505
		2506
	b DBTXTSTR,\$08,\$2e	2507
	w T04F2	2508
	b DBTXTSTR,\$08,\$38	2509
	w :t2	2510
		2511
	b OK , \$11,\$28	2512
		2513
	b DBTXTSTR,\$08,\$52	2514
	w T04F5	2515
		2516
	b CANCEL , \$11,\$48	2517
		2518
	b NULL	2519
		2520
if LANG = LANG_DE		2521
::t1	b BOLDON,"Überschreiben"	2522
	b PLAINTEXT," ?",0	2523
		2524
::t2	b BOLDON,"Nicht ersetzen"	2525
	b PLAINTEXT," ?",0	2526
endif		2527
if LANG = LANG_EN		2528
::t1	b BOLDON,"Overwrite"	2529
	b PLAINTEXT," ?",0	2530
		2531
::t2	b BOLDON,"Do not replace"	2532
	b PLAINTEXT," ?",0	2533

```

2534 endif
2535
2536 ; Dialogbox: Datei schreibgeschützt.
2537 ; TD13: nichtüberschreibenbox
2538 :D0EDF          b %10000001
2539
2540                b DBVARSTR,$12,$38
2541                b r13L
2542                b DBTXTSTR,$08,$38
2543                w T29D3
2544
2545                b DBTXTSTR,$08,$10
2546                w T29CF
2547
2548                b DBTXTSTR,$08,$20
2549                w T04F0
2550                b DBTXTSTR,$08,$2a
2551                w T04F3
2552
2553                b DBTXTSTR,$08,$44
2554                w :t1
2555
2556                b OK          , $11,$48
2557
2558                b NULL
2559
2560 if LANG = LANG_DE
2561 ::t1            b "Datei wird nicht ersetzt.",0
2562 endif
2563 if LANG = LANG_EN
2564 ::t1            b "File will not be replaced.",0
2565 endif
2566
2567 if LANG = LANG_DE
2568 :T04F0          b "Auf Ziel-Disk vorhanden.",0
2569
2570 :T04F1          b BOLDON
2571 :T04F2          b "Datei:"
2572                b PLAINTEXT,0
2573
2574 :T04F3          b "=> Schreibschutz aktiv!",0
2575
2576 :T04F5          b "Vorgang abbrechen",0
2577
2578 :T04F6          b BOLDON
2579                b "Oder:"
2580                b PLAINTEXT,0
2581 endif
2582 if LANG = LANG_EN
2583 :T04F0          b "Available on target disk.",0
2584
2585 :T04F1          b BOLDON
2586 :T04F2          b "File:"
2587                b PLAINTEXT,0
2588

```

:T04F3	b "=> Write protected!",0	2589
		2590
:T04F5	b "Cancel process",0	2591
		2592
:T04F6	b BOLDON	2593
	b "Or:"	2594
	b PLAINTEXT,0	2595
endif		2596
		2597
; Ordner löschen?		2598
:D0500	b %10000001	2599
		2600
	b DBVARSTR,\$12,\$38	2601
	b r13L	2602
	b DBTXTSTR,\$08,\$38	2603
	w T29D3	2604
		2605
	b DBTXTSTR,\$08,\$10	2606
	w T29CF	2607
		2608
	b DBTXTSTR,\$08,\$20	2609
	w T04F0	2610
	b DBTXTSTR,\$08,\$2a	2611
	w T0510	2612
		2613
	b CANCEL ,,\$11,\$48	2614
	b YES ,,\$01,\$48	2615
	b NO ,,\$09,\$48	2616
		2617
	b NULL	2618
		2619
if LANG = LANG_DE		2620
:T0510	b "Ordner inkl. Inhalt ersetzen?",0	2621
endif		2622
if LANG = LANG_EN		2623
:T0510	b "Delete folder incl. content?",0	2624
endif		2625
		2626
; Datei auf Diskette suchen.		2627
; Übergabe: X = Zeiger auf r10(Source)/r11(Target)		2628
; r9L = 0: Datei auf Disk suchen.		2629
; Rückgabe: X = 0: Kein Fehler => InitForIO		2630
; X > 0: Fehler, DoneWithIO		2631
; TD13: Search_Turbo		2632
:Z0F35	txa	2633
	pha	2634
	jsr DoneWithIO ;I/O abschalten.	2635
	pla	2636
	tax	2637
		2638
	lda r2L ;Status Daten/Puffer	2639
	pha ;zwischenspeichern.	2640
		2641
	jsr Z0F30 ;Vorh.Datei löschen.	2642
		2643

```

2644          pla          ;Status Daten/Puffer
2645          sta    r2L    ;zurücksetzen.
2646
2647          txa
2648          bne     :exit
2649
2650 ; Kopieren initialisieren.
2651          jsr     EnterTurbo
2652          txa
2653          bne     :exit
2654
2655          jsr     InitForIO
2656 ::exit    rts
2657
2658 ; Datei auf Quelle/Ziel suchen.
2659 ; Übergabe: X   = Zeiger auf r10(Source)/r11(Target)
2660 ;             r9L = 0: Datei auf Disk suchen.
2661 ; Rückgabe: X   = Fehler.
2662 :Z0F30    lda    C042A    ;Oberstes Fenster
2663          pha          ;zwischenspeichern.
2664
2665          cpx    #r10      ;Source/Target?
2666          bne     :1        ; => Target-Disk.
2667
2668 ; Source-Disk:
2669          ldy    C042A    ;Oberstes Fenster.
2670          lda    V0A72    ;Lfwk. Source-Disk.
2671          bne     :2
2672
2673 ; Target-Disk:
2674 ::1       ldy    WIN042F    ;Aktives Fenster.
2675          lda    V0A73    ;Lfwk. Target-Disk.
2676
2677 ::2       sty    V10BB    ;Quell-/Ziel-Fens.
2678          sta    V10BA    ;Quell-/Ziel-Lfwk.
2679
2680 ; Hinweis:
2681 ; Der Name der Diskette in r6 wird
2682 ; nicht mehr verwendet.
2683 if FALSE
2684          lda    zpage +0,x    ;Zeiger auf Name von
2685          sta    r6L          ;Source-/Target-Disk.
2686          lda    zpage +1,x
2687          sta    r6H
2688 endif
2689
2690 ;          ldy    V10BB    ;Quell-/Ziel-Fens.
2691          sty    C042A    ;Oberstes Fenster.
2692
2693 ; Hinweis:
2694 ; r0 bis r4 beinhalten keine wichtigen
2695 ; Daten, lediglich r2L wird benötigt.
2696 ; r2L wurde aber bereits auf den Stack
2697 ; zwischengespeichert.
2698 if FALSE

```

```

                                ldy    #(r4H - r0L)                2699
::3                            lda    r0,y                      2700
                                pha                                2701
                                dey                                2702
                                bpl    :3                        2703
endif                                                                    2704
                                                                    2705
                                lda    V10BA                    ;Quell-/Ziel-Lfwk. 2706
                                jsr    S06AB                    ;Laufwerk öffnen. 2707
                                                                    2708
; Hinweis:                                                                2709
; Nicht mehr erforderlich, siehe oben. 2710
if FALSE                                                                2711
                                ldy    #0                      2712
::4                            pla                                2713
                                sta    r0,y                    2714
                                iny                                2715
                                cpy    #(r4H - r0L) +1         2716
                                bcc    :4                      2717
endif                                                                    2718
                                                                    2719
                                pla                                2720
                                sta    C042A                    ;Oberstes Fenster. 2721
                                                                    2722
                                txa                                2723
                                bne    :5                        ; Fehler?         2724
                                                                    ; => Ja, Ende... 2725
                                lda    r9L                      ;Copy in Arbeit? 2726
                                bne    :5                        ; => Ja, weiter... 2727
                                                                    2728
                                jsr    Z0FC8                    ;Datei vorhanden? 2729
                                                                    2730
::5                            rts                                2731
                                                                    2732
:V10BA                        b $00    ;Quell-/Ziel-Laufwerk. 2733
:V10BB                        b $00    ;Quell-/Ziel-Fenster. 2734
                                                                    2735
; TD13: initsearchturbo 2736
; Übergabe: r13 = Zeiger auf Dateiname. 2737
:Z0FC8                        2738
                                                                    2739
; Hinweis: 2740
; Wird nur von Z0F35 aufgerufen. Die 2741
; Routine führt bereits ein OpenDisk, 2742
; OpenPartition oder OpenSubDir aus. 2743
; Kann hier entfallen. 2744
;                                jsr    OpenDisk                2745
;                                txa                                2746
;                                bne    :err                    2747
                                                                    2748
; Register zwischenspeichern. 2749
                                ldy    #(r13H - r9L)            ;Register r9-r13 2750
::11                          lda    r9,y                      ;zwischenspeichern. 2751
                                pha                                2752
                                dey                                2753

```

```

2754          bpl      :11
2755
2756          ldy      #30 -1          ;dirEntryBuf
2757 ::12          lda      dirEntryBuf,y      ;zwischenspeichern.
2758          pha
2759          dey
2760          bpl      :12
2761
2762 ; Datei vorhanden / ersetzen?
2763          jsr      Z10C0          ;Datei suchen.
2764
2765 ; Register zurücksetzen.
2766 ::done        ldy      #0          ;dirEntryBuf wieder
2767 ::21          pla          ;herstellen.
2768          sta      dirEntryBuf,y
2769          iny
2770          cpy      #30
2771          bcc      :21
2772
2773          ldy      #0          ;Register r9-r13
2774 ::22          pla          ;wieder herstellen.
2775          sta      r9,y
2776          iny
2777          cpy      #(r13H - r9L) +1
2778          bcc      :22
2779
2780 ::err          rts
2781
2782 ; Datei vorhanden / ersetzen?
2783 :Z10C0        lda      r13L          ;Zeiger auf Name.
2784          ldx      r13H
2785
2786          sta      fileHeader +0      ;Name für DelFile
2787          stx      fileHeader +1      ;zwischenspeichern.
2788
2789 ;          sta      r6L          ;Zeiger auf Name
2790 ;          stx      r6H          ;für FindFile.
2791          jsr      S1414          ;FindFile.
2792          txa
2793          beq      :found
2794
2795          cpx      #FILE_NOT_FOUND
2796          bne      :cancel
2797          ldx      #NO_ERROR
2798          beq      :cancel          ; => Suche beenden.
2799
2800 ; Ziel-Datei vorhanden.
2801 ::found        jsr      Z10C8          ;Datei ersetzen?
2802          bcs      :cancel          ; => Nein, Ende...
2803
2804          jsr      Z10D0          ;Ziel-Datei löschen.
2805          bcc      :exit          ; => Kein Fehler...
2806
2807 ; Suche beenden.
2808 ; Übergabe: X = Fehler (NO_ERROR/FILE_EXISTS/CANCEL_ERR)

```

```

::cancel      lda    V0B64          ;Fehler-Status?      2809
               beq    :exit          ; => Kein Fehler...   2810
               tax      ; => Diskfehler.                 2811
               ;                                           2812
::exit        rts                                           2813
               ;                                           2814
; TopDesk-Programmcode speichern/laden.                 2815
; COPYMEM_MAX ($1200) bis $7FFF.                         2816
; Verändert: Y                                           2817
;                                                         2818
; Hinweis:                                               2819
; Die Routine ist nur erforderlich,                     2820
; wenn während der Routine ":CopyFile"                 2821
; eine Dialogbox angezeigt werden soll.                2822
; Wenn die Routine ":search_turbo" von                 2823
; ":CopyService" auferufen wird muss                  2824
; der Speicher nicht gerettet werden.                 2825
:Z10BC        ldy    V10CF          ;SwapRAM notwendig?  2826
               bne    :exit          ; => Nein, Ende...    2827
               ;                                           2828
               pha                                           2829
               txa                                           2830
               pha                                           2831
               ;                                           2832
               lda    MP3_64K_DATA    ;Speicherbank für   2833
               sta    :rambank        ;SwapFile.         2834
               ;                                           2835
               ldx    #(r3L - r0L)    2836
::1           lda    r0,x             2837
               pha                                           2838
               lda    :bufsize,x      2839
               sta    r0,x            2840
               dex                     2841
               bpl    :1              2842
               ;                                           2843
               jsr    SwapRAM         2844
               ;                                           2845
               ldx    #0              2846
::2           pla                     2847
               sta    r0,x            2848
               inx                     2849
               cpx    #(r3L - r0L) +1 2850
               bcc    :2              2851
               ;                                           2852
               pla                     2853
               tax                     2854
               pla                     2855
               ;                                           2856
::exit        rts                                           2857
               ;                                           2858
::bufsize     w    COPYMEM_MAX        2859
               w    $0000             2860
               w    (OS_VARS - COPYMEM_MAX)             2861
::rambank     b    $00               2862
               ;                                           2863

```

```

2864 ; Dialogbox: Ziel ersetzen?
2865 :Z10C8      jsr    Z10F0      ;Ordner/Verz./Datei?
2866           stx    V10E1      ;$00 = Datei.
2867
2868           cpx    #%01000000    ;Ordner?
2869           beq    :folder      ; => Ja, weiter...
2870
2871           lda    dirEntryBuf
2872           and    #FMODE_WRPROT ;Schreibschutz?
2873           bne    :wr_prot      ; => Ja, weiter...
2874
2875 ;           txa
2876 ;           bmi    :yes         ; => Verzeichnis.
2877 ;           beq    :file        ; => Datei.
2878
2879 ; Datei/Verzeichnis ersetzen.
2880 ::file_dir   lda    V04B1      ;Ziel ersetzen?
2881           bne    :yes         ; => Ja, weiter...
2882
2883           lda    #< D0EAF      ;überschreiben?
2884           ldx    #> D0EAF
2885           bne    :dodlg
2886
2887 ; Ordner ersetzen.
2888 ::folder     bit    V10CF      ;Ordner => Ordner?
2889           bvs    :skip        ; => Ja, Ende...
2890
2891 ; Ordner wird durch Datei oder ein
2892 ; Verzeichnis ersetzt.
2893           lda    #< D0500      ;Ordner löschen?
2894           ldx    #> D0500
2895           bne    :dodlg
2896
2897 ; Schreibschutz.
2898 ::wr_prot    lda    #< D0EDF    ;Datei ist
2899           ldx    #> D0EDF      ;schreibgeschützt.
2900
2901 ; Datei/Ordner.
2902 ; Übergabe: A/X = Zeiger auf Dialogbox.
2903 ::dodlg      jsr    Z10BC      ;Swap APP_RAM.
2904
2905           jsr    S2080          ;DBox, k. Maustaste.
2906
2907           jsr    Z10BC          ;Swap APP_RAM.
2908
2909           lda    sysDBData
2910           cmp    #YES          ;überschreiben?
2911           beq    :yes          ; => Ja, weiter...
2912           cmp    #CANCEL       ;Abbruch?
2913           bne    :no           ; => Nein, weiter...
2914
2915           ldx    #CANCEL_ERR
2916           stx    V0B64          ;Fehler-Status.
2917           b $2c
2918 ::no         ldx    #FILE_EXISTS ;Datei existiert.

```



```

                                b $2c                                2919
::skip                        ldx    #NO_ERROR                    2920
                                sec                                     2921
                                rts                                2922
                                ; => Suche beenden.                2923
::yes                        clc                                     2924
                                rts                                2925
                                ; => Datei löschen.                2926
; Ziel-Datei löschen.        2927
; Rückgabe: C=1: Fehler, nicht genügend Speicher.                2928
; C=0: Kein Fehler, Datei gelöscht.                                2929
:Z10D0                        lda    WIN042F                      2930
                                ;Aktives Fenster.                2931
                                asl                                     2932
                                tax                                     2933
                                lda    V0527 +0,x                2934
                                ;Anzahl freie Blocks.            2935
                                clc                                     2936
                                adc    dirEntryBuf +28            2937
                                sta    V0527 +0,x                2938
                                lda    V0527 +1,x                2939
                                adc    dirEntryBuf +29            2940
                                sta    V0527 +1,x                2941
                                lda    dirEntryBuf +29            2942
                                pha                                     2943
                                lda    dirEntryBuf +28            2944
                                pha                                     2945
                                lda    V0B62 +0                    2946
                                sta    dirEntryBuf +28            2947
                                lda    V0B62 +1                    2948
                                sta    dirEntryBuf +29            2949
                                ;                                     2950
                                jsr    Z0B38                      2951
                                ;Noch Blocks frei?                2952
                                pla                                     2953
                                sta    dirEntryBuf +28            2954
                                pla                                     2955
                                sta    dirEntryBuf +29            2956
                                ;                                     2957
                                txa                                2958
                                ;Blocks frei?                    2959
                                beq    :1                        2960
                                ; => Ja, weiter...                2961
                                ldx    #NO_BLOCKS                2962
                                stx    V0B64                      2963
                                ;Fehler-Status.                    2964
                                sec                                     2965
                                rts                                2966
::1                            lda    #NO_ERROR                    2967
                                sta    V0B64                      2968
                                ;                                     2969
                                jsr    Z10E0                      2970
                                ;Eintrag löschen.                2971
                                clc                                     2972
                                rts                                2973

```

```

2974
2975 ; Existierende Datei/Ordner/Verzeichnis löschen.
2976 :Z10E0      lda    V10E1          ;$00 = Datei?
2977           beq     :file          ; => Ja, weiter...
2978
2979 ; Verzeichnis/Ordner löschen.
2980 ::1         jsr     Z10BC          ;Swap APP_RAM.
2981
2982           lda     C042A          ;Ziel-Fenster setzen.
2983           pha
2984           lda     WIN042F
2985           sta     C042A
2986
2987           lda     V5C5B          ;Datei-Nr. speichern.
2988           pha
2989
2990           lda     #$00
2991           sta     V5C5A
2992
2993           jsr     S5B38          ;Eintrag löschen.
2994
2995           pla
2996           sta     V5C5B          ;Datei-Nr. setzen.
2997
2998           pla
2999           sta     C042A          ;Fenster setzen.
3000
3001           txa
3002           ora     V5C5A          ;Fehlerstatus
3003           pha          ;addieren und
3004           ;zwischenspeichern.
3005
3006           ldx     V04A9          ;Kop.od.verschieben.
3007           jsr     S55B4          ;Status anzeigen.
3008
3009           jsr     Z10BC          ;Swap APP_RAM.
3010
3011           pla
3012           tax
3013 ;           ldx     #NO_ERROR
3014           rts
3015
3016 ; Datei löschen.
3017 ::file      lda     fileHeader +0
3018           ldx     fileHeader +1
3019           jmp     S0B34          ;Datei löschen.
3020
3021 ; Auf Datei/Ordner/Verzeichnis testen.
3022 :Z10F0      ldx     #%10000000    ;$80 = NM/Dir.
3023
3024           lda     dirEntryBuf     ;Dateityp einlesen.
3025           and     #FTYPE_MODES
3026           cmp     #FTYPE_DIR      ;CMD-Verzeichnis?
3027           beq     :1              ; => Ja, Ende...
3028
3029           ldx     #%01000000      ;$40 = Ordner.

```

```

3029
3030      lda    dirEntryBuf +1      ;Daten vorhanden?
3031      beq     :1                  ; => Kein Ordner.
3032
3033      ldx     %%00000000          ;$00 = Datei.
3034
3035  ::1      rts
3036
3037 ; Dateityp-Flags:
3038 :V10CF      b $00      ;$00 =Datei, SwapRam aktivieren.
3039              ;$80/$40 =Verz./Ordner., kein Swap.
3040 :V10E1      b $00      ;$00 =Datei, $80/$40 =Verz./Ordner.
3041
3042 ; Fensterkoordinaten nach r2-r4.
3043 ; Übergabe: X = Fenster-Nr.
3044 ; TD13: GetWinAdrRec
3045 :Z11A3      jsr     S11A9          ;a0=Fensterdaten.
3046              jmp     Z11BD          ;Fens.daten > r2-r4.
3047
3048 ; Zeiger auf Fensterdaten setzen.
3049 ; Übergabe: X = Fenster-Nr.
3050 ; TD13: GetWinAdr
3051 .S11A9      lda     V0450,x        ;Offset für Tabelle.
3052              clc
3053              adc     WIN0446 +0     ;Zeiger Fensterdaten.
3054              sta     a0L
3055              lda     #$00
3056              adc     WIN0446 +1
3057              sta     a0H
3058              rts
3059
3060 ; Fensterkoord. nach :a4 kopieren.
3061 ; Übergabe: a0 = Fensterdaten
3062 ; TD13: GetWinRec3
3063 :Z11BA      lda     #a4
3064              b $2c
3065
3066 ; Fensterkoord. nach :r2 kopieren.
3067 ; Übergabe: a0 = Fensterdaten
3068 ; TD13: GetWinRec
3069 :Z11BD      lda     #r2
3070              b $2c
3071
3072 ; Fensterkoord. nach :r5 kopieren.
3073 ; Übergabe: a0 = Fensterdaten
3074 ; TD13: GetWinRec2
3075 .S11C0      lda     #r5
3076              sta     :copyto +1
3077
3078              ldy     #6 -1
3079 ::1          lda     (a0),y
3080 ::copyto    sta     $00ff,y        ;LB wird verändert!
3081              dey
3082              bpl     :1
3083              rts

```

```

3084
3085 ; Textgrenzen nach r5-r7 kopieren.
3086 ; TD13: GetClipRec
3087 .S11D0          ldy    #6 -1
3088 ::1            lda    windowTop,y
3089              sta    r5,y
3090              dey
3091              bpl     :1
3092              rts
3093
3094 ; Hinweis:
3095 ; Ab hier beginnt COPYMEM_MAX.
3096 ; Siehe Routine ab L0A74.
3097 ;
3098 :CUR_COPYMEM_MAX
3099 .COPYMEM_MAX     = (( >CUR_COPYMEM_MAX ) +1) *256
3100
3101 ; Fenster zeichnen.
3102 ; Übergabe: a4-a6 = Fenster-Rechteck.
3103 ; TD13: DrawWindow2
3104 ; Aufruf nur über S141D(DrawWindowB) und Z1428(DrawWindowC)
3105 :Z11DC          ldy    #(r4H - r2L)
3106 ::1            lda    a4,y
3107              sta    r2,y
3108              dey
3109              bpl     :1
3110
3111              lda    #$00          ;Vorgabe:
3112              sta    V0454          ;Fenster verdeckt.
3113
3114 ;              lda    #$00          ;Wozu hier
3115 ;              jsr    SetPattern    ;Füllmuster setzen?
3116
3117              jsr    S11D0          ;Textgrenzen r5-r7.
3118              jsr    S1D1C          ;Schnittfläche?
3119              bcc     :2            ; => Ja, weiter...
3120              rts
3121
3122 ; Fenster verdeckt, Ausschnitt neu zeichnen.
3123 ::2            ldy    #(r4H - r2L)    ;Fenster abschneiden?
3124 ::3            lda    r2,y
3125              cmp    a4,y
3126              bne     :4            ; => Ja, weiter...
3127              dey
3128              bpl     :3
3129
3130              lda    #$01          ;Fenster ist
3131              sta    V0454          ;nicht verdeckt.
3132
3133 if ENABLE_DIFFICOL = FALSE
3134 ::4            ldy    #$10
3135              lda    (a9),y          ;Fensterfarbe.
3136 endif
3137 if ENABLE_DIFFICOL = TRUE
3138 ::4            lda    V621A          ;Fensterfarbe.

```

endif				3139
				3140
	jsr	S1561	;Screencolor setzen.	3141
				3142
	lda	#\$00	;Füllmuster.	3143
	jsr	S140C	;Rechteck zeichnen.	3144
				3145
	lda	#< V130A	;Zeiger auf die	3146
	sta	a1L	;Grafikbefehle für	3147
	lda	#> V130A	;das Fenster.	3148
	sta	a1H		3149
				3150
::loop	lda	a2H		3151
	sta	r11L		3152
				3153
	lda	a3L		3154
	sta	r3L		3155
	lda	a3H		3156
	sta	r3H		3157
				3158
	ldy	#2	;Befehl einlesen und	3159
::11	lda	(a1),y	;nach a2H/a3L/a3H	3160
	sta	a2H,y	;übertragen.	3161
	dey			3162
	bpl	::11		3163
				3164
	lda	a3H		3165
	cmp	#\$ff	;Ende erreicht?	3166
	bne	::21	; => Nein, weiter...	3167
	clc			3168
	rts			3169
				3170
; a2H: Y-Offset				3171
; a3L: X-Offset/Lowbyte				3172
; a3H: Befehl und X-Offset/Highbyte				3173
::21	tax			3174
	and	##111100000	;Position.	3175
	sta	a2L		3176
	txa			3177
	and	##00011111	;X-Offset.	3178
	sta	a3H		3179
				3180
	bit	a2L	;Rechter Rand?	3181
	bvs	:right	; => Ja, weiter...	3182
				3183
::left	lda	a5L		3184
	clc			3185
	adc	a3L		3186
	sta	a3L		3187
	lda	a5H		3188
	adc	a3H		3189
	sta	a3H		3190
				3191
	jmp	:cont1		3192
				3193

```

3194 ::right      lda    a6L
3195             sec
3196             sbc    a3L
3197             sta    a3L
3198             lda    a6H
3199             sbc    a3H
3200             sta    a3H
3201
3202 ::cont1       bit    a2L                ;Unterer Rand?
3203             bmi    :bottom            ; => Ja, weiter...
3204
3205 ::top         clc
3206             lda    a4L
3207             adc    a2H
3208             sta    a2H
3209
3210             jmp    :cont2
3211
3212 ::bottom      lda    a4H
3213             sec
3214             sbc    a2H
3215             sta    a2H
3216
3217 ; Linie zeichnen.
3218 ::cont2       lda    a2L
3219             and    #%00100000          ;Linie zeichnen?
3220             beq    :next                ; => Nein, weiter...
3221
3222 ::line        lda    a3L
3223             sta    r4L
3224             lda    a3H
3225             sta    r4H
3226
3227             lda    a2H
3228             sta    r11H
3229
3230             lda    V0454                ;Fenster verdeckt?
3231             bne    :41                  ; => Nein, weiter...
3232
3233 ; Fenster in ClipRect einpassen.
3234             lda    r11L                ;Y-oben/unten testen.
3235             ldx    r11H                ;Koord. tauschen wenn
3236             cmp    r11H                ;Y-oben > Y-unten.
3237             bcc    :31
3238
3239             lda    r11H
3240             ldx    r11L
3241 ::31          sta    r2L
3242             stx    r2H
3243
3244 ;           ldx    r12L
3245           stx    r11L
3246 ;           ldy    r2H
3247           sty    r11H
3248

```

	lda	r3L	;X-li/re testen.	3249
	sec		;Koord. tauschen wenn	3250
	sbc	r4L	;X-links > X-rechts.	3251
	lda	r3H		3252
	sbc	r4H		3253
	bmi	:32		3254
				3255
	ldx	r3L		3256
	lda	r4L		3257
	sta	r3L		3258
	stx	r4L		3259
	ldx	r3H		3260
	lda	r4H		3261
	sta	r3H		3262
	stx	r4H		3263
				3264
::32	jsr	S11D0	;Textgrenzen r5-r7.	3265
				3266
	jsr	S1D1C	;Schnittfläche?	3267
	bcs	:next	; => Nicht vorhanden.	3268
				3269
	lda	r2L	;Y-Koord. evtl.	3270
	sta	r11L	;auf Fens. reduziert.	3271
	lda	r2H		3272
	sta	r11H		3273
				3274
	lda	r11L	;Punkt zeichnen?	3275
	cmp	r11H		3276
	bne	:41		3277
				3278
	lda	r3L		3279
	cmp	r4L		3280
	bne	:41		3281
	lda	r3H		3282
	cmp	r4H		3283
	bne	:41	; => Nein, Linie...	3284
				3285
	lda	#\$00		3286
	sec			3287
	jsr	DrawPoint		3288
	jmp	:next		3289
				3290
::41	lda	#\$00		3291
	sec			3292
	jsr	DrawLine		3293
				3294
; Nächster Befehl.				3295
::next	lda	a1L		3296
	clc			3297
	adc	#3		3298
	sta	a1L		3299
	bcc	:cont3		3300
	inc	a1H		3301
::cont3	jmp	:loop	;Weiterzeichnen.	3302
				3303

```

3304 ; Tabelle mit Fensterdaten.
3305 ; Aufbau: y,x,pos
3306 ; Über YO/YU/XL/XR definiert man von
3307 ; welcher Ecke aus der Offset x/y aus
3308 ; berechnet werden soll.
3309 ; NP setzt eine neue Koordinate und
3310 ; EL zeichnet vom letzten Punkt bis
3311 ; zum neuen Punkt eine Linie.
3312 ; SX/SY definieren die Breite des
3313 ; Titels/Status/Navigationbalken.
3314 ;
3315 ; Hinweis:
3316 ; Wegen des Clippings sollten Linien
3317 ; nur von links nach rechts bzw. von
3318 ; oben nach unten gezeichnet werden!
3319 ; Diagonale Linien können zu Problemen
3320 ; bei der Anzeige führen wenn der Teil
3321 ; des Fensters verdeckt war:
3322 ; In dem Fall werden die Koordinaten
3323 ; xl/xr und yl/yr ggf. getauscht und
3324 ; eine diagonale Linie zeigt dann in
3325 ; die falsche Richtung!
3326
3327 :V130A
3328 ; Rahmen:
3329             b  0, 0,YO!XL!NP
3330             b  0, 0,YO!XR!EL
3331             b  0, 0,YU!XR!EL
3332             b  0, 0,YU!XL!EL
3333             b  0, 0,YO!XL!EL
3334
3335 ; Titelrahmen:
3336             b TY, 0,YO!XL!NP
3337             b TY, 0,YO!XR!EL
3338
3339 ; Unterer Rahmen:
3340             b SY, 0,YU!XL!NP
3341             b SY, 0,YU!XR!EL
3342
3343 ; Rechter Rahmen inkl. Gadget-Rahmen rechts:
3344             b  0,SX,YO!XR!NP
3345             b  0,SX,YU!XR!EL
3346
3347 ; Gadget-Rahmen links/oben:
3348             b  0,13,YO!XL!NP
3349             b SY,13,YO!XL!EL
3350
3351 ; Close-Gadget:
3352 ; (Relativ von links/oben)
3353             b  3, 4,YO!XL!NP
3354             b  3, 8,YO!XL!EL
3355             b  5, 8,YO!XL!EL
3356             b  5, 4,YO!XL!EL
3357             b  3, 4,YO!XL!EL
3358

```



<b>; Gadget-Rahmen links/oben:</b>	3359
<b>b 0,TOL-1,YO!XL!NP</b>	3360
<b>b SY,TOL-1,YO!XL!EL</b>	3361
	3362
<b>; Name-Gadget:</b>	3363
<b>; (Relativ von links/oben)</b>	3364
<b>b 6,15,YO!XL!NP</b>	3365
<b>b 2,15,YO!XL!EL</b>	3366
<b>b 6,19,YO!XL!EL</b>	3367
<b>b 2,19,YO!XL!EL</b>	3368
	3369
<b>; Hide-Gadget:</b>	3370
<b>; (Relativ von rechts/oben)</b>	3371
<b>b 0,TOR-1,YO!XR!NP</b> <b>;Trennlinie.</b>	3372
<b>b TY,TOR-1,YO!XR!EL</b>	3373
	3374
<b>b 2,SX +8,YO!XR!NP</b> <b>;Inaktiv.Fenster.</b>	3375
<b>b 2,SX +6,YO!XR!EL</b>	3376
<b>b 4,SX +6,YO!XR!EL</b>	3377
<b>b 4,SX +8,YO!XR!EL</b>	3378
<b>b 2,SX +8,YO!XR!EL</b>	3379
	3380
<b>b 4,SX +6,YO!XR!NP</b> <b>;Aktives Fenster.</b>	3381
<b>b 4,SX +4,YO!XR!EL</b>	3382
<b>b 6,SX +4,YO!XR!EL</b>	3383
<b>b 6,SX +6,YO!XR!EL</b>	3384
<b>b 4,SX +6,YO!XR!EL</b>	3385
<b>b 5,SX +5,YO!XR!NP</b>	3386
<b>b 5,SX +5,YO!XR!EL</b>	3387
	3388
<b>; Maximize-Gadget:</b>	3389
<b>; (Relativ von rechts/oben)</b>	3390
<b>b 2,SX-2,YO!XR!NP</b>	3391
<b>b 2, 2,YO!XR!EL</b>	3392
<b>b TY-2, 2,YO!XR!EL</b>	3393
<b>b TY-2,SX-2,YO!XR!EL</b>	3394
<b>b 2,SX-2,YO!XR!EL</b>	3395
	3396
<b>; Pfeil nach oben:</b>	3397
<b>; (Relativ von rechts/oben)</b>	3398
<b>b TY+3,SX-3,YO!XR!NP</b>	3399
<b>b TY+7,SX-3,YO!XR!EL</b>	3400
<b>b TY+7,SX-5,YO!XR!EL</b>	3401
<b>b TY+3,SX-5,YO!XR!EL</b>	3402
<b>b TY+2,SX-4,YO!XR!NP</b>	3403
<b>b TY+7,SX-4,YO!XR!EL</b>	3404
<b>b TY+4,SX-2,YO!XR!NP</b>	3405
<b>b TY+4,SX-6,YO!XR!EL</b>	3406
	3407
<b>; Pfeil nach unten:</b>	3408
<b>; (Relativ von rechts/unten)</b>	3409
<b>b SY+3,SX-3,YU!XR!NP</b>	3410
<b>b SY+7,SX-3,YU!XR!EL</b>	3411
<b>b SY+7,SX-5,YU!XR!NP</b>	3412
<b>b SY+3,SX-5,YU!XR!EL</b>	3413

```

3414          b SY+2,SX-4,YU!XR!NP
3415          b SY+7,SX-4,YU!XR!EL
3416          b SY+4,SX-2,YU!XR!NP
3417          b SY+4,SX-6,YU!XR!EL
3418
3419 ; Size-Gadget:
3420 ; (Relativ von rechts/unten)
3421          b 2, 6,YU!XR!NP          ;Rahmen.
3422          b 2, 2,YU!XR!EL
3423          b 6, 2,YU!XR!EL
3424          b 6, 6,YU!XR!EL
3425          b 2, 6,YU!XR!EL
3426
3427          b 6, 3,YU!XR!NP          ;Innere Linie.
3428          b 3, 3,YU!XR!EL
3429          b 3, 6,YU!XR!EL
3430
3431 ; Gadget-Rahmen links/unten:
3432          b 0,10,YU!XL!NP
3433          b SY,10,YU!XL!EL
3434
3435 ; Select-Gadget:
3436 ; (Relativ von links/unten)
3437          b 4, 3,YU!XL!NP          ;Select.
3438          b 4, 7,YU!XL!EL
3439          b 6, 5,YU!XL!NP
3440          b 2, 5,YU!XL!EL
3441
3442 ; Gadget-Rahmen links/unten:
3443          b 0,18,YU!XL!NP
3444          b SY,18,YU!XL!EL
3445
3446 ; Partition-Gadget:
3447 ; (Relativ von links/unten)
3448          b 2,12,YU!XL!NP          ;Partition.
3449          b 2,16,YU!XL!EL
3450          b 3,12,YU!XL!NP
3451          b 3,16,YU!XL!EL
3452          b 4,12,YU!XL!NP
3453          b 4,16,YU!XL!EL
3454          b 5,12,YU!XL!NP
3455          b 5,16,YU!XL!EL
3456          b 6,12,YU!XL!NP
3457          b 6,16,YU!XL!EL
3458
3459 ; Ende:
3460          b NULL
3461          w $ffff
3462
3463 ; Textgrenzen auf Panel begrenzen.
3464 :Z13D0      ldy #6 -1
3465 ::1        lda :coord,y
3466           sta windowTop,y
3467           dey
3468           bpl :1

```

	rts	3469
		3470
::coord	b PANEL_Y0 +1,PANEL_Y1 -1	3471
	w PANEL_X0 +1,PANEL_X1 -1	3472
		3473
	; Textausgabe auf Desktop begrenzen.	3474
	; TD13: MaxTextWin	3475
:Z13D3	ldy #6 -1	3476
::1	lda :coord,y	3477
	sta windowTop,y	3478
	dey	3479
	bpl :1	3480
	rts	3481
		3482
::coord	b SCREEN_Y0,SCREEN_Y1	3483
	w DESK_X0 ,DESK_X1	3484
		3485
	; Größe Textfenster setzen.	3486
	; Rückgabe: C = 0: Textgrenzen gesetzt.	3487
	; C = 1: Keine Textgrenzen.	3488
	; TD13: SetTextWin	3489
.S13E5	jsr S11D0 ;Textgrenzen r5-r7.	3490
	jsr S1D1C ;Schnittfläche?	3491
	bcs :exit ; => Nicht vorhanden.	3492
		3493
	ldy #(r4H - r2L)	3494
::1	lda windowTop,y	3495
	sta V0455,y	3496
	lda r2,y	3497
	sta windowTop,y	3498
	dey	3499
	bpl :1	3500
		3501
	clc	3502
::exit	rts	3503
		3504
	; Textgrenzen zurücksetzen.	3505
	; TD13: RestoreTextWin	3506
:Z1400	ldy #6 -1	3507
::1	lda V0455,y	3508
	sta windowTop,y	3509
	dey	3510
	bpl :1	3511
	rts	3512
		3513
	; Ausgefülltes Rechteck zeichnen.	3514
.S140C	jsr SetPattern	3515
	jmp Rectangle	3516
		3517
	; Fenster zeichnen.	3518
	; Übergabe: X = Fenster-Nr.	3519
	; TD13: DrawWindowB	3520
.S141D	cpx C042A ;Oberstes Fenster.	3521
	bne Z1428 ; => and.Fenster.	3522
		3523

```

3524 ; Hinweis:
3525 ; Durch setzen der Fenstergrenzen auf
3526 ; den DeskTop wird das oberste Fenster
3527 ; immer komplett neu gezeichnet!
3528 ; Wird das zu aktivierende Fenster
3529 ; aber durch mehrere Fenster verdeckt,
3530 ; dann muss es neu gezeichnet werden:
3531 ; Es kann nur eine Schnittfläche neu
3532 ; gezeichnet werden, nicht mehrere!
3533             jsr    Z13D3                ;Textgrenzen Desktop.
3534
3535 :Z141E             lda    #0                ;Titel hervorheben.
3536             bcc    $2c
3537
3538 ; Fenster zeichnen.
3539 ; Übergabe: X = Fenster-Nr.
3540 ; TD13: DrawWindowC
3541 ; Aufruf nur über Z1CB6(BackWindow)
3542 :Z1428             lda    #1                ;Nicht hervorheben.
3543             sta    V14C7                ;Titel-Modus setzen.
3544
3545             stx    V14C6                ;Fenster-Nr.
3546
3547             jsr    S61E9                ;a9=Fensterfarben.
3548
3549 ;             ldx    V14C6                ;Fenster-Nr.
3550             jsr    S11A9                ;a0=Fensterdaten.
3551             jsr    Z11BA                ;Fens.daten > a4-a6.
3552
3553 ; Fenster neu zeichnen wenn verdeckt.
3554             jsr    Z11DC                ;Fenster zeichnen.
3555             bcc    :draw                ; => Nicht verdeckt.
3556
3557 ; Fenster war verdeckt, neu gezeichnet.
3558 ;             sec                        ;Fens. war verdeckt.
3559             rts
3560
3561 ; Nicht verdeckt.
3562 ::draw             lda    a4L
3563             sta    r2L                ;Oberer Rand.
3564             clc
3565             adc    # T0U
3566             sta    r2H                ;Unterer Rand.
3567
3568             lda    a5L
3569             clc
3570             adc    #< T0L
3571             sta    r3L
3572             sta    r11L
3573             lda    a5H
3574             adc    #> T0L
3575             sta    r3H                ;Linker Rand.
3576             sta    r11H                ;X-Pos. Textausgabe.
3577
3578             lda    a6L

```

	sec		3579
	sbc	#< TOR	3580
	sta	r4L	3581
	lda	a6H	3582
	sbc	#> TOR	3583
	sta	r4H ;Rechter Rand.	3584
			3585
;	lda	r3L	3586
;	sta	r11L	3587
;	lda	r3H	3588
;	sta	r11H	3589
			3590
	lda	r2L	3591
	clc		3592
	adc	# T0U -1 ;Baseline für	3593
	sta	r1H ;Titeltext.	3594
			3595
	jsr	S13E5 ;Backup/Set TextWin.	3596
	bcs	:1 ; => Titel verdeckt.	3597
			3598
	ldx	V14C6 ;Fenster-Nr.	3599
	inx		3600
	stx	r0L	3601
	lda	#0 ;Highbyte löschen.	3602
	sta	r0H	3603
	lda	#SET_LEFTJUST!SET_SUPPRESS	3604
	jsr	PutDecimal ;Fens.-Nr. ausgeben.	3605
			3606
	lda	#"- "	3607
	jsr	SmallPutChar ;Trennstrich.	3608
			3609
	ldy	#7	3610
	lda	(a0),y	3611
	sta	r0L	3612
	iny		3613
	lda	(a0),y	3614
	sta	r0H	3615
			3616
	ldx	V14C6 ;Fenster-Nr.	3617
	jsr	S14C8 ;String ausgeben.	3618
	jsr	Z1400 ;Restore TextWin.	3619
			3620
::1	ldy	#6 -1	3621
::2	lda	windowTop,y	3622
	pha		3623
	dey		3624
	bpl	:2	3625
			3626
	ldx	V14C6 ;Fenster-Nr.	3627
	jsr	S1E09 ;Arbeitsber.Fenster.	3628
	bcs	:3 ; => Verdeckt...	3629
			3630
	jsr	Z1400 ;Restore TextWin.	3631
			3632
::3	lda	#WN_REDRAW	3633

```

3634          ldx  V14C6          ;Fenster-Nr.
3635          jsr  Z1837          ;Fensterfunktion.
3636
3637          jsr  Z1400          ;Restore TextWin.
3638
3639          ldy  #0
3640 :::4          pla
3641          sta  windowTop,y
3642          iny
3643          cpy  #6
3644          bne  :4
3645
3646          lda  V14C6          ;Fenster gleich
3647          cmp  C042A          ;Oberstes Fenster?
3648          bne  :done          ; => Nein, weiter...
3649
3650          lda  V14C7          ;Titel hervorheben?
3651          bne  :done          ; => Nein, weiter...
3652
3653          jsr  Z1D7A          ;Titel hervorheben.
3654
3655 :::done        clc          ;Fens. n. verdeckt.
3656          rts
3657
3658 :V14C6        b $00  ;Fenster-Nr.
3659 :V14C7        b $00  ;1=Titel nicht hervorheben.
3660
3661 ; Neue PutString-Routine.
3662 ; Zeichen werden geprüft, der Code für
3663 ; TDSPLIT als Trenner für Diskname und
3664 ; Ordner wird übersprungen.
3665 ; Übergabe: r0 = Zeiger auf Text.
3666 ; TD13: NewPutString
3667 .S14C8        ldy  #$00
3668          lda  (r0),y          ;Zeichen einlesen.
3669          beq  :done          ; => Ende.
3670
3671          cmp  #TDSPLIT        ;Trenner für Ordner?
3672          beq  :3              ; => Ja, weiter...
3673
3674          and  #%01111111
3675          cmp  #$7f
3676          beq  :1
3677          cmp  #" "
3678          bcs  :2
3679
3680 :::1          lda  #"*"          ;Zeichen ungültig.
3681 :::2          jsr  SmallPutChar
3682
3683 :::3          inc  r0L
3684          bne  :4
3685          inc  r0H
3686
3687 :::4          jmp  S14C8          ;String ausgeben.
3688

```

```

::done                rts                                3689
                                                            3690
; Rechteck zeichnen wenn nicht verdeckt.                3691
; TD13: NewRectangle                                    3692
.S14EB                jsr    S11D0                        ;Textgrenzen r5-r7.    3693
                    jsr    S1D1C                        ;Schnittfläche?    3694
                    bcs    :1                            ; => Nicht vorhanden. 3695
                    jsr    Rectangle                    3696
                    clc                                    3697
::1                    rts                                3698
                                                            3699
; Bildschirm löschen, mit Ausnahme von Zeile 0-14.      3700
; TD13: ClearScreen                                    3701
.S14F8                jsr    Z13D3                        ;Textgrenzen Desktop. 3702
                                                            3703
                    ldx    #(r4H - r2L)                  3704
::1                    lda    V2179,x                    3705
                    sta    r2,x                          3706
                    dex                                    3707
                    bpl    :1                            3708
                                                            3709
                    jmp    Z20C3                        ;Hintergr. zeichnen. 3710
                                                            3711
; Panel und DoIcons zeichnen.                          3712
.Z1514                jsr    S151A                        ;Panel zeichnen.    3713
                    jmp    Z20DF                        ;Icons anzeigen.    3714
                                                            3715
; Panel zeichnen.                                       3716
.S151A                ldx    #(r4H - r2L)                  3717
::1                    lda    V1560,x                    3718
                    sta    r2,x                          3719
                    dex                                    3720
                    bpl    :1                            3721
                                                            3722
                    lda    COL043A                        3723
                    jsr    S1561                        ;Screencolor setzen. 3724
                    lda    #PANEL_PATTERN                ;Füllmuster.        3725
                    jsr    S140C                        ;Rechteck zeichnen. 3726
                    lda    #%11111111                  3727
                    jsr    FrameRectangle                3728
                                                            3729
                    jsr    setTrashCoord                ;Farbe Trash-Icon. 3730
                                                            3731
                    lda    COL0440                        3732
                    jsr    S1561                        ;Screencolor setzen. 3733
                                                            3734
                    lda    #PRINT_Y0                    ;Farbe Drucker-Icon. 3735
                    sta    r2L                          3736
                    lda    #PRINT_Y1                    3737
                    sta    r2H                          3738
                                                            3739
                    lda    COL0441                        3740
                                                            3741
; Bildschirmfarben setzen.                              3742
.S1561                ldx    C0438                        ;TopDesk in Farbe? 3743

```

```

3744          bne    :1          ; => Ja, weiter...
3745          lda    screencolors ;GEOS-Farben.
3746 ::1        jmp    DirectColor
3747
3748 :V1560      b     PANEL_Y0,PANEL_Y1
3749          w     PANEL_X0,PANEL_X1
3750
3751 ; Alle Fenster neu zeichnen.
3752 ; TD13: RedrawAll
3753 .S157E      jsr    S14F8          ;Bildschirm löschen.
3754
3755          ldy    #4 -1
3756 ::1        tya
3757          pha
3758
3759          lda    C042A,y          ;Fenster-Nr. holen.
3760          tay
3761          tax
3762          lda    C0448,y          ;Fenster geöffnet?
3763          beq    :2          ; => Nein, weiter...
3764
3765          inc    V18AD          ;Fens.-Aufbau-Status.
3766          jsr    S141D          ;Fenster anzeigen.
3767
3768 ::2        pla
3769          tay
3770          dey
3771          bpl    :1
3772          rts
3773
3774 ; TD13: Move1stWin
3775 :Z159B      ldy    #(r4H - r2L)
3776 ::1        lda    r2,y
3777          pha
3778          sta    r5,y
3779          dey
3780          bpl    :1
3781
3782          lda    V04AE          ;Klick auf Gadget?
3783          bmi    :2          ; => Ja, weiter...
3784
3785          jsr    Z1D7D          ;Titel deaktivieren.
3786          jsr    Z11BD          ;Fens.daten > r2-r4.
3787          jsr    Z15DD          ;Hintergrund herst.
3788
3789 ::2        ldy    #0          ;Vorgabe: Kein
3790          sty    V04AE          ;Klick auf Gadget.
3791 ::3        pla
3792          sta    (a0),y          ;Koord. in Fenster-
3793          iny          ;Daten kopieren.
3794          cpy    #(r4H - r2L) +1
3795          bne    :3
3796
3797          inc    V18AD          ;Fens.-Aufbau-Status.
3798

```



	ldx	C042A	;Oberstes Fenster.	3799
	jmp	S141D	;Fenster anzeigen.	3800
				3801
			; Fenster neu zeichnen.	3802
			; TD13: Redraw	3803
:Z15CB	ldx	C042A	;Oberstes Fenster.	3804
	lda	C0448,x	;Fenster geöffnet?	3805
	bne	:1	; => Ja, weiter...	3806
	rts			3807
				3808
::1	inc	V18AD	;Fens.-Aufbau-Status.	3809
	jmp	S141D	;Fenster anzeigen.	3810
				3811
			; Bildschirmbereich wieder herstellen.	3812
			; Übergabe: r2-r4 = Bereich	3813
			; TD13: RestoreRectangle	3814
:Z15DD	lda	a0H		3815
	pha			3816
	lda	a0L		3817
	pha			3818
				3819
	jsr	Z13D3	;Textgrenzen Desktop.	3820
	jsr	S13E5	;Backup/Set TextWin.	3821
	jsr	Z20C3	;Hintergr. zeichnen.	3822
				3823
	ldy	#4 -1	;Max. 4 Fenster.	3824
::loop	tya			3825
	pha			3826
				3827
	lda	C042A,y	;Fenster-Nr. holen.	3828
	tax			3829
	lda	C0448,x	;Fenster geöffnet?	3830
	beq	:next	; => Nein, weiter...	3831
				3832
	jsr	S141D	;Fenster anzeigen.	3833
				3834
::next	pla			3835
	tay			3836
	dey			3837
	bne	:loop		3838
				3839
	jsr	Z13D3	;Textgrenzen Desktop.	3840
				3841
	pla			3842
	sta	a0L		3843
	pla			3844
	sta	a0H		3845
	rts			3846
				3847
			; Fenster verschieben.	3848
			; Übergabe: r2-r4 = Bereich	3849
			; r0 = Routine	3850
			; TD13: DoMove (Initialisierung)	3851
:Z160B	lda	r0L	;Zeiger auf	3852
	ldx	r0H	;Verschieberoutine.	3853

```

3854          jsr    Z16A0          ;Neue Routine setzen.
3855
3856          jsr    Z2515          ;Fens.bereich zeigen.
3857
3858          ldy    #(r4H - r2L)
3859 ::1        lda    r2,y
3860          sta    V16AC,y
3861          dey
3862          bpl    :1
3863
3864          php
3865          sei
3866
3867          lda    mouseXPos +0
3868          sec
3869          sbc    r3L
3870          sta    V16B3 +0
3871          lda    mouseXPos +1
3872          sbc    r3H
3873          sta    V16B3 +1
3874
3875          lda    mouseYPos
3876          sec
3877          sbc    r2L
3878          sta    V16B2
3879
3880          plp
3881
3882 ; Linker Rand = MausXoffset
3883          lda    V16B3 +0
3884          clc
3885          adc    #< DESK_X0
3886          sta    mouseLeft +0
3887          lda    V16B3 +1
3888          adc    #> DESK_X0
3889          sta    mouseLeft +1
3890
3891 ; Rechter Rand = 319 - (xr-xl) + MausXoffset
3892 ;           = 319 - xr + xl + MausXoffset
3893          lda    #< DESK_X1
3894          sec
3895          sbc    V16B0 +0
3896          sta    mouseRight +0
3897          lda    #> DESK_X1
3898          sbc    V16B0 +1
3899          sta    mouseRight +1
3900
3901          lda    V16AE +0
3902          clc
3903          adc    mouseRight +0
3904          sta    mouseRight +0
3905          lda    V16AE +1
3906          adc    mouseRight +1
3907          sta    mouseRight +1
3908

```

```

        lda    V16B3 +0                      3909
        clc                                     3910
        adc    mouseRight                     3911
        sta    mouseRight                     3912
        lda    V16B3 +1                      3913
        adc    mouseRight +1                 3914
        sta    mouseRight +1                 3915
                                           3916
; Oberer Rand = MausYOffset                  3917
        lda    V16B2                         3918
        clc                                     3919
        adc    #DESK_Y0                      3920
        sta    mouseTop                      3921
                                           3922
; Unterer Rand = 199 - (yu-yo) + MausYOffset 3923
        lda    #DESK_Y1                      3924
        sec                                     3925
        sbc    V16AD                         3926
        clc                                     3927
        adc    V16AC                         3928
        clc                                     3929
        adc    V16B2                         3930
        sta    mouseBottom                   3931
                                           3932
        lda    #$ff                          ;$FF = Verschieben. 3933
        sta    V16B7                          ;Gummiband-Modus. 3934
        rts                                  3935
                                           3936
; Zeiger auf Verschiebe-Routine setzen.      3937
:Z16A0      sta    V16B5 +0                   3938
           stx    V16B5 +1                   3939
           rts                                3940
                                           3941
:V16AC      b $00      ;oben                 3942
:V16AD      b $00      ;unten                3943
:V16AE      w $0000    ;links                3944
:V16B0      w $0000    ;rechts               3945
                                           3946
:V16B2      b $00      ;Maus Y               3947
:V16B3      w $0000    ;Maus X               3948
                                           3949
:V16B5      w $0000    ;Verschieberoutine.   3950
:V16B7      b $00      ;$80=Größe, $FF=Verschieben 3951
:V16B8      b $00      ;$00=Nur ganze Cards, $FF=Pixel 3952
                                           3953
; Bewegung eines Rahmens über Mauszeiger-Koordinaten. 3954
; TD13: AppMoveFrame (Mainloop-Routine)      3955
:Z16B9      lda    V16B7                      ;Gummiband-Modus. 3956
           cmp    #$ff                      ;Verschieben?      3957
           beq    :1                          ; => Ja, weiter... 3958
                                           3959
           jmp    Z17D2                      ; => Größe ändern. 3960
                                           3961
::1         php                                     3962
           sei                                3963

```

```

3964
3965         lda    mouseXPos +1
3966         cmp    V1763 +1
3967         bne    :2
3968         lda    mouseXPos +0
3969         cmp    V1763 +0
3970 ::2      bne    :3
3971
3972         lda    mouseYPos
3973         cmp    V1765
3974         bne    :3
3975
3976 ; Ende, Mausposition nicht geändert.
3977         plp
3978         rts
3979
3980 ::3      ldy    #(r4H - r2L)
3981 ::4      lda    V16AC,y
3982         sta    r2,y
3983         dey
3984         bpl    :4
3985
3986         jsr    Z2515                ;Fens.bereich zeigen.
3987
3988         lda    r2H
3989         sec
3990         sbc    r2L
3991         sta    r2H
3992
3993         lda    r4L
3994         sec
3995         sbc    r3L
3996         sta    r4L
3997         lda    r4H
3998         sbc    r3H
3999         sta    r4H
4000
4001         lda    mouseXPos +0
4002         sta    r3L
4003         lda    mouseXPos +1
4004         sta    r3H
4005
4006         lda    r3L
4007         sec
4008         sbc    V16B3 +0
4009         sta    r3L
4010         lda    r3H
4011         sbc    V16B3 +1
4012         sta    r3H
4013
4014         lda    V16B8                ;Nur ganze Cards?
4015         bne    :5                ; => Nein, weiter...
4016
4017         lda    r3L
4018         and    #%11111000        ;Abrunden...

```

```

                                sta    r3L                                4019
                                4020
::5                             lda    r3L                                4021
                                clc                                       4022
                                adc    r4L                                4023
                                sta    r4L                                4024
                                lda    r3H                                4025
                                adc    r4H                                4026
                                sta    r4H                                4027
                                4028
                                lda    mouseYPos                          4029
                                sec                                       4030
                                sbc    V16B2                             4031
                                4032
                                ldy    V16B8                                4033
                                bne    :6                                ;Nur ganze Cards?
                                ; => Nein, weiter...                      4034
                                4035
                                and    #%11111000                        ;Abrunden...
                                4036
::6                             sta    r2L                                4037
                                clc                                       4038
                                adc    r2H                                4039
                                sta    r2H                                4040
                                4041
                                ldy    #(r4H - r2L)                      4042
::7                             lda    r2,y                              4043
                                sta    V16AC,y                          4044
                                dey                                       4045
                                bpl    :7                                4046
                                4047
                                jsr    Z2515                                ;Fens.bereich zeigen.
                                4048
                                4049
                                lda    mouseXPos +0                      ;Mausposition
                                sta    V1763 +0                          ;zwischenspeichern.
                                lda    mouseXPos +1                      4052
                                sta    V1763 +1                          4053
                                4054
                                lda    mouseYPos                          4055
                                sta    V1765                             4056
                                4057
                                plp                                       4058
                                rts                                       4059
                                4060
:V1763                          w $0000    ;Letzte X-Position            4061
:V1765                          b $00     ;Letzte Y-Position            4062
                                4063
; Rahmengröße ändern über Mauszeiger-Koordinaten.                    4064
; TD13: DoRubber (Initialisierung)                                    4065
:Z1766                          lda    r0L                                ;Zeiger auf
                                ldx    r0H                                ;Verschieberoutine.
                                jsr    Z16A0                             ;Neue Routine setzen.
                                4069
                                php                                       4070
                                sei                                       4071
                                4072
                                ldy    #(r4H - r2L)                      4073

```

```

4074 ::1          lda    r2,y
4075             sta    V16AC,y
4076             dey
4077             bpl     :1
4078
4079             lda    r4L
4080             sta    mouseXPos +0
4081             lda    r4H
4082             sta    mouseXPos +1
4083             lda    r2H
4084             sta    mouseYPos
4085
4086             lda    r2L
4087             clc
4088             adc    #MIN_HEIGHT          ;Mindesthöhe.
4089             sta    mouseTop
4090             sta    r2H
4091
4092             lda    #< DESK_X1
4093             sta    mouseRight +0
4094             lda    #> DESK_X1
4095             sta    mouseRight +1
4096
4097             lda    #DESK_Y1
4098             sta    mouseBottom
4099
4100             lda    V16AE +0
4101             clc
4102             adc    #< MIN_WIDTH          ;Mindestbreite.
4103             sta    mouseLeft +0
4104             lda    V16AE +1
4105             adc    #> MIN_WIDTH
4106             sta    mouseLeft +1
4107
4108             lda    mouseYPos
4109             sta    r2H
4110             sta    V16AD
4111
4112             lda    mouseXPos +0
4113             sta    V16B0 +0
4114             sta    r4L
4115             lda    mouseXPos +1
4116             sta    V16B0 +1
4117             sta    r4H
4118
4119             plp
4120
4121             jsr    Z2515          ;Fens.bereich zeigen.
4122
4123             lda    #%10000000          ;Modus Größe ändern.
4124             sta    V16B7          ;Gummiband-Modus.
4125
4126             rts
4127
4128 ; Rahmengröße ändern über Mauszeiger-Koordinaten.

```

```

; TD13: AppRubberFrame (Mainloop-Routine) 4129
:Z17D2      lda    V16B7                    ;Gummiband-Modus. 4130
            cmp    #%100000000             ;Größe ändern? 4131
            bne    :6                      ; => Nein, Ende... 4132
            4133
            php 4134
            sei 4135
            4136
            lda  mouseXPos +1 4137
            cmp  V182E +1 4138
            bne  :1 4139
            lda  mouseXPos +0 4140
            cmp  V182E +0 4141
::1         bne  :2 4142
            4143
            lda  mouseYPos 4144
            cmp  V1830 4145
            beq  :5 4146
            4147
::2         ldy  #(r4H - r2L) 4148
::3         lda  V16AC,y 4149
            sta  r2,y 4150
            dey 4151
            bpl  :3 4152
            4153
            jsr  Z2515                    ;Fens.bereich zeigen. 4154
            4155
            lda  mouseXPos +0 4156
            ora  #%00000111 4157
            sta  r4L 4158
            lda  mouseXPos +1 4159
            sta  r4H 4160
            4161
            lda  mouseYPos 4162
            ora  #%00000111 4163
            sta  r2H 4164
            4165
            ldy  #(r4H - r2L) 4166
::4         lda  r2,y 4167
            sta  V16AC,y 4168
            dey 4169
            bpl  :4 4170
            4171
            jsr  Z2515                    ;Fens.bereich zeigen. 4172
            4173
            lda  mouseXPos +0 4174
            sta  V182E +0 4175
            lda  mouseXPos +1 4176
            sta  V182E +1 4177
            4178
            lda  mouseYPos 4179
            sta  V1830 4180
            4181
::5         plp 4182
::6         rts 4183

```

```

4184
4185 :V182E          w $0000 ;Letzte Maus-X-Position.
4186 :V1830          b $00   ;Letzte Maus-Y-Position.
4187
4188 ; Fensterfunktion ausführen.
4189 ; TD13: SendMessage2
4190 :Z1831          jsr     Z1837          ;Fensterfunktion.
4191                jmp     Z13D3          ;Textgrenzen Desktop.
4192
4193 ; Fensterfunktion ausführen.
4194 ; Übergabe: A = Fenster-Befehl
4195 ;           X = Fenster-Nr.
4196 ; TD13: SendMessage
4197 :Z1837          sta     WIN042E        ;Fenster-Befehl.
4198                stx     WIN042F        ;Aktives Fenster.
4199
4200                ldy     #10
4201                lda     (a0),y
4202                tax
4203                dey
4204                lda     (a0),y
4205                jmp     CallRoutine
4206
4207 ; Das nächste freie Fenster suchen.
4208 ; Rückgabe: C=0 Fenster frei
4209 ;           X = Fenster-Nr.
4210 ;           C=1 Kein Fenster frei.
4211 ; TD13: GetNext
4212 :Z1848          ldx     #0
4213 ::1            lda     C0448,x        ;Fenster geöffnet?
4214                beq     :4            ; => Nein, Ende...
4215                inx
4216
4217                ldy     C0429          ;System in REU?
4218                beq     :2            ; => Nein, weiter...
4219
4220                cpx     #4            ;Max. 4 Fenster?
4221                bne     :1            ; => Nein, suchen...
4222                beq     :3
4223
4224 ::2            cpx     #2            ;Max. 2 Fenster?
4225                bne     :1            ; => Nein, suchen...
4226
4227 ::3            sec                  ;Kein Fenster frei.
4228                rts
4229
4230 ::4            clc                  ;Fenster frei.
4231                rts
4232
4233 ; Fenster X nach vorne holen.
4234 :KB_W1          ldx     #0            ;Fenster #1.
4235                b $2c
4236 :KB_W2          ldx     #1            ;Fenster #2.
4237                b $2c
4238 :KB_W3          ldx     #2            ;Fenster #3.

```



```

                                b $2c                                4239
:KB_W4                        ldx  #3                            ;Fenster #4.    4240
                                4241
                                lda  C0428                      ;Ram-TopDesk?  4242
                                bne  :1                          ; => Ja, weiter... 4243
                                cpx  #3                          ;Fens.3/4?    4244
                                bcc  :1                          ; => Nein, weiter... 4245
::exit                        rts                                4246
                                4247
::1                            lda  C0448,x                    ;Bereits geöffnet? 4248
                                beq  :exit                      ; => Nein, Ende... 4249
                                4250
                                cpx  C042A                      ;Oberstes Fenster? 4251
                                beq  :exit                      ; => Ja, Ende... 4252
                                4253
                                clc                               ;Nicht neu laden. 4254
                                jmp  Z1C10                     ;Fenster nach vorne. 4255
                                4256
; Fenster öffnen.                                4257
; Übergabe: X = Fenster-Nr.                        4258
; Rückgabe: C=1 Fenster bereits geöffnet.          4259
;           C=0 Fenster wurde geöffnet.            4260
;           -> Wird nicht abgefragt!                4261
; TD13: OpenWindow                                4262
:Z1863                        lda  C0448,x                    ;Bereits geöffnet? 4263
                                beq  :1                          ; => Nein, weiter... 4264
                                sec                               4265
                                rts                               4266
                                4267
::1                            txa                               4268
                                pha                               4269
                                4270
; Aufgaben:                                4271
; - Aktuellen Titel invertieren.                    4272
; - Laufwerksfarben löschen.                        4273
                                jsr  Z1D7D                     ;Titel deaktivieren. 4274
                                4275
                                pla                               4276
                                tax                               4277
                                lda  #$ff                      ;Fenster "geöffnet". 4278
                                sta  C0448,x                    ;Fensterstatus 1-4. 4279
                                stx  r0L                        4280
                                4281
                                cpx  C042A                      ;Oberstes Fenster? 4282
                                beq  :exit                      ; => Ja, Ende... 4283
                                4284
                                ldy  #0                          ;Fenster nach oben 4285
                                txa                              ;sortieren.    4286
::loop                        ldx  C042A,y                      ;Fenster-Nr. holen. 4287
                                sta  C042A,y                    4288
                                tya                              4289
                                beq  :next                      4290
                                cpx  r0L                        4291
                                beq  :exit                      4292
::next                        txa                               4293

```

```

4294             iny
4295             cpy    #4
4296             bne    :loop
4297
4298 ::exit         ldx    r0L                ;Neue Fenster-Nr.
4299             clc
4300             rts
4301
4302 .V18AD         b $00    ;>$00 = Fenster im Aufbau.
4303
4304 ; Fenster schließen.
4305 ; TD13: CloseWindow
4306 :Z18AE         lda    V04AE                ;Klick auf Gadget?
4307             bmi    :1                ; => Ja, weiter...
4308
4309 if ENABLE_ANIMATE = TRUE
4310             lda    WIN042F                ;Aktives Fenster.
4311             jsr    AnimateClose
4312 endif
4313
4314 ; Bei einem nicht aktiven Fenster
4315 ; wird hier das Fenster nach CLOSE
4316 ; nicht nach vorne geholt.
4317             jsr    Z1D7D                ;Titel deaktivieren.
4318
4319 ::1            ldx    C042A                ;Oberstes Fenster.
4320
4321             lda    #$00                ;Vorgabe: Fenster
4322             sta    V04AE                ;neu anzeigen.
4323
4324             sta    C050B,x                ;Fensterposition.
4325             sta    C0448,x                ;Fenster inaktiv.
4326
4327             jsr    Z11A3                ;Fens.koord. r2-r4.
4328             jsr    Z15DD                ;Hintergrund herst.
4329
4330             ldy    #0
4331 ::2            ldx    C042A,y                ;Fenster-Nr. holen.
4332             lda    C0448,x                ;Fenster geöffnet?
4333             bne    :3                ; => Ja, weiter...
4334             iny
4335             cpy    #3                ;Fens. durchsucht?
4336             bpl    :2                ; => Nein, weiter...
4337             ldx    #$ff                ;Kein Fenster mehr.
4338 ::3            txa                ;Fenster vorhanden?
4339             bmi    x18EB                ; => Nein, Ende...
4340
4341             clc
4342             jsr    Z1C10                ;Fenster nach vorne.
4343
4344 ; Fenster aktivieren.
4345 ; TD13: SendActivate
4346 :Z18E3         lda    #WN_ACTIVATE
4347             ldx    C042A                ;Oberstes Fenster.
4348             jmp    Z1831                ;Fensterfunktion.

```

```

4349
: x18EB          rts                                4350
4351
; appMain-Routine.                                4352
; TD13: MyAppMain                                4353
.S1160          lda    #< S18EC                    4354
                sta    mouseVector +0              4355
                lda    #> S18EC                    4356
                sta    mouseVector +1              4357
4358
                lda    V04AE                        ;Klick auf Gadget? 4359
                beq    :1                          ; => Nein, weiter... 4360
4361
                dec    V04AE                        ;Resize/Move:      4362
                jsr    Z18F1                        ;Modus beenden.    4363
4364
::1             jsr    Z13D3                        ;Textgrenzen Desktop. 4365
                jsr    Z16B9                        ;Fenster verschieben. 4366
                jsr    Z1A43                        ;DnD auf TrashBox?      4367
4368
                lda    V046F                        ;TaskMan starten? 4369
                beq    :3                          ; => Nein, weiter... 4370
                lda    Flag_TaskAktiv              ;TaskMan aktiv?      4371
                bmi    :2                          ; => Nein, weiter... 4372
4373
                jsr    TaskMan_NewJob               4374
4375
; Hinweis:                                         4376
; Warten bis keine Maustaste gedrückt,           4377
; da sonst nach dem verlassen des Menü           4378
; der TaskManager sofort wieder neu              4379
; gestartet wird.                                4380
::2             jsr    UpdateMouse                 ;Maus aktualisieren. 4381
4382
                bit    mouseData                   ;Taste gedrückt? 4383
                bpl    :2                          ; => Ja, warten.    4384
4385
; Hinweis:                                         4386
; Nur den Maustasten-Status löschen!             4387
                lda    pressFlag                   4388
                and    #%11011111                 ;Bit%5=MTaste neu.    4389
                sta    pressFlag                   4390
4391
                ldx    #$00                        ;Löschen des      4392
                stx    V046F                      ;TaskMan-Flag.    4393
4394
; Hinweis:                                         4395
; ":mouseData" = $80 = Keine Maustaste.          4396
;          dex                                     ;Falsch...      4397
;          ldx    #%100000000                     ;Richtig...        4398
;          stx    mouseData                       ;Ist bereits $80.    4399
4400
::3             lda    V044C +0                   4401
                ldx    V044C +1                   4402
                jmp    CallRoutine                 ;":appMain"-Routine. 4403

```

```

4404
4405 ; Mausabfrage.
4406 ; TD13: NewMouseService
4407 .S18EC          lda    #TRUE          ;Maustaste testen.
4408                b $2c
4409 :Z18F1          lda    #FALSE         ;Mse. nicht testen.
4410
4411                tax
4412                beq     :0              ; => Nein, weiter...
4413
4414                lda     mouseData       ;Maustaste gedrückt?
4415                bmi     :exit           ; => Nein, Ende...
4416
4417 ::0             lda     menuNumber     ;Menü aktiv?
4418                bne     :exit           ; => Ja, Ende...
4419
4420 ; Maustaste aktiv, kein Menü aktiv.
4421                lda     V16B7           ;Gummiband-Modus.
4422                cmp     #$ff            ;Modus Verschieben?
4423                beq     :1              ; => Ja, weiter...
4424                cmp     #%100000000     ;Modus Größe ändern?
4425                bne     :11            ; => Nein, weiter...
4426
4427 ; Gummiband oder verschieben.
4428 ::1             ldy     #(r4H - r2L)    ;Koordinaten für
4429 ::2             lda     V16AC,y         ;Gummiband einlesen.
4430                sta     r2,y
4431                dey
4432                bpl     :2
4433
4434                jsr     Z2515            ;Fens.bereich zeigen.
4435
4436                lda     #$00            ;Modus Beenden.
4437                sta     V16B7           ;Gummiband-Modus.
4438
4439                jsr     Z1962            ;Fenstergrenzen.
4440
4441                lda     V16B5 +0         ;Verschieberoutine.
4442                ldx     V16B5 +1
4443                jmp     CallRoutine
4444
4445 ; GEOS64.Editor/geoHDscsi testen.
4446 ::11            lda     V0423          ;DnD aktiv?
4447                bne     :12            ; => Ja, weiter...
4448
4449                jsr     Z410D           ;Konfig. starten?
4450                txa
4451                beq     :12            ; => Nein, weiter...
4452 ::done          rts                   ; => Nicht gefunden.
4453
4454 ; Fensterbereiche testen.
4455 ::12            jsr     Z1B26          ;Fenster auswerten.
4456
4457                ldx     C042A           ;Oberstes Fenster.
4458                lda     C0448,x         ;Fenster geöffnet?

```

```

                                beq    :exit                ; => Nein, Ende...      4459
                                                                4460
                                lda     V0423                ;DnD aktiv?      4461
                                bne     :13                 ; => Ja, weiter... 4462
                                                                4463
; Gadgets testen.              4464
                                jsr     S11A9                ;a0=Fensterdaten. 4465
                                                                4466
                                lda     #$ff                ;Vorgabe:      4467
                                sta     V1962                ;Gadget ausgeführt. 4468
                                jsr     Z1AA7                ;Erw.Mausabfrage. 4469
                                bit     V1962                ;Gadget ausgeführt? 4470
                                bmi     :done                ; => Ja, Ende... 4471
                                                                4472
                                jsr     Z197B                ;Scrollpfeile.   4473
                                                                4474
; Auf Klick in Arbeitsbereich oder 4475
; Fenstersteuerung testen. Rückkehr nur 4476
; wenn Mausklick ausserhalb.      4477
::13                            jsr     Z1A75                ;Fenster/Steuerung? 4478
                                                                4479
; Mausabfrage an System übergeben. 4480
::exit                          jsr     Z13D3                ;Textgrenzen Desktop. 4481
                                                                4482
                                lda     V045B +0             ;GEOS/mouseVector. 4483
                                ldx     V045B +1             4484
                                jmp     CallRoutine          4485
                                                                4486
; Fenstergrenzen: Gesamter Bildschirmbereich. 4487
:Z1962                          ldx     #6 -1                ;Fenstergrenzen 4488
::1                             lda     V1963,x             ;zurücksetz.      4489
                                sta     mouseTop,x           4490
                                dex                     4491
                                bpl     :1                   4492
                                rts                          4493
                                                                4494
:V1962                          b $00    ;$FF = Mausaktion ausgeführt. 4495
                                                                4496
.V1963                          b SCREEN_Y0,SCREEN_Y1       4497
                                w SCREEN_X0,SCREEN_X1       4498
                                                                4499
; Mausklick auf Quicklink-Bereiche auswerten. 4500
:Z410D                          ldx     #(r4H - r2L)        4501
::1                             lda     :coord,x           4502
                                sta     r2,x                4503
                                dex                     4504
                                bpl     :1                   4505
                                                                4506
                                jsr     IsMseInRegion        ;Klick Quicklink? 4507
                                beq     :exit                ; => Nein, Ende... 4508
                                                                4509
                                jsr     S48B6                ;Langer Mausklick? 4510
                                bcs     :2                   ; => Nein, weiter... 4511
                                                                4512
; Langer Mausklick:              4513

```

```

4514                jmp     Z39B1                ;geoHDscsi.
4515
4516 ; Kurzer Mausklick.
4517 ::2                jmp     Z39AE                ;GEOS64.Editor.
4518
4519 ::exit                ldx     #NO_ERROR
4520                rts
4521
4522 ; Position links neben Lfwk.-Icon.
4523 ;
4524 ; Hinweis:
4525 ; Bei TD5.x wird über die gesamte Höhe
4526 ; des Panels auf einen Klick für die
4527 ; Konfigurationsprogramme GEOS64.Editor
4528 ; und geoHDscsi getestet:
4529 ::coord                b PANEL_Y0,PANEL_Y1
4530 ;                    w PANEL_X0,PANEL_X0 +16 -1
4531 ;
4532 ; Logischer ist es nur die Bereiche
4533 ; neben den Laufwerk-Icons zu testen:
4534 ::coord                b DRIVE_Y0,DRIVE_Y1
4535 ;                    w DRIVE_X0,DRIVE_X0 +16 -1
4536
4537 ; Rechte Navigationsleiste.
4538 ; Inkl. Pfeile und Pos.Anzeige.
4539 :Z1969                jsr     Z2491                ;Koord. Pos.-Anzeige.
4540
4541                sec                                ;Pfeil "Nach oben".
4542                lda     r2L
4543                sbc     #HSC
4544                sta     r2L
4545
4546                lda     r2H                                ;Pfeil "Nach unten".
4547                clc
4548                adc     #HSC
4549                sta     r2H
4550                rts
4551
4552 ; Scrollpfeile angeklickt?
4553 ; Aufruf nur aus ":NewMouseService".
4554 :Z197B                jsr     Z1969                ;Koord.rechte Navbar.
4555                jsr     IsMseInRegion                ;Maus in Bereich?
4556                beq     :exit                ; => Nein, Ende...
4557
4558                pla                                ;Rücksprungadresse
4559                pla                                ;löschen.
4560
4561                jsr     Z4CCF                ;Koord. Füllstand.
4562                jsr     IsMseInRegion                ;Füllst. angeklickt?
4563                bne     :navbar                ; => Ja, weiter...
4564
4565                jsr     Z4CCF                ;Koord. Füllstand.
4566
4567                lda     r2L
4568                pha

```

```

        jsr    Z1969                ;Koord.rechte Navbar.      4569
        pla                                4570
        sta    r2H                    4571
        jsr    IsMseInRegion          ;Pfeil "Nach oben"?      4572
        beq    :1                    ; => Nein, weiter...        4573
                                                4574
::up      lda    #WN_SCROLL_U          4575
        b $2c                        4576
::down    lda    #WN_SCROLL_D          4577
        ldx    C042A                  ;Oberstes Fenster.      4578
        jmp    Z1831                  ;Fensterfunktion.        4579
                                                4580
::1       lda    r2H                    4581
        pha                                4582
        jsr    Z1969                ;Koord.rechte Navbar.      4583
        pla                                4584
        sta    r2L                    4585
        jsr    IsMseInRegion          ;Pfeil "Nach unten"?    4586
        bne    :down                  ; => Ja, weiter...        4587
                                                4588
::exit    rts                        4589
                                                4590
; Klick auf Positionsanzeige.          4591
::navbar  jsr    Z2491                ;Koord. Pos.-Anzeige.    4592
                                                4593
        lda    r2L                    4594
        sta    a4L                    4595
        lda    r2H                    4596
        sta    a4H                    4597
                                                4598
        jsr    Z4CCF                  ;Koord. Füllstand.      4599
                                                4600
        inc    r3L                    4601
        bne    :2                    4602
        inc    r3H                    4603
                                                4604
::2       inc    V16B8                ; => Gummiband/Pixel.    4605
                                                4606
        jsr    Z2515                  ;Fens.bereich zeigen.    4607
                                                4608
        ldy    #0                    4609
::3       lda    r2,y                  4610
        sta    V16AC,y                4611
        iny                                4612
        cpy    #(r4H - r2L) +1        4613
        bne    :3                    4614
                                                4615
        php                                4616
        sei                                4617
                                                4618
        lda    mouseXPos +0           4619
        sec                                4620
        sbc    r3L                    4621
        sta    V16B3 +0               4622
        lda    mouseXPos +1           4623

```

```

4624          sbc    r3H
4625          sta    V16B3 +1
4626
4627          lda    mouseYPos
4628          sec
4629          sbc    r2L
4630          sta    V16B2
4631
4632          lda    mouseYPos
4633          pha
4634          sec
4635          sbc    r2L
4636          clc
4637          adc    a4L
4638          sta    mouseTop
4639          pla
4640          sta    r5L
4641
4642          lda    r2H
4643          sec
4644          sbc    r5L
4645          sta    r5L
4646          sec
4647          lda    a4H
4648          sbc    r5L
4649          sta    mouseBottom
4650
4651          lda    mouseXPos +0
4652          sta    mouseLeft
4653          lda    mouseXPos +1
4654          sta    mouseLeft +1
4655
4656          lda    mouseXPos +0
4657          sta    mouseRight
4658          lda    mouseXPos +1
4659          sta    mouseRight +1
4660
4661          lda    #< Z4D6D          ;Zeiger auf
4662          ldx    #> Z4D6D          ;Verschieberoutine.
4663          jsr    Z16A0             ;Neue Routine setzen.
4664
4665          lda    #$ff              ;Modus Verschieben.
4666          sta    V16B7             ;Fenster verschieben.
4667
4668          plp
4669          rts
4670
4671 ; DnD / Maus auf TrashBox?
4672 :Z1A43          lda    V0423          ;DnD aktiv?
4673          beq    :1                ; => Nein, Ende...
4674
4675          jsr    setTrashCoord        ;Koord. TrashBox.
4676          jsr    IsMseInRegion        ;DnD auf TrashBox?
4677          beq    :1                ; => Nein, Ende...
4678

```



```

; Hinweis: 4679
; Fehlerton ausgeben wenn Mauszeiger 4680
; über TrashBox ist => Warnung. 4681
        lda    V1A74                ;Fehlerton aktiv? 4682
        bne    :2                    ; => Ja, Ende... 4683
                                        4684
        inc    V1A74                ;Fehlerton aktiv. 4685
                                        4686
        lda    #$01                 4687
        jmp    S24C0                ;Fehlerton. 4688
                                        4689
::1      sta    V1A74                ;Fehlerton löschen. 4690
::2      rts                        4691
                                        4692
:V1A74   b $00    ;$01=Fehlerton aktiv. 4693
                                        4694
; Koordinaten für Trashbox setzen. 4695
:setTrashCoord    ldx    #(r4H - r2L) 4696
::1          lda    :coord,x 4697
            sta    r2,x 4698
            dex 4699
            bpl    :1 4700
            rts 4701
            4702
::coord        b TRASH_Y0,TRASH_Y1 4703
            w TRASH_X0,TRASH_X1 4704
            4705
; Mausabfrage: 4706
; Arbeitsbereich oder Steuerung? 4707
; Aufruf nur aus ":NewMouseService". 4708
:Z1A75        ldx    C042A            ;Oberstes Fenster. 4709
            jsr    S1E09            ;Arbeitsber.Fenster. 4710
            jsr    IsMseInRegion    ;Mausklick in Fens.? 4711
            bne    :1                ; => Arbeitsbereich. 4712
            4713
            jsr    Z11BD            ;Fens.daten > r2-r4. 4714
            jsr    IsMseInRegion    ;Mausklick in Fens.? 4715
            bne    :2                ; => Steuerung. 4716
            beq    :3                ; => NewMouseService. 4717
            4718
; Mausabfrage: 4719
; Klick in Arbeitsbereich/Fenster. 4720
::1          lda    mouseXPos +0 4721
            sec 4722
            sbc    r3L 4723
            sta    r3L 4724
            lda    mouseXPos +1 4725
            sbc    r3H 4726
            sta    r3H 4727
            4728
            lda    mouseYPos 4729
            sbc    r2L 4730
            sta    r2L 4731
            4732
            lda    #WN_USER 4733

```

```

4734          ldx  C042A          ;Oberstes Fenster.
4735          jsr  Z1831          ;Fensterfunktion.
4736
4737 ::2          pla
4738          pla
4739 ::3          rts
4740
4741 ; Koordinaten auf Titelzeile.
4742 :Z1ABE      jsr  Z11BD          ;Fens.daten > r2-r4.
4743
4744          lda  r2L
4745          clc
4746          adc  #$08
4747          sta  r2H
4748          bne  x1AC0
4749
4750 ; Koordinaten auf Statuszeile.
4751 :Z1ABF      jsr  Z11BD          ;Fens.daten > r2-r4.
4752
4753          lda  r2H
4754          sec
4755          sbc  #$08
4756          sta  r2L
4757
4758 :x1AC0      lda  r4L
4759          sec
4760          sbc  #$08
4761          rts
4762
4763 ; Erweiterte Mausabfrage.
4764 ; Zusätzliche Bereiche abfragen:
4765 ; - Titelzeile: Disk umbenennen.
4766 ; - Titelzeile: Fenster verschieben.
4767 ; - Statuszeile/links: Dateien wählen.
4768 ; - Statuszeile/rechts: Partition wechseln.
4769 ; - Fenster schließen.
4770 ; - Fenster maximieren.
4771 ; - Fenster verstecken.
4772 ;
4773 :Z1AA7      jsr  Z11BD          ;Fens.daten > r2-r4.
4774
4775 ; Test auf "Diskname ändern".
4776          lda  r2L
4777          clc
4778          adc  #TY -1
4779          sta  r2H
4780
4781          lda  r3L
4782          clc
4783          adc  #14 -1          ;Breite CLOSE-Gadget.
4784          sta  r3L
4785          bcc  :0a
4786          inc  r3H
4787
4788 ::0a      lda  r3L          ;Breite NAME-Gadget.

```

	clc		4789
	adc	#< (8 -1)	4790
	sta	r4L	4791
	lda	r3H	4792
	adc	#> (8 -1)	4793
	sta	r4H	4794
			4795
	jsr	IsMseInRegion ;Diskname ändern?	4796
	beq	:1 ; => Nein, weiter...	4797
			4798
			4799
			4799
			4800
			4801
			4802
			4803
			4804
			4805
			4806
			4807
			4808
			4809
			4810
			4811
			4812
			4813
			4814
			4815
			4816
			4817
			4818
			4819
			4820
			4821
			4822
			4823
			4824
			4825
			4826
			4827
			4828
			4829
			4830
			4831
			4832
			4833
			4834
			4835
			4836
			4837
			4838
			4839
			4840
			4841
			4842
			4843

```

4844 ::3          jsr    Z1ABF          ;Fens.koord. Statusz.
4845
4846          lda    r3L
4847          clc
4848          adc    #SUL -1
4849          jsr    Z1B7E          ;Partition wählen?
4850          bne    x1B31          ; => Ja, weiter...
4851
4852 ; Klick auf "Close"-Gadget?
4853 :Z1ADF          jsr    Z11BD          ;Fens.daten > r2-r4.
4854          jsr    Z1B72          ;Close-Gadget?
4855          bne    x1B3E          ; => Ja, weiter...
4856
4857 ; Klick auf "Maximize"-Gadget?
4858 :Z1AE7          jsr    Z11BD          ;Fens.daten > r2-r4.
4859          jsr    Z1B89          ;Max.-Gadget?
4860          bne    x1B4B          ; => Ja, weiter...
4861
4862 ; Klick auf "Hide"-Gadget?
4863          jsr    Z11BD          ;Fens.daten > r2-r4.
4864          lda    r2L
4865          clc
4866          adc    # T0U+1
4867          sta    r2H
4868          lda    r4L
4869          sec
4870          sbc    # TOR-1
4871          jsr    Z1B95          ;Hide-Gadget?
4872          bne    x1B5E          ; => Ja, weiter...
4873
4874 ; Klick auf "Resize"-Gadget?
4875          jsr    Z1ABF          ;Fens.koord. Statusz.
4876          jsr    Z1B95          ;Größe ändern?
4877          bne    x1B53          ; => Ja, weiter...
4878
4879 ; Klick auf Titelzeile?
4880          jsr    Z11BD          ;Fens.daten > r2-r4.
4881          lda    r2L
4882          clc
4883          adc    #$08
4884          sta    r2H
4885          jsr    IsMseInRegion      ;Fenster verschieben?
4886          bne    x1B1B          ; => Ja, weiter...
4887
4888 ; Ende erw. Mausabfrage.
4889          lda    #$00          ;Kein Gadget
4890          sta    V1962          ;ausgeführt.
4891          rts
4892
4893 ; Mausabfrage:
4894 ; Klick auf Titelzeile/Langer Klick=Fenster verschieben.
4895 :x1B1B          jsr    S48B6          ;Langer Mausklick?
4896          bcs    :1          ; => Nein, weiter...
4897          jsr    Z1B66          ;Zeiger auf Redraw
4898          jsr    Z160B          ;für Fenster versch.

```

```

::1                rts                                4899
                                                         4900
; Mausabfrage:                                       4901
; Klick Statuszeile Partition ändern.              4902
:x1B31             jsr    S48B6                        ;Langer Mausklick? 4903
                                                         bcc    :1      ; => Ja, weiter... 4904
                                                         4905
                                                         jmp    MR3A26   ;Part./Img. wechseln. 4906
                                                         4907
; Vorherige Partition aufrufen.                    4908
;                                                    4909
; Hinweis:                                         4910
; Funktioniert nur bei CMD-Laufwerken,             4911
; nicht bei 64Net oder SD2IEC.                    4912
::1                ldx    C042A                        ;Oberstes Fenster. 4913
                                                         ldy    C04F8,x  ;Lfwk. für Fenster. 4914
                                                         lda    V04B8 -8,y ;CMD-Laufwerk? 4915
                                                         beq    :2      ; => 15xx, weiter... 4916
                                                         bmi    :2      ; => 64Net, weiter... 4917
                                                         4918
                                                         jmp    Z3A8C    ;Vorherige Partition. 4919
                                                         4920
::2                rts                                4921
                                                         4922
; Mausabfrage:                                       4923
; Klick auf Close-Gadget.                          4924
:x1B3E             lda    #$00                        4925
                                                         sta    C050B,x  ;Fensterposition. 4926
                                                         jsr    S524C    ;Dat.-Ausw. aufheben. 4927
                                                         4928
                                                         lda    #WN_CLOSE 4929
                                                         ldx    C042A    ;Oberstes Fenster. 4930
                                                         jmp    Z1831    ;Fensterfunktion. 4931
                                                         4932
; Mausabfrage:                                       4933
; Klick auf Max.-Gadget.                          4934
:x1B4B             jsr    Z2150                        ;Fensgr. Max/Normal. 4935
                                                         jmp    Z159B    4936
                                                         4937
; Mausabfrage:                                       4938
; Klick auf Gummiband-Gadget.                     4939
:x1B53             jsr    Z1B66                        ;Zeiger auf Redraw 4940
                                                         jmp    Z1766    ;für Größe ändern. 4941
                                                         4942
; Mausabfrage:                                       4943
; Klick auf Hide-Gadget.                          4944
:x1B5E             4945
:KB1B60            ldx    C042A                        ;Oberstes Fenster. 4946
                                                         jmp    Z1CB6    4947
                                                         4948
; Zeiger auf Routine für "Gummiband beenden".      4949
:Z1B66             jsr    Z11BD                        ;Fens.daten > r2-r4. 4950
                                                         4951
                                                         lda    #< Z159B ;Zeiger auf Routine 4952
                                                         sta    r0L      ;wenn Gummiband-Modus 4953

```

```

4954          lda    #> Z159B          ;beendet wird.
4955          sta    r0H
4956          rts
4957
4958 ; Auf Mausklick/Gadgets testen.
4959 ; TD13: IsMseOnGad
4960 :BREITEY      = TY
4961 :BREITEX      = TOL -1
4962 :Z1B72        lda    r2L
4963              clc
4964              adc    #BREITEY
4965              sta    r2H
4966
4967              lda    r3L
4968              clc
4969              adc    #BREITEX
4970 :Z1B7E        sta    r4L
4971              lda    r3H
4972              adc    #$00
4973              sta    r4H
4974              jmp    IsMseInRegion
4975
4976 ; Auf Mausklick/Gadgets testen.
4977 ; TD13: CheckMax
4978 :Z1B89        lda    r2L
4979              clc
4980              adc    #BREITEY
4981              sta    r2H
4982
4983              lda    r4L
4984              sec
4985              sbc    #SX
4986
4987 ; Auf Mausklick/Gadgets testen.
4988 ; TD13: Fortsetzung von CheckHide
4989 :Z1B95        sta    r3L
4990              lda    r4H
4991              sbc    #$00
4992              sta    r3H
4993              jmp    IsMseInRegion
4994
4995 ; Auf Mausklick/Gadgets testen.
4996 :Z1B26        lda    V04AE          ;Klick auf Gadget?
4997              bpl    :2            ; => CheckWindows.
4998
4999              cmp    #$ff          ;Größe ändern?
5000              bne    :1            ; => Nein, weiter...
5001              jmp    Z1AE7          ;$FF=Fens.vergrößern.
5002 ::1          jmp    Z1ADF          ;$F0=Fens.schließen.
5003
5004 ; Fenster überprüfen.
5005 ; TD13: CheckWindows
5006 ::2          lda    #0            ;Fenster-Nr. auf
5007              sta    r0L            ;Angang setzen.
5008              tay

```

```

::loop      lda    C042A,y          ;Fenster-Nr. holen.      5009
            tax                                5010
            lda    C0448,x          ;Fenster geöffnet?      5011
            beq    :3                ; => Nein, weiter...      5012
            ;                                5013
            jsr    Z11A3              ;Fens.koord. r2-r4.      5014
            jsr    IsMseInRegion      ;Klick im Fenster?      5015
            bne    :setwin            ; => Ja, weiter...      5016
            ;                                5017
::3          inc    r0L              5018
            ldy    r0L              5019
            cpy    #4                ;Max. 4 Fenster.        5020
            bne    :loop              5021
            ;                                5022
::exit       rts                    5023
            ;                                5024
; Fenster aktivieren?                5025
::setwin     lda    r0L              ;Oberstes Fenster?      5026
            beq    :exit              ; => Ja, Ende...        5027
            ;                                5028
            pla                                5029
            pla                                ;Keine Rückkehr zur 5030
            ;                                ;Mausabfrage.      5031
            ldx    #(r4H - r2L)        ;Fens.koord. retten. 5032
::11         lda    zpage,x           5033
            pha                                5034
            dex                                5035
            bpl    :11                 5036
            ;                                5037
            jsr    Z1B89              ;Maximieren?          5038
            tay                                ;Erg. IsMseInRegion. 5039
            ;                                5040
            ldx    #0                 ;Fens.koord. wieder    5041
::12         pla                                ;zurücksetzen. 5042
            sta    zpage,x           5043
            inx                                5044
            cpx    #(r4H - r2L) +1    5045
            bcc    :12                 5046
            ;                                5047
            tya                                ;Maximieren?    5048
            bne    :done              ; => Ja, weiter...      5049
            ;                                5050
            jsr    Z1B72              ;CLOSE-Gadget?        5051
            beq    :done              ; => Nein, weiter...      5052
            ;                                5053
            lda    #$f0              ;Inaktives Fenster    5054
            ;                                ;schließen...      5055
            sta    V04AE              ;Klick auf Gadget.    5056
            ;                                5057
::done       inc    V04AE              ;Klick auf Gadget.    5058
            ;                                5059
            ldy    r0L              ;Gew. Fenster.          5060
            ldx    C042A,y          ;Fenster-Nr. holen.      5061
            lda    #WN_ACTIVATE2      5062
            jmp    Z1837              ;Fensterfunktion.      5063

```

```

5064
5065 ; Fenster nach vorne holen.
5066 ; Übergabe: X = Fenster-Nr.
5067 ;           C = 1, Fenster immer neu zeichnen.
5068 ; TD13: FrontWindow.
5069 :Z1C10          php
5070
5071                txa
5072                pha
5073                jsr    Z1D7D          ;Titel deaktivieren.
5074                pla
5075                tax
5076                stx    r1L          ;Fenster-Nr.
5077
5078                lda    #$00
5079                sta    r1H          ;Fenster-Zähler.
5080
5081 if ENABLE_OPTWIN = TRUE
5082                sta    V1CB2          ;Verd. Fenster.
5083 endif
5084
5085 ;                ldx    r1L          ;Fenster-Nr.
5086                jsr    S11A9          ;a0=Fensterdaten.
5087                jsr    S11C0          ;Fens.daten > r5-r7.
5088
5089                ldy    #4 -1          ;Max. 4 Fenster.
5090                sty    r0H          ;Stack-Pointer.
5091
5092 ::1                lda    r1L
5093                cmp    C042A,y          ;Fenster-Nr. prüfen.
5094                bne    :2
5095                sty    V1C9F
5096
5097                lda    r1H
5098                sta    r0L
5099                lda    #$00
5100                sta    r1H
5101                beq    :3
5102
5103 ::2                lda    C042A,y          ;Fenster-Nr. holen.
5104                tax
5105                lda    C0448,x          ;Fenster geöffnet?
5106                beq    :3          ; => Nein, weiter...
5107
5108                jsr    Z11A3          ;Fens.koord. r2-r4.
5109                jsr    S1D1C          ;Schnittfläche?
5110                bcs    :3          ; => Nicht vorhanden.
5111
5112 if ENABLE_OPTWIN = TRUE
5113                ldy    r0H          ;Stack-Pointer.
5114                lda    C042A,y
5115                sta    V1CB2          ;Verd. Fenster.
5116 endif
5117
5118                inc    r1H          ;Verd.Fenster +1.

```



			5119
::3	dec	r0H	5120
	ldy	r0H	5121
	bpl	:1	5122
			5123
	lda	r1H	5124
	beq	:4	5125
			5126
	plp		5127
	sec		5128
	php		5129
			5130
::4	ldx	r1L	5131
	ldy	V1C9F	5132
			5133
if ENABLE_OPTWIN = TRUE			5134
	pha		5135
endif			5136
			5137
	jsr	Z1CA0	5138
			5139
if ENABLE_OPTWIN = TRUE			5140
	pla		5141
endif			5142
			5143
	plp		5144
			5145
	bit	V04AE	5146
	bmi	:7	5147
			5148
	bcc	:6	5149
			5150
; Fenster verdeckt, neu zeichnen.			5151
::5			5152
if ENABLE_OPTWIN = TRUE			5153
	cmp	#2	5154
	bcc	:5a	5155
endif			5156
			5157
; Fenster immer neu zeichnen.			5158
	inc	V18AD	5159
::5c	ldx	C042A	5160
	jmp	S141D	5161
			5162
if ENABLE_OPTWIN = TRUE			5163
; Fenster durch ein Fenster verdeckt.			5164
::5a	lda	V18AD	5165
	bne	:5c	5166
			5167
	ldx	V1CB2	5168
	jsr	S11A9	5169
			5170
	ldy	#6 -1	5171
::5b	lda	(a0),y	5172
	sta	windowTop,y	5173

```

5174         dey
5175         bpl      :5b
5176
5177         ldx      C042A           ;Oberstes Fenster.
5178         jmp      Z141E           ;Fenster anzeigen.
5179 endif
5180
5181 ; Fenster nicht verdeckt.
5182 ::6         ldx      C042A           ;Oberstes Fenster.
5183         stx      WIN042F         ;Aktives Fenster.
5184         jsr      S1E09           ;Arbeitsber.Fenster.
5185
5186         jsr      Z1D7A           ;Titel hervorheben.
5187
5188 ::7         jmp      Z13D3           ;Textgrenzen Desktop.
5189
5190 :V1C9F      b $00      ;Zwischensp. Y-Register.
5191
5192 if ENABLE_OPTWIN = TRUE
5193 :V1CB2      b $00      ;Verdeckendes Fenster.
5194 endif
5195
5196 ; Fenster sortieren.
5197 ; Alle vor (y) liegenden Plätze im
5198 ; Fensterstapel rücken um ein Fenster
5199 ; nach hinten.
5200 ; Übergabe: X = Neues FrontWindow
5201 ;           Y = Startfenster.
5202 :Z1CA0      cpx      C042A           ;Oberstes Fenster.
5203         beq      :2
5204 ::1         lda      C042A -1,y       ;Fensterstapel
5205         sta      C042A +0,y         ;umsortieren.
5206         dey
5207         bne      :1
5208         stx      C042A           ;Oberstes Fenster.
5209 ::2         rts
5210
5211 ; Aktuelles Fenster nach hinten setzen.
5212 ; TD13: BackWindow
5213 :Z1CB6      ldy      #4 -1
5214 ::1         lda      C0448,y         ;Fenster geöffnet?
5215         bne      :2                 ; => Ja, weiter...
5216         dey
5217         bne      :1
5218         rts
5219
5220 ::2         jsr      S5842           ;mark.Dat. anzeigen.
5221         jsr      S524C           ;Dat.-Ausw. aufheben.
5222         jsr      Z1D7D           ;Titel deaktivieren.
5223         jsr      Z13D3           ;Textgrenzen Desktop.
5224
5225         ldx      C042A           ;Oberstes Fenster.
5226         jsr      Z11A3           ;Fens.daten > r2-r4.
5227
5228         lda      C042A           ;Oberstes Fenster.

```

	pha		5229
			5230
:Z1CDA	ldy	#\$00	5231
::1	lda	C042A +1,y ;Fensterstapel	5232
	sta	C042A +0,y ;umsortieren.	5233
	iny		5234
	cpy	#4 -1	5235
	bne	:1	5236
			5237
	pla		5238
	sta	C042A +3 ;Unterstes Fenster.	5239
			5240
	tax		5241
	lda	C0448,x ;Fenster geöffnet?	5242
	beq	:4 ; => Nein, weiter...	5243
			5244
	jsr	S13E5 ;Backup/Set TextWin.	5245
			5246
	ldy	#\$02	5247
::2	tya		5248
	pha		5249
	lda	C042A,y ;Fenster-Nr. holen.	5250
	tax		5251
	lda	C0448,x ;Fenster geöffnet?	5252
	beq	:3 ; => Nein, weiter...	5253
	jsr	Z1428 ;Fenster zeichnen.	5254
::3	pla		5255
	tay		5256
	dey		5257
	bpl	:2	5258
			5259
::4	ldx	C042A ;Oberstes Fenster.	5260
	lda	C0448,x ;Fenster geöffnet?	5261
	bne	:5 ; => Ja, weiter...	5262
	txa		5263
	pha		5264
	jmp	Z1CDA	5265
			5266
::5	jsr	Z1D7A ;Titel hervorheben.	5267
	jmp	Z18E3	5268
			5269
	; Schnittfläche zwischen zwei Rechtecken berechnen.		5270
	; Übergabe: r2-r4 = Rechteck 1		5271
	; r5-r7 = Rechteck 2		5272
	; Rückgabe: r2-r4 = Schnittfläche		5273
	; C=1: Keine Schnittfläche		5274
	; TD13: CutRec		5275
.S1D1C	lda	r6L	5276
	sec		5277
	sbc	r3L	5278
	lda	r6H	5279
	sbc	r3H	5280
	bmi	:1	5281
			5282
	lda	r6L	5283

```

5284          sta    r3L
5285          lda    r6H
5286          sta    r3H
5287
5288 :::1          lda    r5L
5289          sec
5290          sbc    r2L
5291          lda    #$00
5292          sbc    #$00
5293          bmi    :2
5294
5295          lda    r5L
5296          sta    r2L
5297
5298 :::2          lda    r7L
5299          sec
5300          sbc    r4L
5301          lda    r7H
5302          sbc    r4H
5303          bpl    :3
5304
5305          lda    r7L
5306          sta    r4L
5307          lda    r7H
5308          sta    r4H
5309
5310 :::3          lda    r5H
5311          sec
5312          sbc    r2H
5313          lda    #$00
5314          sbc    #$00
5315          bpl    :4
5316
5317          lda    r5H
5318          sta    r2H
5319
5320 :::4          lda    r4L
5321          sec
5322          sbc    r3L
5323          lda    r4H
5324          sbc    r3H
5325          bpl    :6
5326 :::5          sec
5327          rts
5328
5329 :::6          lda    r2H
5330          sec
5331          sbc    r2L
5332          lda    #$00
5333          sbc    #$00
5334          bmi    :5
5335          clc
5336          rts
5337
5338 ; Titel invertieren und ggf. Laufwerk anzeigen.

```



```

5394          lda    #> TOL
5395          adc     r3H
5396          sta     r3H
5397
5398          lda     r4L
5399          sec
5400          sbc     #< TOR
5401          sta     r4L
5402          lda     r4H
5403          sbc     #> TOR
5404          sta     r4H
5405
5406          jsr     InvertRectangle
5407
5408 ; Aktives Laufwerk anzeigen.
5409          lda     V1DE1                ; Laufwerk anzeigen?
5410          bne     :exit                ; => Nein, weiter...
5411
5412          jsr     Z1DBE                ; Lfwk. markieren.
5413
5414 ::exit      pla
5415          sta     a0L                ; Zeiger auf Fens.Dat.
5416          pla
5417          sta     a0H                ; zurücksetzen.
5418
5419 ::done      rts
5420
5421 ; Farbe für akt./Fenster/Lfwk. setzen.
5422 :Z1DBE      ldx     C042A                ; Oberstes Fenster.
5423          lda     C04F8,x                ; Lfwk. für Fenster.
5424          sec
5425          sbc     #$08
5426          asl
5427          tay
5428          lda     V1DD9 +0,y            ; y-Oben.
5429
5430          ldx     C0438                ; TopDesk in Farbe?
5431          bne     :1                    ; => Ja, weiter...
5432
5433 ; Laufwerk mit "*" markieren.
5434          clc
5435          adc     #6                    ; Pos. für Anzeige
5436          sta     r1H                    ; aktives Lfwk.
5437
5438          lda     #< PANEL_X1 -$07
5439          sta     r11L
5440          lda     #> PANEL_X1 -$07
5441          sta     r11H
5442
5443          jsr     Z13D0                ; Textgrenzen Screen.
5444
5445          lda     #"*"
5446          jsr     SmallPutChar
5447
5448          jmp     Z13D3                ; Textgrenzen DeskTop.

```

```

5449
; Laufwerk in Farbe anzeigen. 5450
::1      sta    r2L 5451
          lda    V1DD9 +1,y      ;y-Unten. 5452
          sta    r2H 5453
          lda    COL043B      ;Farbe akt.Laufwerk. 5454
          lda    COL043B      ;Farbe akt.Laufwerk. 5455
; Farbe für akt. Laufwerk setzen. 5456
; Übergabe: A = Farbe. 5457
:Z1DED    ldx    #< PANEL_X0 +$10 5458
          stx    r3L 5459
          ldx    #> PANEL_X0 +$10 5460
          stx    r3H 5461
          ldx    #< PANEL_X1 -$08 5462
          stx    r4L 5463
          ldx    #> PANEL_X1 -$08 5464
          stx    r4H 5465
          jmp    S1561      ;Screencolor setzen. 5466
          jmp    S1561      ;Screencolor setzen. 5467
          jmp    S1561      ;Screencolor setzen. 5468
; Bereiche für Laufwerk-Icons. 5469
:V1DD9    b PANEL_Y0 +$08 +$18 *0 5470
          b PANEL_Y0 +$08 +$18 *0 +$18 -1 5471
          b PANEL_Y0 +$08 +$18 *1 5472
          b PANEL_Y0 +$08 +$18 *1 +$18 -1 5473
          b PANEL_Y0 +$08 +$18 *2 5474
          b PANEL_Y0 +$08 +$18 *2 +$18 -1 5475
          b PANEL_Y0 +$08 +$18 *3 5476
          b PANEL_Y0 +$08 +$18 *3 +$18 -1 5477
          b PANEL_Y0 +$08 +$18 *3 +$18 -1 5478
          b PANEL_Y0 +$08 +$18 *3 +$18 -1 5479
          b PANEL_Y0 +$08 +$18 *3 +$18 -1 5480
:V1DE1    b $00      ;$00 = Akt.Laufwerk anzeigen. 5481
          b $00      ;$00 = Akt.Laufwerk anzeigen. 5482
; Ermittlung des Arbeitsbereichs für Fenster. 5483
; Übergabe: X = Fenster-Nr. 5484
; TD13: GetWorkArea 5485
.S1E09    lda    a0H 5486
          pha    5487
          lda    a0L 5488
          pha    5489
          jsr    Z1E36      ;Grenzen Textfenster. 5490
          jsr    Z1E36      ;Grenzen Textfenster. 5491
          ldy    #(r4H - r2L) 5492
          lda    r2,y 5493
          pha    5494
          dey    5495
          bpl    :1 5496
          bpl    :1 5497
          jsr    S13E5      ;Backup/Set TextWin. 5498
          jsr    S13E5      ;Backup/Set TextWin. 5499
          php    5500
          pla    5501
          tax    5502
          tax    5503

```

```

5504
5505         ldy    #0
5506 ::2         pla
5507         sta    r2,y
5508         iny
5509         cpy    #(r4H - r2L) +1
5510         bne    :2
5511
5512         pla
5513         sta    a0L
5514         pla
5515         sta    a0H
5516
5517         txa
5518         pha
5519         plp
5520         rts
5521
5522 ; Hinweis:
5523 ; Aufruf nur aus vorheriger Routine.
5524 ;
5525 ; TODO:
5526 ; Hier werden die Fens.Koord. nach
5527 ; innen versetzt. Zusammen mit L5875
5528 ; und mod.#05/L2C2E wird die obere
5529 ; Y/X-Koordinate mehrfach bearbeitet!
5530 ; TD13: SetTextWin
5531 :Z1E36     jsr    Z11A3                ;Fens.koord. r2-r4.
5532
5533         inc    r3L                ;Linker Rand +1
5534         bne    :1
5535         inc    r3H
5536
5537 ::1         lda    r2L
5538         clc
5539         adc    #TY +1
5540         sta    r2L
5541
5542         lda    r2H                ;Titelzeile.
5543         sec
5544         sbc    #SY +1
5545         sta    r2H
5546
5547         lda    r4L                ;Hier wird der
5548         sec                ;rechte Rand ein 2x
5549         sbc    #< SX +1            ;korrigiert!
5550         sta    r4L
5551         lda    r4H
5552         sbc    #> SX +1
5553         sta    r4H
5554
5555         rts
5556
5557 ; Datenbyte aus Bitmap holen.
5558 ; TD13: UnPackMap

```



```

:Z1E67          ldy    r3L          5559
                iny          5560
                lda     (r0),y      5561
                sty     r3L          5562
                rts          5563
                5564
; Icon mit Clipping ausgeben.      5565
; Übergabe: X = Y-Koordinate.      5566
;           r10 = X-Koordinate.     5567
;           r13L = Breite Bitmap.   5568
;           r13H = Höhe Bitmap.     5569
;           r0 = Zeiger auf Bitmap. 5570
;           windowsTop... = Fenstergrenzen. 5571
;                                   5572
; TODO:                            5573
; Die Routine wird nur 1x in mod.#05 5574
; verwendet. Evtl. den Code in das   5575
; Modul verschieben.                5576
;                                   5577
; TD13: DrawMap                     5578
;---                                5579
.S1E6F          lda     leftMargin +1      ;X-Koordinate im      5580
                cmp     r10H              ;Fenster?          5581
                bcc     :1                 ; => Nein...         5582
                lda     r10L              5583
                cmp     leftMargin +0      5584
                bcs     :2                 ; => Ja...           5585
                5586
::1             lda     leftMargin +0      5587
                ora     #%00000111        5588
                sec          5589
                sbc     r10L              5590
                sta     r14L              5591
                lda     leftMargin +1      5592
                sbc     r10H              5593
                sta     r14H              5594
                bpl     :3                 5595
                5596
::2             lda     #$00              ;Kein Clipping links. 5597
                sta     r14L              5598
                sta     r14H              5599
                beq     :4                 5600
                5601
::3             lsr     r14H              5602
                ror     r14L              5603
                lsr     r14H              5604
                ror     r14L              5605
                lsr     r14H              5606
                ror     r14L              5607
                5608
                inc     r14L              5609
                5610
                lda     leftMargin +0      5611
                and     #%00000111        5612
                eor     #%00000111        5613

```

```

5614          sta    r14H
5615
5616 ; r14L enthält die Anzahl Bytes +1
5617 ; r14H enthält Nr. des Bits ab dem maskiert werden soll.
5618 ::4          lda    rightMargin +0
5619          ora     #%00000111
5620          sec
5621          sbc     r10L
5622          sta     r15L
5623          lda     rightMargin +1
5624          sbc     r10H
5625          sta     r15H
5626          bpl     :5
5627          rts                                ;Keine Icon-Ausgabe.
5628
5629 ; r15L enthält die Anzahl Bytes +1
5630 ; r15H enthält Nr. des Bits ab dem maskiert werden soll.
5631 ::5          lsr     r15H
5632          ror     r15L
5633          lsr     r15H
5634          ror     r15L
5635          lsr     r15H
5636          ror     r15L
5637
5638          inc     r15L
5639
5640          lda     rightMargin +0
5641          and     #%00000111
5642          eor     #%00000111
5643          sta     r15H
5644
5645          lda     #$00
5646          sta     r9H
5647          sta     r3L
5648          sta     r4L
5649
5650          inc     windowBottom
5651
5652 ::6          lda     #$00
5653          sta     V045D                        ;Icon ganz zeigen.
5654
5655          txa                                ;Zeiler unter
5656          pha                                ;oberen Rand?
5657          sec
5658          sbc     windowTop
5659          lda     #$00                        ;0 - (0 + Carry)
5660          sbc     #$00
5661          bmi     :7                        ; => Nein...
5662
5663          pla
5664          pha
5665
5666          sec                                ;Zeile über
5667          sbc     windowBottom                ;unterem Rand?
5668          lda     #$00                        ;0 - (0 + Carry)

```

	sbcb	#\$00		5669
	bmi	:8	; => Ja, weiter...	5670
				5671
	lda	#\$01		5672
	sta	r13H		5673
				5674
::7	lda	#\$01	;"winOutFlag".	5675
	sta	V045D	;Icon teilw. zeigen.	5676
				5677
::8	lda	r13H		5678
	pha			5679
	lda	r13L		5680
	pha			5681
	lda	r14L		5682
	pha			5683
	lda	r15L		5684
	pha			5685
	jsr	Z1F27	;Zeile ggf. zeigen.	5686
	pla			5687
	sta	r15L		5688
	pla			5689
	sta	r14L		5690
	pla			5691
	sta	r13L		5692
	pla			5693
	sta	r13H		5694
	pla			5695
	tax			5696
	inx		;Nächste Zeile...	5697
	dec	r13H		5698
	bne	:6		5699
				5700
	dec	windowBottom		5701
	rts			5702
				5703
				5704
				5705
				5706
				5707
	lda	r5H		5708
	bmi	:1		5709
	rts			5710
				5711
::1	lda	r10H		5712
	clc			5713
	adc	r5H		5714
	sta	r5H		5715
	lda	r10L		5716
	and	##11111000		5717
	sta	r11L		5718
				5719
	tay			5720
	lda	r10L		5721
	and	##00000111		5722
	sta	r11H		5723

```

5724          tax
5725          lda    (r5),y          ;Bits am linken
5726          and    V2051,x        ;Rands maskieren.
5727          pha
5728
5729          jsr     Z1E67           ;Datenbyte holen.
5730
5731          ldx     #$00
5732          ldy     r11H
5733          jsr     Z1FB0          ;Bits verschieben.
5734
5735          ldy     r11L           ;Mit Hintergrund
5736          sta     r11L          ;verknüpfen und als
5737          pla           ;1.Byte schreiben.
5738          ora     r11L
5739          jsr     Z1FDA          ;Bits schreiben.
5740
5741          dec     r13L           ;Restbreite -1.
5742
5743 ::2          tya
5744          clc
5745          adc     #8
5746          tay
5747          bcc     :3
5748          inc     r5H
5749
5750 ::3          sty     r11L
5751
5752          txa
5753          pha
5754          jsr     Z1E67           ;Datenbyte holen.
5755          tay
5756          pla
5757          tax
5758          tya
5759
5760          ldy     r11H
5761          jsr     Z1FB0          ;Bits verschieben.
5762
5763          ldy     r11L
5764          jsr     Z1FDA          ;Bits schreiben.
5765
5766          dec     r13L
5767          bne     :2
5768          tya
5769          clc
5770          adc     #8
5771          tay
5772          bcc     :4
5773          inc     r5H
5774
5775 ::4          sty     r11L
5776          stx     r13H
5777
5778          lda     r11H

```

	eor	#\$07		5779
	tax			5780
	lda	(r5),y		5781
	and	V2048,x		5782
	pha			5783
				5784
	lda	#\$00		5785
	ldx	r13H		5786
	ldy	r11H		5787
	jsr	Z1FB0	;Bits verschieben.	5788
				5789
	ldy	r11L		5790
	sta	r11L		5791
	pla			5792
	ora	r11L		5793
	jsr	Z1FDA	;Bits schreiben.	5794
	sty	r11L		5795
	rts			5796
				5797
				5798
				5799
				5800
				5801
				5802
				5803
				5804
				5805
				5806
				5807
				5808
				5809
				5810
::1	lsr		;Highbyte um y	5811
	ror	V1FD9	;Schritte nach rechts	5812
	bcc	:2	;rotieren.	5813
	ora	##10000000		5814
::2	dey			5815
	bne	:1		5816
				5817
	pha			5818
				5819
	lda	#8	;Lowbyte um (8-y)	5820
	sec		;nach links schieben,	5821
	sbc	V1FD8		5822
	tay			5823
	lda	V1FD9		5824
::3	lsr			5825
	dey			5826
	bne	:3		5827
				5828
	tax			5829
				5830
::4	pla			5831
	rts			5832
				5833

```

5834 :V1FD8          b $00  ;x
5835 :V1FD9          b $00  ;y
5836
5837 ; Byte in Grafik speichern.
5838 ; TD13: StoreBits
5839 :Z1FDA          dec    V045D          ;Iconzeile sichtbar?
5840                bmi    :1              ; => Ja, weiter...
5841                inc    V045D          ;Zeile nicht im
5842                rts                    ;Fenster...
5843
5844 ::1              inc    V045D
5845                dec    r15L
5846                beq    :2
5847                bpl    :3
5848                lda    #$00
5849                sta    r15L
5850                rts
5851
5852 ::2              stx    V2046
5853                sta    r15L
5854
5855                ldx    r15H
5856                lda    (r5),y
5857                and    V2047,x
5858                pha
5859                txa
5860                eor    #$07
5861                tax
5862                lda    r15L
5863                and    V2051,x
5864                sta    r15L
5865                pla
5866                ora    r15L
5867
5868                ldx    #$00
5869                stx    r15L
5870
5871                ldx    V2046
5872 ::3              dec    r14L
5873                beq    :5
5874                bpl    :4
5875
5876                inc    r14L
5877                sta    (r5),y
5878 ::4              rts
5879
5880 ::5              stx    V2046
5881                sta    r14L
5882
5883                lda    r14H
5884                eor    #$07
5885                tax
5886                lda    (r5),y
5887                and    V2050,x
5888                pha

```

	txa		5889
	eor	##00000111	5890
	tax		5891
	lda	r14L	5892
	and	V2048,x	5893
	sta	r14L	5894
	pla		5895
	ora	r14L	5896
	sta	(r5),y	5897
	lda	#\$00	5898
	sta	r14L	5899
	ldx	V2046	5900
	rts		5901
			5902
:V2046	b \$00	;x	5903
			5904
:V2047	b \$00	;Bitmaske 0	5905
			5906
:V2048	b %00000001	;Bitmaske 1	5907
	b %00000011		5908
	b %00000111		5909
	b %00001111		5910
	b %00011111		5911
	b %00111111		5912
	b %01111111		5913
	b %11111111		5914
			5915
:V2050	b \$00	;Bitmaske 3	5916
			5917
:V2051	b %10000000	;Bitmaske 2	5918
	b %11000000		5919
	b %11100000		5920
	b %11110000		5921
	b %11111000		5922
	b %11111100		5923
	b %11111110		5924
	b %11111111		5925
			5926
	; Dialogbox ohne ScreenRecover.		5927
	; Übergabe: A/X = Zeiger auf Dialogbox.		5928
.S2080	jsr	S2090 ;DBox ohne Recover.	5929
			5930
	; Warten bis keine Maustaste gedrückt.		5931
	; TD13: n.v.		5932
.S52AD	lda	mouseData	5933
	bpl	S52AD	5934
			5935
	; Hinweis:		5936
	; Nur den Maustasten-Status löschen!		5937
	lda	pressFlag	5938
	and	##11011111 ;Bit%5=MTaste neu.	5939
	sta	pressFlag	5940
			5941
	rts		5942
			5943

```

5944 ; Dialogbox ohne ScreenRecover.
5945 ; Übergabe: A/X = Zeiger auf Dialogbox.
5946 .S2090          sta    r0L
5947                stx    r0H
5948
5949 ; Dialogbox ohne ScreenRecover.
5950 ; TD13: NewDoDlgBox
5951 .S2095          lda    r6H
5952                pha
5953                lda    r6L
5954                pha
5955                lda    r7L
5956                pha
5957
5958 ; Hinweis:
5959 ; ":MyRecoverService" deaktivieren, da
5960 ; GEOS/MP3 den Bildhintergrund rettet.
5961                lda    RecoverVector +1
5962                pha
5963                lda    RecoverVector +0
5964                pha
5965                ldy    #$00
5966                sty    RecoverVector +0
5967                sty    RecoverVector +1
5968
5969                lda    C_DBoxBack
5970                pha
5971                lda    C0L043C
5972                sta    C_DBoxBack
5973
5974                jsr    DoDlgBox
5975
5976                pla
5977                sta    C_DBoxBack
5978
5979                pla
5980                sta    RecoverVector +0
5981                pla
5982                sta    RecoverVector +1
5983
5984                pla
5985                sta    r7L
5986                pla
5987                sta    r6L
5988                pla
5989                sta    r6H
5990                rts
5991
5992 ; Hintergrund wieder herstellen.
5993 ; Übergabe: r2-r4 = Bereich.
5994 ; TD13: BackgroundRectangle
5995 :Z20C3          lda    C0438                ;TopDesk in Farbe?
5996                beq    :1                ; => Nein, weiter...
5997
5998                lda    sysRAMFlg

```



	and	##00001000	;Hitergrundbild?	5999
	beq	:1	; => Nein, weiter...	6000
				6001
	dec	V238F	;Zugriff auf MP3/SYS.	6002
	jsr	Z21C6	;Hintergrund aus REU.	6003
	inc	V238F	;Zugriff auf MP3/DAT.	6004
	rts			6005
				6006
::1	lda	screencolors	;GEOS-Farben.	6007
	jsr	S1561	;Screencolor setzen.	6008
	lda	BackScrPattern	;System-Füllmuster.	6009
	jmp	S140C	;Rechteck zeichnen.	6010
				6011
			; Icons aus DoIcons-Menü anzeigen.	6012
			; Wird u.a. verwendet nachdem man die	6013
			; GEOS-Farben angepasst hat und der	6014
			; Bildschirm neu aufgebaut wird.	6015
:Z20DF	lda	a0H		6016
	pha			6017
	lda	a0L		6018
	pha			6019
				6020
	lda	V044E +0	;Zeiger auf	6021
	sta	a0L	;DoIcons-Tabelle.	6022
	lda	V044E +1		6023
	sta	a0H		6024
				6025
	ldy	#\$00		6026
	lda	(a0),y	;Anzahl Icons.	6027
	sta	V214F		6028
	iny			6029
	iny		;Mausposition	6030
	iny		;überlesen.	6031
	iny		;1xWORD, 1xBYTE.	6032
				6033
	b	\$2c		6034
				6035
::2	ldy	#\$00		6036
	lda	(a0),y		6037
	sta	r0L	;Bitmap-Adresse.	6038
	jsr	Z214B	;Byte einlesen.	6039
	sta	r0H		6040
				6041
	jsr	Z214B	;Byte einlesen.	6042
	sta	r1L	;X-Koordinate.	6043
	jsr	Z214B	;Byte einlesen.	6044
	sta	r1H	;Y-Koordinate.	6045
	jsr	Z214B	;Byte einlesen.	6046
	sta	r2L	;Breite.	6047
	jsr	Z214B	;Byte einlesen.	6048
	sta	r2H	;Höhe.	6049
				6050
	iny		;Zeiger auf das	6051
	iny		;nächste Icon in	6052
	iny		;Tabelle.	6053

```

6054          tya
6055          clc
6056          adc    a0L
6057          sta    a0L
6058          lda    #$00
6059          adc    a0H
6060          sta    a0H
6061
6062          jsr    BitmapUp          ;Icon ausgeben.
6063
6064          dec    V214F
6065          bne    :2
6066
6067          lda    WIN0446 +0          ;Zeiger Fensterdaten.
6068          sta    a0L
6069          lda    WIN0446 +1
6070          sta    a0H
6071
6072          lda    #WN_RESTORE
6073          ldx    #0
6074          jsr    Z1837          ;Fensterfunktion.
6075
6076          pla
6077          sta    a0L
6078          pla
6079          sta    a0H
6080
6081          ldy    #$00
6082          rts
6083
6084 ; Nächstes Byte einlesen.
6085 :Z214B          iny
6086          lda    (a0),y
6087          rts
6088
6089 :V214F          b $00 ;Anzahl Icons in Tabelle.
6090
6091 ; Klick auf Maximize-Gadget:
6092 ; Neue Fenstergröße ermitteln.
6093 ; Wird nur 1x bei x1B4B verwendet.
6094 ;
6095 ; Aktives Fenster auf Maximalgröße
6096 ; setzen bzw. zurück auf zurücksetzen.
6097 ; neue Größe wird in r2-r4 übergeben.
6098 ;
6099 ; TD13: MaxWindow
6100 :Z2150          lda    WIN042F          ;Zeiger auf Speicher
6101          asl          ;für Fensterdaten
6102          sta    r2L          ;berechnen.
6103          asl
6104          clc
6105          adc    r2L
6106          tax
6107
6108          ldy    #6 -1

```

```

::1          lda    (a0),y          ;Fenstergröße =          6109
              cmp    V2179,y        ;volle Größe?          6110
              bne    x2168          6111
              dey          6112
              bpl    :1             6113
              6114
; Fenster auf die gespeicherte Größe          6115
; zurücksetzen.          6116
              ldy    #0             ;Ja, Fenstergröße          6117
::2          lda    V217F,x         ;zurücksetzen.          6118
              sta    r2,y          6119
              inx          6120
              iny          6121
              cpy    #(r4H - r2L) +1 6122
              bcc    :2             6123
              rts                 6124
              6125
; Fenster auf volle Größe setzen und          6126
; Fenstergröße zwischenspeichern.          6127
:x2168       ldy    #0             6128
::1          lda    (a0),y          6129
              sta    V217F,x        6130
              lda    V2179,y        6131
              sta    r2,y          6132
              inx          6133
              iny          6134
              cpy    #(r4H - r2L) +1 6135
              bcc    :1             6136
              6137
              rts                 6138
              6139
; Desktop/Bereichsgrenzen für Fenster.          6140
; Max. Fenstergröße.          6141
:V2179       b DESK_Y0,DESK_Y1      6142
              w DESK_X0,DESK_X1      6143
              6144
; Zwischenspeicher Fenstergröße.          6145
; Größe vor Fenster/maximieren.          6146
:V217F       b DESK_Y0,DESK_Y1      ;Fenster #1.          6147
              w DESK_X0,DESK_X1      6148
              6149
              b DESK_Y0,DESK_Y1      ;Fenster #2.          6150
              w DESK_X0,DESK_X1      6151
              6152
              b DESK_Y0,DESK_Y1      ;Fenster #3.          6153
              w DESK_X0,DESK_X1      6154
              6155
              b DESK_Y0,DESK_Y1      ;Fenster #4.          6156
              w DESK_X0,DESK_X1      6157
              6158
; Bildhintergrund wieder herstellen.          6159
; TD13: MyRecoverService          6160
.S2185       lda    Flag_ExtRAMinUse 6161
              and    #%10111111      6162
              sta    Flag_ExtRAMinUse 6163

```

```

6164
6165         lda    #< V0460           ;Zeiger auf Buffer1
6166         sta    r1L                 ;für Fenstergrenzen.
6167         lda    #> V0460
6168         sta    r1H
6169
6170         ldx    V046C               ;Buffer1 od. Buffer2?
6171         beq    :1                  ; => Buf1, weiter...
6172
6173         lda    #< V0466           ;Zeiger auf Buffer2
6174         sta    r1L                 ;für Fenstergrenzen.
6175         lda    #> V0466
6176         sta    r1H
6177
6178 :::1      ldy    #(r4H - r2L)
6179 :::2      lda    (r1),y
6180         sta    r2,y
6181         dey
6182         bpl    :2
6183
6184         jsr    Z21C6               ;Hintergrund aus REU.
6185
6186         lda    #$ff
6187         sta    V046C               ;Buffer voll.
6188         rts
6189
6190 ; Hintergrund in REU speichern.
6191 ; Wird nur bei Z3BBB/Z3BBE verwendet.
6192 :Z21B5    lda    V046C             ;Buffer frei?
6193         bpl    :1                  ; => Ja, weiter...
6194
6195 ; Buffer belegt, Ende.
6196         rts
6197
6198 :::1      lda    Flag_ExtRAMinUse
6199         ora    #%01000000
6200         sta    Flag_ExtRAMinUse
6201
6202         ldy    #%10010000         ;StashRAM
6203         b $2c
6204
6205 ; Bildschirmbereich aus REU wieder herstellen.
6206 :Z21C6    ldy    #%10010001       ;FetchRAM
6207         sty    V2390              ;Jobcode für DoRAMOp.
6208
6209         php
6210         sei
6211
6212         ldx    #(r15H - r0L)       ;Register r0-r15
6213 :::1      lda    r0,x              ;zwischenspeichern.
6214         pha
6215         dex
6216         bpl    :1
6217
6218         lda    dispBufferOn        ;Nur Vordergrund-

```

	pha		;grafik verwenden.	6219
	lda	#ST_WR_FORE		6220
	sta	dispBufferOn		6221
				6222
	lda	r3H	;Erstes Card in	6223
	lsr		;Zeile berechnen.	6224
	lda	r3L		6225
	and	##11111000		6226
	sta	r3L		6227
	ror			6228
	lsr			6229
	lsr			6230
	sta	V2391	;1.Card in Zeile.	6231
				6232
	lda	r2L	;Zeile berechnen.	6233
	lsr			6234
	lsr			6235
	lsr			6236
	sta	V2392	;Erste Card-Zeile.	6237
	sta	V2397	;Bildschirmzeile.	6238
				6239
	lda	r2H		6240
	lsr			6241
	lsr			6242
	lsr			6243
	sec			6244
	sbc	V2397	;Bildschirmzeile.	6245
	clc			6246
	adc	#\$01		6247
	sta	V2393	;Anzahl Card-Zeilen.	6248
				6249
	lda	r4L	;Breite Ausschnitt	6250
	ora	##00000111	;berechnen.	6251
	sta	r4L		6252
	sec			6253
	sbc	r3L		6254
	sta	r0L		6255
	lda	r4H		6256
	sbc	r3H		6257
	sta	r0H		6258
				6259
	inc	r0L		6260
	bne	:2		6261
	inc	r0H		6262
				6263
::2	lda	r0L	;Breite der Zeile	6264
	sta	V2395 +0	;in Pixel = Anzahl	6265
	lda	r0H	;Bytes in Zeile.	6266
	sta	V2395 +1		6267
				6268
	ldx	#r0	;Anzahl Cards	6269
	ldy	#3	;berechnen.	6270
	jsr	DShiftRight		6271
				6272
	lda	r0L		6273

```

6274          sta    V2394          ;Anzahl Cards/Zeile.
6275
6276 ::3          jsr    Z22FF          ;Zeiger auf Farben.
6277          ldy    V2390          ;Jobcode für DoRAMOp.
6278          jsr    DoRAMOp          ;RAM laden/speichern.
6279
6280          jsr    Z225B          ;Zeiger auf Grafik.
6281          ldy    V2390          ;Jobcode für DoRAMOp.
6282          jsr    DoRAMOp          ;RAM laden/speichern.
6283
6284          inc     V2397          ;Bildschirmzeile.
6285
6286          dec     V2393          ;Anzahl Card-Zeilen.
6287          bne     :3
6288
6289          pla
6290          sta     dispBufferOn
6291
6292          ldx     #0              ;Register r0-r15
6293 ::4          pla              ;wieder herstellen.
6294          sta     r0,x
6295          inx
6296          cpx     #(r15H - r0L) +1
6297          bcc     :4
6298
6299          plp
6300          rts
6301
6302 ; Zeiger auf Grafikspeicher berechnen.
6303 ; Buf1: MP3_64K_DATA   = $0000-$1FFF
6304 ; Buf2: MP3_64K_SYSTEM = R2A_BS_GRAFX
6305 :Z225B          ldx     V2397          ;Bildschirmzeile.
6306          lda     V2398,x          ;Offset Zeile für
6307          clc              ;Bildschirmspeicher.
6308          adc     #< SCREEN_BASE
6309          sta     r0L
6310          lda     V23B1,x
6311          adc     #> SCREEN_BASE
6312          sta     r0H
6313
6314          bit     V238F          ;Zugriff auf MP3/DAT?
6315          bpl     :1              ; => Ja, weiter...
6316
6317          lda     V2398,x          ;Zeiger auf Daten für
6318          clc              ;Hintergrundbild.
6319          adc     #< R2A_BS_GRAFX
6320          sta     r1L
6321          lda     V23B1,x
6322          adc     #> R2A_BS_GRAFX
6323          sta     r1H
6324
6325          jmp     Z228E
6326
6327 ::1          lda     V2398,x          ;RAM in REU für
6328          sta     r1L              ;Zwischenspeicher.

```

	lda	V23B1,x		6329
	sta	r1H		6330
				6331
:Z228E	ldx	V2391	;1.Card in Zeile.	6332
	lda	V23CA,x	;Offset innerhalb	6333
	clc		;Zeile berechnen.	6334
	adc	r0L		6335
	sta	r0L		6336
	lda	V23F2,x		6337
	adc	r0H		6338
	sta	r0H		6339
				6340
	lda	V23CA,x	;Offset in REU.	6341
	clc			6342
	adc	r1L		6343
	sta	r1L		6344
	lda	V23F2,x		6345
	adc	r1H		6346
	sta	r1H		6347
				6348
	lda	V2395 +0	;Breite der Zeile	6349
	sta	r2L	;in Pixel = Anzahl	6350
	lda	V2395 +1	;Bytes in Zeile.	6351
	sta	r2H		6352
				6353
	jmp	Z2380	;Speicherbank setzen.	6354
				6355
; Zeiger auf Farbspeicher		berechnen.		6356
:Z22FF	ldx	V2397	;Bildschirmzeile.	6357
	lda	V241A,x		6358
	clc			6359
	adc	#< COLOR_MATRIX		6360
	sta	r0L		6361
	lda	V2433,x		6362
	adc	#> COLOR_MATRIX		6363
	sta	r0H		6364
				6365
	bit	V238F	;Zugriff auf MP3/DAT?	6366
	bpl	:1	; => Ja, weiter...	6367
				6368
	lda	V241A,x	;Zeiger auf Daten für	6369
	clc		;Hintergrundbild.	6370
	adc	#< R2A_BS_COLOR		6371
	sta	r1L		6372
	lda	V2433,x		6373
	adc	#> R2A_BS_COLOR		6374
	sta	r1H		6375
				6376
	jmp	Z2337		6377
				6378
::1	lda	V241A,x	;Zeiger auf Speicher	6379
	clc		;hinter Grafikdaten.	6380
	adc	#< 25*40*8		6381
	sta	r1L		6382
	lda	V2433,x		6383

```

6384          adc    #> 25*40*8
6385          sta    r1H
6386
6387 :Z2337          lda    V2391                ;1.Card in Zeile.
6388          clc                    ;Offset innerhalb
6389          adc    r0L                ;Zeile berechnen.
6390          sta    r0L
6391          lda    #$00
6392          adc    r0H
6393          sta    r0H
6394
6395          lda    V2391                ;1.Card in Zeile.
6396          clc                    ;Offset in REU.
6397          adc    r1L
6398          sta    r1L
6399          lda    #$00
6400          adc    r1H
6401          sta    r1H
6402
6403          lda    V2394                ;Anzahl Cards/Zeile.
6404          sta    r2L
6405          lda    #$00
6406          sta    r2H
6407
6408 ; Speicherbank setzen.
6409 :Z2380          lda    MP3_64K_SYSTEM        ;REU-Speicherbank für
6410                                     ;Hintergrundbild.
6411
6412          bit    V238F                ;Zugriff auf MP3/SYS?
6413          bmi    :1                ; => Ja, weiter...
6414
6415          lda    MP3_64K_DATA            ;REU-Speicherbank für
6416                                     ;Zwischenspeicher.
6417
6418 ::1          sta    r3L
6419          rts
6420
6421 :V238F          b $00                ;$00=MP3/DAT, $FF=MP3/SYS.
6422 :V2390          b $00                ;Jobcode für DoRAMOp.
6423 :V2391          b $00                ;1.Card in Zeile.
6424 :V2392          b $00                ;Erste Card-Zeile.
6425 :V2393          b $00                ;Anzahl Card-Zeilen.
6426 :V2394          b $00                ;Anzahl Cards/Zeile.
6427 :V2395          w $0000            ;Anzahl Cardsx8 = Anzahl Bytes.
6428 :V2397          b $00                ;Bildschirmzeile.
6429
6430 ; Offset Grafikzeile.
6431 :V2398          b < 0*40*8, < 1*40*8, < 2*40*8, < 3*40*8
6432          b < 4*40*8, < 5*40*8, < 6*40*8, < 7*40*8
6433          b < 8*40*8, < 9*40*8, <10*40*8, <11*40*8
6434          b <12*40*8, <13*40*8, <14*40*8, <15*40*8
6435          b <16*40*8, <17*40*8, <18*40*8, <19*40*8
6436          b <20*40*8, <21*40*8, <22*40*8, <23*40*8
6437          b <24*40*8
6438

```



```

:V23B1      b > 0*40*8, > 1*40*8, > 2*40*8, > 3*40*8      6439
              b > 4*40*8, > 5*40*8, > 6*40*8, > 7*40*8      6440
              b > 8*40*8, > 9*40*8, >10*40*8, >11*40*8      6441
              b >12*40*8, >13*40*8, >14*40*8, >15*40*8      6442
              b >16*40*8, >17*40*8, >18*40*8, >19*40*8      6443
              b >20*40*8, >21*40*8, >22*40*8, >23*40*8      6444
              b >24*40*8      6445
; Offset Grafikspalte.      6446
:V23CA      b < 0*8, < 1*8, < 2*8, < 3*8, < 4*8      6448
              b < 5*8, < 6*8, < 7*8, < 8*8, < 9*8      6449
              b <10*8, <11*8, <12*8, <13*8, <14*8      6450
              b <15*8, <16*8, <17*8, <18*8, <19*8      6451
              b <20*8, <21*8, <22*8, <23*8, <24*8      6452
              b <25*8, <26*8, <27*8, <28*8, <29*8      6453
              b <30*8, <31*8, <32*8, <33*8, <34*8      6454
              b <35*8, <36*8, <37*8, <38*8, <39*8      6455
:V23F2      b > 0*8, > 1*8, > 2*8, > 3*8, > 4*8      6457
              b > 5*8, > 6*8, > 7*8, > 8*8, > 9*8      6458
              b >10*8, >11*8, >12*8, >13*8, >14*8      6459
              b >15*8, >16*8, >17*8, >18*8, >19*8      6460
              b >20*8, >21*8, >22*8, >23*8, >24*8      6461
              b >25*8, >26*8, >27*8, >28*8, >29*8      6462
              b >30*8, >31*8, >32*8, >33*8, >34*8      6463
              b >35*8, >36*8, >37*8, >38*8, >39*8      6464
; Offset Farbzeile.      6465
:V241A      b < 0*40, < 1*40, < 2*40, < 3*40      6467
              b < 4*40, < 5*40, < 6*40, < 7*40      6468
              b < 8*40, < 9*40, <10*40, <11*40      6469
              b <12*40, <13*40, <14*40, <15*40      6470
              b <16*40, <17*40, <18*40, <19*40      6471
              b <20*40, <21*40, <22*40, <23*40      6472
              b <24*40      6473
:V2433      b > 0*40, > 1*40, > 2*40, > 3*40      6475
              b > 4*40, > 5*40, > 6*40, > 7*40      6476
              b > 8*40, > 9*40, >10*40, >11*40      6477
              b >12*40, >13*40, >14*40, >15*40      6478
              b >16*40, >17*40, >18*40, >19*40      6479
              b >20*40, >21*40, >22*40, >23*40      6480
              b >24*40      6481
; Ermittlung einer Stringlänge in Zeichen und in Pixel.      6482
; Übergabe: r0 = Zeiger auf Sstring, max. 256 Zeichen.      6483
;                               Ende mit NULL oder $A0.      6484
; Rückgabe: X  = Anzahl Zeichen.      6485
;                               r1 = Breite in Pixel.      6486
; TD13: StringLen      6487
.S244C      lda    #$00      6488
              sta    r1L      6489
              sta    r1H      6490
              sta    r13H      6491
              sta    r13L      6492
              sta    r13L      6493

```

```

6494
6495 ::1          ldy    r13L
6496             lda     (r0),y
6497             beq     :3
6498             cmp     #$a0
6499             beq     :3
6500             inc     r13H
6501             ldx     currentMode
6502             jsr     GetRealSize
6503             tya
6504             clc
6505             adc     r1L
6506             sta     r1L
6507             bcc     :2
6508             inc     r1H
6509
6510 ::2          inc     r13L
6511             bne     :1
6512
6513 ::3          ldx     r13H
6514             rts
6515
6516 ; Füllstandsanzeige.
6517 :Z2478        jsr     Z2491          ;Koord. Pos.-Anzeige.
6518             jsr     S13E5          ;Backup/Set TextWin.
6519
6520             lda     #0
6521             jsr     SetPattern
6522             jsr     S14EB          ;Rechteck zeichnen.
6523             jsr     Z4CCF          ;Koord. Füllstand.
6524
6525             lda     #1
6526             jsr     SetPattern
6527             jmp     S14EB          ;Rechteck zeichnen.
6528
6529 ; Koordinaten für Positionsanzeige.
6530 ; Bereich zwischen den Scroll U/D-
6531 ; Pfeilen am rechten Rand.
6532 :Z2491        ldx     WIN042F      ;Aktives Fenster.
6533             jsr     Z11A3          ;Fens.koord. r2-r4.
6534
6535             lda     r2L
6536             clc
6537             adc     #TY +11
6538             sta     r2L
6539
6540             lda     r2H
6541             sec
6542             sbc     #SY +11
6543             sta     r2H
6544
6545             lda     r4L
6546             bne     :1
6547             dec     r4H
6548 ::1          dec     r4L

```

```

6549
6550      lda    r4L
6551      sec
6552      sbc    #< SX -2
6553      sta    r3L
6554      lda    r4H
6555      sbc    #> SX -2
6556      sta    r3H
6557
6558      rts
6559
6560 ; Fehlerton ausgeben.
6561 ; Übergabe: A = Anzahl Töne.
6562 .S24C0      pha
6563
6564      jsr    SCPU_Pause
6565
6566      ldx    #50                      ;Ton-Frequenz.
6567      jsr    Z24DD                    ;Signalton ausgeben.
6568
6569 :Z24CF      pla
6570      sec
6571      sbc    #1
6572      beq    :1
6573
6574 ; Hinweis:
6575 ; Die Beep-Routine verändert die Ton-
6576 ; Frequenz. Sollen mehrere Signaltöne
6577 ; über die Routine ausgegeben werden,
6578 ; dann haben die folgenden Töne eine
6579 ; andere Start-Ton-Frequenz als der
6580 ; erste Signalton.
6581 ; In TopDesk 1.3 bleibt die Frequenz
6582 ; für alle Signaltöne gleich.
6583      pha
6584      jsr    Z24E0                    ;Ton erneut ausgeben.
6585      jmp    Z24CF                    ;Nächster Signalton.
6586
6587 ::1      rts
6588
6589 ; Signalton ausgeben.
6590 :Z24DD      stx    Z24F7 +1          ;Ton-Frequenz.
6591
6592 :Z24E0      jsr    InitForIO
6593
6594      lda    #$11
6595      sta    $d404
6596      lda    #$00
6597      sta    $d405
6598      lda    #$f7
6599      sta    $d406
6600      lda    #$0f
6601      sta    $d418
6602
6603 :Z24F7      lda    #50                ;Wird verändert.

```

```

6604          sec
6605          sbc    #10
6606          sta    Z24F7 +1
6607          sta    $d400          ;Ton-Frequenz
6608          sta    $d401          ;fest legen.
6609
6610          jsr    SCPU_Pause
6611
6612          lda    #$10
6613          sta    $d404
6614          lda    #$00
6615          sta    $d418
6616
6617          jmp    DoneWithIO
6618
6619 ; Fensterbereich anzeigen.
6620 ; Übergabe: r2L = Y-oben
6621 ;           r2H = Y-unten
6622 ;           r3  = X-links
6623 ;           r4  = X-rechts
6624 ; Rückgabe: -
6625 ; Verändert: r0,r1,r5-r7L,A,X,Y
6626 ; TD13: InvFrame
6627 :Z2515      lda    r8H
6628            pha
6629            lda    r8L
6630            pha
6631
6632            lda    r11H
6633            pha
6634            lda    r11L
6635            pha
6636
6637            lda    r2H
6638            pha
6639
6640            lda    r2L
6641            sta    r11L
6642            jsr    InvertLine      ;Obere Linie.
6643            pla
6644            sta    r11L
6645            jsr    InvertLine      ;Untere Linie.
6646
6647            pla
6648            sta    r11L
6649            pla
6650            sta    r11H
6651
6652            pla
6653            sta    r8L
6654            pla
6655            sta    r8H
6656
6657            lda    r2L
6658            and    #%11111000

```

	sta	r0L	6659
			6660
	lda	r2H	6661
	and	11111000	6662
	sta	r0H	6663
			6664
	ldx	r0L	6665
	jsr	GetScanLine	6666
			6667
	lda	r3L	6668
		;Linke Linie.	
	ldy	r3H	6669
	jsr	Z255A	6670
		;Vertikale Linie.	
			6671
	lda	r4L	6672
		;Rechte Linie.	
	ldy	r4H	6673
:Z255A	tax		6674
	and	11111000	6675
	clc		6676
	adc	r5L	6677
	sta	r6L	6678
	tya		6679
	adc	r5H	6680
	sta	r6H	6681
			6682
	txa		6683
	and	00000111	6684
	tax		6685
	lda	V25C6,x	6686
	sta	r1L	6687
			6688
; Invertieren des oberen Restes.			6689
	lda	r2L	6690
	and	00000111	6691
	tay		6692
	jsr	Z25B1	6693
		;Bits invertieren.	
	tya		6694
	sec		6695
	sbc	#8	6696
	sta	r1H	6697
			6698
; Mitte.			6699
	lda	r0H	6700
	sec		6701
	sbc	r0L	6702
	lsr		6703
	lsr		6704
	lsr		6705
	tax		6706
	dex		6707
	beq	:2	6708
			6709
::1	ldy	r1H	6710
	jsr	Z25B1	6711
		;Bits invertieren.	
	dex		6712
	bne	:1	6713

```

6714
6715 ; Invertieren des unteren Restes.
6716 ::2          lda    r2H
6717             and    #%00000111
6718             cmp    r1H
6719             bcc    :4
6720
6721             sta    r7L
6722
6723             ldy    r1H
6724 ::3          jsr    Z25A8             ;Bits invertieren.
6725             cpy    r7L
6726             beq    :3
6727             bcc    :3
6728
6729 ::4          rts
6730
6731 ; Jedes zweite Bit invertieren.
6732 :Z25A8        lda    (r6),y
6733             eor    r1L
6734             sta    (r6),y
6735             iny
6736             iny
6737             rts
6738
6739 ; Max. 8 Bits invertieren.
6740 :Z25B1        jsr    Z25A8             ;Bits invertieren.
6741             cpy    #8
6742             bcc    Z25B1
6743
6744             lda    #< 40*8
6745             clc
6746             adc    r6L
6747             sta    r6L
6748             lda    #> 40*8
6749             adc    r6H
6750             sta    r6H
6751             rts
6752
6753 :V25C6        b $80,$40,$20,$10
6754             b $08,$04,$02,$01
6755
6756 :V25CE        b $00 ;Zwischensp. Laufwerk für Dokument.
6757 .V25CF        b $00 ;$00/$80=Datei öffnen/drucken.
6758             ;$01=Quicklink öffnen (mod.#09).
6759 .V25D0        b $00 ;GEOS-Dateityp für "Datei öffnen".
6760
6761 :V25D1        s 18 ;Zwischensp. Diskname/GetFile.
6762 :V25E3        s 14 ;Zwischensp. GEOS-Klasse/FindFTypes.
6763 :V25F1        s 17 ;Zwischensp. Dateiname/FindFTypes.
6764
6765 :V2602        b $ff
6766
6767 ; Dateien suchen.
6768 ; Übergabe:

```



```

6824          sta    r14L          ;Offset löschen.
6825 ;          sta    r14H
6826
6827 ; Ergänzung:
6828 ; r15L: Flag: Überlauf aufgetreten.
6829 ; r15H: Kopie Offset für Berechnung
6830 ; Gesamtanzahl Dateien.
6831 ;          lda    #$00
6832          sta    r15L          ;>$00: BFR_OVERFLOW.
6833          lda    r11H
6834          sta    r15H          ;Kopie von Offset.
6835
6836 ; TurboDOS und I/O einschalten.
6837          jsr    Z27CA          ;Zeiger erster Block.
6838          txa
6839          bne    :2            ; => Fehler, Abbruch.
6840
6841 ; Dateien einlesen.
6842          jsr    Z2651          ;Dir lesen/filtern.
6843          txa
6844          bne    :2            ; => Fehler, Abbruch.
6845
6846 ;          ldx    #NO_ERROR
6847          lda    r11L          ;Speicher voll?
6848          beq    :2            ; => Ja, weiter...
6849          lda    isGEOS        ;GEOS-Diskette?
6850          beq    :2            ; => Nein, weiter...
6851
6852 ; Dateien Borderblock einlesen.
6853 ;---
6854          lda    curDirHead +171
6855 ; Hinweis:
6856 ; Einige MP3-Versionen testen immer
6857 ; das Hauptverzeichnis auf GEOS-Disk,
6858 ; aber der Borderblock ist in einem
6859 ; Unterverzeichnis nicht gesetzt.
6860 ; => Tr/Se=$00/$00, Fehler INV_TRACK
6861 ; Problem ist erst in GDOS64 behoben.
6862 ;---
6863          beq    :2            ; => Kein BorderB.
6864 ;---
6865          sta    V266A          ;Dir-Block/Track.
6866          lda    curDirHead +172
6867          sta    V266B          ;Dir-Block/Sektor.
6868 ;---
6869 ; Alternativ:
6870 ; Ließt aber zusätzlich die BAM neu
6871 ; in den Speicher -> Langsam.
6872 ;          jsr    GetBorderBlock ;Borderblock?
6873 ;          txa
6874 ;          bne    :2            ; => Fehler, Abbruch.
6875 ;
6876 ;          tya
6877 ;          beq    :2            ; => Kein BorderB.
6878 ;

```



```

;          lda    r1L                                6879
;          sta    V266A                                ;Dir-Block/Track. 6880
;          lda    r1H                                6881
;          sta    V266B                                ;Dir-Block/Sektor. 6882
;---                                              6883
;                                              6884
;          jsr    Z2651                                ;Dir lesen/filtern. 6885
;          txa                                          6886
;          bne    :2                                ; => Fehler, Abbruch. 6887
;                                              6888
::2          jsr    DoneWithIO                        6889
;                                              6890
;          cpx    #TDLIST_RELOAD                    ;Neustart Dateisuche. 6891
;          bne    :3                                ; => Nein, weiter... 6892
;                                              6893
; Hinweis:                                          6894
; Workaround für SD2IEC wenn bei einer            6895
; schreibgeschützten Diskette eine der            6896
; Temporär-Dateien gelöscht wurde:                6897
; In dem Fall wird ein Fehler $20 vom             6898
; Laufwerk gemeldet, der Fehler wird              6899
; erst durch ExitTurbo gelöscht.                  6900
; Wird beim Neustart der Suche dann               6901
; wieder aktiviert.                               6902
;          jsr    PurgeTurbo                        ;TurboDOS entfernen. 6903
;          jsr    ExitTurbo                        ;TurboDOS aus.      6904
;                                              6905
;          jmp    Z260D                                ; => Dateisuche... 6906
;                                              6907
::3          rts                                      6908
;                                              6909
; Verzeichniskette einlesen/filtern.              6910
:Z2651      jsr    Z27EF                                ;Dir-Block einlesen. 6911
;          txa                                          6912
;          bne    :1                                6913
;                                              6914
;          jsr    Z266C                                ;Einträge anpassen. 6915
;          jsr    Z26E4                                ;Dateien filtern. 6916
;          txa                                          ;Fehler? 6917
;          bne    :1                                ; => Ja, Abbruch... 6918
;                                              6919
;          lda    V266A                                ;Weiterer Dir-Block? 6920
;          bne    Z2651                                ; => Ja, weiter... 6921
;                                              6922
;          ldx    #NO_ERROR                        6923
::1          rts                                      6924
;                                              6925
:V2666      b $00,$00 ;Anzahl Dateien einlesen. 6926
:V2667      b $00 ;$FF = Temp. nicht löschen. 6927
:V2668      w $0000 ;Zeiger Speicher Dateieinträge. 6928
:V266A      b $00 ;Verzeichnisblock/Track 6929
:V266B      b $00 ;Verzeichnisblock/Sektor 6930
;                                              6931
; Dateieinträge anpassen. 6932
; TD13: SortDirFiles 6933

```

```

6934 :Z266C      lda    diskBlkBuf +32      ;Ordner-Nr. für die
6935           sta    diskBlkBuf +1        ;erste Datei holen.
6936
6937           lda    #< diskBlkBuf
6938           sta    r0L
6939           lda    #> diskBlkBuf
6940           sta    r0H                    ;Zeiger Eintrag.
6941
6942           lda    #< diskBlkBuf +1
6943           sta    r5L
6944           lda    #> diskBlkBuf +1
6945           sta    r5H                    ;Zeiger Ordner-Nr.
6946
6947 ; Hinweis:
6948 ; Hier werden die Dateieinträge im
6949 ; Verzeichnisblock umkopiert:
6950 ; Alle Bytes ab Offset #1 (Ordner-Nr.)
6951 ; nach Offset #0 kopieren.
6952           ldx    #8 -1                  ;Einträge/Sektor.
6953
6954 ::1          ldy    #31
6955           jsr    Z280C                  ;Eintrag einlesen.
6956
6957           ldy    #1
6958           jsr    Z26CD                  ;DIR=Kein Infoblock.
6959
6960           lda    #32
6961           clc
6962           adc    r0L
6963           sta    r0L
6964           bcc    :2
6965           inc    r0H
6966
6967 ::2          lda    #32
6968           clc
6969           adc    r5L
6970           sta    r5L
6971           bcc    :3
6972           inc    r5H
6973
6974 ::3          dex
6975           bpl    :1
6976           rts
6977
6978 ; Auf Verzeichnis prüfen.
6979 ; -> "Kein Infoblock" setzen.
6980 :Z26CD      lda    (r0),y
6981           and    #FTYPE_MODES
6982           cmp    #FTYPE_DIR            ;Verzeichnis?
6983           bne    :1                    ; => Nein, Ende...
6984
6985           tya                          ;Zeiger auf Sektor
6986           clc                          ;für Infoblock.
6987           adc    #1+2+16 +1            ;Typ/Tr+Se/Name +1.
6988           tay

```

```

        lda    #$ff                ;Durch $00,$ff      6989
        sta    (r0),y              ;ersetzen.          6990
        dey                                     6991
        lda    #$00                6992
        sta    (r0),y              6993
                                     6994
::1      rts                        6995
                                     6996
; Dateien aus Liste filtern.          6997
; TD13: SortInBuffer                  6998
:Z26E4   lda    #< diskBlkBuf        6999
        sta    r5L                  7000
        lda    #> diskBlkBuf        7001
        sta    r5H                  7002
                                     7003
        ldx    #8 -1                ;Einträge/Sektor.  7004
                                     7005
:Z26EE   lda    r3L                  7006
        sta    r0L                  7007
        lda    r3H                  7008
        sta    r0H                  7009
                                     7010
; Hinweis:                           7011
; Pufferüberlauf nur bei sortierter  7012
; Text-Anzeige der Dateien möglich.  7013
; Im Icon-/Text-Modus werden nur die 7014
; ersten 16 Dateien eingelesen, bei  7015
; sortiertem Text-Modus 254 Dateien.  7016
        cmp    MHB04AC              ;HB Ende CopyBuf.  7017
        bcc    :1                    7018
                                     7019
        ldx    #BFR_OVERFLOW         7020
        rts                          7021
                                     7022
::1      bit    r10L                 ;Alle Ordner?      7023
        bmi    :6                    ; => Ja, weiter... 7024
                                     7025
        ldy    #1                    7026
        lda    (r5),y                ;Gelöscht?        7027
        beq    :5                    ; => Ja, weiter... 7028
                                     7029
        ldy    #$17                  7030
        lda    (r5),y                7031
        cmp    #TEMPORARY            ;Temp. Datei?      7032
        bne    :5                    ; => Nein, weiter... 7033
                                     7034
; TEMPORARY-Datei löschen.            7035
        bit    V2667                 ;Löschen möglich?  7036
        bmi    :5                    ; => Nein, weiter... 7037
                                     7038
        ldy    #4                    ;Name temp.Datei    7039
::2      lda    (r5),y                ;einlesen.        7040
        sta    V0470 -4,y            7041
        cmp    #$a0                  7042
        beq    :3                    7043

```

```

7044          iny
7045          cpy    #4 +16          ;Dateiname kopiert?
7046          bne    :2              ; => Nein, weiter...
7047 ::3          lda    #NULL        ;Ende kennzeichnen.
7048          sta    V0470 -4,y
7049
7050          jsr    DoneWithIO
7051
7052          jsr    S0B30            ;Datei löschen.
7053 ;          txa
7054          beq    :4
7055
7056          lda    #$ff            ;Löschen von Temp-
7057          sta    V2667            ;Datei nicht möglich.
7058
7059 ::4          jsr    InitForIO
7060
7061          ldx    #TDLIST_RELOAD   ;Neustart Dateisuche.
7062          rts
7063
7064 ; Hinweis: Bei gelöschten Dateien ist
7065 ; die Ordner-Nr. = $00!
7066 ::5          ldy    #0
7067          lda    (r5),y
7068          cmp    r10L            ;Stimmt Ordner-Nr.?
7069          bne    :next           ; => Nein, weiter..
7070
7071 ::6          bit    r12H         ;Alle Dateitypen?
7072          bmi    :7              ; => Ja, weiter...
7073
7074          ldy    #23
7075          lda    (r5),y          ;Datei vom gesuchten
7076          cmp    r12H            ;GEOS-Dateityp?
7077          bne    :next           ; => Nein, weiter...
7078
7079 ; *TDFOLDERFIX*
7080 ;
7081 ; mod.#09/MakeDirList sucht alle Ordner
7082 ; im Verzeichnis. Die Laufwerkstreiber
7083 ; von GDOS64 sind vom Typ#11=TDFOLDER
7084 ; und werden ohne den Fix ebenfalls in
7085 ; die Liste mit Ordner-Infoblöcken für
7086 ; die Ordner-Suche aufgenommen.
7087 ; Würde im Infoblock des Treibers ab
7088 ; Ordner-Offset ein Wert <>0 stehen,
7089 ; dann würde dieser Wert als Ordner-Nr.
7090 ; interpretiert und damit wäre diese
7091 ; Ordner-Nr. dann "belegt".
7092 ; Für die GDOS64-Laufwerkstreiber ist
7093 ; das (aktuell) kein Problem, da hier
7094 ; $00 steht und ein Ordner #0 nicht
7095 ; erstellt werden kann (=ROOT-Dir).
7096 ;---
7097          cmp    #TDFOLDER        ;Ordner suchen?
7098          bne    :7              ; => Nein, weiter...

```

```

        ldy    #2                ;Zeiger auf Tr/Se.          7099
        lda    (r5),y            ;Tr/Se= $00/$ff?          7100
        bne    :next            ; => Dat. ignorieren.      7101
; ---
                                           7102
                                           7103
::7          ldy    #1                7104
        lda    (r5),y            ;Datei gelöscht?          7105
        beq    :8                ; => Ja, weiter...        7106
                                           7107
        bit    r12L              7108
        bpl    :10              7109
        bvc    :next            7110
        bvs    :10                ; => Nein, weiter...      7111
                                           7112
; Gelöschte Dateien.                    7113
::8          ldy    #23              7114
        lda    (r5),y            7115
        cmp    #TEMPORARY        ;Temp.-Datei?          7116
        beq    :next            ; => Dat. ignorieren.      7117
                                           7118
        bit    r12L              7119
        bpl    :10                ; => Alle Einträge.      7120
        bvs    :next            ; => Ohne gelöschte.      7121
                                           7122
; r12L=%10xxxxxx                      7123
        ldy    #4                7124
        lda    (r5),y            ;Dateiname vorhanden?    7125
        beq    :next            ; => Dat. ignorieren.      7126
        bne    :10              ; => Ja, weiter...        7127
;                                           7128
;                                           7129
:::9          lda    V046D          ;Nur gelöschte Dat.?    7129
        bne    :next            ; => Dat. ignorieren.      7130
                                           7131
:::10         lda    r11H           ;Einträge überlesen?    7132
        beq    :11              ; => Nein, weiter...        7133
                                           7134
        dec    r11H              ;Überlesen -1.          7135
        jmp    :next            ; => Dat. ignorieren.      7136
                                           7137
:::11         lda    r13H           ;Zeiger auf            7138
        clc                      ;Dateiname.              7139
        adc    r5L                7140
        sta    r5L                7141
        bcc    :11a              7142
        inc    r5H                7143
                                           7144
; Hinweis:                            7145
; Die Original-Routine liest mehr als  7146
; 255 Dateien obwohl Offset + Anzahl    7147
; nicht >255 sein dürfen (Überlauf).    7148
:::11a        lda    r14L           ;Dateien ab Offset.    7149
        clc                      7150
        adc    r15H              ;Offset.                  7151
        cmp    #MAXFILES         ;Max. 254 Dateien?      7152
        bne    :12              ; => Nein, weiter...      7153

```

```

7154
7155             sta    r15L                ;Flag: BFR_OVERFLOW
7156             beq    :14                ; => Puffer voll.
7157
7158 ::12         inc    r14L                ;Zähler +1.
7159             bne    :13                ;Nur 8Bit da sonst
7160             ;      inc    r14H          ;Überlauf möglich!
7161
7162 ::13         ldy    r11L                ;Max. Anzahl Dateien?
7163             beq    :14                ; => Ja, weiter...
7164
7165             ldy    r13L
7166             jsr    Z280C                ;Eintrag einlesen.
7167
7168             dec    r11L                ;Dateien -1.
7169
7170             lda    r13L                ;Zeiger auf den
7171             clc                        ;nächsten Dateinamen.
7172             adc    r3L
7173             sta    r3L
7174             bcc    :14
7175             inc    r3H
7176
7177 ::14         lda    r5L                ;Zeiger auf Eintrag
7178             sec                        ;zurücksetzen.
7179             sbc    r13H
7180             sta    r5L
7181             bcs    :next
7182             dec    r5H
7183
7184 ::next       lda    #< 32              ;Zeiger auf
7185             clc                        ;nächsten Eintrag.
7186             adc    r5L
7187             sta    r5L
7188             lda    #> 32
7189             adc    r5H
7190             sta    r5H
7191
7192             dex                        ;Alle geprüft?
7193             bmi    :done                ; => Ja, Ende...
7194
7195             jmp    Z26EE                ; => Weitersuchen...
7196
7197 ::done       ldx    #NO_ERROR
7198             rts
7199
7200 ; Zeiger auf ersten Verzeichnisblock
7201 ; und I/O aktivieren.
7202 ; Rückgabe: X = Fehler (I/O ist aktiv!)
7203 ;      r4 = diskBlkBuf
7204 ;      V266A/B = Tr/Se Block.
7205 ; TD13: DiskInitTurboIO
7206 :Z27CA       lda    #< diskBlkBuf
7207             sta    r4L
7208             lda    #> diskBlkBuf

```

```

        sta    r4H                                7209
                                                7210
        jsr    EnterTurbo                        7211
        txa                                         7212
        bne    :exit                             7213
                                                7214
        lda    curDirHead +1                     7215
        sta    V266B                             ;Dir-Block/Sektor. 7216
        lda    curDirHead +0                     7217
        sta    V266A                             ;Dir-Block/Track. 7218
        bne    :exit                             ; => Gültig, OK... 7219
                                                7220
        ldx    #INV_TRACK                        7221
::exit    jmp    InitForIO                       ;I/O aktivieren. 7222
                                                7223
; Verzeichnisblock einlesen.                    7224
; Übergabe: V266A = Tr/Se Block.                7225
; Rückgabe: V266A = Tr/Se nächster Block.      7226
; TD13: ReadTrScBlock                          7227
:Z27EF    lda    V266A                             ;Dir-Block/Track. 7228
        sta    r1L                                7229
        lda    V266B                             ;Dir-Block/Sektor. 7230
        sta    r1H                                7231
        jsr    ReadBlock                        7232
        txa                                         7233
        bne    :1                                7234
                                                7235
        lda    diskBlkBuf +0                     7236
        sta    V266A                             ;Dir-Block/Track. 7237
        lda    diskBlkBuf +1                     7238
        sta    V266B                             ;Dir-Block/Sektor. 7239
                                                7240
::1        rts                                    7241
                                                7242
; Dateieintrag kopieren.                      7243
; Übergabe: Y = Anzahl Bytes.                  7244
;         r5 = Source                          7245
;         r0 = Target                          7246
; TD13: CopyFW                                7247
:Z280C    sty    :next +1                        7248
                                                7249
        ldy    #0                                7250
::1        lda    (r5),y                          7251
        sta    (r0),y                          7252
        iny                                         7253
::next    cpy    #$00                             ;Wird verändert. 7254
        bne    :1                                7255
                                                7256
        rts                                    7257
                                                7258
; Datei in Ordner verschieben.                7259
; Übergabe: r6 = Zeiger auf Dateiname.        7260
;         r10L = Nummer Ziel-Ordner.          7261
; Rückgabe: r1 = Tr/Se Verzeichnisblock.      7262
;         r5 = Zeiger auf Verzeichniseintrag. 7263

```

```

7264 ; (CBM-Dateityp, nicht TD-Ordner!)
7265 ; TD13: MoveFileInDir
7266 .S2832      jsr    S1412      ;Datei suchen.
7267           txa
7268           beq     :1
7269           rts
7270
7271 ::1          ldy     r5L
7272           cpy     #2           ;Eintrag #0?
7273           bne     :2           ; => Nein, weiter...
7274           ldy     #32 +1       ;Sonderbehandlung.
7275 ::2          dey           ;Zeiger Ordner-Nr.
7276
7277 ; Der Verzeichnisblock liegt in
7278 ; diskBlkBuf. Der Zugriff über r5 ist
7279 ; daher nicht erforderlich.
7280           lda     r10L         ;Ordner-Nr.
7281 ;---
7282           dey
7283           dey
7284           sta     (r5),y
7285 ;---
7286           sta     diskBlkBuf,y ;schreiben.
7287 ;---
7288
7289           jmp     PutBlock_dskBuf
7290
7291 ; Neuen Ordner in Tabelle eintragen.
7292 ; Übergabe: r0 = Zeiger auf Ordnerliste
7293 ;           A/X = Zeiger auf Dateiname
7294 ; Rückgabe: r10L = Ordner-Nr (bei Fehler undefiniert)
7295 ;           Y = Schachteltiefe.
7296 ;           X = 0/Kein Fehler.
7297 ; TD13: SubDirNrInList
7298 :Z286D      jsr     Z28BE      ;Infoblock einlesen.
7299           txa
7300           bne     :2           ; => Fehler, Abbruch.
7301
7302           lda     diskBlkBuf +OFF_DIR_NUM
7303           sta     r10L
7304
7305           ldy     #$00
7306 ::1          lda     (r0),y     ;Freier Eintrag?
7307           bmi     :4           ; => Ja, weiter...
7308           cmp     r10L         ;Ordner vorhanden?
7309           bne     :3           ; => Nein, weiter...
7310
7311           ldx     #TDSUB_EXISTS ;Fehler: REKURSIV
7312 ::2          rts
7313
7314 ::3          iny
7315           cpy     #MAXDIRSIZE -1
7316           bne     :1
7317
7318           ldx     #TDLIST_FULL ;Fehler: FULL

```



	rts	7319
		7320
::4	lda r10L	7321
	sta (r0),y	7322
	rts	7323
		7324
	; Zum vorherigen Ordner wechseln.	7325
	; Übergabe: r0 = Zeiger auf Ordner-Liste.	7326
	; Rückgabe: r10L = Neue Ordner-Nr.	7327
	; TD13: UpperDir	7328
:Z288D	ldy #MAXDIRSIZE -1	7329
::1	lda (r0),y	7330
	bpl :2	7331
	dey	7332
	bne :1	7333
		7334
	lda #\$00	7335
	sta (r0),y	7336
	beq :3	7337
		7338
::2	lda #\$ff	7339
	sta (r0),y	7340
	dey	7341
	lda (r0),y	7342
::3	sta r10L	7343
	rts	7344
		7345
	; Ordner-Liste löschen.	7346
	; Übergabe: r0 = Zeiger auf Ordner-Liste.	7347
	; TD13: ClearList	7348
.S28A7	ldy #MAXDIRSIZE -1	7349
	lda #\$ff	7350
::1	sta (r0),y	7351
	dey	7352
	bne :1	7353
		7354
	; lda #\$00	7355
	tya	7356
	sta (r0),y	7357
	rts	7358
		7359
	; Datei suchen/Infoblock einlesen.	7360
	; Übergabe: A/X = Zeiger auf Dateiname.	7361
	; TD13: getInfoBlock	7362
	; Hinweis:	7363
	; Aufruf nur von Z28B5!	7364
:Z28BE	jsr S1414 ;Datei suchen.	7365
	txa	7366
	bne x28DA	7367
		7368
	; Infoblock einlesen.	7369
	; TD13: GetInfo	7370
:Z28C4	ldy #19	7371
	lda (r5),y	7372
	sta r1L	7373

```

7374             iny
7375             lda    (r5),y
7376             sta    r1H
7377             jsr    GetBlock_dskBuf
7378 :x28DA             rts
7379
7380 ; Max. Anzahl Laufwerke setzen.
7381 ;
7382 ; TD13: SetNumDrives
7383 ;
7384 ; TODO:
7385 ; Wozu ist das noch erforderlich?
7386 .S28DB             jsr    Z28E2             ;Anz. Lfwk. holen.
7387             sta    numDrives
7388             rts
7389
7390 ; Anzahl Laufwerke ermitteln.
7391 ; TD13: GetMaxDrives
7392 :Z28E2             ldy    numDrives
7393             cpy    #$02
7394             bcs    :2
7395
7396             lda    curDrive             ;Nur auf Laufwerk
7397             eor    #$01             ;A/B oder C/D suchen.
7398             tax
7399             lda    driveType -8,x
7400             beq    :1
7401             iny
7402 ::1               tya
7403             rts
7404
7405 ::2               ldy    #$02
7406             bit    sysRAMFlg
7407             bvs    :4
7408
7409             lda    driveType
7410             cmp    driveType +2
7411             bne    :3
7412             iny
7413             cmp    driveType +3
7414             bne    :3
7415             iny
7416 ::3               tya
7417             rts
7418
7419 ::4               lda    driveType +2
7420             beq    :3
7421             iny
7422             lda    driveType +3
7423             beq    :3
7424             iny
7425             bne    :3
7426             rts
7427
7428 ; Max. Anzahl Laufwerk setzen.

```



```

7484          bcc      :3
7485
7486          ldx      #ERR_UNKNOWN      ;Fehler: Unbekannt.
7487 ::2          rts
7488
7489 ::3          bit    V25CF            ;Drucken?
7490          bpl      :4                ; => Nein, weiter...
7491
7492          cpy      #APPL_DATA        ;Dokument?
7493          beq      :4                ; => Ja, weiter...
7494
7495          ldx      #ERR_NO_DOC        ;Fehler: kein Dok.
7496          rts
7497
7498 ::4          lda    V298D,y          ;Lowbyte.
7499          ldx      V297D,y          ;Highbyte.
7500          jmp      CallRoutine
7501
7502 ; TD13: JumpTab
7503 :V297D      b >R299D ;Nicht-GEOS
7504          b >R299D ;BASIC
7505          b >R2A1C ;Assembler
7506          b >R29BF ;Datenfile
7507          b >R29BF ;Systemdatei
7508          b >R2A19 ;DeskAccessory
7509          b >R2A1C ;Anwendung
7510          b >R2A3E ;Dokument
7511          b >R29BF ;Zeichensatz
7512          b >R3A06 ;Druckertreiber
7513          b >R3A06 ;Eingabetreiber
7514          b >R2A22 ;Laufwerkstreiber/Ordner
7515          b >R29BF ;Startprogramm
7516          b >R29BF ;Temporär
7517          b >R2A1C ;Autostart
7518          b >R29BF ;Eingabetreiber 128
7519
7520 :V298D      b <R299D ;Nicht-GEOS
7521          b <R299D ;BASIC
7522          b <R2A1C ;Assembler
7523          b <R29BF ;Datenfile
7524          b <R29BF ;Systemdatei
7525          b <R2A19 ;DeskAccessory
7526          b <R2A1C ;Anwendung
7527          b <R2A3E ;Dokument
7528          b <R29BF ;Zeichensatz
7529          b <R3A06 ;Druckertreiber
7530          b <R3A06 ;Eingabetreiber
7531          b <R2A22 ;Laufwerkstreiber/Ordner
7532          b <R29BF ;Startprogramm
7533          b <R29BF ;Temporär
7534          b <R2A1C ;Autostart
7535          b <R29BF ;Eingabetreiber 128
7536
7537 ; Nicht-GEOS/BASIC laden.
7538 :R299D      ldy     #4 -1

```

```

::1          lda    :com,y          7539
              sta    SCREEN_BASE,y  7540
              dey    7541
              bpl    :1              7542
              7543
              lda    #< SCREEN_BASE 7544
              sta    r0L              7545
              lda    #> SCREEN_BASE  7546
              sta    r0H              7547
              7548
              lda    #< $0801         7549
              sta    r7L              7550
              lda    #> $0801         7551
              sta    r7H              7552
              jmp    ToBasic           7553
              7554
::com        b "RUN",NULL            7555
              7556
; Datei kann nicht geöffnet werden.  7557
; TD13: XDispErr                      7558
:R29BF       lda    #< T29D0          7559
              ldx    #> T29D0          7560
:R29C0       sta    r6L              7561
              stx    r6H              7562
              7563
              jsr    Z13D3             ;Textgrenzen Desktop. 7564
              7565
              lda    #< D29CD          7566
              ldx    #> D29CD          7567
              jmp    S2090             ;DBox ohne Recover. 7568
              7569
; TD13: XErrorDial                    7570
:D29CD       b %10000001              7571
              7572
              b OK      , $11, $4c     7573
              7574
              b DBTXTSTR, $08, $10     7575
              w T29CF                  7576
              7577
              b DBVARSTR, $08, $20     7578
              b r6L                    7579
              7580
              b DBTXTSTR, $08, $2c     7581
              w T29D3                  7582
              b DBVARSTR, $12, $2c     7583
              b r15L                   7584
              7585
              b NULL                   7586
              7587
; Hinweis:                            7588
; Wird in mod.09/10/12 verwendet.    7589
if LANG = LANG_DE                      7590
.T29CF       b BOLDON, "Achtung!", PLAINTEXT, 0 7591
endif                                              7592
if LANG = LANG_EN                      7593

```

```

7594 .T29CF          b BOLDON,"Warning!",PLAINTEXT,0
7595 endif
7596
7597 if LANG = LANG_DE
7598 :T29D0          b "Datei kann nicht geöffnet werden.",0
7599 endif
7600 if LANG = LANG_EN
7601 :T29D0          b "The file cannot be opened.",0
7602 endif
7603
7604 ; Hinweis:
7605 ; Wird auch in mod.#01 verwendet.
7606 .T29D3          b BOLDON
7607                b "?"
7608                b PLAINTEXT,0
7609
7610 ; Hinweis:
7611 ; Wird auch in mod.#02/04/08/09/12 verwendet.
7612 .T29D5          b BOLDON
7613                b ">"
7614                b PLAINTEXT,0
7615
7616 ; Assembler/Anwendung/Autostart laden.
7617 ; TD13: XAssembly/XAutoExec/XAppl
7618 ;
7619 ; TODO:
7620 ; Wozu ist das noch erforderlich?
7621 ;
7622 ; Anwendungen die numDrives auswerten
7623 ; werden damit auf max. zwei Laufwerke
7624 ; reduziert.
7625 ;
7626 :R2A1C          jsr    Z291E          ;Anzahl Lfwk. setzen.
7627
7628 ; DeskAccessory laden.
7629 ; TD13: XDAcc
7630 :R2A19          jmp    Z2B24          ;Datei laden.
7631
7632 ; Ordner öffnen.
7633 :R2A22          lda    dirEntryBuf +$15
7634                beq    :ok
7635
7636                jmp    R29BF          ; => Kein Ordner.
7637
7638 ::ok            lda    r15L
7639                sta    r6L
7640                ldx    r15H
7641                stx    r6H
7642                jsr    Z286D          ;Ordner in Liste.
7643                txa
7644                bne    :1
7645
7646                ldx    #ERR_TDFOLDER ;Fehler: SUB_DIR.
7647 ::1             rts
7648

```

```

; Dokument laden. 7649
; Übergabe: r5 = Zeiger auf dirEntryBuf 7650
; TD13: XDocument 7651
:R2A3E      jsr    Z28C4      ;Infoblock einlesen. 7652
7653
; GEOS-Klasse testen. 7654
      lda    diskBlkBuf +$75 7655
      bne    :ok      ; => OK, weiter... 7656
      jmp    R29BF      ;Fehler:Dat.ungültig. 7657
7658
::ok      lda    curDrive      ;Laufwerk für 7659
      sta    V25CE      ;Dokument speichern. 7660
7661
      ldx    #r0L      ;Diskname kopieren. 7662
      jsr    GetPtrCurDkNm 7663
7664
; Diskname kopieren. 7665
if TRUE 7666
      ldy    #18 -1      ;Diskname in 7667
::1      lda    (r0),y      ;Zwischenspeicher. 7668
      sta    V25D1,y 7669
      dey 7670
      bpl    :1 7671
else 7672
      lda    #< V25D1      ;Zwischenspeicher 7673
      sta    r1L      ;für Diskname. 7674
      lda    #> V25D1 7675
      sta    r1H 7676
7677
      lda    #< 18      ;Länge Diskname. 7678
      sta    r2L 7679
      lda    #> 18 7680
      sta    r2H 7681
7682
      jsr    MoveData 7683
endif 7684
7685
      jsr    i_MoveData 7686
      w diskBlkBuf +$75      ;GEOS-Klasse. 7687
      w V25E3 7688
;      w 13 ;13 Zeichen ??? 7689
      w 12 ;Standard = 12Z. 7690
7691
; Suche nach Application für das 7692
; Dokument zuerst auf RAMDisk suchen. 7693
      ldy    #8 7694
::2      lda    driveType -8,y      ;RAMDisk? 7695
      bpl    :3      ; => Nein, weiter... 7696
7697
      tya 7698
      pha 7699
      jsr    Z2B03      ;Datei/Klasse suchen. 7700
      pla 7701
      tay 7702
7703

```

```

7704          txa          ;Appl. gefunden?
7705          beq    Z2AC0    ; => Ja, öffnen...
7706
7707 ::3          iny
7708          cpy    #12          ;Nächstes Laufwerk?
7709          bne    :2          ; => weitersuchen...
7710
7711 ; Appl. auf Laufwerk mit Dokument suchen.
7712 ::5          lda    V25CE          ;Laufwerk Dokument.
7713          jsr    Z2B03          ;Lfwk öffnen und
7714          ;Datei/Klasse suchen.
7715          txa          ;Appl. gefunden?
7716          beq    Z2AC0    ; => Ja, weiter...
7717
7718 ; Appl./Klasse auf Diskette suchen.
7719          ldy    #8
7720 ::6          lda    driveType -8,y
7721          beq    :7          ; => Kein Laufwerk.
7722          bmi    :7          ; => RAMDisk.
7723          cpy    V25CE          ;Laufwerk Dokument?
7724          beq    :7
7725
7726          tya
7727          pha
7728          jsr    Z2B03          ;Datei/Klasse suchen.
7729          pla
7730          tay
7731
7732          txa          ;Appl. Gefunden?
7733          beq    Z2AC0    ; => Ja, öffnen.
7734
7735 ::7          iny
7736          cpy    #12
7737          bne    :6
7738
7739          ldx    #ERR_NO_APPL
7740          jsr    S5CD0          ;Fehler A-D ausgeben.
7741
7742          lda    V25CE          ;Startlaufwerk
7743          jmp    SetDevice      ;zurücksetzen.
7744
7745 ; Fortsetzung Dokument laden.
7746 ; Übergabe: V25F1 = Application-Name.
7747 ; TD13: XApplLoad
7748 :Z2AC0      lda    #< V25F1          ;Zeiger auf Name
7749          ;          sta    r6L          ;Application.
7750          ldx    #> V25F1
7751          ;          stx    r6H
7752          ;          jsr    FindFile      ;Appl. suchen.
7753          jsr    S1414          ;FindFile.
7754          txa
7755          bne    :err          ; => Fehler...
7756
7757 ; Infoblock zur Application-Datei in
7758 ; ":dirEntryBuf" einlesen.

```



	jsr	Z2B5C	;Nur C128/80Z-Modus?	7759
	txa			7760
	beq	:1	; => Nein, weiter...	7761
::err	rts			7762
				7763
::1	bit	V25CF	;Drucken?	7764
	bpl	:2	; => Nein, weiter...	7765
				7766
	lda	##11000000	;Dok. laden/drucken.	7767
	b	\$2c		7768
::2	lda	##10000000	;Dokument laden.	7769
	sta	r0L		7770
				7771
	lda	#< V25D1	;Name Diskette.	7772
	sta	r2L		7773
	lda	#> V25D1		7774
	sta	r2H		7775
				7776
	lda	r15L	;Name Dokument.	7777
	sta	r3L		7778
	lda	r15H		7779
	sta	r3H		7780
				7781
	lda	#< V25F1	;Zeiger auf Name	7782
	sta	r6L	;Application.	7783
	lda	#> V25F1	;FindFile verändert	7784
	sta	r6H	;r6 => neu setzen!	7785
				7786
	jsr	Z37DC	;TopDesk.win -> REU.	7787
				7788
; TODO:				7789
; Wozu ist das noch erforderlich?				7790
;				7791
; Anwendungen die numDrives auswerten				7792
; werden damit auf max. zwei Laufwerke				7793
; reduziert.				7794
;				7795
	jsr	Z291E	;Anzahl Lfwk. setzen.	7796
				7797
	jmp	Z2B4A	;Datei laden.	7798
				7799
; Datei über GEOS-Klasse suchen.				7800
; Übergabe: A = Laufwerk.				7801
:Z2B03	jsr	SetDevice		7802
	jsr	OpenDisk		7803
	txa			7804
	bne	:exit	; => Fehler...	7805
				7806
; TD13: XSearchAppl				7807
	lda	#< V25F1	;Zwischenspeicher	7808
	sta	r6L	;für Dateiname.	7809
	lda	#> V25F1		7810
	sta	r6H		7811
				7812
	lda	#< V25E3	;GEOS-Klasse.	7813

```

7814          sta    r10L
7815          lda    #> V25E3
7816          sta    r10H
7817
7818          lda    #1
7819          sta    r7H
7820          lda    #APPLICATION
7821          sta    r7L
7822
7823          jsr    FindFTypes
7824          txa
7825          bne    :exit          ; => Fehler...
7826          lda    r7H
7827          beq    :exit
7828          ldx    #FILE_NOT_FOUND
7829 ::exit      rts
7830
7831 ; DeskAccessory laden.
7832 ; Assembler/Anwendung/Autostart laden.
7833 ;
7834 ; TODO:
7835 ; Für TD64 unter GEOS128 existiert hier
7836 ; kein Wechsel des Grafikmodus!
7837 ; TD13: DA0
7838 :Z2B24      jsr    Z2B5C          ;Nur C128/80Z-Modus?
7839          txa
7840          beq    :1              ; => Nein, weiter...
7841          rts
7842
7843 ::1         jsr    Z37DC          ;TopDesk.win -> REU.
7844
7845          lda    r15L
7846          sta    r6L
7847          lda    r15H
7848          sta    r6H
7849
7850          lda    #$00
7851          sta    r0L
7852          sta    r10L
7853
7854          lda    V25D0          ;Dateityp für öffnen.
7855          cmp    #DESK_ACC      ;DeskAccessory?
7856          bne    :2              ; => nein, weiter...
7857
7858          jmp    Z3B8F          ;DeskAccessory laden.
7859
7860 ::2         lda    #%00000000    ;Nur Anwendung laden.
7861          sta    r0L
7862
7863 ; Bildschirm löschen und Datei laden.
7864 :Z2B4A      lda    #> EnterDeskTop -1
7865          pha
7866          lda    #< EnterDeskTop -1
7867          pha
7868

```

```

                                ldx    #(r10L - r0L)                7869
::1                            lda     r0,x                      7870
                                pha                                7871
                                dex                                7872
                                bpl     :1                        7873
                                7874
                                jsr     ResetScreen              ;Bildschirm löschen. 7875
                                7876
                                ldx     #0                      7877
::2                            pla                                7878
                                sta     r0,x                    7879
                                inx                                7880
                                cpx     #(r10L - r0L) +1        7881
                                bne     :2                      7882
                                7883
                                jmp      GetFile                 ;Datei laden.        7884
                                7885
; Auf C128/80Z-Modus testen. 7886
:Z2B5C                        lda     #< dirEntryBuf           7887
                                sta     r9L                    7888
                                lda     #> dirEntryBuf           7889
                                sta     r9H                    7890
                                jsr     GetFHdrInfo              7891
                                7892
                                lda     fileHeader +96          7893
                                cmp     #%11000000             ;Nur 80Z-Modus?      7894
                                bne     :1                      7895
                                ldx     #BAD_SCRN_RES            ;Falscher Modus.    7896
                                rts                                7897
                                7898
::1                            ldx     #NO_ERROR                7899
                                rts                                7900
                                7901
.CORE_END                    7902
                                7903
; Hinweis:                7904
; Ab hier liegen die Module zur Anzeige 7905
; von Icons oder Text der Fenster.    7906
; Der Bereich kann nicht am Ende des 7907
; Hauptmodules wie bei anderen VLIR- 7908
; Modulen liegen, da im Text-Modus der 7909
; Speicher zum einlesen der Dateien   7910
; genutzt wird um diese dann ggf. zu 7911
; sortieren.                7912
; Im Icon-Modus werden nur Name, der 7913
; Tr/Se des Infoblock und das Icon in 7914
; einen der beiden DIRBUF-Speicher    7915
; eingelesen.                7916
;                                7917
; BASE_MOD_B                7918
;                                7919
; --- mod.#05/#06          7920
.A2BA6                        ;Init, Inhalt anzeigen.          7921
.A2BA9 = A2BA6 +3            ;Inhalt anzeigen.                7922
.A2BAC = A2BA6 +6            ;Init, Mausklick auswerten.      7923

```

```
7924 .A2BAF = A2BA6 +9      ;Mausklick auswerten.
7925 .A2BB2 = A2BA6 +12     ;Rechteck für Eintrag ermitteln.
7926 .A2BB5 = A2BA6 +15     ;Anz. Einträge/Seite berechnen.
7927 .A2BB8 = A2BA6 +18     ;Zeiger auf DnD-Icon.
7928 ;--- mod.#06
7929 .A2BBB = A2BA6 +21     ;Dateien sortieren.
7930 .A2BBE = A2BA6 +24     ;mark.Datei suchen.
7931
7932          s SIZE_FSERVICE
7933
7934 .V4CC3          s 4      ;Anzahl Zeilen.
7935 .V4CC7          s 4      ;Anzahl Spalten.
7936 .V4CCB          s 4      ;Anzahl Icons/Seite od. Zeilen/Fens.
7937
```

```

; 7938
; 7939
; SourceCode: inc.mod#00b.s 7940
; 7941
; 7942
; Hinweis: 7943
; Modul#11 (Partitionswechsel) legt 7944
; ab hier Daten zu den Partitionen ab. 7945
; Das Modul darf daher keine Routinen 7946
; in diesem Bereich aufrufen! 7947
; 7948
; Auch dürfen hier keine Zwischen- 7949
; oder Parameterspeicher abgelegt 7950
; werden, da der Bereich in die REU 7951
; ausgelagert wird. 7952
; 7953
; TD13: Start 7954
;L3136 7955
.MAININIT      bit    c128Flag      ;C128? 7956
               bmi    :panic        ; => Ja, Abbruch... 7957
               7958
               lda    MP3_CODE +0    ;Auf MP3/GD testen. 7959
               cmp    #"M"           7960
               bne    :panic        7961
               lda    MP3_CODE +1    7962
               cmp    #"P"           7963
               beq    :ok             ; => OK, weiter... 7964
               7965
::panic        jsr    Panic          ;Abbruch! 7966
               7967
; Hinweis: 7968
; Nach der Rückkehr über EnterDeskTop 7969
; ist die Mausabfrage nicht aktiv, aber 7970
; der Status evtl. noch gesetzt. 7971
::ok           lda    #%10000000     ;Status löschen. 7972
               sta    mouseData      7973
               asl                    7974
;             lda    #%00000000     7975
               sta    pressFlag      7976
               7977
               lda    #< V32E0        7978
               sta    r0L             7979
               lda    #> V32E0        7980
               sta    r0H             7981
               jsr    LoadCharSet     ;Zeichensatz laden. 7982
               7983
               lda    #< BASE_MOD_A   ;Ladeadresse für 7984
               sta    V0425 +0        ;Modul. 7985
               lda    #> BASE_MOD_A   7986
               sta    V0425 +1        7987
               7988
               lda    #ST_WR_FORE     ;Nur Vordegrund- 7989
               sta    dispBufferOn    ;grafik verwenden. 7990
               7991
; Systemvorgaben übernehmen. 7992

```

```

7993 ; Nur beim ersten Start wenn noch keine
7994 ; Konfiguration gespeichert wurde!
7995         lda    C_GEOS_BACK          ;GEOS-Farben.
7996         sta    COL0439
7997         lda    C_PulldMenu          ;Menü-Farben.
7998         sta    COL0444
7999
8000 if ENABLE_ORIGCOL = FALSE
8001         lda    C_DBoxBack           ;Dialogbox-Farben.
8002         sta    COL043C
8003 endif
8004
8005         lda    C0429                 ;System in REU?
8006         bne    :init                 ; => Ja, weiter...
8007
8008 ; TopDesk von Disk gestartet.
8009 ::1      lda    curDrive             ;Systemlaufwerk =
8010         sta    C0424                 ;TopDesk/Laufwerk.
8011
8012         ldy    #0                   ;Dateiname.
8013 ::2      lda    dirEntryBuf +3,y
8014         cmp    #$a0
8015         beq    :3
8016         sta    C0400,y
8017         iny
8018         cpy    #16
8019         bne    :2
8020
8021 ::3      lda    #NULL                ;TopDesk-Dateiname
8022         sta    C0400,y               ; + NULL-Byte.
8023
8024 ; Konfiguration einlesen.
8025 ::init   jsr    Z0956                ;TD-Konfig. einlesen.
8026
8027 ; Hinweis:
8028 ; Bei RAM-TopDesk wird die Uhr evtl.
8029 ; nicht korrekt angezeigt, wenn Datum
8030 ; und Stunde/Minute unverändert sind.
8031 ; Daher Minute ungültig machen und die
8032 ; Werte für Datum+Uhrzeit werden dann
8033 ; vollständig angezeigt.
8034         lda    #$ff                 ;Minute auf $FF =
8035         sta    V61E7                 ;Uhr aktualisieren.
8036
8037         jmp    Z3A03                 ;TopDesk-Reset.
8038
8039 ; TopDesk initialisieren, Teil #2.
8040 ;
8041 ; Diese Routine muss im Hauptmodul
8042 ; bleiben, da Submodule zum einlesen
8043 ; von "TopDesk.win", RAM-TopDesk und
8044 ; der Suche nach der aktiven TD-Part.
8045 ; geladen werden.
8046 ; Aufruf nur aus mod.#07.
8047 ; TD13: Start2

```

```

.S3186          lda    #$00          ;Keine neue          8048
                sta    V04AF         ;PadColorPref.      8049
                ;                                     8050
                lda    C0429         ;System in REU?      8051
                bne    :2            ; => Ja, weiter...    8052
                lda    C0428         ;RAM-TopDesk?        8053
                beq    :1            ; => Nein, weiter...    8054
                ;                                     8055
; Aufruf VLIR-Modul!          8056
                jsr    Z31CB         ;TD-Part. ermitteln.  8057
                ;                                     8058
                lda    #$00          8059
                sta    C0428         ;Kein RAM-TopDesk.    8060
                ;                                     8061
; TODO:                      8062
; Speicherbank nicht löschen. 8063
; Wird das Programm mit RAM-TopDesk 8064
; aktiv erneut gestartet, dann findet 8065
; sich hier die Speicherbank in der 8066
; die EnterDeskTop-Routine liegt. 8067
;                               8068
                sta    C0427         ;64K-Bank RAMDesk.    8069
; Aufruf VLIR-Modul!          8070
                jsr    Z3A41         ;RAM-TopDesk ein/aus.  8071
                ;                                     8072
; TopDesk.win einlesen.      8073
                lda    C0427         ;64K-Bank RAMDesk?    8074
                beq    :1            ; => Nein, weiter...    8075
                ;                                     8076
; Hinweis:                    8077
; Hier ist V0422=mod.#07. Im Anschluss 8078
; wird mod.#10 geladen, das Modul wird 8079
; daher immer von Disk geladen.    8080
;                               8081
                lda    #$00          ;Kein VLIR-Modul mehr 8081
                sta    V0422         ;im Speicher!        8082
                ;                                     8083
; Hinweis:                    8084
; Durch das anpassen von V0429 wird die 8085
; TopDesk.win immer von Disk gelesen. 8086
                dec    C0429         ;Kein System in REU.    8087
                ;                                     8088
; Aufruf VLIR-Modul!          8089
                jsr    Z3A44         ;TopDesk.win lesen.    8090
                ;                                     8091
                inc    C0429         ;System in REU.        8092
                rts                  8093
                ;                                     8094
; TopDesk.win von Disk einlesen. 8095
::1              lda    #$00          8096
                sta    C0428         ;Kein RAM-TopDesk.    8097
                ;                                     8098
::2              lda    C0424         ;TopDesk/Laufwerk.    8099
                jsr    SetDevice      8100
                jsr    OpenDisk       8101
                ;                                     8102

```

```

8103          lda    C0429          ;System in REU?
8104          bne     :3            ; => Ja, weiter...
8105
8106 ; Aufruf VLIR-Modul!
8107          jsr     Z31CB          ;TD-Part. ermitteln.
8108
8109 ; Aufruf VLIR-Modul!
8110 ::3          jmp     Z3A44          ;TopDesk.win lesen.
8111
8112 ; TopDesk-Partition ermitteln.
8113 :Z31CB       ldx     #$00
8114             ldy     curDrive
8115             lda     V04B8 -8,y    ;Geräteinfo einlesen.
8116             beq     :1            ; => 1541/71/81
8117
8118             jsr     Z3A83          ;CMD-Part. suchen.
8119
8120             ldx     #$00
8121             lda     C04FC,x        ;Part. für Fenster.
8122 ::1          sta     C050A          ;TopDesk/Partition.
8123
8124 ; Hinweis:
8125 ; Fenster-Nr. ist hier immer 0.
8126             ldx     #$00
8127             jmp     S09B5          ;CMD-Info löschen.
8128
8129 ; Preferences/PadColorPref suchen und laden.
8130 ; Übergabe: X = Fenster-Nr.
8131 :Z31FD       lda     #$00          ;Keine neue
8132             sta     V04AF          ;PadColorPref.
8133
8134 ;             ldx     #$ff          ;Fenster-Nr.
8135             jsr     S3280          ;PadColorPref laden.
8136             jsr     S327F          ;Prefs. laden.
8137
8138 ::exit       rts
8139
8140 ; PadColorPref suchen/laden.
8141 ; Übergabe: X = Fenster-Nr.
8142 ; TD13: GetPrefs
8143 .S3280       jsr     S61E9          ;a9=Fensterfarben.
8144
8145             lda     #< F324E      ;"Pad Color Pref"
8146             ldx     #> F324E
8147             jsr     Z325D          ;Datenblock laden.
8148             bcs     :exit         ; => Fehler, Ende...
8149
8150 ::1          ldx     #$00
8151             stx     r1L
8152
8153 ::2          lda     diskBlkBuf +2,x
8154             pha
8155             and     #%00001111
8156             asl
8157             asl

```



	asl		8158
	asl		8159
	ldy	r1L	8160
	cmp	(a9),y	8161
	beq	:3	8162
	sta	(a9),y	8163
			8164
	lda	#\$01	8165
	sta	V04AF ;Neue PadColorPref.	8166
			8167
::3	pla		8168
	and	111110000	8169
	inc	r1L	8170
	ldy	r1L	8171
	cmp	(a9),y	8172
	beq	:4	8173
	sta	(a9),y	8174
			8175
	lda	#\$01	8176
	sta	V04AF ;Neue PadColorPref.	8177
			8178
::4	inc	r1L	8179
	inx		8180
	cpx	#8	8181
	bne	:2	8182
			8183
::exit	rts		8184
			8185
	; Preferences suchen/laden.		8186
	; TD13: GetPrefs		8187
.S327F	lda	< F32C5 ;"Preferences"	8188
	ldx	> F32C5	8189
	jsr	Z325D ;Datenblock laden.	8190
	bcs	:exit ; => Fehler, Ende...	8191
			8192
	ldy	#3 -1	8193
::2	lda	diskBlkBuf +2,y	8194
	sta	maxMouseSpeed,y	8195
	dey		8196
	bpl	:2	8197
			8198
	jsr	InitForIO	8199
			8200
	lda	diskBlkBuf +5	8201
	ora	diskBlkBuf +6	8202
	sta	screenColors ;GEOS-Farben.	8203
			8204
	lda	diskBlkBuf +5	8205
	sta	\$d021 ;Hintergrundfarbe.	8206
	lda	diskBlkBuf +7	8207
	sta	\$d027 ;Spritefarbe.	8208
	lda	diskBlkBuf +71	8209
	sta	\$d020 ;Rahmenfarbe.	8210
			8211
	jsr	DoneWithIO	8212

```

8213
8214         ldy    #63 -1
8215 :::3         lda    diskBlkBuf +8,y
8216         sta    mousePicData,y
8217         dey
8218         bpl     :3
8219
8220 :::exit      rts
8221
8222 ; Dateinamen.
8223 :F324E       b "Pad Color Pref",0
8224 :F32C5       b "Preferences",0
8225
8226 ; Datei suchen und ersten Block einlesen.
8227 ; => PadColorPref / Preferences
8228 ; Übergabe: A/X = Zeiger auf Dateiname.
8229 :Z325D
8230 ;           sta    r6L
8231 ;           stx    r6H
8232 ;           jsr    FindFile           ;Datei suchen.
8233 ;           jsr    S1414             ;FindFile.
8234 ;           txa
8235 ;           bne    :1                 ; => Fehler...
8236
8237 ;           jsr    S09BC             ;1.Block > r1L/r1H.
8238
8239 ; Block einlesen.
8240 ;           lda    #< diskBlkBuf     ;Zeiger auf
8241 ;           sta    r4L               ;Zwischenspeicher.
8242 ;           lda    #> diskBlkBuf
8243 ;           sta    r4H
8244 ;           jsr    GetBlock_dskBuf
8245 ;           txa
8246 ;           beq    :2
8247
8248 :::1         sec                     ;Fehler.
8249         rts
8250
8251 :::2         clc                     ;Kein Fehler.
8252         rts
8253
8254 ; Mini-Zeichensatz DE.
8255 if LANG = LANG_DE
8256 :V32E0       b $04                 ;Baseline
8257             w $0030                 ;Length of Bitstream
8258             b $06                   ;Font height
8259             w $0008                 ;Pointer index table
8260             w $00ca                 ;Pointer bitstream
8261
8262 :::indextab  b $00,$00,$03,$00,$05,$00,$09,$00
8263             b $0f,$00,$13,$00,$19,$00,$1f,$00
8264             b $21,$00,$24,$00,$27,$00,$2d,$00
8265             b $31,$00,$34,$00,$38,$00,$3a,$00
8266             b $3f,$00,$43,$00,$46,$00,$4a,$00
8267             b $4e,$00,$52,$00,$56,$00,$5a,$00

```

	b \$5e,\$00,\$62,\$00,\$66,\$00,\$68,\$00	8268
	b \$6b,\$00,\$6e,\$00,\$72,\$00,\$75,\$00	8269
	b \$79,\$00,\$7d,\$00,\$81,\$00,\$85,\$00	8270
	b \$89,\$00,\$8d,\$00,\$91,\$00,\$95,\$00	8271
	b \$9a,\$00,\$9e,\$00,\$a0,\$00,\$a3,\$00	8272
	b \$a7,\$00,\$ab,\$00,\$b1,\$00,\$b6,\$00	8273
	b \$ba,\$00,\$be,\$00,\$c2,\$00,\$c6,\$00	8274
	b \$ca,\$00,\$ce,\$00,\$d2,\$00,\$d6,\$00	8275
	b \$dc,\$00,\$e0,\$00,\$e4,\$00,\$e8,\$00	8276
	b \$ec,\$00,\$f0,\$00,\$f4,\$00,\$f8,\$00	8277
	b \$fd,\$00,\$00,\$01,\$04,\$01,\$08,\$01	8278
	b \$0b,\$01,\$0f,\$01,\$13,\$01,\$16,\$01	8279
	b \$1a,\$01,\$1e,\$01,\$20,\$01,\$23,\$01	8280
	b \$27,\$01,\$29,\$01,\$2f,\$01,\$33,\$01	8281
	b \$37,\$01,\$3b,\$01,\$3f,\$01,\$42,\$01	8282
	b \$46,\$01,\$49,\$01,\$4d,\$01,\$51,\$01	8283
	b \$57,\$01,\$5b,\$01,\$5f,\$01,\$63,\$01	8284
	b \$67,\$01,\$6b,\$01,\$6f,\$01,\$74,\$01	8285
	b \$74,\$01	8286
		8287
::line1	b \$0a,\$94,\$4c,\$90,\$94,\$a8,\$00,\$00	8288
	b \$45,\$98,\$5c,\$9c,\$88,\$00,\$03,\$19	8289
	b \$31,\$b3,\$b9,\$95,\$2a,\$88,\$a4,\$98	8290
	b \$98,\$dd,\$55,\$55,\$57,\$55,\$52,\$02	8291
	b \$04,\$02,\$04,\$11,\$28,\$80,\$00,\$00	8292
	b \$01,\$00,\$00,\$00,\$0a,\$aa,\$e0,\$00	8293
		8294
::line2	b \$0a,\$be,\$6d,\$24,\$a2,\$71,\$00,\$02	8295
	b \$ac,\$44,\$d1,\$05,\$54,\$05,\$d0,\$a2	8296
	b \$aa,\$2a,\$22,\$15,\$2a,\$8d,\$b5,\$55	8297
	b \$55,\$09,\$55,\$55,\$51,\$27,\$05,\$01	8298
	b \$36,\$26,\$48,\$d8,\$0a,\$b4,\$c4,\$c6	8299
	b \$4d,\$aa,\$aa,\$aa,\$e6,\$40,\$90,\$00	8300
		8301
::line3	b \$08,\$14,\$c2,\$1e,\$22,\$fb,\$87,\$04	8302
	b \$a4,\$89,\$59,\$88,\$8d,\$28,\$09,\$3b	8303
	b \$b2,\$2b,\$ba,\$dd,\$2c,\$8a,\$ad,\$59	8304
	b \$58,\$89,\$55,\$52,\$22,\$55,\$50,\$00	8305
	b \$55,\$4a,\$ad,\$55,\$2c,\$aa,\$aa,\$aa	8306
	b \$99,\$2a,\$aa,\$4a,\$4a,\$aa,\$e0,\$00	8307
		8308
::line4	b \$00,\$3e,\$65,\$a4,\$22,\$71,\$00,\$08	8309
	b \$a5,\$05,\$c5,\$49,\$44,\$05,\$d0,\$3a	8310
	b \$aa,\$2a,\$22,\$55,\$2a,\$88,\$a5,\$51	8311
	b \$54,\$49,\$55,\$55,\$24,\$75,\$50,\$00	8312
	b \$55,\$4a,\$c9,\$55,\$2c,\$aa,\$aa,\$aa	8313
	b \$8d,\$2a,\$aa,\$4a,\$8a,\$aa,\$90,\$00	8314
		8315
::line5	b \$08,\$14,\$c9,\$9c,\$22,\$a8,\$10,\$50	8316
	b \$45,\$d8,\$58,\$88,\$89,\$20,\$01,\$0a	8317
	b \$b1,\$b3,\$a1,\$d5,\$4a,\$e8,\$a4,\$90	8318
	b \$d5,\$88,\$c8,\$a5,\$27,\$52,\$30,\$00	8319
	b \$36,\$26,\$68,\$d5,\$2a,\$aa,\$a4,\$c6	8320
	b \$98,\$99,\$14,\$a6,\$e6,\$46,\$e0,\$00	8321
		8322

```

8323 ::line6          b $00,$00,$40,$00,$14,$00,$20,$00
8324                  b $00,$00,$00,$00,$00,$40,$00,$30
8325                  b $00,$00,$00,$00,$00,$00,$00,$00
8326                  b $40,$00,$00,$00,$00,$00,$00,$78
8327                  b $00,$00,$01,$80,$40,$00,$00,$82
8328                  b $00,$00,$00,$04,$00,$00,$80,$00
8329 endif
8330
8331 ; Mini-Zeichensatz EN.
8332 if LANG = LANG_EN
8333 :V32E0             b $04                      ;Baseline
8334                   w $0030                     ;Length of Bitstream
8335                   b $06                      ;Font height
8336                   w $0008                     ;Pointer index table
8337                   w $00ca                     ;Pointer bitstream
8338
8339 ::indextab          b $00,$00,$03,$00,$05,$00,$09,$00
8340                  b $0f,$00,$13,$00,$19,$00,$1f,$00
8341                  b $21,$00,$24,$00,$27,$00,$2d,$00
8342                  b $31,$00,$34,$00,$38,$00,$3a,$00
8343                  b $3f,$00,$43,$00,$46,$00,$4a,$00
8344                  b $4e,$00,$52,$00,$56,$00,$5a,$00
8345                  b $5e,$00,$62,$00,$66,$00,$68,$00
8346                  b $6b,$00,$6e,$00,$72,$00,$75,$00
8347                  b $79,$00,$7e,$00,$82,$00,$86,$00
8348                  b $8a,$00,$8e,$00,$92,$00,$96,$00
8349                  b $9b,$00,$9f,$00,$a1,$00,$a4,$00
8350                  b $a8,$00,$ac,$00,$b2,$00,$b7,$00
8351                  b $bb,$00,$bf,$00,$c3,$00,$c7,$00
8352                  b $cb,$00,$cf,$00,$d3,$00,$d7,$00
8353                  b $dd,$00,$e1,$00,$e5,$00,$e9,$00
8354                  b $ec,$00,$f1,$00,$f4,$00,$f8,$00
8355                  b $fd,$00,$00,$01,$04,$01,$08,$01
8356                  b $0b,$01,$0f,$01,$13,$01,$16,$01
8357                  b $1a,$01,$1e,$01,$20,$01,$23,$01
8358                  b $27,$01,$29,$01,$2f,$01,$33,$01
8359                  b $37,$01,$3b,$01,$3f,$01,$42,$01
8360                  b $46,$01,$49,$01,$4d,$01,$51,$01
8361                  b $57,$01,$5b,$01,$5f,$01,$63,$01
8362                  b $67,$01,$6b,$01,$6f,$01,$74,$01
8363                  b $74,$01
8364
8365 ::line1             b $0a,$94,$4c,$90,$94,$a8,$00,$00
8366                  b $45,$98,$5c,$9c,$88,$00,$03,$18
8367                  b $98,$d9,$dc,$ca,$95,$44,$52,$4c
8368                  b $4c,$6e,$aa,$aa,$ab,$b0,$32,$02
8369                  b $04,$02,$04,$11,$28,$80,$00,$00
8370                  b $01,$00,$00,$00,$02,$48,$00,$00
8371
8372 ::line2             b $0a,$be,$6d,$24,$a2,$71,$00,$02
8373                  b $ac,$44,$d1,$05,$54,$05,$d0,$a5
8374                  b $55,$15,$11,$0a,$95,$46,$da,$aa
8375                  b $aa,$84,$aa,$aa,$a8,$a4,$15,$01
8376                  b $36,$26,$48,$d8,$0a,$b4,$c4,$c6
8377                  b $4d,$aa,$aa,$aa,$e4,$44,$50,$00

```

```

8378
::line3      b $08,$14,$c2,$1e,$22,$fb,$87,$04      8379
              b $a4,$89,$59,$88,$8d,$28,$09,$35      8380
              b $d9,$15,$dd,$6e,$96,$45,$56,$ac      8381
              b $ac,$44,$aa,$a9,$29,$22,$10,$00      8382
              b $55,$4a,$ad,$55,$2c,$aa,$aa,$aa      8383
              b $99,$2a,$aa,$4a,$48,$42,$a0,$00      8384
              8385
::linbe4     b $00,$3e,$65,$a4,$22,$71,$00,$08      8386
              b $a5,$05,$c5,$49,$44,$05,$d0,$3d      8387
              b $55,$15,$11,$2a,$95,$44,$52,$a8      8388
              b $aa,$24,$aa,$aa,$92,$21,$10,$00      8389
              b $55,$4a,$c9,$55,$2c,$aa,$aa,$aa      8390
              b $8d,$2a,$aa,$4a,$84,$44,$00,$00      8391
              8392
::line5      b $08,$14,$c9,$9c,$22,$a8,$10,$50      8393
              b $45,$d8,$58,$88,$89,$20,$01,$21      8394
              b $58,$d9,$d0,$ea,$a5,$74,$52,$48      8395
              b $6a,$c4,$64,$52,$93,$a0,$90,$00      8396
              b $36,$26,$68,$d5,$2a,$aa,$a4,$c6      8397
              b $98,$99,$14,$a6,$e4,$44,$00,$00      8398
              8399
::line6      b $00,$00,$40,$00,$14,$00,$20,$00      8400
              b $00,$00,$00,$00,$00,$00,$40,$00,$18      8401
              b $00,$00,$00,$00,$00,$00,$00,$00,$00      8402
              b $20,$00,$00,$00,$00,$00,$30,$30,$78      8403
              b $00,$00,$01,$80,$40,$00,$00,$00,$82      8404
              b $00,$00,$00,$04,$02,$48,$00,$00      8405
endif      8406
            8407
; Zeiger auf Speicher Laufwerk-Icons.      8408
.V72DD      w V34D6      8409
            w V3526      8410
            w V3576      8411
            w V35C6      8412
            8413
; Speicher für Laufwerk-Icons.      8414
:V34D6      s DRVICON_SIZE      8415
:V3526      s DRVICON_SIZE      8416
:V3576      s DRVICON_SIZE      8417
:V35C6      s DRVICON_SIZE      8418
            8419
; DoIcons-Tabelle      8420
; TD13: IconTab      8421
.MENU_ICONS = 3 ;Icons für Drucker/TrashBox/TaskMan.      8422
.DI3616     b $00      8423
            w $0000      8424
            b $00      8425
            8426
            w GR3692      ;Trash-Icon      8427
            b (PANEL_X0+$10)/8,PANEL_Y0+$70      8428
            b $03,$18      8429
            w IR5A9C      8430
            8431
            w GR36DB      ;Drucker-Icon      8432

```

```

8433      b (PANEL_X0+$10)/8,PANEL_Y0+$90
8434      b $03,$18
8435      w IR4E1C
8436
8437      w GR3724                      ;GEOS-Icon
8438      b TASK_X0/8,TASK_Y0
8439      b TASK_XW/8,TASK_YH
8440      w IR3A9F
8441
8442 :V3630      w V34D6                      ;Laufwerk #1
8443      b (PANEL_X0+$10)/8,PANEL_Y0+$08
8444      b DRVICON_X,DRVICON_Y
8445      w IR3AAE
8446
8447      w V3526                      ;Laufwerk #2
8448      b (PANEL_X0+$10)/8,PANEL_Y0+$20
8449      b DRVICON_X,DRVICON_Y
8450      w IR3AAB
8451
8452      w V3576                      ;Laufwerk #3
8453      b (PANEL_X0+$10)/8,PANEL_Y0+$38
8454      b DRVICON_X,DRVICON_Y
8455      w IR3AA8
8456
8457      w V35C6                      ;Laufwerk #4
8458      b (PANEL_X0+$10)/8,PANEL_Y0+$50
8459      b DRVICON_X,DRVICON_Y
8460      w IR3AA5
8461
8462 ; Disk-Icon (für DnD).
8463 if ENABLE_DRVGHST = FALSE
8464 :GR3652      b $80 +3*21
8465      b %01111111,%11111111,%11111110
8466      b %01000000,%00000000,%00000010
8467      b %01011111,%10000000,%00000110
8468      b %01010000,%10000000,%00000100
8469      b %01010000,%10000000,%00000110
8470      b %01010000,%10000000,%00000010
8471      b %01011111,%10011000,%00000010
8472      b %01000000,%00111100,%00000010
8473      b %01000000,%01100110,%00000010
8474      b %01000000,%01000010,%00000010
8475      b %01000000,%01100110,%10000010
8476      b %01000000,%00111100,%00000010
8477      b %01000000,%00011000,%00000010
8478      b %01000000,%00000000,%00000010
8479      b %01000000,%00011000,%00000010
8480      b %01000000,%00100100,%00000010
8481      b %01000000,%00100100,%00000010
8482      b %01000000,%00011000,%00000010
8483      b %01000000,%00000000,%00000010
8484      b %01111111,%11111111,%11111110
8485      b %00000000,%00000000,%00000000
8486 endif
8487

```

; Trash-Icon.			8488
:GR3692	b \$80 +3*24		8489
	b %00000000,%00111100,%00000000		8490
	b %00000000,%01100110,%00000000		8491
	b %00111111,%11111111,%11111100		8492
	b %00100000,%00000000,%00000100		8493
	b %00111111,%11111111,%11111100		8494
	b %00011111,%11111111,%11111000		8495
	b %00010101,%00100100,%10101000		8496
	b %00010101,%00100100,%10101000		8497
	b %00010101,%00100100,%10101000		8498
	b %00010101,%00100100,%10101000		8499
	b %00010101,%00100100,%10101000		8500
	b %00010101,%00100100,%10101000		8501
	b %00010101,%00100100,%10101000		8502
	b %00010101,%00100100,%10101000		8503
	b %00010101,%00100100,%10101000		8504
	b %00010101,%00100100,%10101000		8505
	b %00010101,%00100100,%10101000		8506
	b %00010101,%00100100,%10101000		8507
	b %00010101,%00100100,%10101000		8508
	b %00010101,%00100100,%10101000		8509
	b %00010101,%00100100,%10101000		8510
	b %00011111,%11111111,%11111000		8511
	b %00110000,%00000000,%00001100		8512
	b %00111111,%11111111,%11111100		8513
			8514
; Drucker-Icon.			8515
:GR36DB	b \$80 +3*24		8516
	b %11111111,%00000000,%00001110		8517
	b %10111001,%10011100,%00011011		8518
	b %10000000,%10110110,%11110001		8519
	b %10111000,%11100011,%10000000		8520
	b %10000000,%10000000,%00000000		8521
	b %10111001,%10011111,%11111111		8522
	b %11111111,%00100000,%00000000		8523
	b %00000000,%01001011,%11011001		8524
	b %00000000,%10000000,%00000010		8525
	b %00000111,%00111100,%11100111		8526
	b %00001101,%00000000,%00001001		8527
	b %00011010,%01101110,%10010011		8528
	b %00110100,%00000000,%00100111		8529
	b %01101111,%11111111,%11001101		8530
	b %11000000,%00000000,%00011001		8531
	b %11111111,%11111111,%11110001		8532
	b %10000000,%00000000,%00010001		8533
	b %10000000,%11111100,%00010001		8534
	b %10111000,%11111100,%00010011		8535
	b %10111000,%00000000,%00010110		8536
	b %10000000,%00000000,%00011100		8537
	b %01111111,%11111111,%11111000		8538
	b %00100000,%00000000,%00110000		8539
	b %00111111,%11111111,%11100000		8540
			8541
; GEOS-Icon (für TaskManager).			8542

```

8543 if FALSE
8544 :GR3724          b $80 +2*16
8545                  b %11111111,%11111111
8546                  b %10000000,%00000001
8547                  b %10111111,%11110001
8548                  b %10111111,%00010001
8549                  b %10100000,%00010001
8550                  b %10100000,%00010001
8551                  b %10101111,%11111101
8552                  b %10101111,%10011101
8553                  b %10101000,%00000101
8554                  b %10111011,%11110101
8555                  b %10001010,%00010101
8556                  b %10001011,%11110101
8557                  b %10001000,%000000101
8558                  b %10001111,%11111101
8559                  b %10000000,%00000001
8560                  b %11111111,%11111111
8561 else
8562 :GR3724          b $80 +2*16
8563                  b %11111111,%11111111
8564                  b %11010101,%01010101
8565                  b %10111111,%11111101
8566                  b %11111110,%01110101
8567                  b %10100000,%00011011
8568                  b %11100000,%00010101
8569                  b %10101111,%11111111
8570                  b %11101111,%10011101
8571                  b %10101000,%00000111
8572                  b %11111000,%00000101
8573                  b %10101000,%00000111
8574                  b %11011000,%00000101
8575                  b %10101000,%00000111
8576                  b %11011111,%11111101
8577                  b %10101010,%10101011
8578                  b %11111111,%11111111
8579 endif
8580
8581 ; Hauptmenü aktivieren.
8582 ; Ist kein Menü aktiv (z.B. nach der
8583 ; Rückkehr aus dem RAM-TopDesk), dann
8584 ; führt der Aufruf von GotoFirstMenu
8585 ; zum Absturz, da keine Menü-Daten in
8586 ; GEOS hinterlegt sind.
8587 :Z377A          bit   mouseOn          ;Menü aktiv?
8588                  bvc   :1              ; => Nein, weiter...
8589                  jsr   GotoFirstMenu
8590 ::1            rts
8591
8592 ; VLIR-Modul nachladen.
8593 ; Benötigt ein geöffnetes Fenster und
8594 ; öffnet das Hauptmenü.
8595 ; Übergabe: A = Sprungziel.
8596 ;           X = Modul-Nr.
8597 ; TD13: JmpSub2

```



:Z377B	pha		;Sprungziel	8598
			;zwischenspeichern.	8599
	txa		;Modul-Nr.	8600
	pha		;zwischenspeichern.	8601
				8602
	jsr	Z377A	; => GotoFirstMenu	8603
				8604
	ldx	C042A	;Oberstes Fenster.	8605
	lda	C0448,x	;Fenster geöffnet?	8606
	bne	x3790	; => Ja, weiter...	8607
				8608
	pla			8609
	pla			8610
	rts			8611
				8612
; VLIR-Modul nachladen.				8613
; Übergabe: A = Sprungziel.				8614
; X = Modul-Nr.				8615
; TD13: JmpSub				8616
:Z378F	pha		;Sprungziel	8617
			;zwischenspeichern.	8618
	txa		;Modul-Nr.	8619
	pha		;zwischenspeichern.	8620
				8621
	jsr	Z377A	; => GotoFirstMenu	8622
				8623
:x3790	pla		;Modul-Nr.	8624
	jsr	Z072C	;Neues Modul laden.	8625
	bcs	:err	; => Fehler...	8626
				8627
; TODO:				8628
; Sprungziel direkt als 0,3,6... dann				8629
; ist keine Berechnung erforderlich.				8630
				8631
	pla		;Sprungziel x 3 =	8632
	sta	r0L	;JMP-Zieladresse.	8633
	clc			8634
	adc	r0L		8635
	adc	r0L		8636
	adc	V0425 +0		8637
	sta	r0L		8638
	lda	V0425 +1		8639
	adc	#\$00		8640
	sta	r0H		8641
				8642
	jmp	(r0)		8643
				8644
::err	pla			8645
	rts			8646
				8647
; Anzeigemodus: Text.				8648
; Übergabe: A = Icon-/Text-Modus.				8649
; TD13: GetTextService				8650
:Z37AF	ldx	#VMod06	;Anzeige: Text.	8651
	b	\$2c		8652

```

8653
8654 ; Anzeigemodus: Icons.
8655 ; Übergabe: A = Icon-/Text-Modus.
8656 ; TD13: GetIconService
8657 .S37B2          ldx    #VMod05          ;Anzeige: Icons.
8658                pha
8659                txa
8660                pha
8661
8662                jsr     Z377A              ; => GotoFirstMenu
8663
8664                lda     #< BASE_MOD_B      ; => $2BA6
8665                sta     V0425 +0
8666                lda     #> BASE_MOD_B      ;Ladeadresse für
8667                sta     V0425 +1          ;Anzeige-Service.
8668
8669                pla
8670                jsr     Z072C              ;Neues Modul laden.
8671
8672                lda     #< BASE_MOD_A
8673                sta     V0425 +0
8674                lda     #> BASE_MOD_A      ;Ladeadresse für
8675                sta     V0425 +1          ;VLIR-Module.
8676
8677                bcs     :err              ; => Fehler...
8678
8679                pla
8680                sta     C04B2              ;Icon-/Text-Modus.
8681                rts
8682
8683 ::err          pla
8684                rts
8685
8686 ; Hinweis:
8687 ; Diese Routine wird u.a. beim laden
8688 ; eines Hilfsmittels ausgeführt.
8689 ;
8690 ; TopDesk.win/Konfig. in REU speichern.
8691 ; TD13: StashMain
8692 :Z37DC          lda     C0428              ;RAM-TopDesk?
8693                beq     :exit              ; => Nein, Ende...
8694
8695                lda     #$00              ;Kein VLIR-Modul mehr
8696                sta     V0422              ;im Speicher!
8697
8698                ldx     #(r3L - r0L)
8699 ::1            lda     r0,x
8700                pha
8701                dex
8702                bpl     :1
8703
8704 ; Hinweis:
8705 ; Beim speichern der Fensterdaten wird
8706 ; nur MoveData verwendet.
8707 ; => Verändert: r0 bis r2

```



```

8763      iny
8764      inx
8765      cpx    #(r2H - r0L) +1
8766      bcc    :2
8767
8768      tya
8769      pha
8770      jsr    MoveData
8771      pla
8772      tay
8773
8774      cpy    #tdwin_data_s
8775      bcc    :1
8776
8777 ; Menü-Optionen definieren.
8778      lda    MT3E11                ;Anzeige Block/Kb.
8779      sta    TDWIN_SIZEMODE
8780      lda    C04B2                ;Icon-/Text-Modus.
8781      sta    TDWIN_VIEWMODE
8782
8783      rts
8784
8785 ; Daten für TOPDESK.WIN.
8786 :tdwin_data
8787 ::00      w C624F                ;Fenster-Status.
8788           w TDWIN_SUBDIR
8789           w 4 * MAXDIRSIZE
8790
8791 ::01      w C0448                ;Fenster-Status.
8792           w TDWIN_WINDOWS
8793           w 4
8794
8795 ::02      w C04BC                ;Größe Fenster #1.
8796           w TDWIN_WSIZE1
8797           w 6
8798
8799 ::03      w C04C7                ;Größe Fenster #2.
8800           w TDWIN_WSIZE2
8801           w 6
8802
8803 ::04      w C04D2                ;Größe Fenster #3.
8804           w TDWIN_WSIZE3
8805           w 6
8806
8807 ::05      w C04DD                ;Größe Fenster #4.
8808           w TDWIN_WSIZE4
8809           w 6
8810
8811 ::06      w C042A                ;Fensterstapel.
8812           w TDWIN_WSTACK
8813           w 4
8814
8815 ::07      w START_TITLE_DATA     ;Fenstertitel #1-#4.
8816           w TDWIN_TITLE
8817           w SIZE_TITLE_DATA

```

```

8818
; TODO: 8819
; Warum wird hier die temp.Ordnerliste 8820
; in der TD.WIN gespeichert? 8821
; Die temporäre Ordnerliste wird nur 8822
; in mod.#09 beim anlegen eines neuen 8823
; Ordners verwendet. 8824
::08      w C051B          ;Aktive Ordner-Nr. 8825
          w TDWIN_SDINFO  8826
          w 12            8827
8828
::09      w C050B          ;Fensterposition. 8829
          w TDWIN_WINPOS  8830
          w 4             8831
8832
::10      w C04F8          ;Lfwk./Part. für 8833
          w TDWIN_DRVPART ;Fenster #1-#4. 8834
          w 4+4           8835
8836
:tdwin_data_e 8837
:tdwin_data_s = ( tdwin_data_e - tdwin_data ) 8838
8839
;*** Menü-Routinen. 8840
; TD13: DeskDosNew 8841
:MR396E      lda  #VJump2          ;Disk/löschen. 8842
8843
; TODO: 8844
; Wozu wird hier das aktuelle VLIR-Mod. 8845
; verändert? Das Modul ändert keine 8846
; Programmroutinen und kann mehrfach 8847
; nacheinander ausgeführt werden. 8848
;          sta  V0422          ;Aktuelles VLIR-Mod. 8849
;          b $2c              8850
; TD13: DValidate 8851
:MR3962      lda  #VJump1          ;Disk/aufräumen. 8852
          ldx  #VMod01          8853
          jmp  Z377B          ;VLIR-Modul laden. 8854
8855
; Hinweis: 8856
; "Datei retten" löscht das akt. VLIR- 8857
; Modul, damit wird das Modul zur 8858
; Fortsetzung immer neu geladen um die 8859
; Variablen zu initialisieren. 8860
8861
; TD13: n.v. 8862
.S3997      ldx  #VMod01          ;Dateien retten. 8863
          b $2c              8864
8865
;*** Menü-Routinen. 8866
:MR399E      ldx  #VMod02          ;TopDesk-Info. 8867
          b $2c              8868
:MR3A2F      ldx  #VMod12          ;Farben ändern. 8869
          lda  #VJump0          8870
          jmp  Z378F          ;VLIR-Modul laden. 8871
8872

```

```

8873 ; TD13: DeskDuplicate
8874 ;
8875 ; *FEHLER*
8876 ; Die Routine hat ein grundsätzliches
8877 ; Problem wenn zwischen zwei markierten
8878 ; Dateien ein gelöschter Dateieintrag
8879 ; liegt: Wird die erste markierte Datei
8880 ; dupliziert, dann wird im nächsten
8881 ; Durchlauf der neue Eintrag als zweite
8882 ; markierte Datei behandelt.
8883 ; => Hier wird dann eine andere Datei
8884 ; dupliziert als zuvor markiert.
8885 ;
8886 ; Fix:Nur einzelne Dateien duplizieren.
8887 :MR39A4      jsr    Z4E67          ;Test MultiFile.
8888             bcc    :1             ; => OK, weiter...
8889             rts
8890
8891 ::1          lda    #VJump2        ;Datei/duplizieren.
8892             b $2c
8893
8894 ; TD13: DeskRename
8895 :MR39A1      lda    #VJump1        ;Datei/umbenennen.
8896             b $2c
8897
8898 ; TD13: DeskDuplicate, Fortsetzung.
8899 .S39A7       lda    #VJump3        ;Datei/duplizieren.
8900             ldx    #VMod02
8901             jmp    Z377B          ;VLIR-Modul laden.
8902
8903 ;*** Menü-Routinen.
8904 :MR39B9      lda    #VJump0        ;Neues Verzeichnis
8905             b $2c
8906 :Z39BC       lda    #VJump1        ;Verzeichnis löschen.
8907             b $2c
8908 :Z39BF       lda    #VJump3        ;Hilfsm./Anwendungen.
8909             b $2c
8910 :Z39C2       lda    #VJump4        ;Verz. kopieren.
8911             b $2c
8912 .S39C5       lda    #VJump5        ;Verzeichnis dupl.
8913             b $2c
8914 :MR39C8      lda    #VJump7        ;Neuer Ordner.
8915             ldx    #VMod09
8916             jmp    Z377B          ;VLIR-Modul laden.
8917
8918 ; Shortcur: CTRL+X.
8919 :KB39CD      lda    C0428          ;RAM-TopDesk?
8920             beq    :1             ; => Nein, weiter...
8921             rts
8922
8923 ::1          lda    #VJump5        ;Programm beenden.
8924             b $2c
8925
8926 ;*** Menü-Routinen.
8927 :MR39CF      lda    #VJump4        ;GEOS beenden.

```

```

                                pha                                8928
                                                                8929
                                jsr    S09DD                      ;Konfig in REU/Datei. 8930
                                jsr    Z37DC                      ;TopDesk.win -> REU. 8931
                                                                8932
                                ldx     #VMod02                   8933
                                pla                                8934
                                jmp     Z378F                    ;VLIR-Modul laden. 8935
                                                                8936
;*** Menü-Routinen.                                              8937
:MR39DF                        ldx     #VMod12                   8938
                                lda     #VJump1                  ;Uhrzeit ändern. 8939
                                jmp     Z378F                    ;VLIR-Modul laden. 8940
                                                                8941
; Quicklink (kurzer Mausklick).                                    8942
:Z39AE                        lda     #$ff                      ;GEOS64.Editor. 8943
                                sta     V4191                   8944
                                                                8945
;*** Menü-Routinen.                                              8946
:Z39B6                        ldx     #VMod09                    ;Quicklink öffnen. 8947
                                b $2c                            8948
:MR39E9                        ;Disk/formatieren.              8949
                                                                8950
;*** Shortcut.                                                    8951
:KB39EC                        ldx     #VMod12                    ;Disk/formatieren. 8952
                                lda     #VJump2                  8953
                                jmp     Z378F                    ;VLIR-Modul laden. 8954
                                                                8955
;*** Menü-Routinen.                                              8956
:MR39F9                        lda     #VJump1                  ;Disk/umbenennen. 8957
                                ldx     #VMod04                   8958
                                jmp     Z377B                    ;VLIR-Modul laden. 8959
                                                                8960
:Z3A00                        lda     #VJump0                    ;Laufwerk-Icons. 8961
                                b $2c                            8962
:Z3A03                        lda     #VJump1                    ;TopDesk-Reset. 8963
                                b $2c                            8964
:R3A06                        lda     #VJump2                    ;Treiber laden. 8965
                                ldx     #VMod07                   8966
                                jmp     Z378F                    ;VLIR-Modul laden. 8967
                                                                8968
; Laufwerke tauschen.                                            8969
:Z3A0D                        stx     V3A1A                      ;Quell-Laufwerk. 8970
                                sta     V3A1B                    ;Ziel-Laufwerk. 8971
                                                                8972
                                ldx     #VMod07                   8973
                                lda     #VJump3                  ;Laufwerke tauschen. 8974
                                jmp     Z378F                    ;VLIR-Modul laden. 8975
                                                                8976
; GEOS-Laufwerke für "Laufwerk tauschen".                        8977
; Wird nur in mod.#07 verwendet.                                  8978
.V3A1A                        b $00 ;Quell-Laufwerk.            8979
.V3A1B                        b $00 ;Ziel-Laufwerk.            8980
                                                                8981
;*** Menü-Routinen.                                              8982

```

```

8983 :MR3A1C          ldx    #VMod08          ;Disk/kopieren.
8984                b $2c
8985 :MR3A1F          ldx    #VMod03          ;Directory sortieren.
8986                lda    #VJmp1
8987                jmp     Z377B              ;VLIR-Modul laden.
8988
8989 ;*** Menü-Routinen.
8990 :MR3A2C          ldx    curDrive
8991                lda    RealDrvMode -8,x
8992                and    #SET_MODE_SD2IEC
8993                bne    :1                  ; => SD2IEC.
8994
8995                jmp     GotoFirstMenu
8996
8997 ::1              ldx    #VMod08          ;Image erstellen.
8998                b $2c
8999 :MR3A29          ldx    #VMod04          ;Datei/Info.
9000                lda    #VJmp0
9001                jmp     Z377B              ;VLIR-Modul laden.
9002
9003 ; Fortsetzung Kopierfunktion.
9004 ; Aufruf nur aus Routine L5537 nachdem
9005 ; Unterverz./Datei ausgeschlossen sind.
9006 :Z3A37          ldx    #VMod09          ;Ordner kopieren.
9007                lda    #VJmp6
9008                jmp     Z378F              ;VLIR-Modul laden.
9009
9010 ;*** Menü-Routinen.
9011 :MR3A3E
9012 :Z3A41          lda    #VJmp0          ;RAM-TopDesk ein/aus.
9013                b $2c
9014 :Z3A44          lda    #VJmp1          ;TD.WIN einlesen.
9015                b $2c
9016 :MR3824          lda    #VJmp2          ;TD.WIN speichern.
9017                ldx    #VMod10
9018                jmp     Z378F              ;VLIR-Modul laden.
9019
9020 ; 64Net-/CMD-Partition oder DiskImage wechseln.
9021 :MR3A26          ldx    C042A          ;Oberstes Fenster.
9022                lda    C0448,x          ;Fenster geöffnet?
9023                beq     :exit            ; => Nein, Ende...
9024
9025                ldy    C04F8,x          ;Lfwk. für Fenster.
9026                lda    V04B8 -8,y        ;Geräteinfo einlesen.
9027                bne    :part            ; => CMD/64Net.
9028
9029                lda    RealDrvMode -8,y
9030                and    #SET_MODE_SD2IEC
9031                beq     :exit            ; => Kein SD2IEC...
9032
9033                ldx    #VMod03          ;DiskImage wechseln.
9034                lda    #VJmp0
9035                jmp     Z377B              ;VLIR-Modul laden.
9036
9037 ::exit          jmp     GotoFirstMenu

```



```

; CMD-/64Net-Partition wählen.                                9038
; Übergabe: Y = Laufwerk 8-11.                                9039
::part      tya                                                9040
            jsr    S06AB          ;Laufwerk öffnen.          9041
            jsr    Z3A89          ;Partition wählen.          9042
            bit    V3A4B          ;64Net-Part. wählen?       9043
            bpl    :done          ; => Nein, Ende...           9044
; 64Net ändert ggf. den Laufwerkstyp,                          9045
; daher Laufwerke aktualisieren.                              9046
            jsr    Z3A00          ;Laufwerk-Icons.            9047
            jsr    Z1514          ;Panel/Icons zeigen.        9048
            jsr    Z1DBE          ;Farbe Lfwk. setzen.        9049
            lda    C042A          ;Oberstes Fenster.         9050
            sta    WIN042F        ;Aktives Fenster.          9051
            lda    #WN_ACTIVATE   9052
            sta    WIN042E        ;Fenster-Befehl.           9053
            jmp    Z4923          ;Window-Handler.            9054
::done      rts                                                9055
; CMD-Partition suchen.                                       9056
:Z3A83      stx    V3A94          ;Fenster-Nr.                9057
            lda    #VJump2        ;CMD-Part. suchen.         9058
            b $2c                9059
; CMD-Partition wählen.                                       9060
:Z3A89      lda    #VJump0        ;Neue Partition.            9061
            b $2c                9062
; Vorherige CMD-Partition für Fenster öffnen.                 9063
:Z3A8C      lda    #VJump1        ;Part./Fens. öffnen.       9064
            ldx    #VMod11        9065
            jmp    Z378F          ;VLIR-Modul laden.          9066
;*** Menü-Routinen:                                           9067
if ENABLE_GEOHELP = TRUE                                       9068
:MR3A95     ldx    #VMod12        ;Hilfesystem starten.      9069
            lda    #VJump3        9070
            jmp    Z378F          ;VLIR-Modul laden.          9071
endif                                                  9072
;*** Icon-Routine: TaskManager starten.                       9073
:IR3A9F     lda    #$ff          9074
            sta    V046F          9075
            rts                  9076
; Shortcut und DoIcons.                                       9077
; TD13: OpenD8 bis OpenD11                                     9078

```

```

9093 :IR3AA5          lda    #11          ;F7 / Klick auf D:
9094                b $2c
9095 :IR3AA8          lda    #10          ;F5 / Klick auf C:
9096                b $2c
9097 :IR3AAB          lda    #9           ;F3 / Klick auf B:
9098                b $2c
9099 :IR3AAE          lda    #8           ;F1 / Klick auf A:
9100
9101 ; Hinweis:
9102 ; Ist der DND-Modus aktiv und wird
9103 ; auf ein DoIcons-Laufwerk geklickt,
9104 ; dann wird das Laufwerk getauscht.
9105 ; Laufwerk öffnen.
9106 ; Übergabe: A = Laufwerk.
9107 ; TD13: OpenDA
9108 .S3AB0            pha                ;Laufwerk retten.
9109
9110                lda    numDrives
9111                cmp    #$02
9112                bcc    :2
9113
9114                lda    V04B3           ;ShortCut-Flag?
9115                bne    :1              ; => Ja, weiter...
9116
9117                jsr    S48B6           ;Langer Mausklick?
9118                bcs    :2              ; => Nein, weiter...
9119
9120 ; DnD-Modus für Laufwerke tauschen.
9121 ::1               pla
9122
9123                bit    V0423           ;DnD bereits aktiv?
9124                bmi    :3              ; => Ja, Ende...
9125
9126                sta    V3B29           ;Quell-Laufwerk.
9127
9128 if ENABLE_DRVGHST = FALSE
9129                ldx    #< GR3652 +1    ;Disk-Icon für DnD.
9130                ldy    #> GR3652 +1
9131 endif
9132 if ENABLE_DRVGHST = TRUE
9133                sec
9134                sbc    #8
9135                asl
9136                tay
9137                lda    V72DD +0,y
9138                clc
9139                adc    #< 1
9140                tax
9141                lda    V72DD +1,y
9142                adc    #> 1
9143                tay
9144 endif
9145
9146                jsr    Z4F9E            ;DnD aktivieren.
9147

```

```

; Hinweis: 9148
; DnD-Sprite anzeigen / positionieren. 9149
; Ansonsten erscheint das DnD-Sprite 9150
; wenn Maustaste nicht mehr gedrückt. 9151
        jsr    R3B2A                ;Sprite anzeigen. 9152
                                           9153
        lda    #$00                ;Löschen des 9154
        sta    V04B3                ;ShortCut-Flag. 9155
        jmp    S52AD                ;Maustaste loslassen. 9156
                                           9157
; Kurzer Mausklick auf Laufwerk. 9158
::2      lda    V3B29                ;Quell-Laufwerk. 9159
        beq    :4                    ; => Nicht gesetzt... 9160
                                           9161
        lda    V0423                ;DnD aktiv? 9162
        bpl    :4                    ; => Nein, weiter... 9163
                                           9164
        ldx    V3B29                ;Quell-Laufwerk. 9165
        pla                    ;Ziel-Laufwerk. 9166
        cmp    V3B29                ;Quelle=Ziel? 9167
        beq    :3                    ; => Ja, DnD beenden. 9168
                                           9169
        jsr    Z3A0D                ;Laufwerk tauschen. 9170
                                           9171
        lda    #$00                9172
        sta    V3B29                ;Laufwerk löschen. 9173
                                           9174
        jsr    Z1514                ;Panel/Icons zeigen. 9175
        jmp    S157E                ;Alle Fens. anzeigen. 9176
                                           9177
::3      jmp    Z3B54                ;DnD-Modus beenden. 9178
                                           9179
::4      ldx    numDrives            ;Bei numDrives=1 9180
        dex                    ;Kein Lfwk.wechsel. 9181
        bne    :5                    9182
        pla                    9183
        cmp    curDrive            9184
        beq    :6                    9185
::8      rts                        9186
                                           9187
::5      pla                    9188
        tay                    9189
        ldx    driveType -8,y      9190
        beq    :8                    9191
                                           9192
::6      sta    V3B28                9193
        jsr    SetDevice            ;Laufwerk setzen. 9194
        jsr    OpenDisk            ;Diskette öffnen. 9195
        txa                    9196
        beq    :7                    ; => OK, Ende... 9197
                                           9198
        ldy    curDrive            9199
        lda    RealDrvMode -8,y    9200
        and    #SET_MODE_SD2IEC    9201
        beq    :err                ; => Keine Disk... 9202

```

```

9203
9204             ldx    #VMod03             ;DiskImage wechseln.
9205             lda    #VJump0
9206             jsr    Z378F             ;VLIR-Modul laden.
9207
9208 ; Hinweis:
9209 ; Workaround für SD2IEC: Bei einem
9210 ; Laufwerksfehler bleibt der Fehler im
9211 ; SD2IEC aktiv bis PurgeTurbo oder
9212 ; ExitTurbo ausgeführt wird.
9213 ; Sollte der DiskImage-Wechsel mit
9214 ; einem Fehler beendet werden, hier den
9215 ; Fehlerstatus des SD2IEC löschen.
9216 ;---
9217             jsr    ExitTurbo             ;TurboDOS aus.
9218 ;---
9219
9220             lda    V3B28
9221             jsr    SetDevice             ;Laufwerk setzen.
9222             jsr    OpenDisk             ;Diskette öffnen.
9223             txa
9224             beq    :7                 ; => OK, Ende...
9225
9226 ::err             jmp    S5CE1             ;Fehler ohne Refresh.
9227 ::7             jmp    Z44D1             ;Fens./Lfwk. öffnen.
9228
9229 :V3B29             b $00 ;Laufwerk für DnD/Lfwk. tauschen.
9230 :V3B28             b $00 ;Neues Laufwerk.
9231
9232 if ENABLE_ANIMATE = TRUE
9233
9234 ; Damit die Animation auch mit SuperCPU
9235 ; und TC64 sichtbar ist, muss der Takt
9236 ; auf 1MHz gesetzt werden.
9237 ; Die Ausführung von SCPU_Pause würde
9238 ; die Animation stark verlangsamen.
9239 ;
9240 ; Speed-Register für SuperCPU/TC64:
9241 :SCPU_HW_NORMAL    = $d07a
9242 :SCPU_HW_SPEED     = $d0b8
9243 :TC64_HW_EN_DIS    = $d0fe
9244 :TC64_HW_SPEED     = $d0f3
9245
9246 ; CPU-Takt auf 1MHz.
9247 :_slowCPU           ldx    #$00
9248                     b $2c
9249
9250 ; CPU-Takt zurücksetzen.
9251 :_resetCPU          ldx    #$ff
9252
9253                     php
9254                     sei
9255
9256                     lda    CPU_DATA             ;I/O einschalten.
9257                     pha

```

```

        lda    #IO_IN                      9258
        sta    CPU_DATA                    9259
                                           9260
        lda    #$2a                        ;TC64/HW ein. 9261
        sta    TC64_HW_EN_DIS             9262
                                           9263
        txa                                9264
        bne    :resetTC64                 9265
                                           9266
; CPU-Takt auf 1MHz setzen.                9267
::slowTC64    lda    TC64_HW_SPEED         9268
              sta    speedBufTC64         ;CPU-Takt speichern. 9269
              and    #%01111111          9270
              sta    TC64_HW_SPEED         ;CPU auf 1MHz setzen. 9271
                                           9272
::slowSCPU    ldy    #1                   9273
              bit    SCPU_HW_SPEED        9274
              bpl    :1                   9275
              dey                                9276
::1           sty    speedBufSCPU         ;CPU-Takt speichern. 9277
                                           9278
              sta    SCPU_HW_NORMAL       ;CPU auf 1MHz setzen. 9279
              bpl    :exit                9280
                                           9281
; CPU-Takt zurücksetzen.                  9282
::resetTC64   lda    speedBufTC64         ;TC64 zurücksetzen. 9283
              sta    TC64_HW_SPEED        9284
                                           9285
::resetSCPU   ldy    speedBufSCPU         ;SCPU zurücksetzen. 9286
              sta    SCPU_HW_NORMAL,y     9287
                                           9288
::exit        lda    #$ff                 ;TC64/HW aus. 9289
              sta    TC64_HW_EN_DIS       9290
                                           9291
              pla                                9292
              sta    CPU_DATA              ;I/O abschalten. 9293
                                           9294
              plp                                9295
              rts                             9296
                                           9297
:speedBufSCPU b    $00                   9298
:speedBufTC64 b    $00                   9299
                                           9300
; Fensteranimation.                      9301
; Übergabe: A = Fenster-Nr.              9302
:AnimateOpen  ldy    #$00                 9303
              b    $2c                    9304
:AnimateClose ldy    #$ff                 9305
                                           9306
              ldx    MT4048                9307
              cpx    #"*"                 9308
              beq    :animate              9309
              rts                             9310
                                           9311
::animate     sty    r1L                   9312

```

```

9313          pha
9314          tax
9315
9316          lda    V0450,x          ;Offset Fensterdaten.
9317          tay
9318
9319          ldx    #0
9320 :::1        lda    C04BC,y          ;Fensterkoord. nach
9321          sta    r10,x          ;r10 bis r12.
9322          iny
9323          inx
9324          cpx    #(r12H - r10L) +1
9325          bcc    :1
9326
9327          pla
9328          tax
9329          lda    C04F8,x
9330          sec
9331          sbc    #8
9332          asl
9333          asl
9334          asl
9335          tax
9336
9337          lda    V3630 +2,x          ;Koordinaten für
9338          ldy    #0          ;Laufwerk-Icon.
9339          asl
9340          asl
9341          asl
9342          bcc    :2
9343          iny
9344 :::2        sta    r3L
9345          sty    r3H
9346
9347 ;          lda    r3L
9348          clc
9349          adc    #< (24 -1)
9350          sta    r4L
9351          tya
9352          adc    #> (24 -1)
9353          sta    r4H
9354
9355          lda    V3630 +3,x
9356          sta    r2L
9357
9358          clc
9359          adc    # (21 -1)
9360          sta    r2H
9361
9362 ; r2 -r4 = Koordinaten Laufwerk-Icon.
9363 ; r10-r12 = Koordinaten für Fenster.
9364          bit    r1L
9365          bpl    SpeedFrame
9366
9367          ldx    #0

```

```

::3          lda    r2,x          9368
              pha          9369
              inx          9370
              cpx    #(r4H - r2L) +1 9371
              bcc    :3          9372
              9373
              jsr    SpeedFrameCalc 9374
              9375
              ldx    #(r4H - r2L) 9376
::4          lda    r2,x          9377
              sta    r10,x        9378
              pla          9379
              sta    r2,x        9380
              dex          9381
              bpl    :4          9382
              9383
              ldx    #0          9384
::5          lda    r2,x          9385
              pha          9386
              lda    r10,x        9387
              sta    r2,x        9388
              pla          9389
              sta    r10,x        9390
              inx          9391
              cpx    #(r4H - r2L) +1 9392
              bcc    :5          9393
              9394
;            jmp    SpeedFrame    9395
endif          9396
              9397
if ENABLE_ANIMATE = TRUE 9398
:astep = 15    9399
              9400
; SpeedFrame - Animation zeichnen. 9401
; Übergabe: r2L - y-oben          9402
;          r2H - y-unten          9403
;          r3  - x-links          9404
;          r4  - x-rechts         9405
;          r10L - Ziely-oben      9406
;          r10H - Ziely-unten     9407
;          r11 - Zielx-links      9408
;          r12 - Zielx-rechts     9409
; Zerstört: r0 bis r13           9410
:SpeedFrame    lda    #TRUE       9411
              b    $2c          9412
:SpeedFrameCalc lda    #FALSE     9413
              sta    r7H        9414
              9415
              jsr    _calcFrame    ;Animation berechnen. 9416
              9417
              bit    r7H          ;Animation anzeigen? 9418
              bpl    :3          ; => Nein, weiter... 9419
              9420
              jsr    _slowCPU      ;CPU auf 1MHz. 9421
              9422

```

```

9423          ldx    #0
9424 ::1        lda    r2,x
9425          pha
9426          inx
9427          cpx    #(r4H - r2L) +1
9428          bcc    :1
9429
9430          jsr    _drawFrame          ;Rahmen zeichnen.
9431
9432          ldx    #(r4H - r2L)
9433 ::2        pla
9434          sta    r2,x
9435          dex
9436          bpl    :2
9437
9438          jsr    _resetCPU          ;CPU zurücksetzen.
9439
9440 ::3        jmp    _drawFrame          ;Rahmen zeichnen.
9441
9442 ; Angaben für Animation berechnen.
9443 :_calcFrame  lda    r10H          ;Differenz zwischen
9444          sec          ;y-unten und
9445          sbc    r2H          ;Ziely-unten (16Bit).
9446          sta    r13L
9447          lda    #0
9448          sbc    #0
9449          sta    r13H
9450
9451          lda    r10L          ;Differenz zwischen
9452          sec          ;y-oben und
9453          sbc    r2L          ;Ziely-oben (16Bit).
9454          sta    r10L
9455          lda    #0
9456          sbc    #0
9457          sta    r10H
9458
9459          lda    r11L          ;Differenz zwischen
9460          sec          ;x-links und
9461          sbc    r3L          ;Zielx-links.
9462          sta    r11L
9463          lda    r11H
9464          sbc    r3H
9465          sta    r11H
9466
9467          lda    r12L          ;Differenz zwischen
9468          sec          ;x-rechts und
9469          sbc    r4L          ;Zielx-rechts.
9470          sta    r12L
9471          lda    r12H
9472          sbc    r4H
9473          sta    r12H
9474
9475          lda    #< astep          ;Schreittweite.
9476          sta    r0L          ;(16Bit-Wert)
9477          lda    #> astep

```



	sta	r0H	9478
			9479
; Werte in r10-r13 / Schrittweite.			9480
; r0 = Anzahl Schritte			9481
; r10 = Delta y-oben (16Bit)			9482
; r11 = Delta x-links			9483
; r12 = Delta x-rechts			9484
; r13 = Delta y-unten (16Bit)			9485
	ldx	#r10	9486
	ldy	#r0	9487
::1	jsr	DSdiv	9488
	inx		9489
	inx		9490
	cpx	#r14	9491
	bcc	:1	9492
			9493
	lda	r13L	9494
	sta	r10H	9495
	rts		9496
			9497
; Rahmen zeichnen/löschen.			9498
; r2L = y-oben			9499
; r2H = y-unten			9500
; r3 = x-links			9501
; r4 = x-rechts			9502
; r10L = Offset nach oben			9503
; r10H = Offset nach unten			9504
; r11 = Offset nach links			9505
; r12 = Offset nach rechts			9506
:_drawFrame	lda	#astep	9507
	sta	r8L	9508
			9509
::1	bit	r7H	9510
	bpl	:2	9511
			9512
	jsr	Z2515	9513
		;InvFrame	9514
::2	lda	r2L	9515
	clc		9516
	adc	r10L	9517
	sta	r2L	9518
			9519
	lda	r2H	9520
	clc		9521
	adc	r10H	9522
	sta	r2H	9523
			9524
	lda	r3L	9525
	clc		9526
	adc	r11L	9527
	sta	r3L	9528
	lda	r3H	9529
	adc	r11H	9530
	sta	r3H	9531
			9532

```

9533          lda    r4L
9534          clc
9535          adc    r12L
9536          sta    r4L
9537          lda    r4H
9538          adc    r12H
9539          sta    r4H
9540
9541          dec    r8L
9542          bne    :1
9543
9544          rts
9545 endif
9546
9547 ; appMain-Routine für DnD-Sprite-Anzeige.
9548 ; TD13: DeskMain
9549 .R3B2A          lda    V0423          ;DnD aktiv?
9550          beq    :exit          ; => Nein, weiter...
9551          bmi    :1          ;DnD/Laufwerk.
9552
9553          ldy    #0          ;DnD/MultiFile.
9554          b $2c
9555 ::1          ldy    #6          ;DnD/Laufwerk.
9556
9557          ldx    #0
9558 ::2          lda    :data,y          ;Mausgrenzen setzen.
9559          sta    mouseTop,x
9560          iny
9561          inx
9562          cpx    #6
9563          bcc    :2
9564
9565          lda    mouseXPos +0
9566          sec
9567          sbc    #< 10
9568          sta    r4L
9569          lda    mouseXPos +1
9570          sbc    #> 10
9571          sta    r4H
9572
9573          lda    mouseYPos
9574          sec
9575          sbc    #10
9576          sta    r5L
9577
9578          lda    #1
9579          sta    r3L
9580          jsr    PosSprite
9581          jsr    EnablSprite
9582
9583 ::exit          rts
9584
9585 ::data          b DESK_Y0 +10,    DESK_Y1 -10
9586          w SCREEN_X0 +10,    SCREEN_X1 -13
9587

```

```

                b DRIVE_Y0 +14,  DRIVE_Y1 -18          9588
                w DRIVE_X0 +10,  DRIVE_X1 -13          9589
                                                    9590
; otherPressVec-Routine für DnD-Sprite-Anzeige.      9591
; TD13: DeskOther                                  9592
.R3B53          lda  mouseData          ;Maustaste gedrückt? 9593
                bmi  :1                  ; => Nein, Ende...    9594
                                                    9595
                jsr  Z5B60                ;Koord. für Uhr.    9596
                jsr  IsMseInRegion        ;Klick auf Uhr?     9597
                beq  Z3B54                ; => Nein, weiter...  9598
                                                    9599
                jmp  MR39DF                ;Datum/Uhrzeit.    9600
                                                    9601
::1             rts                      9602
                                                    9603
:Z3B54          jsr  S52AD                ;Maustaste loslassen. 9604
                jmp  Z42DB                ;DnD-Modus beenden. 9605
                                                    9606
; Koordinaten für Uhr setzen.                      9607
:Z5B60          ldx  #(r4H - r2L)         ;Mausklick auf    9608
::1             lda  :coord,x            ;Datum/Uhrzeit?    9609
                sta  r2,x                9610
                dex                      9611
                bpl  :1                  9612
                rts                      9613
                                                    9614
::coord         b   TIME_Y0,TIME_Y1      9615
                w   TIME_X0,TIME_X1      9616
                                                    9617
; Laufwerk über ShortCut öffnen.                  9618
; TD13: KS8-KS11                                  9619
:KB3B7C         lda  #8                  ;F1                 9620
                b   $2c                  9621
:KB3B7F         lda  #9                  ;F3                 9622
                b   $2c                  9623
:KB3B82         lda  #10                 ;F5                 9624
                b   $2c                  9625
:KB3B85         lda  #11                 ;F7                 9626
                ldx  #$01                ;Setzen des        9627
                stx  V04B3                ;ShortCut-Flag.    9628
                jmp  S3AB0                ;Laufwerk öffnen.   9629
                                                    9630
; DeskAccessory laden.                            9631
:Z3B8F          lda  screencolors        ;GEOS-Farben.      9632
                sta  :col                9633
                jsr  i_FillRam            9634
                w 1000                   9635
                w COLOR_MATRIX           9636
::col           b   $00                  9637
                                                    9638
; TD13: DA_Call2                                  9639
                jsr  GetFile              ;Datei laden.      9640
                txa                      ;Fehler?            9641
                beq  S3B9B                ; => Nein, weiter... 9642

```

```

9643
9644             jsr    S5CF3                ;Fehler ausgeben.
9645
9646 ; TD13: DAReturn
9647 .S3B9B             jsr    InitForIO
9648
9649             lda    C_GEOS_FRAME        ;Rahmenfarbe setzen.
9650             sta    $d020
9651
9652             jsr    DoneWithIO
9653
9654             ldx    C042A                ;Oberstes Fenster.
9655             jsr    Z31FD                ;Prefs./PadCol laden.
9656
9657             jsr    S437D                ;TopDesk-Oberfläche.
9658             jsr    S5255                ;Mark.Dat=0/Anzeige.
9659
9660             jsr    S14F8                ;Bildschirm löschen.
9661             jsr    Z1514                ;Panel/Icons zeigen.
9662             jsr    S4840                ;Lfwk/Prnt/Inpt-Info.
9663
9664             jsr    S4321                ;Alle Fenster neu.
9665
9666             jsr    S6118                ;Datum/Uhr anzeigen.
9667
9668             ldx    #NO_ERROR
9669             rts
9670
9671 ; Hintergrund in REU speichern.
9672 ; Hinweis:
9673 ; Aufruf mit JSR! Die Rücksprungadresse
9674 ; dient als Zeiger auf die Fenster-
9675 ; koordinaten zum speichern der Grafik.
9676 ; -> SubMenu Dokumente.
9677 ; -> SubMenu Andere Dateien.
9678 :Z3BBB             lda    #0                ;Buffer1 verwenden.
9679                     b $2c
9680 ; -> Menu Disk.
9681 ; -> Menu Datei.
9682 ; -> Menu Anzeige.
9683 ; -> Menu Fenster.
9684 ; -> Menu Speziell.
9685 ; -> Menu geos.
9686 :Z3BBE             lda    #1                ;Buffer2 verwenden.
9687                     sta    V046C            ;Buffer setzen.
9688
9689                     pla                    ;Rücksprungadresse
9690                     clc                    ;als Zeiger auf die
9691                     adc    #< $0001        ;Koordinaten.
9692                     sta    r0L
9693                     pla
9694                     adc    #> $0001
9695                     sta    r0H
9696
9697                     lda    #< V0460        ;Zeiger auf Buffer1

```

	ldx	#> V0460	;für Fenstergrenzen.	9698
				9699
	ldy	V046C	;Buffer1 od. Buffer2?	9700
	beq	:1	; => Buf1, weiter...	9701
				9702
	lda	#< V0466	;Zeiger auf Buffer2	9703
	ldx	#> V0466	;für Fenstergrenzen.	9704
				9705
::1	sta	r1L		9706
	stx	r1H		9707
				9708
	ldy	#{r4H - r2L)		9709
::2	lda	(r0),y		9710
	sta	r2,y		9711
	sta	(r1),y		9712
	dey			9713
	bpl	:2		9714
				9715
	jsr	Z21B5	;Hintergrund in REU.	9716
				9717
	ldy	#{r4H - r2L)		9718
::3	lda	(r0),y		9719
	sta	r2,y		9720
	dey			9721
	bpl	:3		9722
				9723
;	ldy	r2L	;Wozu y-Koordinate	9724
;	iny		;korrigieren?	9725
;	sty	r2L		9726
				9727
	lda	COL0444	;Farbe GEOS-Menü.	9728
	jmp	S1561	;Screencolor setzen.	9729
				9730
;- - - Hauptmenü:				9731
; TD13: HauptMenu				9732
:mnu0_x0 = MENU_X0				9733
:mnu0_x1 = MENU_X1				9734
:mnu0_y0 = MENU_Y0				9735
:mnu0_y1 = MENU_Y1				9736
:DM3C04	b	mnu0_y0,mnu0_y1		9737
	w	mnu0_x0,mnu0_x1		9738
				9739
::_x3C0A	b	6 ! HORIZONTAL ! UN_CONSTRAINED		9740
				9741
::m1	w	MT5352	;geos	9742
	b	DYN_SUB_MENU		9743
	w	DS404E		9744
				9745
::m2	w	MT3C3D	;Datei	9746
	b	DYN_SUB_MENU		9747
	w	DS3CE2		9748
				9749
::m3	w	MT3C29	;Anzeige	9750
	b	DYN_SUB_MENU		9751
	w	DS3D87		9752

```

9753
9754 : :m4                w MT3C43                ;Disk
9755                     b DYN_SUB_MENU
9756                     w DS3C48
9757
9758 : :m5                w MT3C34                ;Fenster
9759                     b DYN_SUB_MENU
9760                     w DS3E86
9761
9762 : :m6                w MT3C2E                ;Optionen
9763                     b DYN_SUB_MENU
9764                     w DS3F9E
9765
9766 if LANG = LANG_DE
9767 :MT5352              b "geos",0
9768 :MT3C3D              b "Datei",0
9769 :MT3C29              b "Anzeige",0
9770 :MT3C43              b "Disk",0
9771 :MT3C34              b "Fenster",0
9772 :MT3C2E              b "Optionen",0
9773 endif
9774 if LANG = LANG_EN
9775 :MT5352              b "geos",0
9776 :MT3C3D              b "file ",0
9777 :MT3C29              b "view ",0
9778 :MT3C43              b "disk ",0
9779 :MT3C34              b "windows",0
9780 :MT3C2E              b "options",0
9781 endif
9782
9783 ;--- Hauptmenü/Disk:
9784 ; TD13: Disk_Menü
9785 if LANG = LANG_DE
9786 :mnu4_of = $0060
9787 endif
9788 if LANG = LANG_EN
9789 :mnu4_of = $0050
9790 endif
9791 :mnu4_xw = $0078
9792 :mnu4_x0 = mnu0_x0 +mnu4_of
9793 :mnu4_x1 = mnu4_x0 +mnu4_xw -1
9794 :mnu4_y0 = $10
9795 :mnu4_y1 = $77
9796 :mnu4_t1 = mnu4_x1 -20
9797 :mnu4_t2 = mnu4_x1 -47
9798 :DS3C48              jsr   Z3BBE                ;Grafik in REU/Buf2.
9799
9800                     b mnu4_y0,mnu4_y1
9801                     w mnu4_x0,mnu4_x1
9802
9803 : :_x3c51            b 7 ! VERTICAL ! MENU_VMODE
9804
9805 : :_x3c52            w MT3C75                ;Umbenennen
9806                     b MENU_ACTION
9807                     w MR39F9

```

		9808
w MT3C87	;Löschen	9809
b MENU_ACTION		9810
w MR396E		9811
		9812
w MT3C96	;Formatieren	9813
b MENU_ACTION		9814
w MR39E9		9815
		9816
w MT3CA9	;Kopieren	9817
b MENU_ACTION		9818
w MR3A1C		9819
		9820
w MT3CB9	;Aufräumen	9821
b MENU_ACTION		9822
w MR3962		9823
		9824
w MT3CCA	;Part./Img. wechseln	9825
b MENU_ACTION		9826
w MR3A26		9827
		9828
w MT5338	;Img. erstellen	9829
b MENU_ACTION		9830
w MR3A2C		9831
		9832
if LANG = LANG_DE		9833
:MT3C75	b "Umbenennen"	9834
	b GOTOX	9835
	w mnu4_t1	9836
	b \$80,BOLDON,"N",PLAINTEXT,0	9837
		9838
:MT3C87	b "Inhalt löschen"	9839
	b GOTOX	9840
	w mnu4_t1	9841
	b \$80,BOLDON,"E",PLAINTEXT,0	9842
		9843
:MT3C96	b "Formatieren"	9844
	b GOTOX	9845
	w mnu4_t1	9846
	b \$80,BOLDON,"F",PLAINTEXT,0	9847
		9848
:MT3CA9	b "Kopieren"	9849
	b GOTOX	9850
	w mnu4_t1	9851
	b \$80,BOLDON,"K",PLAINTEXT,0	9852
		9853
:MT3CB9	b "Aufräumen"	9854
	b GOTOX	9855
	w mnu4_t1	9856
	b \$80,BOLDON,"V",PLAINTEXT,0	9857
		9858
:MT3CCA	b "Part/Image wechseln"	9859
	b GOTOX	9860
	w mnu4_t1	9861
	b \$80,BOLDON,"J",PLAINTEXT,0	9862

```

9863
9864 :MT5338          b "Neues Image"
9865                b GOTOX
9866                w mnu4_t2
9867                b $80,BOLDON,"Shift I",PLAINTEXT,0
9868 endif
9869 if LANG = LANG_EN
9870 :MT3C75          b "Rename"
9871                b GOTOX
9872                w mnu4_t1
9873                b $80,BOLDON,"N",PLAINTEXT,0
9874
9875 :MT3C87          b "Erase content"
9876                b GOTOX
9877                w mnu4_t1
9878                b $80,BOLDON,"E",PLAINTEXT,0
9879
9880 :MT3C96          b "Format"
9881                b GOTOX
9882                w mnu4_t1
9883                b $80,BOLDON,"F",PLAINTEXT,0
9884
9885 :MT3CA9          b "Copy"
9886                b GOTOX
9887                w mnu4_t1
9888                b $80,BOLDON,"K",PLAINTEXT,0
9889
9890 :MT3CB9          b "Validate"
9891                b GOTOX
9892                w mnu4_t1
9893                b $80,BOLDON,"V",PLAINTEXT,0
9894
9895 :MT3CCA          b "Swap partition/image"
9896                b GOTOX
9897                w mnu4_t1
9898                b $80,BOLDON,"J",PLAINTEXT,0
9899
9900 :MT5338          b "Create image"
9901                b GOTOX
9902                w mnu4_t2
9903                b $80,BOLDON,"Shift I",PLAINTEXT,0
9904 endif
9905
9906 ;--- Hauptmenü/Datei:
9907 ; TD13: Datei_Menü
9908 :mnu2_of = $0018
9909 if LANG = LANG_DE
9910 :mnu2_xw = $0070
9911 endif
9912 if LANG = LANG_EN
9913 :mnu2_xw = $0060
9914 endif
9915 :mnu2_x0 = mnu0_x0 +mnu2_of
9916 :mnu2_x1 = mnu2_x0 +mnu2_xw -1
9917 :mnu2_y0 = $10

```



```

:mnu2_y1 = $77 9918
:mnu2_t1 = mnu2_x1 -22 9919
:DS3CE2      jsr    Z3BBE      ;Grafik in REU/Buf2. 9920
9921
          b mnu2_y0,mnu2_y1 9922
          w mnu2_x0,mnu2_x1 9923
9924
::_x3CEB      b 7 ! VERTICAL ! MENU_VMODE 9925
9926
::_x3CEC      w MT3D0F      ;Öffnen 9927
          b MENU_ACTION 9928
          w MR4E34 9929
9930
          w MT3D1D      ;Duplizieren 9931
          b MENU_ACTION 9932
          w MR39A4 9933
9934
          w MT3D30      ;Umbenennen 9935
          b MENU_ACTION 9936
          w MR39A1 9937
9938
          w MT3D42      ;Info 9939
          b MENU_ACTION 9940
          w MR3A29 9941
9942
          w MT3D4E      ;Drucken 9943
          b MENU_ACTION 9944
          w MR4E31 9945
9946
          w MT3D5D      ;Löschen 9947
          b MENU_ACTION 9948
          w MR5AB6 9949
9950
          w MT3D6C      ;Directory sortieren 9951
          b MENU_ACTION 9952
          w MR3A1F 9953
9954
if LANG = LANG_DE 9955
:MT3D0F      b "Öffnen" 9956
          b GOTOX 9957
          w mnu2_t1 9958
          b $80,BOLDON,"Z",PLAINTEXT,0 9959
9960
:MT3D1D      b "Duplizieren" 9961
          b GOTOX 9962
          w mnu2_t1 9963
          b $80,BOLDON,"H",PLAINTEXT,0 9964
9965
:MT3D30      b "Umbenennen" 9966
          b GOTOX 9967
          w mnu2_t1 9968
          b $80,BOLDON,"M",PLAINTEXT,0 9969
9970
:MT3D42      b "Info" 9971
          b GOTOX 9972

```

```

9973      w mnu2_t1
9974      b $80,BOLDON,"Q",PLAINTEXT,0
9975
9976 :MT3D4E      b "Drucken"
9977      b GOTOX
9978      w mnu2_t1
9979      b $80,BOLDON,"P",PLAINTEXT,0
9980
9981 :MT3D5D      b "Löschen"
9982      b GOTOX
9983      w mnu2_t1
9984      b $80,BOLDON,"D",PLAINTEXT,0
9985
9986 :MT3D6C      b "Directory sortieren"
9987      b GOTOX
9988      w mnu2_t1
9989      b $80,BOLDON,"T",PLAINTEXT,0
9990 endif
9991 if LANG = LANG_EN
9992 :MT3D0F      b "Open"
9993      b GOTOX
9994      w mnu2_t1
9995      b $80,BOLDON,"Z",PLAINTEXT,0
9996
9997 :MT3D1D      b "Duplicate"
9998      b GOTOX
9999      w mnu2_t1
10000      b $80,BOLDON,"H",PLAINTEXT,0
10001
10002 :MT3D30      b "Rename"
10003      b GOTOX
10004      w mnu2_t1
10005      b $80,BOLDON,"M",PLAINTEXT,0
10006
10007 :MT3D42      b "Info"
10008      b GOTOX
10009      w mnu2_t1
10010      b $80,BOLDON,"Q",PLAINTEXT,0
10011
10012 :MT3D4E      b "Print"
10013      b GOTOX
10014      w mnu2_t1
10015      b $80,BOLDON,"P",PLAINTEXT,0
10016
10017 :MT3D5D      b "Delete"
10018      b GOTOX
10019      w mnu2_t1
10020      b $80,BOLDON,"D",PLAINTEXT,0
10021
10022 :MT3D6C      b "Sort directory"
10023      b GOTOX
10024      w mnu2_t1
10025      b $80,BOLDON,"T",PLAINTEXT,0
10026 endif
10027

```

```

;--- Hauptmenü/Anzeige:                                     10028
; TD13: Anzeige_Menü                                       10029
if LANG = LANG_DE                                           10030
:mnu3_of = $0038                                           10031
:mnu3_xw = $0050                                           10032
endif                                                       10033
if LANG = LANG_EN                                           10034
:mnu3_of = $0030                                           10035
:mnu3_xw = $0048                                           10036
endif                                                       10037
:mnu3_x0 = mnu0_x0 +mnu3_of                                10038
:mnu3_x1 = mnu3_x0 +mnu3_xw -1                             10039
:mnu3_y0 = $10                                              10040
:mnu3_y1 = $87                                              10041
;mnu3_t1 = mnu3_x1 -22                                       10042
:DS3D87             jsr    Z3BBE                ;Grafik in REU/Buf2. 10043
                                                           10044
                                                           b mnu3_y0,mnu3_y1      10045
                                                           w mnu3_x0,mnu3_x1      10046
                                                           10047
::_x3D91            b 8 ! VERTICAL ! MENU_VMODE            10048
                                                           10049
::_x3D92            w MT3DB9                ;Icon-Modus        10050
                                                           b MENU_ACTION          10051
                                                           w MR3E1E                10052
                                                           10053
                                                           w MT3DC6                ;Text-Modus            10054
                                                           b MENU_ACTION          10055
                                                           w MR3E1E                10056
                                                           10057
                                                           w MT3DD3                ;Text/Name            10058
                                                           b MENU_ACTION          10059
                                                           w MR3E1E                10060
                                                           10061
                                                           w MT3DE0                ;Text/Datum          10062
                                                           b MENU_ACTION          10063
                                                           w MR3E1E                10064
                                                           10065
                                                           w MT3DED                ;Text/Größe          10066
                                                           b MENU_ACTION          10067
                                                           w MR3E1E                10068
                                                           10069
                                                           w MT3DFA                ;Text/Typ            10070
                                                           b MENU_ACTION          10071
                                                           w MR3E1E                10072
                                                           10073
                                                           w MT3E11                ;Anzeige/KByte      10074
                                                           b MENU_ACTION          10075
                                                           w MR3E24                10076
                                                           10077
                                                           w MT3E05                ;Anzeige/Blocks     10078
                                                           b MENU_ACTION          10079
                                                           w MR3E34                10080
                                                           10081
if LANG = LANG_DE                                           10082

```

```

10083 :MT3DB9          b "*" Icon-Modus",0
10084 :MT3DC6          b "  Text-Modus",0
10085 :MT3DD3          b "  nach Name",0
10086 :MT3DE0          b "  nach Datum",0
10087 :MT3DED          b "  nach Größe",0
10088 :MT3DFA          b "  nach Typ",0
10089 endif
10090 if LANG = LANG_EN
10091 :MT3DB9          b "*" Icon mode",0
10092 :MT3DC6          b "  Text mode",0
10093 :MT3DD3          b "  by name",0
10094 :MT3DE0          b "  by date",0
10095 :MT3DED          b "  by size",0
10096 :MT3DFA          b "  by type",0
10097 endif
10098
10099 .MT3E05          b ""
10100 :T3E09           b "  KBytes",0
10101 ;--- TD.WIN
10102 .MT3E11          b "  "
10103 ;---
10104 :T3E15           b "  Blocks",0
10105
10106 ; Anzeige: Icon-/Text-Modus.
10107 :MR3E1E          jsr   S3E48                ;Icon-/Text-Modus.
10108                  jmp   S4321                ;Alle Fenster neu.
10109
10110 ; Anzeige: Größe in KBytes.
10111 :MR3E34          lda   #""                  ;Anzeige/Blocks
10112                  ldx   #" "
10113                  bne   Z3E28
10114
10115 ; Anzeige: Größe in Blocks.
10116 :MR3E24          lda   #" "                  ;Anzeige/KByte
10117                  ldx   #""
10118 :Z3E28           sta   MT3E05
10119                  stx   MT3E11
10120                  jsr   GotoFirstMenu
10121                  jmp   S157E                ;Alle Fens. anzeigen.
10122
10123 ; Anzeige-Menü initialisieren.
10124 ; Übergabe: A = Icon-/Text-Modus.
10125 ; Aufruf aus Hauptmodul und mod.#10.
10126 .S3E48           tax
10127
10128                  ldy   #" "                  ;Modus löschen:
10129                  sty   MT3DB9                ;Icon-Modus.
10130                  sty   MT3DC6                ;Text-Modus.
10131                  sty   MT3DD3                ;Nach Name.
10132                  sty   MT3DE0                ;Nach Datum.
10133                  sty   MT3DED                ;Nach Größe.
10134                  sty   MT3DFA                ;Nach Typ.
10135
10136                  asl                           ;Zeiger auf neuen
10137                  tay                           ;Modus berechnen.

```

```

        lda    :data +0,y          10138
        sta    r0L                  10139
        lda    :data +1,y          10140
        sta    r0H                  10141
                                     10142
        ldy    #$00                 10143
        lda    #""                  10144
        sta    (r0),y               ;Modus aktivieren. 10145
                                     10146
        txa                          ;Icon-/Text-Modus? 10147
        bne    :1                   ; => Text-Modus. 10148
                                     10149
        jmp    S37B2                ;Anzeige: Icons. 10150
::1      jmp    Z37AF                ;Anzeige: Text. 10151
                                     10152
; Tabelle mit Zeigern auf Anzeige-Modi. 10153
::data   w MT3DB9                  10154
        w MT3DC6                  10155
        w MT3DD3                  10156
        w MT3DE0                  10157
        w MT3DED                  10158
        w MT3DFA                  10159
                                     10160
;--- Hauptmenü/Fenster: 10161
; TD13: WindowMenü 10162
if LANG = LANG_DE 10163
:mnu5_of = $0078 10164
:mnu5_xw = $0090 10165
endif 10166
if LANG = LANG_EN 10167
:mnu5_of = $0070 10168
:mnu5_xw = $0080 10169
endif 10170
:mnu5_x0 = mnu0_x0 +mnu5_of 10171
:mnu5_x1 = mnu5_x0 +mnu5_xw -1 10172
                                     10173
if ENABLE_ANIMATE = FALSE 10174
:mnu5_o1 = 0 10175
else 10176
:mnu5_o1 = 1 10177
endif 10178
if ENABLE_DIFFICOL = FALSE 10179
:mnu5_o2 = 0 10180
endif 10181
if ENABLE_DIFFICOL = TRUE 10182
:mnu5_o2 = 1 10183
endif 10184
:mnu5_me = 7 +mnu5_o1 +mnu5_o2 10185
                                     10186
:mnu5_y0 = $10 10187
:mnu5_y1 = (mnu5_y0 +mnu5_me*14)!%00000111 10188
                                     10189
:mnu5_t1 = mnu5_x1 -22 10190
:DS3E86      jsr    Z3BBE          ;Grafik in REU/Buf2. 10191
                                     10192

```

```

10193      b mnu5_y0,mnu5_y1
10194      w mnu5_x0,mnu5_x1
10195
10196 ::_x3E8F      b mnu5_me ! VERTICAL ! MENU_VMODE
10197
10198 ::_x3E90      w MT3EB3      ;Fenster platzieren
10199      b MENU_ACTION
10200      w MR5652
10201
10202      w MT3EC5      ;Alle schließen
10203      b MENU_ACTION
10204      w MR4277
10205
10206      w MT3EDB      ;Alle Dateien wählen
10207      b MENU_ACTION
10208      w MR589D
10209
10210      w MT3EF6      ;Dateien wählen
10211      b MENU_ACTION
10212      w MR58A0
10213
10214      w MT3F19      ;Einstell. speichern
10215      b MENU_ACTION
10216      w MR3824
10217
10218      w MT3F38      ;Autom. platzieren
10219      b MENU_ACTION
10220      w MR3F6E
10221
10222      w MT3F58      ;Versch. Farben
10223      b MENU_ACTION
10224      w MR3F86
10225
10226 if ENABLE_DIFFICOL = TRUE
10227      w MT3F59      ;Versch. Iconfarben.
10228      b MENU_ACTION
10229      w MR3F87
10230 endif
10231
10232 if ENABLE_ANIMATE = TRUE
10233      w MT4048      ;Animation EIN/AUS
10234      b MENU_ACTION
10235      w MR404A
10236 endif
10237
10238 if LANG = LANG_DE
10239 :MT3EB3      b "Fenster platzieren"
10240      b GOTOX
10241      w mnu5_t1
10242      b $80,BOLDON,"S",PLAINTEXT,0
10243
10244 :MT3EC5      b "Alle Fenster schließen"
10245      b GOTOX
10246      w mnu5_t1
10247      b $80,BOLDON,"C",PLAINTEXT,0

```

		10248
:MT3EDB	b "Alle Dateien wählen"	10249
	b GOTOX	10250
	w mnu5_t1	10251
	b \$80,BOLDON,"W",PLAINTEXT,0	10252
		10253
:MT3EF6	b "Dateien im Fenster wählen"	10254
	b GOTOX	10255
	w mnu5_t1	10256
	b \$80,BOLDON,"X",PLAINTEXT,0	10257
		10258
:MT3F19	b "Einstellungen speichern"	10259
	b GOTOX	10260
	w mnu5_t1	10261
	b \$80,BOLDON,"I",PLAINTEXT,0	10262
		10263
:MT3F38	b " "	10264
	b " Automatisch platzieren"	10265
	b GOTOX	10266
	w mnu5_t1	10267
	b \$80,BOLDON,"L",PLAINTEXT,0	10268
endif		10269
if LANG!ENABLE_DIFFICOL = LANG_DE!FALSE		10270
.MT3F58	b " "	10271
	b " Verschiedene Farben",0	10272
endif		10273
if LANG!ENABLE_DIFFICOL = LANG_DE!TRUE		10274
.MT3F58	b " "	10275
	b " Verschiedene Fensterfarben",0	10276
.MT3F59	b " "	10277
	b " Verschiedene Iconfarben",0	10278
endif		10279
if LANG!ENABLE_ANIMATE = LANG_DE!TRUE		10280
:MT4048	b " "	10281
	b " Fensteranimation",0	10282
endif		10283
		10284
if LANG = LANG_EN		10285
:MT3EB3	b "Arrange windows"	10286
	b GOTOX	10287
	w mnu5_t1	10288
	b \$80,BOLDON,"S",PLAINTEXT,0	10289
		10290
:MT3EC5	b "Close all windows"	10291
	b GOTOX	10292
	w mnu5_t1	10293
	b \$80,BOLDON,"C",PLAINTEXT,0	10294
		10295
:MT3EDB	b "Select all files"	10296
	b GOTOX	10297
	w mnu5_t1	10298
	b \$80,BOLDON,"W",PLAINTEXT,0	10299
		10300
:MT3EF6	b "Select page files"	10301
	b GOTOX	10302

```

10303          w mnu5_t1
10304          b $80,BOLDON,"X",PLAINTEXT,0
10305
10306 :MT3F19          b "Save settings"
10307          b GOTOX
10308          w mnu5_t1
10309          b $80,BOLDON,"I",PLAINTEXT,0
10310
10311 :MT3F38          b " "
10312          b " Place automatically"
10313          b GOTOX
10314          w mnu5_t1
10315          b $80,BOLDON,"L",PLAINTEXT,0
10316 endif
10317 if LANG!ENABLE_DIFFICOL = LANG_EN!FALSE
10318 .MT3F58          b " "
10319          b " Different colors",0
10320 endif
10321 if LANG!ENABLE_DIFFICOL = LANG_EN!TRUE
10322 .MT3F58          b " "
10323          b " Different window colors",0
10324 .MT3F59          b " "
10325          b " Different icon colors",0
10326 endif
10327 if LANG!ENABLE_ANIMATE = LANG_EN!TRUE
10328 :MT4048          b " "
10329          b " Window animation",0
10330 endif
10331
10332 ; Hinweis:
10333 ; ReDoMenu nicht möglich, da die Farben
10334 ; nicht zurückgesetzt werden!
10335 ;
10336 ; Fenster automatisch platzieren.
10337 ;
10338 ; Hinweis:
10339 ; Ist "Automatisch platzieren" aktiv,
10340 ; dann werden die Fenster am Ende neu
10341 ; gezeichnet und nicht "geöffnet".
10342 ; Dadurch funktioniert aktuell die
10343 ; Animation nicht mit dieser Option.
10344 ;
10345 ; TD13: AutoSwap
10346 :MR3F6E          jsr   GotoFirstMenu
10347
10348 if ENABLE_ANIMATE = TRUE
10349          ldy   MT4048          ;Animation EIN.
10350 endif
10351          lda   MT3F38          ;Autom. platzieren.
10352          cmp   #""
10353          beq   :1
10354
10355 if ENABLE_ANIMATE = TRUE
10356          ldy   #" "          ;Animation AUS.
10357 endif

```



```

        lda    #""          ; => Einschalten.          10358
        b $2c                10359
::1      lda    #" "         ; => Ausschalten.          10360
        sta    MT3F38        ;Autom. platzieren.        10361
if ENABLE_ANIMATE = TRUE                10362
        sty    MT4048        ;Animation EIN/AUS.        10363
endif                                      10364
                                      10365
        jsr    S09DD         ;Konfig in REU/Datei.        10366
        jmp    Z5655         ;Fenster platzieren.        10367
                                      10368
; Verschiedene Fensterfarben für Fenster #1-#4.          10369
; TD13: n.v.                                              10370
:MR3F86      jsr    GotoFirstMenu          10371
                                      10372
        lda    MT3F58        ;Versch. Farben.          10373
        cmp    #""          10374
        beq    :1            10375
        lda    #""          ; => Einschalten.          10376
        b $2c                10377
::1      lda    #" "         ; => Ausschalten.          10378
        sta    MT3F58        ;Versch. Farben.          10379
                                      10380
if ENABLE_DIFFICOL = TRUE                10381
        sta    MT3F59        ;Versch. Iconfarben.        10382
endif                                      10383
                                      10384
        jsr    S09DD         ;Konfig in REU/Datei.        10385
                                      10386
; Hinweis:                                              10387
; Werden die Fenster nur neu angezeigt,                  10388
; dann wird bei "Verschiedene Farben"                    10389
; nicht nach PadColorPref gesucht und                    10390
; das Fenster ggf. mit falschen Farben                    10391
; angezeigt.                                              10392
;                                                         10393
        jmp    S157E         ;Alle Fens. anzeigen.        10393
        jmp    S4320         ;Fenster neu laden.          10394
                                      10395
; Verschiedene Iconfarben für Fenster #1-#4.              10396
; TD13: n.v.                                              10397
if ENABLE_DIFFICOL = TRUE                10398
:MR3F87      jsr    GotoFirstMenu          10399
                                      10400
        lda    MT3F59        ;Versch. Iconfarben.        10401
        cmp    #""          10402
        beq    :1            10403
        lda    #""          ; => Einschalten.          10404
        b $2c                10405
::1      lda    #" "         ; => Ausschalten.          10406
        sta    MT3F59        ;Versch. Iconfarben.        10407
                                      10408
        jsr    S09DD         ;Konfig in REU/Datei.        10409
                                      10410
; Hinweis:                                              10411
; Werden die Fenster nur neu angezeigt,                  10412

```

```

10413 ; dann wird bei "Verschiedene Farben"
10414 ; nicht nach PadColorPref gesucht und
10415 ; das Fenster ggf. mit falschen Farben
10416 ; angezeigt.
10417 ;                               jmp    S157E                ;Alle Fens. anzeigen.
10418 ;                               jmp    S4320                ;Fenster neu laden.
10419 endif
10420
10421 ; Fensteranimation ein/aus.
10422 ;
10423 ; Hinweis:
10424 ; Ist "Automatisch platzieren" aktiv,
10425 ; dann werden die Fenster am Ende neu
10426 ; gezeichnet und nicht "geöffnet".
10427 ; Dadurch funktioniert aktuell die
10428 ; Animation nicht mit dieser Option.
10429 ;
10430 if ENABLE_ANIMATE = TRUE
10431 :MR404A                jsr    GotoFirstMenu
10432
10433 ;                               ldy    MT3F38                ;Autom. platzieren.
10434 ;                               lda    MT4048                ;Animation EIN/AUS:
10435 ;                               cmp    #""
10436 ;                               beq    :1
10437
10438 ;                               ldy    #" "                  ;Autom. platz. AUS.
10439 ;                               lda    #"*"                  ; => Einschalten.
10440 ;                               b $2c
10441 ::1                    ;                               lda    #" "                  ; => Ausschalten.
10442 ;                               sta    MT4048                ;Modus speichern.
10443 ;                               sty    MT3F38                ;Autom. platzieren.
10444
10445 ;                               jmp    S09DD                ;Konfig in REU/Datei.
10446 endif
10447
10448 ;--- Hauptmenü/Optionen:
10449 ; TD13: Speziell_Menü
10450 if ENABLE_GEOHELP = TRUE
10451 :mnu6_nr = 9
10452 :mnu6_yh = $7f
10453 else
10454 :mnu6_nr = 8
10455 :mnu6_yh = $77
10456 endif
10457 if LANG = LANG_DE
10458 :mnu6_of = $0098
10459 endif
10460 if LANG = LANG_EN
10461 :mnu6_of = $00a0
10462 endif
10463 :mnu6_xw = $0068
10464 :mnu6_x0 = mnu0_x0 +mnu6_of
10465 :mnu6_x1 = mnu6_x0 +mnu6_xw -1
10466 :mnu6_y0 = $10
10467 :mnu6_y1 = mnu6_y0 +mnu6_yh

```

```

:mnu6_t1 = mnu6_x1 -20                                     10468
:DS3F9E      lda    #" "                                     10469
              ldx    C0428                                   ;RAM-TopDesk?      10470
              beq    :1a                                     ; => Nein, weiter... 10471
              lda    #"*"                                    10472
::1a         sta    MT3FFC                                   ;Menü anpassen.    10473
              ldx    C04B5                                   ;Kopieren-Dialog?  10476
              beq    :1b                                     ; => Nein, weiter... 10477
              lda    #"*"                                    10478
::1b         sta    MT3FFD                                   ;Menü anpassen.    10479
              jsr    Z3BBE                                   ;Grafik in REU/Buf2. 10481
              b mnu6_y0,mnu6_y1                             10483
              w mnu6_x0,mnu6_x1                             10484
::_x3FB1     b mnu6_nr ! VERTICAL ! MENU_VMODE             10486
::_x3FB2
if ENABLE_GEOHELP = TRUE                                   10490
              w MT4031                                       ;Hilfe aufrufen    10491
              b MENU_ACTION                                  10492
              w MR3A95                                       10493
endif                                                  10494
              w MT4017                                       ;Reset             10496
              b MENU_ACTION                                  10497
              w MR432B                                       10498
              w MT3FDA                                       ;Neuer Ordner      10500
              b MENU_ACTION                                  10501
              w MR39C8                                       10502
              w MT56FE                                       ;Neues Verzeichnis 10504
              b MENU_ACTION                                  10505
              w MR39B9                                       10506
              w MT3FEE                                       ;Uhrzeit ändern    10508
              b MENU_ACTION                                  10509
              w MR39DF                                       10510
              w MT4040                                       ;Farben ändern     10512
              b MENU_ACTION                                  10513
              w MR3A2F                                       10514
              w MT3FFD                                       ;Kopieren-Dialog   10516
              b MENU_ACTION                                  10517
              w MR4024                                       10518
              w MT3FFC                                       ;RAM-TopDesk       10520
              b MENU_ACTION                                  10521
              w MR3A3E                                       10522

```

```

10523
10524          w MT4023          ;GEOS beenden
10525          b MENU_ACTION
10526          w MR39CF
10527
10528 if LANG!ENABLE_GEOHELP = LANG_DE!TRUE
10529 :MT4031          b "GeoHelpView starten",0
10530 endif
10531 if LANG!ENABLE_GEOHELP = LANG_EN!TRUE
10532 :MT4031          b "Open GeoHelpView",0
10533 endif
10534
10535 if LANG = LANG_DE
10536 :MT4017          b "Fenster-Reset"
10537          b GOTOX
10538          w mnu6_t1
10539          b $80,BOLDON,"R",PLAINTEXT,0
10540
10541 :MT3FDA          b "Neuer Ordner"
10542          b GOTOX
10543          w mnu6_t1
10544          b $80,BOLDON,"O",PLAINTEXT,0
10545
10546 :MT56FE          b "Neues Verzeichnis"
10547          b GOTOX
10548          w mnu6_t1
10549          b $80,BOLDON,"U",PLAINTEXT,0
10550
10551 :MT3FEE          b "Datum/Uhrzeit"
10552          b GOTOX
10553          w mnu6_t1
10554          b $80,BOLDON,"A",PLAINTEXT,0
10555
10556 :MT4040          b "Farben ändern",0
10557
10558 :MT3FFD          b " Kopieren-Dialogbox",0
10559
10560 :MT3FFC          b " RAM-DeskTop",0
10561
10562 :MT4023          b BOLDON
10563          b "GEOS beenden",PLAINTEXT,0
10564 endif
10565 if LANG = LANG_EN
10566 :MT4017          b "Reset windows"
10567          b GOTOX
10568          w mnu6_t1
10569          b $80,BOLDON,"R",PLAINTEXT,0
10570
10571 :MT3FDA          b "New folder"
10572          b GOTOX
10573          w mnu6_t1
10574          b $80,BOLDON,"O",PLAINTEXT,0
10575
10576 :MT56FE          b "New subdirectory"
10577          b GOTOX

```

```

                                w mnu6_t1                                10578
                                b $80,BOLDON,"U",PLAINTEXT,0          10579
                                                                10580
:MT3FEE                        b "Set date/time"                      10581
                                b GOTOX                                10582
                                w mnu6_t1                                10583
                                b $80,BOLDON,"A",PLAINTEXT,0          10584
                                                                10585
:MT4040                        b "Change colors",0                    10586
                                                                10587
:MT3FFD                        b "  Ask for copy",0                  10588
                                                                10589
:MT3FFC                        b "  RAM desktop",0                    10590
                                                                10591
:MT4023                        b BOLDON                                10592
                                b "Exit GEOS",PLAINTEXT,0            10593
endif                                                                    10594
                                                                10595
; Kopieren-Dialog ein/aus.                                            10596
:MR4024                        ldx  #" "                                10597
                                                                10598
                                lda  C04B5                            ;Kopieren-Dialog? 10599
                                eor  #$ff                                10600
                                sta  C04B5                                10601
                                beq  :1                                ; => Nein, weiter... 10602
                                                                10603
                                ldx  #"*"                                10604
::1                            stx  MT3FFD                            ;Menü anpassen. 10605
                                                                10606
                                jsr  S09DD                            ;Konfig in REU/Datei. 10607
                                                                10608
                                jmp  GotoFirstMenu                    ;Menü anzeigen. 10609
                                                                10610
;--- Hauptmenü/geos:                                                10611
; TD13: geos_Menü                                                    10612
:mnu1_of = $0000                                                        10613
:mnu1_xw = $0050                                                        10614
:mnu1_x0 = mnu0_x0 +mnu1_of                                            10615
:mnu1_x1 = mnu1_x0 +mnu1_xw -1                                         10616
:mnu1_y0 = $10                                                         10617
:mnu1_y1 = $57                                                         10618
:mnu1_t1 = mnu1_x1 -12                                                 10619
:DS404E                        jsr  Z3BBE                            ;Grafik in REU/Buf2. 10620
                                                                10621
                                b mnu1_y0,mnu1_y1                    10622
                                w mnu1_x0,mnu1_x1                    10623
                                                                10624
::_x4057                      b 5 ! VERTICAL ! MENU_VMODE            10625
                                                                10626
::_x4058                      w MT4071                            ;TopDesk-Info 10627
                                b MENU_ACTION                        ;DoMenu-Eintrag: 0 10628
                                w MR399E                                10629
                                                                10630
                                w MT4080                            ;Hilfsmittel 10631
                                b MENU_ACTION                        ;DoMenu-Eintrag: 1 10632

```

```

10633          w MR40BD
10634
10635          w MT408C          ;Anwendungen
10636          b MENU_ACTION    ;DoMenu-Eintrag: 2
10637          w MR40BD
10638
10639          w MT4098          ;Dokumente
10640          b DYN_SUB_MENU
10641          w DS41AE
10642
10643          w MT40A9          ;Sonstige
10644          b DYN_SUB_MENU
10645          w DS4249
10646
10647 if LANG = LANG_DE
10648 :MT4071          b "Info DTopDesk64",0
10649
10650 :MT4080          b "Hilfsmittel",0
10651
10652 :MT408C          b "Anwendungen",0
10653
10654 :MT4098          b "Dokumente"
10655          b GOTOX
10656          w mnu1_t1
10657          b BOLDON,">>",PLAINTEXT,0
10658
10659 :MT40A9          b "Sonstige"
10660          b GOTOX
10661          w mnu1_t1
10662          b BOLDON,">>",PLAINTEXT,0
10663 endif
10664 if LANG = LANG_EN
10665 :MT4071          b "Info DTopDesk64",0
10666
10667 :MT4080          b "Desk accessories",0
10668
10669 :MT408C          b "Applications",0
10670
10671 :MT4098          b "Documents"
10672          b GOTOX
10673          w mnu1_t1
10674          b BOLDON,">>",PLAINTEXT,0
10675
10676 :MT40A9          b "Other files"
10677          b GOTOX
10678          w mnu1_t1
10679          b BOLDON,">>",PLAINTEXT,0
10680 endif
10681
10682 ;*** Menü-Routinen.
10683 :MR40BD          pha          ;Hilfsm./Anwendungen.
10684          jsr    GotoFirstMenu
10685          pla
10686
10687 ; Datei öffnen:

```

```

; - OFF_OPEN_APPL :      1=Anwendungen      10688
; - OFF_OPEN_DA   :      2=Hilfsmittel      10689
; - OFF_OPEN_DOC  :      3- 9=Dokumente      10690
; - OFF_OPEN_PRNT :      10=Druckertreiber   10691
; - OFF_OPEN_INPT :      11=Eingabetreiber   10692
; - OFF_OPEN_OTHER: 12/13=AutoExec/NichtGEOS 10693
:Z40C1          sta    V40C2                ;Dateityp speichern. 10694
                                           10695
                        ldx    C042A          ;Oberstes Fenster. 10696
                        lda    C0448,x        ;Fenster geöffnet? 10697
                        beq    :exit          ; => Nein, Ende... 10698
                                           10699
                        jmp     Z39BF          ;Modul laden.      10700
                                           10701
::exit          rts                        10702
                                           10703
;--- geos/Dokumente: 10704
:mnu11_of = mnu1_xw 10705
:mnu11_xw = $0040 10706
:mnu11_x0 = mnu0_x0 +mnu11_of 10707
:mnu11_x1 = mnu11_x0 +mnu11_xw -1 10708
:mnu11_y0 = $40 10709
:mnu11_y1 = $a7 10710
;mnu11_t1 = mnu11_x1 -20 10711
:DS41AE          jsr    Z3BBB                ;Grafik in REU/Buf1. 10712
                                           10713
                        b mnu11_y0,mnu11_y1 10714
                        w mnu11_x0,mnu11_x1 10715
                                           10716
::_x41B7          b MAX_OPEN_DOC ! VERTICAL ! MENU_VMODE 10717
                                           10718
::_x41B9          w MT5082                    ;Alle Dokumente. 10719
                        b MENU_ACTION          ;DoMenu-Eintrag: 0 10720
                        w MR4209              10721
                                           10722
                        w MT41DB                ;GeoWrite 10723
                        b MENU_ACTION          ;DoMenu-Eintrag: 1 10724
                        w MR4209              10725
                                           10726
                        w MT41E4                ;GeoPaint 10727
                        b MENU_ACTION          ;DoMenu-Eintrag: 2 10728
                        w MR4209              10729
                                           10730
                        w MT60D7                ;GeoFile 10731
                        b MENU_ACTION          ;DoMenu-Eintrag: 3 10732
                        w MR4209              10733
                                           10734
                        w MT41ED                ;GeoCalc 10735
                        b MENU_ACTION          ;DoMenu-Eintrag: 4 10736
                        w MR4209              10737
                                           10738
                        w MT41F5                ;GeoPublish 10739
                        b MENU_ACTION          ;DoMenu-Eintrag: 5 10740
                        w MR4209              10741
                                           10742

```

```

10743          w MT4200          ;GeoChart
10744          b MENU_ACTION      ;DoMenu-Eintrag: 6
10745          w MR4209
10746
10747 if LANG = LANG_DE
10748 :MT5082          b "Alle",0
10749 endif
10750 if LANG = LANG_EN
10751 :MT5082          b "All docs",0
10752 endif
10753
10754 :MT60D7          b "GeoFile",0
10755 :MT41DB          b "GeoWrite",0
10756 :MT41E4          b "GeoPaint",0
10757 :MT41ED          b "GeoCalc",0
10758 :MT41F5          b "GeoPublish",0
10759 :MT4200          b "GeoChart",0
10760
10761 ;*** Menü-Routinen.
10762 :MR4209          clc          ;geos/Dokumente.
10763          adc    #OFF_OPEN_DOC
10764
10765 :Z420C          pha
10766          jsr    DoPreviousMenu
10767          pla
10768          jmp    Z40C1          ;Datei wählen/öffnen.
10769
10770 ;--- geos/Sonstige:
10771 :mnu12_of = mnu1_xw
10772 :mnu12_xw = $0040
10773 :mnu12_x0 = mnu0_x0 +mnu12_of
10774 :mnu12_x1 = mnu12_x0 +mnu12_xw -1
10775 :mnu12_y0 = $48
10776 :mnu12_y1 = $67
10777 ;:mnu12_t1 = mnu12_x1 -20
10778 :DS4249          jsr    Z3BBB          ;Grafik in REU/Buf1.
10779
10780          b mnu12_y0,mnu12_y1
10781          w mnu12_x0,mnu12_x1
10782
10783 ::_x4252          b MAX_OPEN_OTHER ! VERTICAL ! MENU_VMODE
10784
10785 ::_x4253          w MT425D          ;AutoExec
10786          b MENU_ACTION      ;DoMenu-Eintrag: 0
10787          w MR4271
10788
10789          w MT4266          ;BASIC-Programme
10790          b MENU_ACTION      ;DoMenu-Eintrag: 1
10791          w MR4271
10792
10793 if LANG = LANG_DE
10794 :MT425D          b "AUTOEXEC",0
10795 :MT4266          b "nicht GEOS",0
10796 endif
10797 if LANG = LANG_EN

```



```

:MT425D      b "AUTOEXEC",0      10798
:MT4266      b "not GEOS",0      10799
endif      10800
      10801
;*** Menü-Routinen.      10802
:MR4271      clc      ;geos/Sonstige.      10803
      adc      #OFF_OPEN_OTHER      10804
      jmp      Z420C      10805
      10806
; Alle Fenster schließen.      10807
; TD13: CloseAll      10808
:MR4277      jsr      GotoFirstMenu      ;Alle schließen      10809
      10810
      jsr      Z1D7D      ;Titel deaktivieren.      10811
      10812
      ldx      #4 -1      ;Max. 4 Fenster.      10813
;      lda      #$00      ;Fensterinfo löschen.      10814
::1      jsr      S09B5      ;CMD-Info löschen.      10815
      sta      C0448,x      ;Fenster geschlossen.      10816
      dex      10817
      bpl      :1      10818
      10819
      jsr      S524C      ;Dat.-Ausw. aufheben.      10820
      jmp      S157E      ;Alle Fens. anzeigen.      10821
      10822
; Oberstes Fenster aktualisieren.      10823
;      10824
; Hinweis:      10825
; Im Textmodus wird der Bereich für      10826
; VLIR-Module überschrieben!      10827
.S4296      ldx      C042A      ;Oberstes Fenster.      10828
      10829
; Fenster aktualisieren.      10830
; Übergabe: X = Fenster-Nr.      10831
:Z4299      10832
      10833
; Optional:      10834
; PadColorPref vom aktuellen Laufwerk      10835
; neu einlesen.      10836
if FALSE      10837
      txa      10838
      pha      ;Fenster-Nr.      10839
      jsr      Z4330      ;Prefs./PadCol laden.      10840
      pla      10841
      tax      10842
endif      10843
      10844
      jsr      S5A3B      ;Gleiche Fens.suchen.      10845
      10846
      ldx      #4 -1      ;Max. 4 Fenster.      10847
::1      lda      a6,x      ;>0 = Fenster gleich?      10848
      bne      :2      ; => Ja, weiter...      10849
      dex      10850
      bpl      :1      10851
      10852

```

```

10853 ; Keine gleichen Fenster.
10854         jmp     Z42CD                ;Oberstes Fens. neu.
10855
10856 ; Inhalt gleicher Fenster aktualisieren.
10857 ::2         ldx     #4 -1
10858 ::3         lda     a6,x                ;0 = Fens. ungleich?
10859         beq     :5                ; => Ja, weiter...
10860
10861         stx     WIN042F                ;Aktives Fenster.
10862
10863         jsr     Z4576                ;Dat. neu einlesen.
10864         bcc     :4                ; => OK, weiter...
10865
10866 ; Hinweis:
10867 ; Wenn die Dateien nicht eingelesen
10868 ; wurden, dann ist der Dateispeicher
10869 ; leer -> Leeres Fenster.
10870 ; Fehler ausgeben, Fenster schließen,
10871 ; weiter mit dem nächsten Fenster.
10872 if FALSE
10873         jmp     S5CF3                ;Fehler ausgeben.
10874 else
10875 ; Benötigt Fenster-Nr. in WIN042F!
10876         jsr     Z5CEB                ;Fehler ausgeben und
10877         ;Fenster schließen.
10878 endif
10879
10880 ::4         ldx     WIN042F                ;Aktives Fenster.
10881
10882 ::5         dex
10883         bpl     :3                ; => Weiter...
10884
10885         ldx     C042A                ;Oberstes Fenster.
10886         stx     WIN042F                ;Aktives Fenster.
10887         jsr     Z4576                ;Dat. neu einlesen.
10888 if FALSE
10889         bcc     :ok                ; => OK, weiter...
10890
10891 ; Fehler ausgeben und alle Fenster
10892 ; neu anzeigen.
10893         jsr     S5CF3                ;Fehler ausgeben.
10894 ::ok        jmp     S157E                ;Alle Fens. anzeigen.
10895 else
10896         bcs     x42CE                ; => Fehler...
10897         jmp     S157E                ;Alle Fens. anzeigen.
10898 endif
10899
10900 ; Oberstes Fenster aktualisieren.
10901 :Z42CD      ldx     C042A                ;Oberstes Fenster.
10902         stx     WIN042F                ;Aktives Fenster.
10903         jsr     Z4576                ;Dat. neu einlesen.
10904 if FALSE
10905         bcc     :ok                ; => Kein Fehler...
10906
10907 ; Fehler ausgeben und oberstes Fenster

```

```

; neu anzeigen. 10908
                jsr    S5CF3                ;Fehler ausgeben. 10909
::ok            jmp    Z15CB                ;Oberstes Fens. neu. 10910
else            10911
                bcs    x42CE                ; => Fehler... 10912
                jmp    Z15CB                ;Oberstes Fens. neu. 10913
                10914
; Hinweis: 10915
; Wenn die Dateien nicht eingelesen 10916
; wurden, dann ist der Dateispeicher 10917
; leer -> Leeres Fenster. 10918
; Fehler ausgeben, Fenster schließen. 10919
; 10920
; Benötigt Fenster-Nr. in WIN042F! 10921
:x42CE          jmp    Z5CEB                ;Fehler ausgeben und 10922
                ;Fenster schließen. 10923
endif 10924
                10925
; DnD-Modus beenden. 10926
; Sprite#1 abschalten, Fenstergrenzen zurücksetzen. 10927
; TD13: EndGhost 10928
; 10929
; Hinweis: 10930
; Darf Inhalt von Akku nicht verändern! 10931
; Die Routine ":Handler" testet den 10932
; Inhalt des Akku auf Fensterbefehle. 10933
:Z42DB          pha 10934
                10935
                jsr    Z1962                ;Fenstergrenzen. 10936
                10937
                ldx    #$00 10938
                stx    V0423                ;DnD deaktivieren. 10939
                inx 10940
;                ldx    #1 10941
                stx    r3L 10942
                10943
                jsr    DisablSprite 10944
                10945
                pla 10946
                rts 10947
                10948
; Kopiere Name für Datei-Nr. in DIRBUF in Speicher. 10949
; Übergabe: A = Dateinummer 0-254 10950
; Rückgabe: r0 = Zeiger auf Eintrag in DIRBUF. 10951
;                r1 = Zeiger auf Dateiname, Ende NULL-Byte. 10952
; TD13: GetName2 10953
:Z42EC          sta    r0L                ;Datei-Nr. 10954
                lda    #$00 10955
                sta    r0H 10956
                10957
;                lda    #> 18                ;16Z. + Tr/Se für 10958
                sta    r1H                ;Infoblock = 18. 10959
                lda    #< 18 10960
                sta    r1L 10961
                10962

```

```

10963         ldx    #r0
10964         ldy    #r1
10965         jsr    DMult
10966
10967         ldx    C042A           ;Oberstes Fenster.
10968         jsr    S599A           ;Zeiger auf DIRBUF.
10969
10970         lda    r1L             ;Zeiger auf Dateiname
10971         clc                     ;berechnen.
10972         adc    r0L
10973         sta    r0L
10974         lda    r1H
10975         adc    r0H
10976         sta    r0H
10977
10978         jmp    Z46CA           ;Dateiname kopieren.
10979
10980 ;*** Menü-Routinen.
10981 :MR432B      jsr    GotoFirstMenu ;Reset.
10982
10983 ; Alle Fenster neu zeichnen.
10984 .S4320       ldx    C042A           ;Oberstes Fenster.
10985             lda    C0448,x         ;Fenster geöffnet?
10986             beq    S4321           ; => Nein, weiter...
10987
10988             jsr    Z4330           ;Prefs./PadCol laden.
10989
10990 .S4321       jsr    S14F8           ;Bildschirm löschen.
10991
10992             ldx    #4 -1           ;Max. 4 Fenster.
10993 ::chk        lda    C0448,x         ;Fenster geöffnet?
10994             bne    :reset          ; => Ja, anzeigen...
10995             dex
10996             bpl    :chk
10997
10998 ::exit       rts
10999
11000 ::reset      ldx    #4 -1           ;Max. 4 Fenster.
11001 ::loop       txa
11002             pha
11003
11004             lda    C042A,x         ;Fenster-Nr. holen.
11005             pha
11006
11007             tax
11008             lda    C0448,x         ;Fenster geöffnet?
11009             beq    :2              ; => Nein, weiter...
11010
11011             jsr    Z4576           ;Dat. neu einlesen.
11012             bcc    :1              ; => OK, weiter...
11013
11014             pla
11015             tax                    ;Disk nicht bereit,
11016                                     ;Fenster schließen.
11017             jsr    S09B5           ;CMD-Info löschen.

```

```

;                                11018
;                                11019
;                                11020
;                                11021
;                                11022
;                                11023
::1                               11024
;                                11025
;                                11026
;                                11027
;                                11028
;                                11029
;                                11030
;                                11031
;                                11032
;                                11033
;                                11034
::2                               11035
;                                11036
::3                               11037
;                                11038
;                                11039
;                                11040
;                                11041
;                                11042
;                                11043
;                                11044
;                                11045
;                                11046
;                                11047
;                                11048
; Verschiedene Farben/Prefs laden. 11049
; Übergabe: X = Fenster-Nr.        11050
:Z4330                             11051
;                                11052
:Z4331                             11053
;                                11054
if ENABLE_DIFFICOL = FALSE         11055
;                                11056
;                                11057
endif                               11058
if ENABLE_DIFFICOL = TRUE          11059
;                                11060
;                                11061
endif                               11062
;                                11063
;                                11064
;                                11065
;                                11066
::skip                             11067
;                                11068
; Menüzeile vorbereiten.           11069
; TD13: RedrawHead                 11070
.S437D                             11071
;                                11072

```

```

11073 ; Hinweis:
11074 ; Evtl. leere Bereiche innerhalb der
11075 ; Menüzeile nicht löschen.
11076 if FALSE
11077         ldx    #(r4H - r2L)
11078 :::1      lda    :coord1,x
11079         sta    r2,x
11080         dex
11081         bpl    :1
11082
11083         lda    BackScrPattern    ;Füllmuster.
11084         jsr    S140C              ;Rechteck zeichnen.
11085         lda    screencolors      ;GEOS-Farben.
11086         jsr    S1561              ;Screencolor setzen.
11087 endif
11088
11089         jsr    S54F0              ;Hauptmenü starten.
11090
11091         ldx    #(r4H - r2L)      ;Koordinate für
11092 :::2      lda    :coord2,x        ;Anz. mark. Dateien.
11093         sta    r2,x
11094         dex
11095         bpl    :2
11096
11097         lda    COL0442           ;Farbe mark.Dateien.
11098         jsr    Z6100             ;Grafik löschen.
11099
11100         ldx    #(r4H - r2L)      ;Koordinate für
11101 :::3      lda    :coord3,x        ;TaskManager-Icon.
11102         sta    r2,x
11103         dex
11104         bpl    :3
11105
11106         lda    COL043D           ;Farbe GEOS-Icon.
11107         jsr    S1561              ;Screencolor setzen.
11108
11109 ; TD13: InitClock
11110 :::x60A5  jsr    Z5B60           ;Koord. für Uhr.
11111
11112         lda    COL0443           ;Farbe Datum/Uhrzeit.
11113         jsr    Z6100             ;Grafik löschen.
11114
11115 ; Hinweis:
11116 ; Optional zur sofortigen Anzeige von
11117 ; Datum/Uhrzeit nach dem Start.
11118 if FALSE
11119         jsr    S60DF              ;Uhr aktualisieren.
11120         jsr    S6118              ;Datum/Uhr anzeigen.
11121 endif
11122
11123         lda    #< :proc          ;Prozesstabelle.
11124         sta    r0L
11125         lda    #> :proc
11126         sta    r0H
11127

```

```

        lda    #$01                                11128
        jsr    InitProcesses                        11129
                                                11130
        ldx    #$00                                11131
        jmp    RestartProcess                      11132
                                                11133
; Hinweis:                                         11134
; Evtl. leere Bereiche innerhalb der              11135
; Menüzeile nicht löschen.                        11136
if FALSE                                         11137
::coord1      b MENU_Y0,MENU_Y1                    ;Menüzeile löschen. 11138
               w SCREEN_X0,SCREEN_X1                11139
endif                                              11140
                                                11141
::coord2      b SLCT_Y0,SLCT_Y1                    ;Mark.Dat. löschen. 11142
               w SLCT_X0,SLCT_X1                    11143
                                                11144
::coord3      b TASK_Y0,TASK_Y1                    ;Task-Icon löschen. 11145
               w TASK_X0,TASK_X1                    11146
                                                11147
; Prozess-Tabelle.                               11148
; TD13: ProzessTab                               11149
::proc        w S60DF                               11150
               w 50                                  11151
                                                11152
; Grafikbereich löschen.                         11153
; Übergabe: A = Farbe.                           11154
;           r2-r4 = Koordinaten.                  11155
:Z6100        pha                                   11156
                                                11157
        lda    #$00                                11158
        jsr    S140C                                ;Rechteck zeichnen. 11159
        lda    #%11111111                          11160
        jsr    FrameRectangle                      ;Rahmen zeichnen. 11161
                                                11162
        pla                                   ;Farbe einlesen. 11163
        jmp    S1561                                ;Screencolor setzen. 11164
                                                11165
; Routine für ":keyVector".                       11166
; TD13: KeyHandler                               11167
.R43DA        lda    menuNumber                    11168
               beq    :1                            11169
::exit        rts                                  11170
                                                11171
::1           tay                                   11172
                                                11173
::loop        lda    KeyTab,y                      11174
               beq    :2                            11175
               cmp    keyData                        11176
               beq    :exec                          11177
               iny                                   11178
               bne    :loop                          11179
                                                11180
; Hinweis:                                         11181
; Läßt man den bpl-Vergleich weg, dann            11182

```

```

11183 ; lassen sich Dateien mit den Tasten
11184 ; 0-9 markieren und ggf. öffnen bzw.
11185 ; für Drag'n'Drop auswählen.
11186 ; Mit bpl-Vergleich muss die C= Taste
11187 ; gedrückt werden um mit 0-9 Dateien
11188 ; zu markieren.
11189 ::2          lda    keyData
11190 ;           bpl     :exit          ;Nicht notwendig.
11191           and     #%01111111
11192
11193           cmp     #"0"
11194           bcc     :exit
11195           cmp     #"9" +1
11196           bcs     :exit
11197
11198           sec
11199           sbc     #"1"
11200           bpl     :3
11201
11202           lda     #9
11203
11204 ::3          ldx     C042A          ;Oberstes Fenster.
11205           ldy     C0448,x          ;Fenster geöffnet?
11206           beq     :exit          ; => Nein, Ende...
11207           cmp     V4CCB,x          ;Icon sichtbar?
11208           bcs     :exit          ; => Nein, Ende...
11209
11210           jmp     Z449B          ;Datei markieren.
11211
11212 ::exec
11213
11214 ; Hinweis:
11215 ; Tastenstatus nicht löschen, da sonst
11216 ; weitere Tasten im Tastaturpuffer
11217 ; nicht abgearbeitet werden.
11218 ;---
11219 ;           lda     pressFlag
11220 ;           and     #%01111111          ;Tastenstatus
11221 ;           sta     pressFlag          ;löschen.
11222 ;---
11223
11224           tya
11225           asl
11226           tay
11227
11228           lda     KeyServiceTab +0,y
11229           ldx     KeyServiceTab +1,y
11230           jmp     CallRoutine
11231
11232 ; TD13: KeyTab
11233 :KeyTab      b $e9    ; C= + I
11234             b $ea    ; C= + J
11235             b $ed    ; C= + M
11236             b $e4    ; C= + D
11237             b $f2    ; C= + R

```



b \$f1	; C= + Q	11238
b \$f6	; C= + V	11239
b \$f3	; C= + S	11240
b \$ee	; C= + N	11241
b \$e6	; C= + F	11242
b \$e5	; C= + E	11243
b \$fa	; C= + Y	11244
b \$e8	; C= + H	11245
b \$eb	; C= + K	11246
b \$f0	; C= + P	11247
b \$ef	; C= + O	11248
b \$a8	; C= + SHIFT + 8	11249
b \$a9	; C= + SHIFT + 9	11250
b \$bd	; C= + SHIFT + 0	11251
b \$a1	; C= + SHIFT + 1	11252
b \$01	; F1	11253
b \$03	; F3	11254
b \$05	; F5	11255
b \$0e	; F7	11256
b \$81	; C= + F1	11257
b \$83	; C= + F3	11258
b \$85	; C= + F5	11259
b \$8e	; C= + F7	11260
b \$f7	; C= + W	11261
b \$f8	; C= + X	11262
b \$e3	; C= + C	11263
b \$ec	; C= + L	11264
b \$f4	; C= + T	11265
b \$e1	; C= + A	11266
b \$10	; CRSR-UP	11267
b \$11	; CRSR-DOWN	11268
b \$94	; C= + <-	11269
b \$e2	; C= + B	11270
b \$f5	; C= + U	11271
b \$c9	; C= + SHIFT + I	11272
b \$1e	; CRSR-RIGHT	11273
b \$08	; CRSR-LEFT	11274
b \$12	; HOME	11275
b \$13	; CLR	11276
b \$02	; F2 (Quicklink#1 starten/ändern)	11277
b \$04	; F4 (Quicklink#2 starten/ändern)	11278
b \$06	; F6 (Quicklink#3 starten/ändern)	11279
b \$0f	; F8 (Quicklink#4 starten/ändern)	11280
b \$d8	; C= + SHIFT + X	11281
b \$14	; <-	11282
b \$21	; SHIFT + 1	11283
b \$22	; SHIFT + 2	11284
b \$40	; SHIFT + 3	11285
b \$24	; SHIFT + 4	11286
b \$18	; CTRL + X	11287
b \$16	; RUN/STOP	11288
b NULL		11289
:EOK		11290
.MAX_SHORTCUT	= EOK - KeyTab	11291
		11292

```

11293 ; TD13: KeyServiceTab
11294 :KeyServiceTab      w MR3824 ; C= + I
11295                    w MR3A26 ; C= + J
11296                    w MR39A1 ; C= + M
11297                    w MR5AB6 ; C= + D
11298                    w MR432B ; C= + R
11299                    w MR3A29 ; C= + Q
11300                    w MR3962 ; C= + V
11301                    w MR5652 ; C= + S
11302                    w MR39F9 ; C= + N
11303                    w KB39EC ; C= + F
11304                    w MR396E ; C= + E
11305                    w MR4E34 ; C= + Y
11306                    w MR39A4 ; C= + H
11307                    w MR3A1C ; C= + K
11308                    w MR4E31 ; C= + P
11309                    w MR39C8 ; C= + O
11310                    w IR3AAE ; C= + SHIFT + 8
11311                    w IR3AAB ; C= + SHIFT + 9
11312                    w IR3AA8 ; C= + SHIFT + 0
11313                    w IR3AA5 ; C= + SHIFT + 1
11314                    w IR3AAE ; F1
11315                    w IR3AAB ; F3
11316                    w IR3AA8 ; F5
11317                    w IR3AA5 ; F7
11318                    w KB3B7C ; C= + F1
11319                    w KB3B7F ; C= + F3
11320                    w KB3B82 ; C= + F5
11321                    w KB3B85 ; C= + F7
11322                    w MR589D ; C= + W
11323                    w MR58A0 ; C= + X
11324                    w MR4277 ; C= + C
11325                    w MR3F6E ; C= + L
11326                    w MR3A1F ; C= + T
11327                    w MR39DF ; C= + A
11328                    w KB490F ; CRSR-UP
11329                    w KB4912 ; CRSR-DOWN
11330                    w KB4EDA ; C= + <-
11331                    w KB1B60 ; C= + B
11332                    w MR39B9 ; C= + U
11333                    w MR3A2C ; C= + SHIFT + I
11334                    w KB4495 ; CRSR-RIGHT
11335                    w KB4907 ; CRSR-LEFT
11336                    w KB4901 ; HOME
11337                    w KB48F7 ; CLR
11338                    w KB4173 ; F2 (Quicklink#1 starten/ändern)
11339                    w KB4176 ; F4 (Quicklink#2 starten/ändern)
11340                    w KB4179 ; F6 (Quicklink#3 starten/ändern)
11341                    w KB417C ; F8 (Quicklink#4 starten/ändern)
11342                    w KB58A3 ; C= + SHIFT + X
11343                    w KB4E72 ; <- Ordner verlassen
11344                    w KB_W1 ; SHIFT + 1
11345                    w KB_W2 ; SHIFT + 2
11346                    w KB_W3 ; SHIFT + 3
11347                    w KB_W4 ; SHIFT + 4

```

```

                                w KB39CD ; CTRL + X                11348
                                w Z3B54  ; RUN/STOP                11349
                                                                    11350
; Datei im Akku markieren.                11351
; TD13: SelectFileA                        11352
:Z449B                                pha                        11353
                                ldy C042A                        ;Oberstes Fenster. 11354
                                lda C0448,y                    ;Fenster geöffnet? 11355
                                bne :1                          ; => Ja, weiter... 11356
                                pla                            11357
                                rts                            11358
                                                                    11359
::1                                sty WIN042F                ;Aktives Fenster. 11360
                                jsr Z4214                      ;X/Y für erstes Icon. 11361
                                pla                            11362
                                pha                            11363
                                jsr A2BB2                      ;Rechteck f.Eintrag. 11364
                                pla                            11365
                                tax                            11366
                                jmp Z4F1D                      ;Auswahl auswerten. 11367
                                                                    11368
; Nächstes Fenster öffnen.                11369
; TD13: OpenNext                          11370
:Z44D1                                ldx C042A                ;Oberstes Fenster. 11371
                                lda C0448,x                    ;Fenster geöffnet? 11372
                                beq :1                          ; => Nein, weiter... 11373
                                stx WIN042F                    ;Aktives Fenster. 11374
                                jsr S5842                      ;mark.Dat. anzeigen. 11375
                                jsr S524C                      ;Dat.-Ausw. aufheben. 11376
                                                                    11377
::1                                jsr Z1848                    ;Fenster frei? 11378
                                bcc :2                          ; => Ja, weiter... 11379
                                                                    11380
                                ldx #ERR_NO_WINDOW              ;Kein Fenster frei. 11381
                                jmp S5CE1                      ;Fehler ohne Refresh. 11382
                                                                    11383
::2                                11384
                                                                    11385
if ENABLE_ANIMATE = TRUE                11386
                                ldy #$ff                      ;Animation zeigen. 11387
                                b $2c                          11388
endif                                    11389
                                                                    11390
; Fenster erneut öffnen.                11391
; Übergabe: X = Fenster-Nr.                11392
.S44E8                                11393
                                                                    11394
if ENABLE_ANIMATE = TRUE                11395
                                ldy #$00                      ;Keine Animation. 11396
                                sty animFlag                    11397
endif                                    11398
                                                                    11399
                                txa                            ;Fenster-Nr. 11400
                                pha                            ;zwischenspeichern. 11401
                                lda C04FC,x                    ;Part. für Fenster. 11402

```

```

11403          bne      :1          ; => Vorhanden...
11404
11405          ldy      curDrive
11406          lda      V04B8 -8,y    ;Geräteinfo einlesen.
11407          beq      :1          ; => 1541/71/81
11408
11409          jsr      Z3A83        ;CMD-Part. suchen.
11410          txa
11411          beq      :1          ; => Gefunden...
11412
11413          pla
11414          jmp      S5CF3        ;Fehler ausgeben.
11415
11416 ::1          pla          ;Fenster-Nr.
11417          pha
11418          tax
11419          jsr      Z4675        ;Diskname einlesen.
11420          txa
11421          beq      :2          ; => Kein Fehler...
11422
11423          pla
11424          jmp      S5CF3        ;Fehler ausgeben.
11425
11426 ::2          ldx      curType    ;1541?
11427          dex
11428          bne      :3          ; => Nein, weiter...
11429
11430          lda      curDirHead +3 ;Doppelseitig?
11431          bpl      :3          ; => Nein, weiter...
11432
11433          pla
11434          ldx      #ERR_41DBLSIDE ;Dbl.side in 1541.
11435          jmp      S5CF3        ;Fehler ausgeben.
11436
11437 ::3          pla          ;Fenster-Nr.
11438          pha
11439          jsr      Z481C        ;Diskinfo/Größe.
11440
11441          pla          ;Fenster-Nr.
11442          pha
11443          tax
11444          lda      curDrive
11445          sta      C04F8,x      ;Lfwk. für Fenster.
11446          lda      #$00
11447          sta      C050B,x      ;Fensterposition.
11448          jsr      S598F        ;Zeiger Ordnerliste.
11449          jsr      S28A7        ;Ordnerliste löschen.
11450          pla
11451          tax          ;Fenster-Nr.
11452
11453          lda      #$00        ;Haupt-Ordner öffnen.
11454
11455 ; Nächsten Ordner öffnen.
11456 ; Übergabe: X = Fenster-Nr.
11457 ;          A = Ordner-Nr.

```



```

11513 if ENABLE_DIFFICOL = TRUE
11514         lda    MT3F59                ;Versch. Iconfarben?
11515         cmp    #"*"
11516         beq    :3                    ; => Ja, weiter...
11517 endif
11518
11519         lda    V04AF                ;Neue PadColorPref ?
11520         beq    :3                    ; => Nein, weiter...
11521
11522 ; Wenn einheitliche Farben und es eine
11523 ; neue PadColorPref gibt -> Alle neu.
11524         jmp    S157E                ;Alle Fens. anzeigen.
11525
11526 ; Nur ein Fenster neu.
11527 ::3         inc    V18AD                ;Fens.-Aufbau-Status.
11528
11529         ldx    C042A                ;Oberstes Fenster.
11530         jmp    S141D                ;Fenster anzeigen.
11531
11532 if ENABLE_ANIMATE = TRUE
11533 :animFlag    b $00    ;$FF = Anzeigen.
11534 endif
11535
11536 ; Speicher für Verzeichnis löschen.
11537 :Z4570        lda    #< SIZE_DIRBUF    ;Speicher für
11538                sta    r0L                ;Dateinamen und
11539                lda    #> SIZE_DIRBUF    ;Icons löschen.
11540                sta    r0H
11541                jmp    ClearRam
11542
11543 ; Dateien neu einlesen.
11544 ; Übergabe: X = Fenster-Nr.
11545 ; TD13: Reload2
11546 ;
11547 ; Hinweis:
11548 ; Im Textmodus wird der Bereich für
11549 ; VLIR-Module überschrieben!
11550 :Z4576        jsr    S599A                ;Zeiger auf DIRBUF.
11551                pha                    ;Fenster-Nr.
11552
11553 ;         lda    #< SIZE_DIRBUF    ;Speicher für
11554 ;         sta    r0L                ;Dateinamen und
11555 ;         lda    #> SIZE_DIRBUF    ;Icons löschen.
11556 ;         sta    r0H
11557 ;         jsr    ClearRam
11558 ;         jsr    Z4570                ;Speicher löschen.
11559
11560                pla
11561                tax                    ;Fenster-Nr.
11562
11563 ; Dateien einlesen.
11564 ; Übergabe: X = Fenster-Nr.
11565 ; TD13: Reload
11566 ;
11567 ; Hinweis:

```

```

; Im Textmodus wird der Bereich für
; VLIR-Module überschrieben!
:Z4587      txa
            pha
            jsr   Z5CB3          ;Diskette öffnen.
            bcc   :1
            pla
            rts
::1         pla
if FALSE
; Hinweis:
; Prefs/PadColorPref erst suchen, wenn
; die Dateien eingelesen wurden.
; Ausserdem nur im Icon-Modus suchen.
;
            pha
            tax          ;Neue Fenster-Nr.
            jsr   Z31FD      ;Prefs./PadCol laden.
            pla
endif
            tax
            jsr   S599A      ;Zeiger auf DIRBUF.
            pha
            lda   r1H
            pha
            lda   r1L
            pha
            lda   #< SIZE_DIRNAM ;Speicher für
            sta   r0L          ;Dateinamen löschen.
            lda   #> SIZE_DIRNAM
            sta   r0H
            jsr   ClearRam
            pla
            sta   r3L
            pla
            sta   r3H
            pla
            pha
            tax
            lda   C051B,x      ;Aktive Ordner-Nr.
            sta   r10L          ;Dat./Ordner suchen.
            lda   C050B,x      ;Fensterposition.
            sta   r11H          ;Dateien überlesen.
            lda   #MAXWINFILES ;Max. 16 Dateien.
            sta   r11L
            lda   #%11000000    ;Keine gelöschten

```

```

11623          sta  r12L          ;oder leere Einträge.
11624          lda  #%10000000    ;Alle Dateien.
11625          sta  r12H
11626
11627          ldx  #$04
11628          lda  C04B2          ;Icon-/Text-Modus?
11629          beq  :3            ; => Icon, weiter...
11630
11631 ; Hinweis:
11632 ; Im Textmodus wird der Bereich für
11633 ; VLIR-Module überschrieben!
11634          lda  #$00          ;Kein VLIR-Modul mehr
11635          sta  V0422          ;im Speicher!
11636
11637          lda  #$00
11638          sec
11639          sbc  #< BASE_MOD_A
11640          sta  r0L
11641          lda  MHB04AC        ;HB Ende CopyBuf.
11642          sbc  #> BASE_MOD_A
11643          sta  r0H
11644
11645          lda  #< BASE_MOD_A
11646          sta  r1L
11647          lda  #> BASE_MOD_A
11648          sta  r1H
11649
11650          jsr  ClearRam
11651
11652          lda  #< BASE_MOD_A +2
11653          sta  r3L
11654          lda  #> BASE_MOD_A +2
11655          sta  r3H
11656
11657          ldx  #$01
11658          ldy  C04B2          ;Icon-/Text-Modus?
11659          dey
11660          beq  :2            ; => Text-Modus.
11661
11662          lda  #MAXFILES      ;Max. 255 Dateien.
11663          sta  r11L
11664          lda  #0             ;0=Keine überlesen.
11665          sta  r11H
11666
11667 ::2          lda  #30        ;Dateieintrag.
11668          b $2c
11669 ::3          lda  #16 +2     ;Name +Tr/Se Infob.
11670          sta  r13L          ;Anzahl Bytes lesen.
11671
11672          stx  r13H          ;Bytes überlesen.
11673
11674          jsr  S2603          ;Dateien suchen.
11675          txa                ;Fehler?
11676          beq  :11          ; => Nein, weiter...
11677

```



```

; Hinweis: 11678
; Bei sortierter Text-Anzeige ist im 11679
; Speicher Platz für ca.150 Dateien. 11680
; Sind mehr Dateien auf Disk, dann 11681
; können diese nicht sortiert werden 11682
; und das Fenster wird geschlossen. 11683
        cpx    #BFR_OVERFLOW 11684
        bne    :4 11685
        jsr    S5CE1 ;Fehler ohne Refresh. 11686
        ldx    #NO_ERROR 11687
::4      pla 11688
        tay 11689
        lda    #$00 11690
        sta    C0448,y ;Fenster geschlossen. 11691
        txa 11692
        sec 11693
        rts 11694
; Fensterdaten aktualisieren. 11695
::11     lda    r15L ;BFR_OVERFLOW? 11696
        beq    :12 ; => Nein, weiter... 11697
        pla ;Fehler nur am 11698
        pha ;Fensteranfang 11699
        tax ;anzeigen. 11700
        lda    C050B,x ;Offset = 0? 11701
        bne    :12 ; => Nein, weiter... 11702
; Hinweis: 11703
; Der Aufruf einer Dialogbox kann die 11704
; Werte von r0 bis r11 verändern. Im 11705
; folgenden wird aber r11L noch genutzt 11706
; um die Anzahl Dateien zu ermitteln. 11707
; Register zwischenspeichern. 11708
; 11709
; Korrektur: r11L/r14L sichern. 11710
        lda    r11L ;Max. Anzahl an 11711
        pha ;Dateien einlesen. 11712
        lda    r14L ;Anzahl Dateien ab 11713
        pha ;Offset auf Disk. 11714
        ldx    #BFR_OVERFLOW 11715
        jsr    S5CE1 ;Fehler ohne Refresh. 11716
        pla 11717
        sta    r14L 11718
        pla 11719
        sta    r11L 11720
; 11721
; lda    #MAXFILES ;Nicht notwendig, 11722
; sta    r14L ;da immer < MAXFILES. 11723
::12     pla 11724

```

```

11733          tax
11734          lda    r14L
11735          sta    V0517,x          ;Anz.Dat. ab Offset.
11736          clc
11737          adc    C050B,x          ;Fensterposition.
11738          sta    V0513,x          ;Anz.Dat. auf Disk.
11739
11740          ldy    C04B2            ;Icon-/Text-Modus?
11741          beq    :13              ; => Icon, weiter...
11742          dey
11743          bne    :14              ; => Text, sortiert.
11744 ::13          sta    V0517,x          ;Anz.Dat. ab Offset.
11745
11746 ::14          lda    #MAXWINFILES
11747          sec
11748          sbc    r11L
11749          sta    V050F,x          ;Anz.Dat. im Fenster.
11750
11751          txa
11752          pha
11753          jsr    Z481C            ;Diskinfo/Größe.
11754
11755          pla
11756          pha
11757          tax
11758          lda    C04B2            ;Icon-/Text-Modus?
11759          beq    :15              ; => Icon, weiter...
11760
11761          lda    C050B,x          ;Fensterposition.
11762          jsr    A2BBB            ;Dateien sortieren.
11763
11764 ::15          pla
11765          tax
11766          lda    C04B2            ;Icon-/Text-Modus.
11767          sta    V0537,x          ;Fens.Mod. Icon/Text.
11768
11769 ; Hinweis:
11770 ; Prefs./PadColorPref nur im Icon-
11771 ; Modus suchen. Allerdings werden die
11772 ; Voreinstellungen auch beim blättern
11773 ; im Verzeichnis gesucht!
11774 if FALSE
11775          bne    :done
11776
11777 ;          ldx    #$ff            ;Fenster-Nr.
11778          jsr    Z31FD            ;Prefs./PadCol laden.
11779 endif
11780
11781 ::done        ldx    #NO_ERROR
11782          clc
11783          rts
11784
11785 ; Diskname einlesen.
11786 ; Übergabe: X = Fenster-Nr.
11787 ; TD13: GetDiskName

```

:Z4675	txa		11788
	pha		11789
	jsr	OpenDisk	11790
	txa		11791
	beq	:1	11792
	pla		11793
	rts		11794
			11795
::1	pla		11796
	jsr	S5979 ;Zeiger Fenstertitel.	11797
			11798
	lda	curDrive	11799
	clc		11800
	adc	#"A" -8	11801
	ldy	#\$00	11802
	sta	(r1),y	11803
			11804
	lda	#\$02	11805
	clc		11806
	adc	r1L	11807
	sta	r1L	11808
	bcc	:2	11809
	inc	r1H	11810
			11811
::2	ldy	#0	11812
::3	lda	(r5),y	11813
	cmp	#\$a0 ;Ende Diskname?	11814
	bne	:4 ; => Nein, weiter...	11815
	lda	#TDSPLIT ;Trenner für Ordner.	11816
::4	sta	(r1),y	11817
	iny		11818
	cpy	#16	11819
	bne	:3	11820
			11821
	lda	#NULL	11822
	sta	(r1),y	11823
			11824
	ldx	#NO_ERROR	11825
	rts		11826
			11827
	; Ordner-Name in Fenstertitel.		11828
	; TD13: GetSubName		11829
:Z46B0	lda	C042A ;Oberstes Fenster.	11830
			11831
	; ZD13: GetSubName2		11832
:Z46B3	jsr	S5979 ;Zeiger Fenstertitel.	11833
	lda	#"/" ! \$80	11834
	sta	(r1),y	11835
			11836
	iny		11837
	tya		11838
	clc		11839
	adc	r1L	11840
	sta	r1L	11841
	bcc	Z46C5	11842

```

11843             inc    r1H
11844
11845 ; Name kopieren.
11846 ; Übergabe: X = Zeiger auf Register.
11847 :Z46C5        ldx    #r0
11848             b $2c
11849 :Z46CA        ldx    #r1
11850
11851             lda    #< V0470
11852             sta    zpage +0,x
11853             lda    #> V0470
11854             sta    zpage +1,x
11855
11856             lda    #16 +1                ;Dateiname in
11857             sta    r2L                  ;Zwischenspeicher.
11858 ;             jmp    Z281B                ;String anpassen.
11859
11860 ; Datei-/Diskname kürzen.
11861 ; Entfernt $A0, setzt NULL-Byte.
11862 ; Übergabe: r0 = Zeiger auf Eintrag in DIRBUF.
11863 ;             r1 = Zeiger auf Zwischenspeicher.
11864 ;             r2L = Max.Länge (inkl. NULL-Byte)
11865 ; Rückgabe: r1 = Dateiname, Ende NULL-Byte.
11866 ; TD13: FormString
11867 :Z281B        ldy    #0
11868 ::1          cpy    r2L
11869             beq    :2
11870             lda    (r0),y
11871             cmp    #$a0
11872             beq    :3
11873             sta    (r1),y
11874             iny
11875             bne    :1
11876
11877 ::2          dey
11878
11879 ::3          lda    #NULL
11880             sta    (r1),y
11881             rts
11882
11883 ; Letzten Ordner aus Titelzeile entfernen.
11884 ; TD13: RemSubName
11885 .S46D4        lda    C042A                ;Oberstes Fenster.
11886             jsr    S5979                ;Zeiger Fenstertitel.
11887
11888 ::1          lda    (r1),y
11889             cmp    #"/" ! $80
11890             beq    :2
11891             dey
11892             bne    :1
11893             sec
11894             rts
11895
11896 ::2          lda    #NULL
11897             sta    (r1),y

```

```

                                clc                                11898
                                rts                                11899
                                                                11900
; Dateiname für markierte Datei holen.                            11901
; Übergabe: A = Datei-Nr. (0-254)                                11902
; Rückgabe: r0 = Adresse (V0470) Dateiname.                      11903
;           C = 0: Datei gefunden.                               11904
;           C = 1: Nicht gefunden.                               11905
; TD13: GetFileName                                              11906
.S46EB    pha                                                        11907
          jsr    S5C89                ;Hole Disk/Fenster.        11908
          tax                                11909
          beq    :2                                11910
          pla                                11911
::1       sec                                ;Nicht gefunden.    11912
          rts                                11913
                                                                11914
::2       stx    V0470                                11915
          inx                                11916
;         ldx    #1                ;Nur eine Datei.            11917
          stx    r11L                                11918
                                                                11919
          pla                                ;Anzahl Einträge    11920
          sta    r11H                                ;überlesen.  11921
                                                                11922
          ldx    C042A                ;Oberstes Fenster.        11923
                                                                11924
          lda    #< V0470                ;Zeiger auf Speicher    11925
          sta    r3L                ;für Dateiname.            11926
          lda    #> V0470                11927
          sta    r3H                                11928
                                                                11929
          ldy    #4 -1                ;r12/r13 setzen.          11930
::3       lda    V4736,y                                11931
          sta    r12,y                                11932
          dey                                11933
          bpl    :3                                11934
                                                                11935
          lda    C051B,x                ;Aktive Ordner-Nr.        11936
          sta    r10L                                11937
                                                                11938
          jsr    S2603                ;Dateien suchen.          11939
                                                                11940
          lda    V0470                ;Name gefunden?          11941
          beq    :1                ; => Nein, Ende...            11942
                                                                11943
          ldy    #16 -1                                11944
::4       lda    V0470,y                                11945
          cmp    #$a0                                11946
          bne    :5                                11947
          dey                                11948
          bpl    :4                                11949
                                                                11950
::5       lda    #NULL                                11951
          sta    V0470 +1,y                                11952

```

```

11953
11954             clc
11955             rts
11956
11957 ; Daten für FindDirFiles.
11958 :V4736          b %11000000 ;r12L: keine leeren/gelöschte
11959                b %10000000 ;r12H: Alle Dateitypen
11960                b 16        ;r13L: 16 Zeichen einlesen
11961                b 4          ;r13H: 4 Bytes überlesen
11962
11963 ; Fensterstaus anzeigen.
11964 ; TD13: PrintDiskInfo
11965 :Z473A          lda    r0H
11966                pha
11967                lda    r0L
11968                pha
11969
11970                ldx    WIN042F          ;Aktives Fenster.
11971                jsr    S1E09          ;Arbeitsber.Fenster.
11972                bcs    :1
11973
11974                jsr    Z1400          ;Restore TextWin.
11975
11976 ::1            lda    r4L          ;Grenze Rechter Rand.
11977                bne    :1a
11978                dec    r4H
11979 ::1a           dec    r4L
11980
11981                ldx    r2H
11982                inx
11983                inx
11984                stx    r2L          ;Grenze oberer Rand.
11985
11986                txa
11987                clc
11988                adc    #6          ;Grenze unterer Rand.
11989                sta    r2H
11990
11991                lda    r3L
11992                clc
11993                adc    #< SUL +1    ;Grenze linker Rand.
11994                sta    r11L
11995                lda    r3H
11996                adc    #> SUL +1
11997                sta    r11H
11998
11999                ldx    r2H
12000                dex
12001                stx    r1H
12002
12003                jsr    S13E5          ;Backup/Set TextWin.
12004                bcs    :3
12005
12006                jsr    Z13E6          ;Part.-Nr. ausgeben.
12007                jsr    Z13E7          ;GEOS-Disk-Status.

```

			12008
	lda	WIN042F ;Aktives Fenster.	12009
	asl		12010
	pha		12011
	tax		12012
	lda	V0527 +0,x ;Anzahl freie Blocks.	12013
	sta	r0L	12014
	lda	V0527 +1,x	12015
	sta	r0H	12016
	jsr	Z47F4 ;Ausgabe Blks./Kb.	12017
			12018
	lda	#< T3E09 ;"KBytes"	12019
	ldx	#> T3E09	12020
			12021
	ldy	MT3E11	12022
	cpy	#" " ;Blocks/KByte?	12023
	beq	:2 ; => KByte...	12024
			12025
	lda	#< T3E15 ;"Blocks".	12026
	ldx	#> T3E15	12027
			12028
::2	sta	r0L	12029
	stx	r0H	12030
	jsr	S14C8 ;String ausgeben.	12031
			12032
	lda	#< T52D5 ;" frei, ".	12033
	sta	r0L	12034
	lda	#> T52D5	12035
	sta	r0H	12036
	jsr	S14C8 ;String ausgeben.	12037
			12038
	pla		12039
	tax		12040
	lda	V052F +0,x ;Anzahl Blocks/Disk.	12041
	sec		12042
	sbc	V0527 +0,x ;Anzahl freie Blocks.	12043
	sta	r0L	12044
	lda	V052F +1,x	12045
	sbc	V0527 +1,x	12046
	sta	r0H	12047
	jsr	Z47F4 ;Ausgabe Blks./Kb.	12048
			12049
	lda	#< T480B ;"belegt".	12050
	sta	r0L	12051
	lda	#> T480B	12052
	sta	r0H	12053
	jsr	S14C8 ;String ausgeben.	12054
			12055
::4	jsr	Z1400 ;Restore TextWin.	12056
			12057
::3	pla		12058
	sta	r0L	12059
	pla		12060
	sta	r0H	12061
	rts		12062

```

12063
12064 if LANG = LANG_DE
12065 :T52D5          b " frei, ",0
12066 :T480B          b " belegt",0
12067 endif
12068 if LANG = LANG_EN
12069 :T52D5          b " free, ",0
12070 :T480B          b " used",0
12071 endif
12072
12073 ; Zahl in Blocks oder KByte ausgeben.
12074 :Z47F4          lda    MT3E11          ;Anzeige in Blocks?
12075                cmp    #""
12076                beq    :1              ; => Nein, weiter...
12077
12078                lsr    r0H              ;In KByte umwandeln.
12079                ror    r0L
12080                lsr    r0H
12081                ror    r0L
12082
12083 ::1            lda    #SET_LEFTJUST!SET_SUPRESS
12084                jmp    S590A            ;Zahl ausgeben.
12085
12086 ; Partitions-Nr. ausgeben.
12087 :Z13E6          ldx    WIN042F          ;Fenster-Nr.
12088                ldy    C04F8,x          ;Lfwk. für Fenster.
12089                lda    V04B8 -8,y       ;CMD-Laufwerk?
12090                beq    :1              ; => 1541/71/81.
12091                bmi    :1              ; => 64Net.
12092
12093 ;                ldx    WIN042F          ;Fenster-Nr.
12094                lda    C04FC,x          ;Partitions-Nr.
12095                pha
12096
12097                lda    #"P"            ;"P" ausgeben.
12098                jsr    SmallPutChar
12099
12100                pla
12101                sta    r0L              ;Partitions-Nr.
12102                lda    #$00
12103                sta    r0H
12104
12105                lda    #SET_LEFTJUST!SET_SUPRESS
12106                jsr    S590A            ;Zahl ausgeben.
12107
12108                lda    #":"
12109                jsr    SmallPutChar
12110                lda    #" "
12111                jsr    SmallPutChar
12112
12113 ::1            rts
12114
12115 ; "GEOS-Disk"-Status anzeigen.
12116 :Z13E7          ldx    WIN042F          ;Aktives Fenster.
12117                lda    V4820,x          ;GEOS-Diskette?

```



```

                                bpl    :exit                ; => Nein, Ende...      12118
                                                                12119
                                lda     #< T480C            ;"GEOS".              12120
                                sta     r0L                 12121
                                lda     #> T480C            12122
                                sta     r0H                 12123
                                jsr     S14C8                ;String ausgeben.      12124
                                                                12125
::exit                          rts                        12126
                                                                12127
:T480C                          b "G! ",0                 12128
                                                                12129
; Info für aktuelles Fenster setzen.      12130
; TD13: GetAkltDiskInfo                  12131
:Z4819                          lda     C042A              ;Oberstes Fenster.    12132
                                                                12133
; Info für Fenster einlesen.             12134
; Übergabe: A = Fenster-Nr.              12135
; TD13: GetDiskInfo                      12136
:Z481C                          pha                        12137
                                lda     #< curDirHead      12138
                                sta     r5L                 12139
                                lda     #> curDirHead      12140
                                sta     r5H                 12141
                                jsr     CalcBlksFree        12142
                                jsr     ChkDkGEOS           ;GEOS-Diskette?      12143
                                pla                        12144
                                tay                        12145
                                asl                        12146
                                tax                        12147
                                lda     r4L                 ;Anzahl freie Blocks.    12148
                                sta     V0527 +0,x          12149
                                lda     r4H                 12150
                                sta     V0527 +1,x          12151
                                lda     r3L                 ;Anzahl Blocks/Disk.    12152
                                sta     V052F +0,x          12153
                                lda     r3H                 12154
                                sta     V052F +1,x          12155
                                lda     isGEOS              ;GEOS-Diskette.      12156
                                sta     V4820 ,y            12157
                                rts                        12158
                                                                12159
; GEOS-Disk-Flag.                       12160
:V4820                          s 4                        12161
                                                                12162
; Geräte-Informationen anzeigen.          12163
; TD13: PrintDriveNames                  12164
.S4840                          jsr     Z13D0              ;Textgrenzen Screen.  12165
                                                                12166
                                ldx     numDrives           12167
::1                              dex                        12168
                                stx     a7L                 12169
                                                                12170
; Laufwerksbuchstabe.                   12171
                                lda     V48B2,x             ;Y-Koordinate.      12172

```

```

12173          sta    r1H
12174
12175          lda    #< PANEL_X0 +6          ;X-Koordinate.
12176          sta    r11L
12177          lda    #> PANEL_X0 +6
12178          sta    r11H
12179
12180          txa
12181          clc
12182          adc    #"A"
12183          jsr    PutChar
12184          lda    #":"
12185          jsr    PutChar
12186
12187 ; Laufwerkmodus.
12188 if ENABLE_DRVMODE = TRUE
12189          lda    r1H          ;Y-Koordinate.
12190          clc
12191          adc    #7
12192          sta    r1H
12193
12194          lda    #< PANEL_X0 +6          ;X-Koordinate.
12195          sta    r11L
12196          lda    #> PANEL_X0 +6
12197          sta    r11H
12198
12199          ldx    a7L
12200          lda    driveType,x
12201          and    #ST_DM0DES
12202          asl
12203          tax
12204          lda    V48B3 +1,x
12205          pha
12206          lda    V48B3 +0,x
12207          jsr    PutChar
12208          pla
12209          jsr    PutChar
12210 endif
12211
12212          ldx    a7L
12213          bne    :1
12214
12215 ; Anzeigebereich löschen.
12216          lda    #PANEL_PATTERN
12217          jsr    SetPattern
12218
12219          jsr    i_Rectangle
12220          b PANEL_Y1 -15, PANEL_Y1 -1
12221          w PANEL_X0 +1 , PANEL_X1 -1
12222
12223 ; Eingabetreiber ausgeben.
12224          jsr    i_PutString
12225          w PANEL_X0 +2
12226          b PANEL_Y1 -8
12227          b NULL

```



```

12283 ; Behandlung der Messages Close/Restore.
12284 ; TD13: NormHandler
12285 :Z48CC          lda    WIN042E          ;Fenster-Befehl.
12286                cmp    #WN_CLOSE        ;Fens. schließen?
12287                bne    :3                ; => Nein, weiter...
12288
12289                jsr    S5842              ;mark.Dat. anzeigen.
12290                jsr    S524C              ;Dat.-Ausw. aufheben.
12291                jsr    S09AC              ;Fens.dat. löschen.
12292                jsr    Z18AE              ;Fenster schließen.
12293
12294 ; Das nächste offene Fenster suchen.
12295                ldy    #0
12296 ::1              lda    C042A,y          ;Fenster-Nr. holen.
12297                tax
12298                lda    C0448,x            ;Fenster geöffnet?
12299                bne    :2                ; => Ja, weiter...
12300                iny
12301                cpy    #4
12302                bne    :1
12303
12304                ldx    #$ff              ;Kein Fens. geöffnet.
12305 ::2              txa
12306                bmi    :4                ; => Kein Fenster.
12307
12308 ; Laufwerk für neues Fenster öffnen.
12309                clc
12310                jsr    Z1C10              ;Fenster nach vorne.
12311
12312                ldx    C042A              ;Oberstes Fenster.
12313                lda    C04F8,x            ;Lfwk. für Fenster.
12314                jmp    S06AB              ;Laufwerk öffnen.
12315
12316 ::3              cmp    #WN_RESTORE
12317                bne    :4
12318
12319                jsr    S4840              ;Lfwk/Prnt/Inpt-Info.
12320 ::4              rts
12321
12322 ;*** Shortcut: CLR/HOME.
12323 :KB48F7          ldx    WIN042F          ;Aktives Fenster.
12324                lda    #$00
12325                sta    C050B,x            ;Fensterposition.
12326                beq    KB490F
12327
12328 ;*** Shortcut: Home.
12329 :KB4901          ldx    WIN042F          ;Aktives Fenster.
12330                lda    #$00              ;Anfang Dateiliste.
12331                jmp    Z4DB7
12332
12333 ;*** Shortcut: Cursor left.
12334 :KB4907          lda    #$ff              ;Maustest abschalten.
12335                sta    V4A22
12336
12337 ;*** Shortcut: Cursor up.

```

```

:KB490F      lda    #WN_SCROLL_U      ;Cursor Up/Left.      12338
              bne    execCsrKeys      12339
              12340
;*** ShortCut: Cursor right.      12341
:KB4495      lda    #$ff              ;Maustest abschalten.  12342
              sta    V4A22             12343
              12344
;*** Shortcut: Cursor down.      12345
:KB4912      lda    #WN_SCROLL_D      ;Cursor Down/Right.  12346
              12347
;*** Shortcut: Cursor-Tasten.    12348
:execCsrKeys sta    WIN042E           ;Fenster-Befehl.    12349
              lda    C042A            ;Oberstes Fenster.  12350
              sta    WIN042F          ;Aktives Fenster.    12351
              tax                     12352
              lda    C0448,x          ;Fenster geöffnet?    12353
              bne    Z4923            ; => Ja, weiter...    12354
              rts                     12355
              12356
; Window-Handler.                12357
; TD13: Handler                  12358
:Z4923       ldx    WIN042F           ;Aktives Fenster.    12359
              jsr    S599A            ;Zeiger auf DIRBUF.  12360
              jsr    Z31EE            ;Kopiere r1 -> r0.    12361
              12362
              lda    WIN042E           ;Fenster-Befehl.    12363
              cmp    #WN_ACTIVATE      12364
              beq    :1               12365
              cmp    #WN_ACTIVATE2     12366
              bne    :6               12367
              12368
; Fensterfunktion: WN_ACTIVATE2   12369
              lda    V0423             ;DnD aktiv?      12370
              beq    :1               ; => Nein, weiter...  12371
              bpl    :5               ; => MultiFile.    12372
              12373
; Fensterfunktion: WN_ACTIVATE    12374
::1          lda    WIN042F           ;Aktives Fenster.  12375
              pha                     12376
              lda    C042A            ;Oberstes Fenster.  12377
              sta    WIN042F          ;Aktives Fenster.    12378
              jsr    S5842            ;mark.Dat. anzeigen. 12379
              pla                     12380
              sta    WIN042F          ;Aktives Fenster.    12381
              12382
              jsr    S524C            ;Dat.-Ausw. aufheben. 12383
              jsr    Z42DB            ;DnD-Modus beenden.  12384
              12385
              ldx    WIN042F           ;Aktives Fenster.  12386
              lda    V0537,x          ;Fens.Mod. Icon/Text. 12387
              cmp    C04B2            ;Icon-/Text-Modus?    12388
              beq    :3               ; => Unverändert.    12389
              12390
              jsr    Z4576            ;Dat. neu einlesen.  12391
              bcc    :2               ; => Kein Fehler.    12392

```

```

12393
12394             jmp     S5CF3             ;Fehler ausgeben.
12395
12396 ; Inhalt geändert.
12397 ::2             sec
12398             bcs     :4
12399
12400 ; Inhalt unverändert.
12401 ::3             clc
12402
12403 ; Fenster nach vorne holen.
12404 ::4             jsr     Z1C10             ;Fenster nach vorne.
12405
12406             ldx     C042A             ;Oberstes Fenster.
12407             lda     C04F8,x             ;Lfwk. für Fenster.
12408             jmp     S06AB             ;Laufwerk öffnen.
12409
12410 ::5             lda     #$00             ;Vorgabe: Kein
12411             sta     V04AE             ;Klick auf Gadget.
12412             jmp     Z5357
12413
12414 ;---
12415 ::6             cmp     #WN_USER
12416             bne     :7
12417
12418 ; Fensterfunktion: WN_USER
12419             jmp     Z49E2
12420
12421 ;---
12422 ::7             jsr     Z42DB             ;DnD-Modus beenden.
12423
12424             cmp     #WN_REDRAW
12425             beq     :11
12426             cmp     #WN_SCROLL_D
12427             bne     :8
12428
12429 ; Fensterfunktion: WN_SCROLL_D
12430             jmp     Z4A11
12431
12432 ;---
12433 ::8             cmp     #WN_SCROLL_U
12434             bne     :9
12435
12436 ; Fensterfunktion: WN_SCROLL_U
12437             jmp     Z4AF4
12438
12439 ;---
12440 ::9             cmp     #WN_CLOSE
12441             bne     :10
12442
12443 ; Fensterfunktion: WN_CLOSE
12444             jmp     Z4EEF
12445
12446 ::10            jmp     Z48CC             ;Message-Handler.
12447

```

; Fensterfunktion: WN_REDRAW.			12448
::11	jsr	Z473A ;DiskInfo anzeigen.	12449
			12450
	ldy	#6 -1	12451
::12	lda	windowTop,y	12452
	sta	V49DC,y	12453
	dey		12454
	bpl	:12	12455
			12456
	ldx	WIN042F ;Aktives Fenster.	12457
	jsr	S1E09 ;Arbeitsber.Fenster.	12458
	bcs	:13	12459
			12460
	jsr	A2BA6 ;Anzeige Icons/Text	12461
	txa	;initialisieren und	12462
	beq	:13 ;Inhalt anzeigen.	12463
			12464
; Benötigt Fenster-Nr. in WIN042F!			12465
	jsr	Z5CEB ;Fehler ausgeben und	12466
		;Fenster schließen.	12467
	jmp	Z13D3 ;Textgrenzen Desktop.	12468
			12469
::13	ldy	#6 -1 ;Textausgabe auf	12470
::14	lda	V49DC,y ;Fenster begrenzen.	12471
	sta	windowTop,y	12472
	dey		12473
	bpl	:14	12474
			12475
	jsr	Z2478 ;Füllstandsanzeige.	12476
	jmp	Z13D3 ;Textgrenzen Desktop.	12477
			12478
:V49DC	s	6 ;Zwischenspeicher Textgrenzen.	12479
			12480
; Fensterfunktion: WN_USER.			12481
:Z49E2	jsr	A2BAC ;Mausklick auswerten.	12482
	txa		12483
	bmi	:1 ; => Kein Icon...	12484
			12485
	pha		12486
	jsr	Z4214 ;X/Y für erstes Icon.	12487
	pla		12488
			12489
	pha		12490
	jsr	A2BB2 ;Rechteck f.Eintrag.	12491
	pla		12492
			12493
	tax		12494
	jmp	Z4F1D ;Auswahl auswerten.	12495
			12496
::1	jsr	S51CE ;C=-Taste gedrückt?	12497
	bcs	:2 ; => Ja, weiter...	12498
			12499
	jsr	Z42DB ;DnD-Modus beenden.	12500
	jsr	S5842 ;mark.Dat. anzeigen.	12501
	jsr	S524C ;Dat.-Ausw. aufheben.	12502

```

12503
12504 ::2                rts
12505
12506 ; Fensterfunktion: WN_SCROLL_D
12507 :Z4A11             jsr   A2BB5                ;Anz.Ze/Sp und Icons.
12508                 jsr   S5C89                ;Hole Disk/Fenster.
12509                 bcs   :exit
12510
12511                 ldx   WIN042F                ;Aktives Fenster.
12512                 lda   V0513,x                ;Anz.Dat. auf Disk.
12513                 bne   :1                    ; => Weiter...
12514
12515 ::exit            rts
12516
12517 ::1                lda   r0H
12518                 pha
12519                 lda   r0L
12520                 pha
12521
12522                 bit   V4A22                ;Maus/Tastatur?
12523                 bmi   Z4A24                ; => Tastatur...
12524
12525                 jsr   S48B6                ;Langer Mausklick?
12526                 bcs   Z4A45                ; => Nein, weiter...
12527
12528 ; WN_SCROLL_D: Langer Mausklick oder Shortcut.
12529 :Z4A24             jsr   Z31E5                ;Kopiere r0 -> r1.
12530
12531 ;                 lda   #< SIZE_DIRBUF        ;Speicher für
12532 ;                 sta   r0L                  ;Dateinamen und
12533 ;                 lda   #> SIZE_DIRBUF        ;Icons löschen.
12534 ;                 sta   r0H
12535 ;                 jsr   ClearRam
12536 ;                 jsr   Z4570                ;Speicher löschen.
12537
12538                 lda   #$00                ;Mausabfrage.
12539                 sta   V4A22
12540
12541                 ldx   WIN042F                ;Aktives Fenster.
12542                 lda   V4CCB,x                ;Anzahl Icons/Seite.
12543                 jmp   Z4AC7                ;Eine Seite runter.
12544
12545 ; WN_SCROLL_D: Kurzer Mausklick.
12546 :Z4A45             pla
12547                 sta   r0L
12548                 pla
12549                 sta   r0H
12550
12551 ;                 lda   r0H
12552 ;                 pha
12553 ;                 lda   r0L
12554 ;                 pha
12555
12556                 jsr   Z31E0                ;r0 + SIZE_DIRNAM ->
12557                 ;Kopiere r0 -> r1.

```



			12558
	ldx	WIN042F ;Aktives Fenster.	12559
	ldy	V4CC7,x ;Anzahl Icons/Zeile.	12560
			12561
::1	lda	#< 64	12562
	clc		12563
	adc	r0L	12564
	sta	r0L	12565
	lda	#> 64	12566
	adc	r0H	12567
	sta	r0H	12568
	dey		12569
	bne	:1	12570
			12571
	lda	r1L	12572
	sta	r2L	12573
	lda	r1H	12574
	sta	r2H	12575
			12576
	lda	#< SIZE_DIRICON	12577
	clc		12578
	adc	r2L	12579
	sta	r2L	12580
	lda	#> SIZE_DIRICON	12581
	adc	r2H	12582
	sta	r2H	12583
			12584
;	lda	r2H	12585
	pha		12586
	lda	r2L	12587
	pha		12588
			12589
	lda	r2L	12590
	sec		12591
	sbc	r0L	12592
	sta	r2L	12593
	lda	r2H	12594
	sbc	r0H	12595
	sta	r2H	12596
	jsr	MoveData	12597
			12598
	lda	r2L	12599
	clc		12600
	adc	r1L	12601
	sta	r1L	12602
	lda	r2H	12603
	adc	r1H	12604
	sta	r1H	12605
			12606
	pla		12607
	sta	r0L	12608
	pla		12609
	sta	r0H	12610
			12611
	lda	r0L	12612

```

12613          sec
12614          sbc    r1L
12615          sta    r0L
12616          lda    r0H
12617          sbc    r1H
12618          sta    r0H
12619          jsr    ClearRam
12620
12621          ldx    WIN042F          ;Aktives Fenster.
12622          lda    V4CC7,x          ;Anzahl Icons/Zeile.
12623
12624 ; Seite oder Zeile runter.
12625 ; Übergabe: A = Anzahl Icons/Zeile.
12626 :Z4AC7          ldx    WIN042F          ;Aktives Fenster.
12627          clc
12628          adc    C050B,x          ;Fensterposition.
12629          sta    C050B,x          ;Fensterposition.
12630          sta    r11H
12631          jsr    S1E09          ;Arbeitsber.Fenster.
12632
12633          ldx    C042A          ;Oberstes Fenster.
12634          jsr    Z4587          ;Dateien einlesen.
12635
12636          ldx    WIN042F          ;Aktives Fenster.
12637          lda    V050F,x          ;Anz.Dat. im Fenster.
12638          bne    :1          ; => OK, weiter...
12639
12640          lda    #$00          ;Zurück auf Anfang.
12641          sta    V4CCB,x
12642          sta    C050B,x          ;Fensterposition.
12643
12644          jmp    Z4A24          ; => WN_SCROLL_U.
12645 ::1          jmp    Z4C0A          ;Fens. aktualisieren.
12646
12647 ; Fensterfunktion: WN_SCROLL_U.
12648 :Z4AF4          jsr    A2BB5          ;Anz.Ze/Sp und Icons.
12649          jsr    S5C89          ;Hole Disk/Fenster.
12650          bcs    :exit
12651
12652          ldx    WIN042F          ;Aktives Fenster.
12653          lda    V0513,x          ;Anz.Dat. auf Disk.
12654          beq    :exit          ; => Ende...
12655
12656          cmp    V4CC7,x          ;Anzahl Icons/Zeile.
12657          beq    :exit
12658          bcs    :1
12659 ::exit          rts
12660
12661 ::1          lda    r0H
12662          pha
12663          lda    r0L
12664          pha
12665
12666          bit    V4A22          ;Maus/Tastatur?
12667          bmi    x4B17          ; => Tastatur...

```

			12668
	jsr	S48B6 ;Langer Mausklick?	12669
	bcs	Z4B7F ; => Nein, weiter...	12670
			12671
	; WN_SCROLL_U: Langer Mausklick oder Shortcut.		12672
:x4B17	jsr	Z31E5 ;Kopiere r0 -> r1.	12673
			12674
;	lda	#< SIZE_DIRBUF ;Speicher für	12675
;	sta	r0L ;Dateinamen und	12676
;	lda	#> SIZE_DIRBUF ;Icons löschen.	12677
;	sta	r0H	12678
;	jsr	ClearRam	12679
	jsr	Z4570 ;Speicher löschen.	12680
			12681
	lda	#\$00 ;Mausabfrage.	12682
	sta	V4A22	12683
			12684
	ldx	WIN042F ;Aktives Fenster.	12685
	lda	C050B,x ;Fensterposition.	12686
	beq	:1 ; => Weiter...	12687
	cmp	V4CCB,x ;Anz. Icons/Seite.	12688
	bcs	:2	12689
			12690
	lda	#\$00 ;Pos. auf Anfang.	12691
	jmp	Z4BFC ;Neue Fenster-Pos.	12692
			12693
	; Zum Ende springen.		12694
::1	lda	V0513,x ;Anz.Dat. auf Disk.	12695
::2	sec		12696
	sbc	V4CCB,x ;Anz. Icons/Seite.	12697
	bcc	:exit ; => Nicht möglich...	12698
	jmp	Z4BFC ;Neue Fenster-Pos.	12699
			12700
	; Nicht möglich, Ende...		12701
::exit	pla		12702
	pla		12703
	rts		12704
			12705
	; WN_SCROLL_U: Kurzer Mausklick.		12706
:Z4B7F	jsr	Z31E0 ;r0 + SIZE_DIRNAM ->	12707
		;Kopieren r0 -> r1.	12708
			12709
	ldx	WIN042F ;Aktives Fenster.	12710
	ldy	V4CC7,x ;Anzahl Icons/Zeile.	12711
			12712
::1	lda	#< 64 ;Icon-Größe:	12713
	clc	;\$bf + 63 Byte Daten.	12714
	adc	r1L	12715
	sta	r1L	12716
	lda	#> 64	12717
	adc	r1H	12718
	sta	r1H	12719
			12720
	dey		12721
	bne	:1	12722

```

12723
12724         lda    r0L
12725         sta    r2L
12726         lda    r0H
12727         sta    r2H
12728
12729         lda    #< SIZE_DIRICON
12730         clc
12731         adc    r2L
12732         sta    r2L
12733         lda    #> SIZE_DIRICON
12734         adc    r2H
12735         sta    r2H
12736
12737         lda    r2L
12738         sec
12739         sbc    r1L
12740         sta    r2L
12741         lda    r2H
12742         sbc    r1H
12743         sta    r2H
12744         jsr    MoveData
12745
12746         lda    r1L
12747         sec
12748         sbc    r0L
12749         sta    r1L
12750         lda    r1H
12751         sbc    r0H
12752         sta    r1H
12753
12754         lda    r1H                ;r0 mit r1 tauschen.
12755         pha
12756         lda    r1L
12757         pha
12758         jsr    Z31E5                ;Kopiere r0 -> r1.
12759         pla
12760         sta    r0L
12761         pla
12762         sta    r0H
12763         jsr    ClearRam
12764
12765         ldx    WIN042F                ;Aktives Fenster.
12766         lda    C050B,x                ;Fensterposition.
12767         beq    :2                    ; => Am Anfang...
12768         cmp    V4CC7,x                ;Anzahl Icons/Zeile.
12769         bcs    :3
12770
12771         pla
12772         sta    r1L
12773         pla
12774         sta    r1H
12775
12776 ;         lda    r1H
12777         pha

```

```

        lda    r1L                                12778
        pha                                12779
                                                12780
;        lda    #< SIZE_DIRBUF                ;Speicher für      12781
;        sta    r0L                            ;Dateinamen und    12782
;        lda    #> SIZE_DIRBUF                ;Icons löschen.     12783
;        sta    r0H                            12784
;        jsr    ClearRam                        12785
;        jsr    Z4570                            ;Speicher löschen. 12786
                                                12787
        ldx    WIN042F                            ;Aktives Fenster.    12788
        lda    #$00                            ;Pos. auf Anfang.    12789
        beq    Z4BFC                            ;Neue Fenster-Pos.    12790
                                                12791
; Zum Ende springen.                            12792
::2        lda    V0513,x                        ;Anz.Dat. auf Disk.    12793
::3        sec                                12794
        sbc    V4CC7,x                        ;Anzahl Icons/Zeile. 12795
                                                12796
; Neue Fensterposition setzen.                    12797
; Übergabe: X = Fenster-Nr.                    12798
;        A = Datei-Nr.                        12799
:Z4BFC        sta    C050B,x                    ;Fensterposition.    12800
        sta    r11H                            12801
        jsr    S1E09                            ;Arbeitsber.Fenster. 12802
                                                12803
        ldx    C042A                            ;Oberstes Fenster. 12804
        jsr    Z4587                            ;Dateien einlesen. 12805
                                                12806
; Fenster aktualisieren.                        12807
:Z4C0A        pla                                12808
        sta    r0L                            12809
        pla                                12810
        sta    r0H                            12811
                                                12812
        inc    V18AD                            ;Fens.-Aufbau-Status. 12813
                                                12814
        jsr    A2BA6                            ;Anzeige Icons/Text 12815
        txa                                ;initialisieren und 12816
        beq    :1                            ;Inhalt anzeigen.   12817
                                                12818
; Benötigt Fenster-Nr. in WIN042F!                12819
        jmp    Z5CEB                            ;Fehler ausgeben und 12820
                                                ;Fenster schließen. 12821
                                                12822
::1        jsr    Z13D3                        ;Textgrenzen Desktop. 12823
        jmp    Z2478                            ;Füllstandsanzeige.   12824
                                                12825
; Hinweis:                                        12826
; CURSOR_RECHTS und CURSOR_LINKS                12827
; simulieren einen langen Mausklick              12828
; auf die Navigationspfeile.                    12829
:V4A22        b    $00 ;$00 = Mausklick, $FF = Shortcut. 12830
                                                12831
; Füllstandsanzeige.                            12832

```

```

12833 :Z4CCF      jsr    A2BB5      ;Anz.Ze/Sp und Icons.
12834          jsr    Z2491      ;Koord. Pos.-Anzeige.
12835
12836          lda    r4L
12837          bne    :0a
12838          dec    r4H
12839 ::0a        dec    r4L
12840
12841          inc    r3L
12842          bne    :0b
12843          inc    r3H
12844
12845 ::0b        ldx    WIN042F      ;Aktives Fenster.
12846          lda    V0517,x        ;Anz.Dat. ab Offset.
12847          sta    r5L
12848          lda    V4CCB,x        ;Anz.Dat. im Fenster.
12849          sta    r6L
12850
12851          lda    #$00          ;Highbyte löschen.
12852          sta    r5H
12853          sta    r6H
12854
12855          ldx    #r5          ;Anteil Dateien/Fens.
12856          ldy    #r6          ;von Gesamtanzahl der
12857          jsr    Ddiv          ;Dateien berechnen.
12858
12859          lda    r5L          ;Anteil = 0 ?
12860          beq    :1          ; => Ja, weiter...
12861
12862          lda    r8L          ;Rest?
12863          beq    :2          ; => Nein, weiter...
12864
12865 ::1        inc    r5L          ;Anteil +1.
12866
12867 ::2        lda    r2H          ;Höhe Balken
12868          sec          ;berechnen.
12869          sbc    r2L
12870          sta    r6L
12871          lda    #$00          ;Highbyte immer 0.
12872          sta    r6H
12873
12874          ldx    #r6          ;Balkenhöhe/Pixel
12875          ldy    #r5          ;durch Anteil Dateien
12876          jsr    Ddiv          ;dividieren.
12877
12878          lda    r6L
12879          cmp    #8          ;Balken < 8 Pixel?
12880          bcs    :3          ; => Nein, weiter...
12881
12882          lda    #8          ;Mindesthöhe setzen:
12883          sta    r6L          ;mind. 8 Pixel hoch.
12884
12885 ::3        lda    r2H          ;Höhe des Rest
12886          sec          ;vom Füllbalken
12887          sbc    r2L          ;berechnen:

```

	sbcb	r6L	;Geamthöhe - H.Balken	12888
	sta	r5L	;Resthöhe speichern.	12889
	sta	V4C53	;Komplette Höhe?	12890
	beq	:4	; => Ja, weiter...	12891
				12892
	ldx	WIN042F	;Aktives Fenster.	12893
	lda	C050B,x	;Fensterposition.	12894
	sta	r10L		12895
				12896
;	lda	#\$00	;Highbyte wird durch	12897
;	sta	r5H	;BBMult gelöscht.	12898
;	sta	r10H	; => Nicht verwendet.	12899
				12900
	ldx	#r5L	;Oberen Versatz des	12901
	ldy	#r10L	;Balkens berechnen.	12902
	jsr	BBMult		12903
				12904
	ldx	WIN042F	;Aktives Fenster.	12905
	lda	V0517,x	;Anz.Dat. ab Offset.	12906
	sta	r10L		12907
				12908
	ldx	#r5		12909
	ldy	#r10		12910
	jsr	Ddiv		12911
				12912
	lda	r2L		12913
	clc			12914
	adc	r5L		12915
	sta	r2L	;Obere Pos. Balken.	12916
	adc	r6L		12917
	sta	r2H	;Untere Pos. Balken.	12918
::4	rts			12919
				12920
:V4C53	b	\$00	;Resthöhe Füllstand ohne Balken.	12921
				12922
; Verschieberoutine für Fenster.				12923
:Z4D6D	dec	V16B8		12924
				12925
	ldx	WIN042F	;Aktives Fenster.	12926
	lda	V4C53	;Resthöhe > 0?	12927
	beq	Z4DB7	; => Ja, zum Anfang.	12928
				12929
; Neue Fensterposition berechnen.				12930
	lda	r2L		12931
	pha			12932
	jsr	Z2491	;Koord. Pos.-Anzeige.	12933
	pla			12934
	sec			12935
	sbcb	r2L		12936
	sta	r0L		12937
				12938
	ldx	WIN042F	;Aktives Fenster.	12939
	lda	V0517,x	;Anz.Dat. ab Offset.	12940
	sta	r10L		12941
				12942

```

12943      ldx    #r10L
12944      ldy    #r0L
12945      jsr    BBMult
12946
12947      lda    V4C53      ;Resthöhe Füllstand
12948      sta    r0L      ;ohne Balken.
12949      lda    #$00
12950      sta    r0H
12951
12952      ldx    #r10L
12953      ldy    #r0L
12954      jsr    Ddiv
12955
12956      ldx    WIN042F      ;Aktives Fenster.
12957      lda    r10L
12958      cmp    V0517,x      ;Anz.Dat. ab Offset.
12959      bcc    Z4DB7      ; => OK, weiter...
12960
12961      ldy    V0517,x      ;Anz.Dat. ab Offset.
12962      dey
12963      tya      ;Ende Dateiliste.
12964
12965      ; Neue Fensterposition setzen.
12966      ; Übergabe: A = Datei-Nr.
12967      ;           X = Fenster-Nr.
12968      :Z4DB7      sta    C050B,x      ;Fensterposition.
12969      jsr    Z4576      ;Dat. neu einlesen.
12970
12971      ldx    WIN042F      ;Aktives Fenster.
12972      jsr    S599A      ;Zeiger auf DIRBUF.
12973      jsr    Z31EE      ;Kopiere r1 -> r0.
12974
12975      inc    V18AD      ;Fens.-Aufbau-Status.
12976
12977      jsr    A2BA6      ;Anzeige Icons/Text
12978      txa      ;initialisieren und
12979      beq    :1      ;Inhalt anzeigen.
12980
12981      ; Benötigt Fenster-Nr. in WIN042F!
12982      jmp    Z5CEB      ;Fehler ausgeben und
12983      ;Fenster schließen.
12984
12985      ::1      jsr    Z13D3      ;Textgrenzen Desktop.
12986      jmp    Z2478      ;Füllstandsanzeige.
12987

```



```

12988
;
12989
; SourceCode: inc.mod#00c.s
12990
;
12991
12992
;*** Icon-Routine: Drucker wechseln/Dateien drucken.
12993
; - Dateien per DnD drucken.
12994
; - Kurzer Klick: Drucker wählen.
12995
; - Langer Klick: Eingabegerät wählen.
12996
:IR4E1C      lda    V0423          ;DnD aktiv?
12997
                bne    MR4E31      ; => Ja, weiter...
12998
12999
                jsr    S48B6      ;Langer Mausklick?
13000
                bcc    :1         ; => Ja, weiter...
13001
13002
; Der AKKU enthält ab hier einen Index
13003
; auf die Dateityp-Tabelle in mod.#09.
13004
                lda    #OFF_OPEN_PRNT    ;Drucker wechseln.
13005
                b $2c
13006
::1          lda    #OFF_OPEN_INPT      ;Input wechseln.
13007
                jsr    Z40C1            ;Datei wählen/öffnen.
13008
                jmp    S3B9B
13009
13010
; Menü: Datei -> drucken
13011
; TD13: DateiDrucken
13012
:MR4E31      lda    #$ff              ;Datei/drucken.
13013
                b $2c
13014
13015
; Menü: Datei -> öffnen
13016
; TD13: Dateiöffnen
13017
:MR4E34      lda    #$00              ;Datei/öffnen.
13018
                sta    V4C54            ;PrintFlag setzen.
13019
13020
                jsr    Z4E67            ;Test MultiFile.
13021
                bcs    :exit           ; => Abbruch...
13022
13023
; Nur eine Datei markiert.
13024
::1          jsr    S5842              ;mark.Dat. anzeigen.
13025
                jsr    S5262            ;Suche mark.Datei.
13026
                jsr    S46EB            ;Name mark.Datei.
13027
                bcs    :exit           ; => Nicht gefunden.
13028
13029
                lda    V4172            ;Quicklink-Modus?
13030
                bne    :2              ; => Ja, weiter...
13031
13032
                jmp    Z5017            ;Datei öffnen.
13033
13034
::2          ldx    #NO_ERROR
13035
                clc
13036
13037
; Bei Abbruch ist hier das C-Flag gesetzt!
13038
::exit      rts
13039
13040
; Test auf Multifile.
13041
:Z4E67      jsr    GotoFirstMenu
13042

```

```

13043
13044             ldx    V04B7             ;Anzahl mark.Dateien.
13045             dex                     ;Nur eine Datei?
13046             beq     :ok              ; => Ja, weiter..
13047
13048             cpx     #$ff             ;Mind. eine Datei?
13049             beq     :err             ; => Nein, Ende...
13050
13051             lda     #< D4E68         ;Fehler: Mehrere
13052             ldx     #> D4E68         ;Dateien ausgewählt.
13053             jsr     S2090            ;DBox ohne Recover.
13054
13055 ; MultiFile, Abbruch...
13056             jsr     S524C            ;Dat.-Ausw. aufheben.
13057             jsr     Z42CD            ;Oberstes Fens. neu.
13058
13059 ::err         sec
13060             rts
13061
13062 ::ok          clc
13063             rts
13064
13065 ; Fehlermeldung: Multifile!
13066 :D4E68        b %10000001
13067
13068             b DBTXTSTR,$08,$10
13069             w T7464
13070
13071             b DBTXTSTR,$08,$20
13072             w :1
13073             b DBTXTSTR,$08,$2a
13074             w :2
13075
13076             b OK          , $11,$48
13077
13078             b NULL
13079
13080 if LANG = LANG_DE
13081 ::1          b "Mehrfachauswahl von Dateien",0
13082 ::2          b "wird nicht unterstützt!",0
13083 endif
13084 if LANG = LANG_EN
13085 ::1          b "Multiple file selection is",0
13086 ::2          b "not supported!",0
13087 endif
13088
13089 ; Hinweis:
13090 ; Wird in mod.#00/02/09 verwendet.
13091 if LANG = LANG_DE
13092 .T7464       b BOLDON
13093             b "Fehler!"
13094             b PLAINTEXT,0
13095 endif
13096 if LANG = LANG_EN
13097 .T7464       b BOLDON

```

```

                                b "Error!"                                13098
                                b PLAINTEXT,0                            13099
endif                                                                    13100
                                                                    13101
; Quicklink starten/ändern.                                            13102
; Ist eine Datei markiert, dann Datei                                  13103
; als neuen Quicklink speichern.                                       13104
; Ist keine Datei markiert, dann wird                                  13105
; der vorhandene Quicklink gestartet.                                  13106
; Quicklink (langer Mausklick).                                         13107
:Z39B1      lda    #17 *0                ;Zeiger geoHDscsi.           13108
                                b $2c                                    13109
:KB4173      lda    #17 *1                ;Zeiger Quicklink#1.         13110
                                b $2c                                    13111
:KB4176      lda    #17 *2                ;Zeiger Quicklink#2.         13112
                                b $2c                                    13113
:KB4179      lda    #17 *3                ;Zeiger Quicklink#3.         13114
                                b $2c                                    13115
:KB417C      lda    #17 *4                ;Zeiger Quicklink#4.         13116
                                sta    V4191                            13117
                                                                    13118
; Quicklink öffnen oder definieren?                                     13119
                                lda    V04B7                            13120
                                beq     :open                            13121
                                                                    13122
; Neuen Quicklink speichern.                                            13123
                                lda    #$01                             13124
                                sta    V4172                            13125
                                                                    13126
; Hinweis:                                                              13127
; Sprung zur "Datei öffnen"-Routine,                                    13128
; die zuerst das Hauptmenü aktiviert.                                   13129
; Ist mehr als eine Datei markiert,                                    13130
; dann MultiFile-Fehler anzeigen und                                    13131
; Funktion abbrechen.                                                  13132
                                jsr     MR4E34                            13133
                                bcc     :save                            13134
                                rts                                       13135
                                                                    13136
; Neuen Quicklink übernehmen.                                            13137
::save      ldy     #$00                                                  13138
                                sty     V4172                            13139
                                                                    13140
                                ldx     V4191                            13141
::1         lda     V0470,y                ;Name kopieren.           13142
                                beq     :3                                13143
                                cmp     #$a0                             13144
                                beq     :2                                13145
                                sta     TD_QLINK_DATA,x                 13146
                                inx                                       13147
                                iny                                       13148
                                cpy     #16                             13149
                                bcc     :1                                13150
                                                                    13151
::2         lda     #NULL                                                13152
                                ;Mit $00 auffüllen.

```

```

13153 ::3          sta    TD_QLINK_DATA,x
13154          inx
13155          iny
13156          cpy    #16 +1
13157          bcc     :3
13158
13159          jmp     S09DD          ;Konfig in REU/Datei.
13160
13161 ; Definierten Quicklink öffnen.
13162 ::open          jmp     Z39B6          ;Quicklink öffnen.
13163
13164 :V4172          b $00    ;>$00="Quicklink Update"-Modus.
13165 .V4191          b $00    ;Zeiger auf Quicklink-Eintrag.
13166
13167 ; Quicklinks.
13168 :TD_QLINK_DATA
13169 :TD_QLINK_SIZE    = 5* 17
13170
13171          t "inc.FKeyApps"
13172
13173          e TD_QLINK_DATA +TD_QLINK_SIZE
13174
13175 ; Ordner verlassen.
13176 ; Aufruf nur aus mod.#08 oder über
13177 ; den ShortCut "<-".
13178 ;
13179 .KB4E72          ldx     C042A          ;Oberstes Fenster.
13180          lda     C0448,x          ;Fenster geöffnet?
13181          beq     exitCLWin          ; => Nein, Ende...
13182
13183          lda     C051B,x          ;Aktive Ordner-Nr.
13184          beq     S4E88          ; => Kein Ordner...
13185
13186 ; Ordner geöffnet.
13187          jsr     S46D4          ;Ordner verlassen.
13188          jsr     S598F          ;Zeiger Ordnerliste.
13189
13190 ;          ldx     C042A          ;Oberstes Fenster.
13191 ::1          jsr     Z288D          ;Vorheriger Ordner.
13192          bne     :1          ; => vorhanden.
13193
13194 ;          lda     #$00
13195          jmp     Z4F10          ; => Kein Ordner.
13196
13197 ; Kein Ordner geöffnet.
13198 ; => CMD-Hauptverzeichnis öffnen.
13199 .S4E88          ldx     C042A          ;Oberstes Fenster.
13200          ldy     C04F8,x          ;Lfwk. für Fenster.
13201          lda     driveType -8,y
13202          and     #ST_DMODES
13203          cmp     #DrvNative          ;NativeMode?
13204          bne     exitCLWin          ; => Nein, weiter...
13205
13206 ;          ldx     C042A          ;Oberstes Fenster.
13207          lda     V04F0,x          ;Dir.T für Fenster.

```

```

        ora    V04F4,x          ;Dir.S für Fenster.      13208
        beq    exitCLWin        ; => Kein SubDir...      13209
                                           13210
;        ldx    C042A            ;Oberstes Fenster.      13211
        jsr    S09B7            ;Tr/Se SubDir = $00.      13212
                                           13213
;        lda    #$00            13214
        sta    C050B,x          ;Fensterposition.        13215
                                           13216
        jsr    S524C            ;Dat.-Ausw. aufheben.     13217
        jsr    Z42CD            ;Oberstes Fens. neu.      13218
                                           13219
:exitCLWin    rts                13220
                                           13221
;*** Shortcut: C= + <-          13222
:KB4EDA        lda    #WN_CLOSE      13223
        sta    WIN042E          ;Fenster-Befehl.        13224
        ldx    C042A            ;Oberstes Fenster.      13225
        stx    WIN042F          ;Aktives Fenster.        13226
                                           13227
        lda    C0448,x          ;Fenster geöffnet?      13228
        beq    exitCLWin        ; => Nein, Ende...        13229
                                           13230
; Fenster schließen.            13231
; Übergabe: X = Fenster-Nr.      13232
:Z4EEF        jsr    S524C          ;Dat.-Ausw. aufheben.  13233
        jsr    S5255            ;Mark.Dat=0/Anzeige.     13234
                                           13235
        jsr    S48B6            ;Langer Mausklick?       13236
                                           13237
        lda    #$00            ;Vorgabe: Kein            13238
        sta    V04AE            ;Klick auf Gadget.       13239
        sta    C050B,x          ;Fensterposition.        13240
                                           13241
; Langer Klick=Hauptverzeichnis öffnen? 13242
        bcc    :root            ; => Ja, weiter...        13243
                                           13244
        ldx    WIN042F          ;Aktives Fenster.        13245
        lda    C051B,x          ;Aktive Ordner-Nr.        13246
        bne    :exit_sub        ; => Ordner...            13247
                                           13248
; Kein Ordner geöffnet.          13249
        lda    C04F8,x          ;Lfwk. für Fenster.      13250
        jsr    SetDevice        13251
                                           13252
; Hinweis:                      13253
; OpenDisk hier nicht aufrufen, da 13254
; bei NativeMode ":OpenRootDir" oder 13255
; ":OpenSubDir" ausgeführt wird.    13256
; Nur bei nicht-NativeMode ausführen. 13257
;        jsr    OpenDisk        13258
                                           13259
        jsr    Z0680            ;Lfwk. = Native?         13260
        bne    :std            ; => Nein, weiter...        13261
                                           13262

```

```

13263         ldx    WIN042F           ;Aktives Fenster.
13264         lda    curDirHead +$22
13265         sta    r1L
13266         sta    V04F0,x           ;Dir.T. für Fenster.
13267         lda    curDirHead +$23
13268         sta    r1H
13269         sta    V04F4,x           ;Dir.S. für Fenster.
13270
13271 ;         lda    r1H
13272         ora    r1L
13273         beq    :root             ;-> Hauptverzeichnis.
13274
13275 ; Elternverzeichnis öffnen (OpenSubDir).
13276         jsr    Z069B             ;-> Unterverzeichnis.
13277         jmp    Z4F13             ;Verz. neu laden.
13278
13279 ; Kein NativeMode -> Disk öffnen.
13280 ::std         jsr    OpenDisk
13281
13282 ; Hauptverzeichnis öffnen (OpenRootDir).
13283 ::root        jsr    Z068B             ;-> Hauptverzeichnis.
13284
13285         lda    C042A             ;Oberstes Fenster.
13286         jsr    S5979             ;Zeiger Fenstertitel.
13287
13288         lda    #"x"             ;Laufwerk im
13289         ldy    #$00             ;Fenstertitel
13290         sta    (r1),y           ;löschen.
13291
13292 ;         lda    #WN_CLOSE        ;Fenster-Befehl ist
13293 ;         sta    WIN042E          ;bereits gesetzt!
13294         jmp    Z48CC             ;Message-Handler.
13295
13296 ; Kurzer Klick=Ordner schließen.
13297 ::exit_sub    jsr    S46D4             ;Ordner verlassen.
13298
13299         ldx    WIN042F           ;Aktives Fenster.
13300         jsr    S598F             ;Zeiger Ordnerliste.
13301         jsr    Z288D             ;Vorheriger Ordner.
13302
13303 ; Neuen Ordner festlegen.
13304 ; Übergabe: X = Fenster-Nr.
13305 ;         A = Ordner-Nr.
13306 :Z4F10        sta    C051B,x       ;Aktive Ordner-Nr.
13307
13308 ; Fensterinhalt neu laden.
13309 :Z4F13        jsr    S524C             ;Dat.-Ausw. aufheben.
13310
13311         jsr    Z5276             ;Verz. neu laden.
13312         txa
13313         beq    :1
13314
13315         jsr    S5CF3             ;Fehler ausgeben.
13316
13317 ::1           rts

```

			13318	
; Auswertung einer Datei-Auswahl.			13319	
; Übergabe: X = Nr. der Datei in der Darstellung (0-15)			13320	
; TD13: File_Selected			13321	
:Z4F1D	lda	dblClickCount	;Doppelklick-Status	13322
	sta	a9L	;einlesen.	13323
				13324
	txa			13325
	sta	a1L	;Datei-Nr. 0-15.	13326
				13327
	ldx	C042A	;Oberstes Fenster.	13328
;	lda	a1L	;Datei-Nr. 0-15.	13329
	cmp	V050F,x	;Anz.Dat. im Fenster.	13330
	bcs	Z4F2E	; => Ungültig.	13331
				13332
	lda	V0423	;DnD aktiv?	13333
	bne	:dnd	; => Ja, weiter...	13334
				13335
	ldx	a1L	;Datei-Nr. 0-15.	13336
	jsr	S51CE	;C=-Taste gedrückt?	13337
	bcs	:1	; => Ja, weiter...	13338
				13339
; Single-Klick.			13340	
	lda	a9L	;Doppelklick?	13341
	bne	:2	; => Ja, weiter...	13342
				13343
	txa			13344
	ldx	C042A	;Oberstes Fenster.	13345
	jsr	Z57D8	;Datei markiert?	13346
	bcs	:2	; => Ja...	13347
				13348
	jmp	Z4F9D	;Pause Doppelklick.	13349
				13350
; C= + Single-Klick.			13351	
::1	jmp	Z5201	;Mehrfach-Auswahl.	13352
				13353
; Auf Doppelklick testen.			13354	
::2	jsr	S5842	;mark.Dat. anzeigen.	13355
	jsr	S524C	;Dat.-Ausw. aufheben.	13356
	jsr	Z51E5	;Dateipos. ermitteln.	13357
				13358
	pha		;Datei-Nr. speichern.	13359
				13360
	jsr	Z5216	;mark.Datei in Tab.	13361
	jsr	S5842	;mark.Dat. anzeigen.	13362
				13363
	lda	a9L	;Doppelklick aktiv?	13364
	bne	:3	; => Ja, weiter...	13365
				13366
	pla		;Neue Datei für	13367
	sta	V4F89	;Doppelklick setzen.	13368
				13369
	lda	#20	;Doppelklick starten.	13370
	sta	dblClickCount		13371
::exit	rts			13372

```

13373
13374 ; Doppelklick.
13375 ::3          pla          ;Doppelklick auf
13376          cmp    V4F89      ;gleiche Datei?
13377          bne    :exit      ; => Nein, Ende...
13378
13379          lda    a1L          ;Datei-Nr. 0-15.
13380          jsr    Z42EC        ;Dateiname einlesen.
13381
13382          lda    #$00
13383          sta    V4C54        ;PrintFlag löschen.
13384
13385          jmp    Z5017        ; => Datei öffnen.
13386
13387 ; DnD auf andere Datei.
13388 ::dnd          lda    V04B7      ;Anzahl mark.Dateien.
13389          cmp    #1          ;Nur eine Datei?
13390          bne    Z4F2E        ; => Nein, weiter...
13391
13392 ; Dateien tauschen.
13393          jsr    Z51E5        ;Dateipos. ermitteln.
13394          cmp    V634F        ;Tab. mark.Dateien.
13395          beq    Z4F2E        ; => Ende...
13396
13397          jsr    Z570E        ;Pos.Datei tauschen.
13398
13399 ; DnD ungültig, Auswahl aufheben.
13400 :Z4F2E          jsr    Z42DB        ;DnD-Modus beenden.
13401          jsr    S5842        ;mark.Dat. anzeigen.
13402          jmp    S524C        ;Dat.-Ausw. aufheben.
13403
13404 :V4F89          b $00      ;Datei-Nr. letzter Mausklick.
13405
13406 ; Pause Doppelklick.
13407 ; TD13: p_d_k
13408 :Z4F9D          jsr    A2BB8        ;Zeiger auf DnD-Icon.
13409
13410          lda    #1          ;DnD-Modus: Dateien.
13411          b $2c
13412 :Z4F9E          lda    #$ff        ;DnD-Modus: Laufwerk.
13413
13414 ; DnD-Sprite erzeugen.
13415          sta    V0423        ;DnD aktivieren.
13416
13417          stx    r4L          ;Adresse DnD-Sprite.
13418          sty    r4H
13419
13420          lda    #1          ;Nummer DnD-Sprite.
13421          sta    r3L
13422
13423          jsr    DrawSprite      ;Sprite einschalten.
13424
13425          jsr    InitForIO
13426
13427          ldx    $d027        ;DnD: Dateien.

```



```

13428
if ENABLE_DRVGHST = TRUE 13429
    bit    V0423           ;DnD für Laufwerke? 13430
    bpl    :1              ; => Nein, weiter... 13431
13432
    ldx    C0L0445         ;DnD: Laufwerke. 13433
endif 13434
13435
::1    stx    $d028        ;Farbe Ghost setzen. 13436
13437
    jmp    DoneWithIO 13438
13439
; Datei öffnen. 13440
; TD13: OpenFile 13441
:Z5017    jsr    S5245      ;Dateien invertieren. 13442
          jsr    Z13D3      ;Textgrenzen Desktop. 13443
          jsr    S5C89      ;Hole Disk/Fenster. 13444
          bcc    :1         13445
          rts 13446
13447
::1    lda    #$00         13448
          sta    a7L        ;Flag: Kurzer Klick. 13449
13450
          jsr    S48B6      ;Langer Mausklick? 13451
          bcs    :2         ; => Nein, weiter... 13452
13453
;          lda    #$01         13454
;          sta    a7L        ;Flag: Langer Klick. 13455
;          inc    a7L        ;Flag: Langer Klick. 13456
13457
; Hinweis: 13458
; Fenster-Nr. in X durch S5C89! 13459
::2    jsr    S598F        ;Zeiger Ordnerliste. 13460
13461
; Hinweis: 13462
; Aufruf aus mod.#09 nachdem ein 13463
; Dokument gewählt wurde. 13464
.S5033    lda    #< V0470    ;Zeiger auf Speicher 13465
          sta    r6L        ;für Dateiname. 13466
          lda    #> V0470    13467
          sta    r6H        13468
13469
          lda    V4C54        ;PrintFlag setzen. 13470
          sta    r1L         13471
13472
; Hinweis: 13473
; Aufruf aus mod.#09 nachdem ein 13474
; Quicklink gewählt wurde. 13475
; Übergabe: r6 = Zeiger auf Dateiname 13476
;          r1L = 1 (Quicklink öffnen) 13477
.S5040    jsr    S2931        ;Datei öffnen. 13478
          txa 13479
          beq    :ok         ; => OK, Ende. 13480
13481
; Fehler beim öffnen der Datei. 13482

```

```

13483 ::err          cpx    #CMD_SUBDIR          ;Verzeichnis öffnen?
13484              beq     :openSubDir
13485              cpx     #ERR_TDFOLDER          ;Fehler: SUB_DIR.
13486              beq     :openFolder
13487              cpx     #BAD_SCRN_RES           ;Nur GEOS128/80Z?
13488              bne     :1                      ; => Nein, weiter...
13489
13490              lda     #< T29D1                ;Inkompatibel.
13491              ldx     #> T29D1
13492              jsr     R29C0                    ;Fehler anzeigen.
13493
13494 ; TODO:
13495 ; Wozu ist das noch erforderlich?
13496 ;
13497 ; Unter GEOS/MP3 ist immer eine REU
13498 ; vorhanden und somit sind immer bis
13499 ; zu vier Laufwerke möglich.
13500 ;
13501 ::1              jmp     S28DB                ;Anz. Lfwk. setzen.
13502
13503 ; CMD-Verzeichnis öffnen.
13504 ::openSubDir     lda     a7L                  ;Langer Mausklick?
13505              bne     :openNew                ; => Ja, weiter...
13506
13507 ; Hinweis:
13508 ; Beim Öffnen des Verzeichnisses muss
13509 ; der Diskname neu eingelesen werden,
13510 ; da der Diskname beim verschieben von
13511 ; Dateien für gleiche Disk oder andere
13512 ; Disk verwendet wird.
13513 ; => Gleiche Disk = Verschieben in
13514 ; einen anderen Ordner.
13515              ldx     C042A                    ;Oberstes Fenster.
13516 ;              lda     #$00
13517              sta     C051B,x                  ;Aktive Ordner-Nr.
13518
13519              jsr     S598F                    ;Zeiger Ordnerliste.
13520              jsr     S28A7                    ;Orderliste löschen.
13521 ;              jsr     S09BC                    ;1.Block > r1L/r1H.
13522              ;(In S2931 gesetzt)
13523              jsr     Z0960                    ;Fens. -> OpenSubDir.
13524
13525              ldx     C042A                    ;Oberstes Fenster.
13526              jsr     Z4675                    ;Diskname einlesen.
13527
13528              jmp     :reload                  ;Inhalt neu laden.
13529
13530 ; Ordner öffnen.
13531 ::openFolder     lda     a7L                  ;Langer Mausklick?
13532              bne     :newFolder                ; => Ja, weiter...
13533
13534              ldx     C042A                    ;Oberstes Fenster.
13535              lda     r10L
13536              sta     C051B,x                  ;Aktive Ordner-Nr.
13537              txa

```

	pha			13538
	jsr	Z46B0	;Ordner in Titel.	13539
	pla			13540
	tax			13541
				13542
::reload	jsr	Z5276	;Verz. neu laden.	13543
	txa			13544
	beq	:ok		13545
	jsr	S5CF3	;Fehler ausgeben.	13546
::ok	rts			13547
				13548
; Hinweis:				13549
; Falls CMD-Verzeichnis aktiv, dann				13550
; Zeiger auf SubDir nach r1L/r1H.				13551
::newFolder	ldx	C042A	;Oberstes Fenster.	13552
	lda	V04F0,x	;Dir.T. für Fenster.	13553
	sta	r1L		13554
	lda	V04F4,x	;Dir.S. für Fenster.	13555
	sta	r1H		13556
				13557
::openNew	jsr	Z1848	;Fenster frei?	13558
	bcc	:newWin	; => Ja, weiter...	13559
				13560
	ldx	C042A	;Oberstes Fenster.	13561
	jsr	S598F	;Zeiger Ordnerliste.	13562
	jmp	Z288D	;Vorheriger Ordner.	13563
				13564
; Ordner in neuem Fenster öffnen.				13565
; Übergabe: X = Fenster-Nr.				13566
; r1 = Zeiger auf CMD-SubDir.				13567
::newWin	stx	V5107	;Neue Fenster-Nr.	13568
				13569
; Hinweis:				13570
; Speichert den Zeiger in r1L/r1H für				13571
; das aktuelle CMD-Verzeichnis in den				13572
; aktuellen Fensterdaten.				13573
	jsr	Z0960	;Fens. -> OpenSubDir.	13574
				13575
	ldx	V5107	;Neue Fenster-Nr.	13576
	jsr	S598F	;Zeiger Ordner-Liste.	13577
				13578
	ldx	C042A	;Oberstes Fenster.	13579
	lda	V0683,x	;Zeiger auf Ordner-	13580
	sta	r1L	;Liste setzen.	13581
	lda	V0687,x		13582
	sta	r1H		13583
				13584
	ldy	#MAXDIRSIZE -1	;Ordner-Liste für	13585
::11	lda	(r1),y	;neues Fens.kopieren.	13586
	sta	(r0),y		13587
	dey			13588
	bpl	:11		13589
				13590
	lda	V5107	;Neue Fenster-Nr.	13591
	jsr	S5979	;Zeiger Fenstertitel.	13592

```

13593          jsr    Z31EE          ;Kopiere r1 -> r0.
13594
13595          lda    C042A          ;Oberstes Fenster.
13596          jsr    S5979          ;Zeiger Fenstertitel.
13597
13598  ::12          lda    (r1),y
13599          sta    (r0),y
13600          dey
13601          bpl    :12
13602
13603          ldx    C042A          ;Oberstes Fenster.
13604          ldy    V5107          ;Neue Fenster-Nr.
13605          lda    C04F8,x        ;Lfwk. für Fenster.
13606          sta    C04F8,y        ;Lfwk. für Fenster.
13607          lda    C04FC,x        ;Part. für Fenster.
13608          sta    C04FC,y        ;Part. für Fenster.
13609
13610          tya
13611          jsr    Z46B3          ;Ordner in Titel.
13612
13613          lda    r10L          ;Ordner-Nr.
13614          pha
13615
13616          ldx    C042A          ;Oberstes Fenster.
13617          jsr    S598F          ;Zeiger Ordnerliste.
13618
13619          jsr    Z288D          ;Vorheriger Ordner.
13620
13621          pla
13622          ldx    V5107
13623          jmp    Z453C          ;Ordner öffnen.
13624
13625  :V5107          b $00 ;Neue Fenster-Nr.
13626  .V4C54          b $00 ;PrintFlag: $00/$80=Öffnen/Drucken.
13627
13628  if LANG = LANG_DE
13629  :T29D1          b "Programm nur mit GEOS128 lauffähig!",0
13630  endif
13631  if LANG = LANG_EN
13632  :T29D1          b "Program only runs under GEOS128!",0
13633  endif
13634
13635  ; Auf C=-Taste prüfen.
13636  ; Rückgabe: C=SEC: C= gedrückt.
13637  ;          C=CLC: C= nicht gedrückt.
13638  ; TD13: TestCBMKey
13639  .S51CE          jsr    InitForIO
13640
13641          lda    %%01111111
13642          sta    $dc00
13643
13644          lda    $dc01
13645          and    %%00100000
13646
13647          pha

```

```

        jsr    DoneWithIO                13648
        pla                                13649
                                           13650
        clc                                13651
        bne    :1                        13652
        sec                                13653
::1      rts                            13654
                                           13655
; Position für Datei ermitteln.          13656
; Übergabe: a1L = Datei-Nr. in der Darstellung (0-15) 13657
; Rückgabe: X   = Datei-Nr. (0-254)      13658
; TD13: GetIndex                        13659
:Z51E5   ldx    C042A                    ;Oberstes Fenster. 13660
        ldy    C04B2                    ;Icon-/Text-Modus. 13661
        beq    :1                        ; => Icons.        13662
        dey                                13663
        beq    :1                        ; => Text.         13664
                                           13665
; Sortierte Anzeige.                    13666
        lda    a1L                      13667
        jsr    A2BBE                    ;mark.Datei suchen. 13668
        tax                                13669
        rts                            13670
                                           13671
; Anzeige Icons oder Text.              13672
::1      lda    a1L                      ;Pos. in der        13673
        clc                                ;Darstellung +    13674
        adc    C050B,x                  ;Fensterposition. 13675
        tax                                ;Dateinummer 0-254. 13676
        rts                            13677
                                           13678
; Mehrfach Dateiauswahl.                13679
; TD13: Multi_Select                    13680
:Z5201   jsr    InvertRectangle          13681
                                           13682
        ldx    C042A                    ;Oberstes Fenster. 13683
        lda    a1L                      13684
        jsr    Z57D8                    ;Datei markiert?    13685
                                           13686
        tax                                ;Datei-Nr.         13687
                                           13688
        bcc    :1                        ; => Ja, abwählen. 13689
                                           13690
        jmp    Z5216                    ;mark.Datei in Tab. 13691
::1      jmp    Z5223                    ;mark.Dat. abwählen. 13692
                                           13693
; Datei in Tabelle mit mark.Dateien aufnehmen. 13694
; Übergabe: X = Nummer der Datei (0-254) 13695
; TD13: MarkFile                        13696
:Z5216   ldy    V04B7                    ;Anzahl mark.Dateien. 13697
        txa                                13698
        sta    V634F,y                  ;Tab. mark.Dateien. 13699
        inc    V04B7                    ;Anzahl mark.Dateien. 13700
        jmp    S5802                    ;Anzahl anzeigen.    13701
                                           13702

```

```

13703 ; Markierte Datei aus Tabelle entfernen.
13704 ; Übergabe: X = Nummer der Datei (0-254)
13705 ; TD13: UnMarkFile
13706 :Z5223          ldy    #MAXFILES
13707                txa
13708 ::1             cmp    V634F -1,y          ;Tab. mark.Dateien.
13709                beq    :2
13710                dey
13711                bne    :1
13712                rts
13713
13714 ::2             lda    V634F +0,y          ;Tab. mark.Dateien.
13715                sta    V634F -1,y          ;Tab. mark.Dateien.
13716                iny
13717                cpy    #MAXFILES
13718                bne    :2
13719
13720                lda    #$ff                ;Letzte Dat.abwählen.
13721                sta    V634F +MAXFILES -1
13722
13723                dec    V04B7                ;Anzahl mark.Dateien.
13724                jmp    S5802                ;Anzahl anzeigen.
13725
13726 ; Dateien invertieren, Dateiauswahl aufheben.
13727 ; Hinweis: X bleibt unverändert!
13728 ; TD13: ClearMultiFile2
13729 .S5245          txa
13730                pha
13731                jsr    S5842                ;mark.Dat. anzeigen.
13732                pla
13733                tax
13734
13735 ; Dateiauswahl aufheben.
13736 ; Hinweis: X bleibt unverändert!
13737 ; TD13: ClearMultiFile
13738 .S524C          ldy    #MAXFILES
13739                tya
13740 ;                lda    #$ff
13741 ::1             sta    V634F -1,y          ;Tab. mark.Dateien.
13742                dey
13743                bne    :1
13744
13745 ; Mark.Dateien auf 0 setzen.
13746 .S5255          lda    #$00
13747                sta    V04B7                ;Anzahl mark.Dateien.
13748                jmp    S5802                ;Anzahl anzeigen.
13749
13750 ; Erste markierte Datei holen,
13751 ; Abbruch wenn Taste gedrückt.
13752 ; Übergabe: A = Datei-Nr,
13753 ; Rückgabe: X = Aktives Fenster.
13754 ;           A = 0-254: mark.Datei.
13755 ;           $FF : Ende.
13756 ;           C = CLC : Mark.Datei gefunden.
13757 ;           SEC  : Keine mark.Datei gefunden, Ende.

```

```

:Z525E          bit   pressFlag          ;Taste gedrückt?      13758
                  bpl   S5262              ; => Nein, weiter...    13759
                  ;                               13760
                  jsr   S524C              ;Dat.-Ausw. aufheben.  13761
                  ;                               13762
; Erste markierte Datei holen.                  13763
; Übergabe: A = Datei-Nr,                      13764
; Rückgabe: X = Aktives Fenster.                13765
;         A = 0-254: mark.Datei.                13766
;         $FF : Ende.                          13767
;         C = CLC : mark.Datei gefunden.        13768
;         SEC  : Ende.                          13769
; TD13: GetMark                                13770
.S5262          lda   V634F                ;Tab. mark.Dateien.    13771
                  cmp   #$ff              ;Datei markiert?    13772
                  beq   :1                ; => Nein, Ende...     13773
                  ;                               13774
                  pha                               13775
                  tax                               13776
                  jsr   Z5223              ;mark.Dat. abwählen. 13777
                  ;                               13778
                  ldx   C042A              ;Oberstes Fenster.  13779
                  pla                               13780
                  ;                               13781
                  clc                       ;Datei gefunden.    13782
                  rts                       13783
                  ;                               13784
::1             sec                       ;Keine Datei.        13785
                  rts                       13786
                  ;                               13787
; Neues Verzeichnis laden.                      13788
; Rückgabe : X = Fehler.                      13789
; TD13: NewDirLoad                             13790
:Z5276          ldx   C042A                ;Oberstes Fenster.  13791
                  lda   #$00              13792
                  sta   C050B,x           ;Fensterposition.    13793
                  ;                               13794
                  jsr   Z4576              ;Dat. neu einlesen.  13795
                  bcs   :1                13796
                  ;                               13797
                  jsr   Z15CB              ;Oberstes Fens. neu.  13798
                  ;                               13799
                  ldx   #NO_ERROR          13800
::1             rts                       13801
                  ;                               13802
; Dialogbox: Dateien überschreiben?            13803
:D5289          b %10000001              13804
                  ;                               13805
                  b DBTXTSTR,$08,$0e      13806
                  w :1                     13807
                  b DBTXTSTR,$08,$18      13808
                  w :2                     13809
                  ;                               13810
                  b YES      , $11,$08    13811
                  ;                               13812

```

```

13813      b DBTXTSTR,$14,$23
13814      w T04F6
13815
13816      b DBTXTSTR,$08,$2e
13817      w :3
13818      b DBTXTSTR,$08,$38
13819      w :4
13820
13821      b OK      , $11,$28
13822
13823      b DBTXTSTR,$08,$52
13824      w T04F5
13825
13826      b CANCEL  , $11,$48
13827
13828      b NULL
13829
13830 if LANG = LANG_DE
13831 ::1      b "Datei(en) auf Ziel-Disk",0
13832 ::2      b "immer "
13833          b BOLDON,"überschreiben"
13834          b PLAINTEXT," ?",0
13835
13836 ::3      b "Vor dem überschreiben",0
13837 ::4      b BOLDON,"Bestätigung"
13838          b PLAINTEXT," anfordern ?",0
13839 endif
13840 if LANG = LANG_EN
13841 ::1      b "Always "
13842          b BOLDON,"overwrite"
13843          b PLAINTEXT," files",0
13844 ::2      b "on destination disk ?",0
13845
13846 ::3      b "Request "
13847          b BOLDON,"confirmation",PLAINTEXT,0
13848 ::4      b "before overwriting ?",0
13849 endif
13850
13851 ; Dialogbox: Dateien kopieren/verschieben?
13852 :D4224      b %10000001
13853
13854      b DBTXTSTR,$08,$0e
13855      w :1
13856      b DBTXTSTR,$08,$18
13857      w :2
13858
13859      b YES      , $11,$08
13860
13861      b DBTXTSTR,$14,$23
13862      w T04F6
13863
13864      b DBTXTSTR,$08,$2e
13865      w :3
13866      b DBTXTSTR,$08,$38
13867      w :4

```



		13868
	b OK , \$11, \$28	13869
		13870
	b DBTXTSTR, \$08, \$52	13871
	w T04F5	13872
		13873
	b CANCEL , \$11, \$48	13874
		13875
	b NULL	13876
		13877
if LANG = LANG_DE		13878
::1	b "Markierte Datei(en) auf", 0	13879
::2	b "die Ziel-Disk "	13880
	b BOLDON, "kopieren"	13881
	b PLAINTEXT, " ?", 0	13882
		13883
::3	b "Datei(en) "	13884
	b BOLDON, "verschieben"	13885
	b PLAINTEXT, " ?", 0	13886
::4	b "(auf Original-Disk löschen)", 0	13887
endif		13888
if LANG = LANG_EN		13889
::1	b BOLDON, "Copy"	13890
	b PLAINTEXT, " selected file(s)", 0	13891
::2	b "to destination disk ?", 0	13892
		13893
::3	b BOLDON, "Move"	13894
	b PLAINTEXT, " selected file(s) ?", 0	13895
::4	b "(delete source files)", 0	13896
endif		13897
		13898
; Dialogbox: Datei mit Schreibschutz löschen?		13899
:D5BF2	b %10000001	13900
		13901
	b DBTXTSTR, \$12, \$38	13902
	w V0470	13903
	b DBTXTSTR, \$08, \$38	13904
	w T29D3	13905
		13906
	b DBTXTSTR, \$08, \$10	13907
	w T29CF	13908
		13909
	b DBTXTSTR, \$08, \$20	13910
	w :1	13911
	b DBTXTSTR, \$08, \$2a	13912
	w T04F3	13913
		13914
	b YES , \$01, \$48	13915
	b NO , \$11, \$48	13916
		13917
	b NULL	13918
		13919
if LANG = LANG_DE		13920
::1	b PLAINTEXT	13921
	b "Wirklich löschen?"	13922

```

13923          b BOLDON,0
13924 endif
13925 if LANG = LANG_EN
13926 ::1          b PLAINTEXT
13927          b "Really delete?"
13928          b BOLDON,0
13929 endif
13930
13931 ; Auf CBM/SHIFT testen.
13932 :Z423B      jsr   InitForIO
13933 ;          lda   #%00001001      ;???
13934          lda   #%00111101      ;CBM/L+R-SHIFT.
13935          sta   $dc00
13936          ldy   $dc01
13937          jmp   DoneWithIO
13938
13939 ; Fortsetzung WindowHandler:
13940 ; Verschieben/Kopieren von Dateien.
13941 ; Rückgabe: a8L = 0: Quelldisk=Zieldisk.
13942 ; TD13: MoveService
13943 :Z5357      ldx   C042A          ;Oberstes Fenster.
13944          lda   C04F8,x          ;Lfwk. für Fenster.
13945          sta   V0A72          ;Lfwk. Source-Disk.
13946
13947          lda   WIN042F          ;Aktives Fenster.
13948          pha
13949          tax
13950          lda   C04F8,x          ;Lfwk. für Fenster.
13951          sta   V0A73          ;Lfwk. Target-Disk.
13952          lda   C042A          ;Oberstes Fenster.
13953          sta   WIN042F          ;Aktives Fenster.
13954          jsr   S5842          ;mark.Dat. anzeigen.
13955          pla
13956          sta   WIN042F          ;Aktives Fenster.
13957
13958          lda   #$00
13959          sta   a8L          ;Quelldisk=Zieldisk.
13960          sta   V04A9          ;Vorgabe: Kopieren.
13961          sta   V04B1          ;Vorg:Nicht ersetzen.
13962          sta   V04B4          ;Modus:Copy/n.ersetz.
13963          sta   V04AB          ;Vorgabe: OK.
13964          sta   V5623          ;Text "Kopiere".
13965          sta   V5624          ;Text "Verschiebe".
13966          sta   V5611          ;Status "Datei".
13967
13968 ; Hinweis:
13969 ; SubDir-Hinweis nur nach dem ersten
13970 ; Programmstart anzeigen.
13971 ;          sta   V5596          ;SubDir-Info löschen.
13972
13973 ; Ist Zieldisk = Quelldisk?
13974          ldx   WIN042F          ;Aktives Fenster.
13975          jsr   Z59F4          ;Diskname für Fens.
13976          bne   :0
13977          jmp   :samedisk      ; => Gleiche Disk...

```

```

13978
; Quelldisk <> Zieldisk. 13979
::0      jsr    S48B6      ;Langer Mausklick? 13980
          bcs    :testkeys ; => Nein, weiter... 13981
13982
; Hinweis: 13983
; Der Kopiervorgang beginnt dann erst 13984
; wenn Maustaste nicht mehr gedrückt. 13985
; Bei langem Mausklick erforderlich, 13986
; da sonst nach der ersten Datei der 13987
; Vorgang beendet wird. 13988
          jsr    S52AD      ;Maustaste loslassen. 13989
13990
          dec    V04B4      ;Modus: Ersetzen. 13991
13992
; Hinweis: 13993
; Langer Mausklick = Kopieren/Ersetzen. 13994
; Ohne den bne-Befehl erscheint dann 13995
; die Abfrage Kopieren / Verschieben? 13996
;--- 13997
          bne    :replace   ;Kopieren/Ersetzen. 13998
;--- 13999
14000
; Hinweis: 14001
; 14002
; TOPDESK: 14003
; Keine Tasten: Dialogbox anzeigen. 14004
; Langer Mausklick: Ersetzen. 14005
; L-SHIFT: Kopieren. 14006
; L-SHIFT+R-SHIFT/CBM: Kopieren/ersetzen. 14007
; CBM: Dateien verschieben. 14008
; CBM+R-SHIFT: Dateien verschieben. 14009
if FALSE 14010
::testkeys jsr    Z423B      ;Tastatur abfragen. 14011
          cpy    #%01111111 ;Tasten gedrückt? 14012
          beq    :copy       ; => L-SHIFT. 14013
          bcc    :replace    ; => L+R-SHIFT/CBM. 14014
14015
          cpy    #%11011111 ;CBM gedrückt? 14016
          beq    :move       ; => CBM. 14017
;          bcc    :move       ; => CBM+R-SHIFT. 14018
endif 14019
14020
; DTOPDESK: 14021
; Keine Tasten: Dialogbox anzeigen. 14022
; L/R-SHIFT: Kopieren/ersetzen. 14023
; CBM: Dateien verschieben. 14024
; CBM+L/R-SHIFT: Verschieben/ersetzen. 14025
if TRUE 14026
::testkeys jsr    Z423B      ;Tastatur abfragen. 14027
          cpy    #%01111111 ;Tasten gedrückt? 14028
          beq    :replace    ; => L-SHIFT. 14029
          cpy    #%11101111 14030
          beq    :replace    ; => R-SHIFT. 14031
14032

```

```

14033          tya
14034          and    %%00100000          ;CBM gedrückt?
14035          bne    :dbox                ; => Nein, weiter...
14036
14037          inc    V04A9                ;Modus: Verschieben.
14038
14039          cpy    %%01011111          ;L/R-SHIFT gedrückt?
14040          beq    :replace              ; => CBM+L-SHIFT.
14041          cpy    %%11001111
14042          beq    :replace              ; => CBM+R-SHIFT.
14043          bne    :dbox_replace        ;DBox anzeigen.
14044      endif
14045
14046      ; Dialogbox: Kopieren/Verschieben.
14047      ::dbox          bit    C04B5          ;Kopieren-Dialog?
14048          bpl    :copy                ; => Nein, weiter...
14049
14050      ; Optional: Bei einer Datei immer Copy.
14051      if FALSE
14052          lda    V04B7                ;Anzahl mark.Dateien.
14053          cmp    #2                  ;Mehr als eine Datei?
14054          bcc    :dbox_replace        ; => Nein, weiter...
14055      endif
14056
14057          lda    #< D4224              ;Dialogbox: Kopieren
14058          ldx    #> D4224              ;oder Verschieben?
14059          jsr    S2080                ;DBox, k. Maustaste.
14060
14061          lda    sysDBData
14062          cmp    #OK                  ;Verschieben?
14063          bne    :2                    ; => Nein, weiter...
14064
14065          inc    V04A9                ;Modus: Verschieben.
14066
14067      ::2          cmp    #CANCEL        ;Abbruch?
14068          beq    :cancel              ; => Ja, Ende...
14069
14070      ; Dialogbox: Dateien überschreiben?
14071      ::dbox_replace  bit    V04B4          ;Kopieren/Ersetzen?
14072          bmi    :replace              ; => Ersetzen.
14073
14074      ; Optional: Bei einer Datei immer Replace.
14075      if FALSE
14076          lda    V04B7                ;Anzahl mark.Dateien.
14077          cmp    #2                  ;Mehr als eine Datei?
14078      ;          bcc    :replace          ; => Nein, weiter...
14079          bcc    :copy                ;Vorgabe: Kopieren.
14080      endif
14081
14082          lda    #< D5289              ;Dialogbox: Dateien
14083          ldx    #> D5289              ;überschreiben?
14084          jsr    S2080                ;DBox, k. Maustaste.
14085
14086          lda    sysDBData
14087          cmp    #CANCEL

```

```

                                bne    :3                                14088
                                14089
; Abbruch Kopiervorgang.                                14090
::cancel                      jsr     S524C                        ;Dat.-Ausw. aufheben. 14091
                                jmp     Z42CD                        ;Oberstes Fens. neu. 14092
                                14093
; Modus: Verschieben (CBM gedrückt).                    14094
::move                        lda     #$01                        14095
                                sta     V04A9                      ;Modus: Verschieben. 14096
                                14097
                                bit     V04B4                      ;Kopieren/Ersetzen? 14098
                                bpl     :copy                      ; => Kopieren. 14099
                                bmi     :replace                    ; => Ersetzen. 14100
                                14101
; Fortsetzung:                                           14102
; Dialogbox: Dateien überschreiben?                      14103
::3                          cmp     #YES                        ;Überschreiben? 14104
                                bne     :copy                      ; => Nein, weiter... 14105
                                14106
; Modus: Dateien kopieren/ersetzen (SHIFT gedrückt).    14107
::replace                     lda     #"/"                        14108
                                sta     V04B1                      ;Modus: Dat.ersetzen. 14109
                                sta     V5623                      ;"Kopiere/Ersetze". 14110
                                sta     V5624                      ;"Versch./Ersetze". 14111
                                14112
; Modus: Dateien kopieren.                                14113
::copy                        lda     #1                          ;Quell- und Zieldisk 14114
                                sta     a8L                        ;sind verschieden. 14115
                                bne     Z5414                      14116
                                14117
; Verschieben von Dateien auf gleicher                   14118
; Disk in einen anderen Ordner.                          14119
; => Markierte Dateien sortieren.                       14120
::samedisk                    jsr     Z5C50                        ;BubbleSort. 14121
                                14122
; Kopiervorgang ausführen.                                14123
; Hinweis:                                               14124
; Tastenstatus löschen, da sonst der                    14125
; Kopiervorgang schon vor der ersten                    14126
; Datei beendet wird wenn ein langer                    14127
; Mausklick oder die Tasten SHIFT /                     14128
; CBM gedrückt werden.                                  14129
:Z5414                        lda     #%00000000                14130
                                sta     pressFlag                  ;Tasten-Stat.löschen. 14131
                                14132
; Alternativ:                                             14133
; Der Kopiervorgang beginnt dann erst                    14134
; wenn Maustaste nicht mehr gedrückt.                    14135
;                                                         14136
                                jsr     S52AD                        ;Maustaste loslassen. 14137
                                14138
::loop                        jsr     Z525E                        ;Mark.Datei suchen. 14138
                                bcs     :done                      ; => Keine Datei... 14139
                                14140
; Hinweis:                                               14141
; Im Akku befindet sich die nächste                     14142

```

```

14143 ; ausgewählte Datei. Diese wurde dabei
14144 ; aus aus der Liste der markierten
14145 ; Dateien entfernt.
14146             ldy    a8L                ;Quelldisk=Zieldisk?
14147             beq    :samedisk          ; => Ja, weiter...
14148
14149 ;--- Quelldisk <> Zieldisk.
14150 ;             lda    #$ff                ;Datei-Nr. im Akku.
14151             jsr    S5537                ;CopyService.
14152             bcc    :loop                ; => OK, weiter...
14153
14154             jsr    S524C                ;Dat.-Ausw. aufheben.
14155
14156             lda    #$00                ;Kein VLIR-Modul mehr
14157             sta    V0422                ;im Speicher!
14158
14159             jsr    S31F7                ;Anfang CopyMem.
14160
14161 ; Hinweis:
14162 ; Da keine Datei mehr markiert ist,
14163 ; führt das fortsetzen des kopierens
14164 ; zum beenden der Routine.
14165             jmp    Z5414                ; => Ende.
14166
14167 ;--- Quelldisk = Zieldisk.
14168 ;Datei-Nr. im Akku!
14169 ::samedisk    jsr    S46EB                ;Name mark.Datei.
14170             bcc    :move                ; => Datei gefunden.
14171
14172 ;--- Fehler, Auswahl aufheben und Ende.
14173 ::exit        jsr    S5CF3                ;Fehler ausgeben.
14174             jsr    S524C                ;Dat.-Ausw. aufheben.
14175             jmp    Z5414
14176
14177 ;--- Datei zum verschieben gefunden.
14178 ::move        ldx    WIN042F                ;Aktives Fenster.
14179             lda    C051B,x                ;Aktive Ordner-Nr.
14180             sta    r10L
14181
14182             jsr    S1412                ;Akt.Datei suchen.
14183             txa
14184             bne    :exit                ; => Fehler, Ende...
14185
14186             lda    dirEntryBuf +22
14187             cmp    #TDFOLDER                ;Ordner?
14188             bne    :notdsb                ; => Nein, weiter...
14189
14190 ; *TDFOLDERFIX*
14191 ;
14192 ; Der VLIR-Test funktioniert hier nicht
14193 ; zuverlässig. Ein Ordner hat als Tr/Se
14194 ; $00/$ff in Byte #1/2, da ein Ordner
14195 ; keine Daten beinhaltet.
14196             lda    dirEntryBuf +1                ;Daten vorhanden?
14197             bne    :notdsb                ; => Kein Ordner.

```

```

;          lda    dirEntryBuf +21                14198
;          cmp    #VLIR                          ;VLIR-Datei?      14199
;          beq    :notdsub                        ; => Ja, weiter... 14200
;                                                  14201
;          jsr    S09C7                          ;Infoblock einlesen. 14202
;                                                  14203
; Hinweis:                                       14204
; Ist ein Ordner im aktuellen Fenster          14205
; bereits geöffnet, dann den Ordner            14206
; nicht verschieben.                          14207
;          lda    diskBlkBuf +OFF_DIR_NUM        14208
;          jsr    Z551E                          ;Ordner geöffnet?  14209
;          bcs    :1                             ; => Ja, weiter... 14210
;                                                  14211
::notdsub   jsr    S2832                          ;Datei in Ordner      14212
;          txa                                ;verschieben.        14213
;          bne    :exit                         ; => Fehler, Ende... 14214
;                                                  14215
::1         jmp    Z5414                          14216
;                                                  14217
; Ende von "MoveService".                      14218
::done      lda    C042A                          ;Oberstes Fenster.  14219
;          pha                                ;(Quelldisk)         14220
;                                                  14221
; Zieldisk öffnen.                            14222
;          ldx    WIN042F                      ;Aktives Fenster.  14223
;          jsr    Z5CB3                        ;Diskette öffnen.  14224
;          txa                                14225
;          beq    :2                             ; => OK, weiter... 14226
;                                                  14227
::error     pla                                14228
;          jmp    S5CF3                        ;Fehler ausgeben.   14229
;                                                  14230
; Dateien auf Zieldisk einlesen.             14231
::2         ldx    WIN042F                      ;Aktives Fenster.  14232
;          jsr    Z4576                        ;Dat. neu einlesen.  14233
;          bcs    :error                       ; => Fehler...      14234
;                                                  14235
; Zielfenster aktualisieren.                  14236
;          ldx    WIN042F                      ;Aktives Fenster.  14237
;          inc    V18AD                        ;Fens.-Aufbau-Status. 14238
;          sec                                14239
;          jsr    Z1C10                        ;Fenster nach vorne. 14240
;                                                  14241
; Quellfenster aktualisieren.                 14242
;          ldy    a8L                          ;Quelle=Ziel?      14243
;          beq    :skip                        ; => Ja, weiter... 14244
;                                                  14245
; Kopieren Quell->Ziel.                      14246
;          pla                                14247
;          tax                                ;Fenster Quelldisk.  14248
;          clc                                14249
;          bcc    :update                      ;Fens. nur anzeigen. 14250
;                                                  14251
; Kopieren Quelle=Ziel.                      14252

```

```

14253 ::skip          pla
14254                pha
14255                tax          ;Fenster Quelldisk.
14256                jsr  Z4576   ;Dat. neu einlesen.
14257                bcs  :error   ; => Fehler...
14258
14259                pla
14260                tax          ;Fenster Quelldisk.
14261                inc  V18AD     ;Fens.-Aufbau-Status.
14262                sec          ;Fens. neu zeichnen.
14263 ::update        jsr  Z1C10   ;Fenster nach vorne.
14264
14265                jsr  Z42DB     ;DnD-Modus beenden.
14266
14267                lda  V04A9     ;Kop.od.verschieben?
14268                beq  S54E3     ; => Kopieren.
14269
14270                ldx  WIN042F   ;Aktives Fenster.
14271                lda  #$00
14272                sta  C050B,x   ;Fensterposition.
14273                jsr  Z4299     ;Update Fenster.
14274
14275 ; Hauptmenü, bei UV Fenster neu.
14276 .S54E3          ldx  WIN042F   ;Aktives Fenster.
14277                lda  V04F0,x   ;Dir.T. für Fenster.
14278                cmp  #$02     ;Unterverzeichnis?
14279                bcc  S54F0     ; => Nein, weiter...
14280                jsr  S4321     ;Alle Fenster neu.
14281
14282 ; Hauptmenü aktivieren.
14283 ; TD13: DoHauptMenu
14284 .S54F0          php
14285                sei
14286
14287                lda  mouseXPos +1
14288                pha
14289                lda  mouseXPos +0
14290                pha
14291                lda  mouseYPos
14292                pha
14293
14294 ; Hinweis:
14295 ; DoMenu löscht leftMargin. Ist aber
14296 ; das Menü ausserhalb des Bereichs von
14297 ; windowTop / windowBottom, dann wird
14298 ; das Menü nicht angezeigt.
14299 ; Die Routine setzt y-oben/unten auf
14300 ; den gesamten Bildschirmbereich.
14301                jsr  Z13D3     ;Textgrenzen löschen.
14302
14303                lda  #< DM3C04
14304                sta  r0L
14305                lda  #> DM3C04
14306                sta  r0H
14307

```



```

        ldy    #(r4H - r2L)                14308
:::1    lda    (r0),y                      14309
        sta    r2,y                        14310
;       sta    windowTop,y                ;Top/Bottom löschen. 14311
        dey                                14312
        bpl    :1                          14313
                                           14314
        lda    COL0444                    ;Farbe GEOS-Menü. 14315
        jsr    S1561                      ;Screencolor setzen. 14316
                                           14317
;       lda    #0                        ;Menu-Pos.ignorieren. 14318
        jsr    DoMenu                    ;Menü starten. 14319
                                           14320
        pla                                14321
        sta    mouseYPos                  14322
        pla                                14323
        sta    mouseXPos +0              14324
        pla                                14325
        sta    mouseXPos +1              14326
                                           14327
        plp                                14328
        rts                               14329
                                           14330
; Ordner-Nr. testen.                      14331
; Übergabe: A = Ordner-Nr.                14332
; Rückgabe: C = SEC / Bereits vorhanden. 14333
;           CLC / Nicht vorhanden.        14334
; TD13: CheckDirNum                      14335
:Z551E    sta    r1L                      14336
                                           14337
        ldx    WIN042F                    ;Aktives Fenster. 14338
        jsr    S598F                      ;Zeiger Ordnerliste. 14339
                                           14340
        ldy    #$00                      14341
:::1    lda    (r0),y                      14342
        bmi    :2                          14343
        cmp    r1L                        14344
        beq    :3                          14345
        iny                                14346
        bne    :1                          14347
                                           14348
:::2    clc                               14349
        rts                               14350
                                           14351
:::3    sec                               14352
        rts                               14353
                                           14354
; Kopierfunktion.                        14355
; Übergabe: A = Aktuelle Datei-Nr.        14356
; Rückgabe: C = 0: Kopieren fortsetzen. 14357
;           C = 1: Kopieren beenden.      14358
;           X = Fehler.                    14359
;                                           14360
;           Nur "Ordner kopieren":        14361
;           A = $FF: Datei nicht im akt.Ordner. 14362

```

```

14363 ; TD13: CopyService
14364 .S5537          sta  V04AA          ;Datei-Nr.
14365              pha
14366              ldx  WIN042F          ;Aktives Fenster.
14367              jsr  Z59F4          ;Diskname für Fens.
14368
14369              ldy  #16 +2
14370 ::1            lda  V0483,y          ;Diskname kopieren.
14371              sta  V0496,y
14372              dey
14373              bpl  :1
14374
14375              jsr  S59F1          ;Diskname akt.Fens.
14376
14377              pla
14378              jsr  S46EB          ;Name mark.Datei.
14379              bcc  :2          ; => Datei gefunden.
14380              txa
14381              bne  :copyErr        ; => Fehler.
14382
14383              lda  #$ff          ;Nicht im Ordner.
14384 ;              sec
14385              rts
14386
14387 ::2            ldx  V04A9          ;Kop.od.verschieben.
14388 ;              ldx  #STAT_COPY      ;Status "Kopiere".
14389 ;              ldx  #STAT_MOVE      ;Status "Versch.".
14390              jsr  S55B4          ;Status anzeigen.
14391
14392              ldx  #(r13H - r10L)    ;r10: Diskname 1.
14393 ::3            lda  V55AE,x          ;r11: Diskname 2.
14394              sta  r10,x            ;r12: Dateiname.
14395              dex                  ;r13: Dateiname.
14396              bpl  :3
14397
14398              ldx  WIN042F          ;Aktives Fenster.
14399              lda  C051B,x          ;Aktive Ordner-Nr.
14400              sta  V0A71          ;Nr. Ziel-Ordner.
14401
14402              lda  r12L
14403 ;              sta  r13L
14404 ;              sta  r6L
14405              ldx  r12H
14406 ;              stx  r13H
14407 ;              stx  r6H
14408 ;              jsr  FindFile        ;Auf Ziel suchen.
14409              jsr  S1414          ;FindFile.
14410              txa
14411              bne  :copyErr        ; => Nicht gefunden.
14412
14413 ; Kopierfunktion.
14414 ; *TDFOLDERFIX*
14415 ;
14416 ; Der VLIR-Test funktioniert hier nicht
14417 ; zuverlässig. Ein Ordner hat als Tr/Se

```

```

; $00/$ff in Byte #1/2, da ein Ordner                                14418
; keine Daten beinhaltet.                                           14419
if TRUE                                                                14420
    jsr    Z10F0                ;Ordner/Verz./Datei?                14421
    txa                                     14422
    beq    :file                ; => Datei.                        14423
else                                                                    14424
    ldx    dirEntryBuf +1        ;Daten vorhanden?                14425
    beq    :no_file              ; => Kein Ordner.                  14426
    ;                                                                    14427
    lda    dirEntryBuf           ;Dateityp auswerten.              14428
    and    #FTYPE_MODES          14429
    cmp    #FTYPE_DIR            14430
    bne    :file                ; => Verzeichnis.                  14431
    ;                                                                    14432
;                               lda    dirEntryBuf +$16              14433
;                               cmp    #TDFOLDER                    ;TopDesk-Ordner?                14434
;                               bne    :file                        ; => Nein, weiter...              14435
;                               ;                                                                    14436
;                               lda    dirEntryBuf +21              14437
;                               cmp    #VLIR                        ;VLIR-Datei?                  14438
;                               beq    :file                        ; => Ja, weiter...              14439
endif                                                                    14440
; Ordner/Verzeichnis kopieren/verschieben.                          14441
::no_file                     stx    V10CF                ;Dateityp speichern.                14443
                             bpl    :no_dir                ; => Ordner...                          14444
                             ;                                                                    14445
                             ldx    WIN042F                ;Ziel-Fenster.                          14446
                             ldy    C04F8,x                ;Lfwk. für Fenster,                      14447
                             ;                                                                    14448
                             ldx    #STRUCT_MISMAT          14449
                             lda    RealDrvMode -8,y        14450
                             and    #SET_MODE_SUBDIR        14451
                             beq    :copyErr                ; => Kein Native...                14452
                             ;                                                                    14453
::no_dir                     lda    #0                    ;Datei auf Ziel                        14454
                             sta    r9L                    ;suchen/löschen.                      14455
                             ;                                                                    14456
                             ldx    #r11                    14457
                             jsr    Z0F30                ;Ziel-Datei suchen.                  14458
                             cpx    #DIR_NOT_EMPTY          ;Verz. n. gelöscht.                  14459
                             beq    :cont                    14460
                             cpx    #FILE_EXISTS            ;0./Verz. n. ersetzt.                14461
                             beq    :4                      14462
                             txa                            ;Fehler?                          14463
                             bne    :copyErr                ; => Ja, Abbruch...                14464
                             ;                                                                    14465
::4                           bit    V10CF                ;Ordner/Verzeichnis?                14466
                             bpl    :5                      ; => Ordner.                          14467
                             ;                                                                    14468
; Hinweis:                                                            14469
; Verzeichnis kopieren nicht aufrufen                                14470
; wenn Ziel nicht gelöscht wurde.                                    14471
                             txa                            ;File exists?                      14472

```

```

14473         bne    :cont          ; => Ja, weiter...
14474
14475         jmp     Z39C2          ;Verz. kopieren.
14476
14477 ; Hinweis:
14478 ; Wenn ein Ordner existiert, dann die
14479 ; Dateien in Ordner kopieren.
14480 ::5         jmp     Z3A37          ;Ordner kopieren.
14481
14482 ; Datei kopieren.
14483 ::file      jsr     S0A74          ;Datei kopieren.
14484           txa
14485           bne     :copyErr
14486
14487 ::cont      lda     #$00
14488           clc          ;Kopieren fortsetzen.
14489           rts
14490
14491 ; Fehler beim kopieren einer Datei.
14492 ::copyErr   cpx     #CANCEL_ERR
14493           beq     :cancel
14494           cpx     #FILE_EXISTS    ;Datei nicht ersetzt.
14495           beq     :cont
14496
14497 ; Hinweis:
14498 ; Fehler WR_PR_ON gilt für Disketten
14499 ; mit einem Schreibschutz.
14500 ; Fehler ERR_WR_PROT gilt für Dateien
14501 ; mit einem Schreibschutz.
14502 ;           cpx     #WR_PR_ON      ;Schreibschutz aktiv.
14503           cpx     #ERR_WR_PROT    ;Schreibschutz aktiv.
14504           beq     :cont
14505
14506           jsr     S5CE1          ;Fehler ohne Refresh.
14507
14508 ; Hinweis:
14509 ; Soll eine Datei ein Verzeichnis mit
14510 ; Dateien ersetzen und das Verzeichnis
14511 ; ist nicht leer, dann wird hier die
14512 ; Datei übersprungen.
14513           cpx     #DIR_NOT_EMPTY  ;Verz.nicht gelöscht.
14514           beq     :cont
14515
14516 ::cancel    jsr     S54F0          ;Hauptmenü starten.
14517
14518           lda     #$00          ;Abbruch und
14519           sec          ;Kopieren beenden.
14520           rts
14521
14522 ; Hinweis:
14523 ; Wird im Hauptmodul initialisiert und
14524 ; nur in mod.#09 verwendet. Muss aber
14525 ; im Hauptmodul verbleiben, da beim
14526 ; duplizieren zw. mod.#09 und mod.#02
14527 ; gewechselt wird.

```

.V5596	b \$00	;\$01=Copy SubDir-Hinweis angezeigt.	14528
.V71CF	b \$ff	;\$00=Umbenennen / \$FF=Duplizieren.	14529
.V71D0	b \$00	;\$01=Name eingegeben.	14530
.V7213	b \$00	;\$Anzahl Dateien im Fenster.	14531
			14532
; Vorgabewerte für r10-r12.			14533
:V55AE	w V0483 +2	;r10: Diskname1 ohne "x:".	14534
	w V0496 +2	;r11: Diskname2 ohne "x:".	14535
	w V0470	;r12: Dateiname.	14536
	w V0470	;r13: Dateiname.	14537
			14538
; Status anzeigen.			14539
; TD13: DispDCFName			14540
	ldx	#STAT_COPY ;Status "Kopiere".	14541
	b	\$2c	14542
;S55B6	ldx	#STAT_MOVE ;Status "Versch.".	14543
	b	\$2c	14544
;S55B5	ldx	#STAT_DEL ;Status "Lösche".	14545
	b	\$2c	14546
;S55B1	ldx	#STAT_VERIFY ;Status "Überprüfe".	14547
	b	\$2c	14548
;S55B2	ldx	#STAT_UNDEL ;Status "Rette".	14549
	b	\$2c	14550
;S55B3	ldx	#STAT_CREATE ;Status "Erstelle".	14551
.S55B4	stx	V5610 ;Status-Code.	14552
			14553
	lda	dispBufferOn	14554
	pha		14555
	lda	#ST_WR_FORE	14556
	sta	dispBufferOn	14557
			14558
	jsr	S55BE ;Status löschen.	14559
			14560
	jsr	i_PutString ;Textcursor setzen.	14561
	w	STATUS_X0 +2	14562
	b	STATUS_Y1 -6	14563
	b	PLAINTEXT ;Textstil löschen.	14564
	b	NULL	14565
			14566
	lda	V5610 ;Status-Code.	14567
	asl		14568
	tay		14569
	lda	V5612 +0,y	14570
	sta	r0L	14571
	lda	V5612 +1,y	14572
	sta	r0H	14573
	jsr	PutString	14574
			14575
	lda	V5611	14576
	asl		14577
	asl		14578
	pha		14579
	tay		14580
	lda	V5611 +1,y ;Zeiger einlesen.	14581
	sta	r0L	14582

```

14583         lda    V5611 +2,y
14584         sta    r0H
14585         jsr    PutString          ;Datei/Disk.
14586
14587 ;         lda    #BOLDON          ;TD-Font in BOLD
14588 ;         jsr    PutChar          ;nicht lesbar!
14589
14590         pla
14591         tay
14592         lda    V5611 +3,y          ;Zeiger einlesen.
14593         sta    r0L
14594         lda    V5611 +4,y
14595         sta    r0H
14596         jsr    PutString          ;Datei-/Diskname.
14597
14598 ;         lda    #PLAINTEXT       ;PLAINTEXT ist
14599 ;         jsr    PutChar          ;bereits aktiv.
14600
14601         pla
14602         sta    dispBufferOn
14603         rts
14604
14605 ; Statusbereich löschen.
14606 .S55BE      ldx    #6 -1
14607 ::1        lda    :coord,x
14608         sta    windowTop,x        ;Textgrenzen.
14609         sta    r2,x              ;Anzeigebereich.
14610         dex
14611         bpl    :1
14612
14613         lda    COL0445            ;Farbe Statusanzeige.
14614         jmp    Z6100             ;Grafik löschen.
14615
14616 ::coord     b STATUS_Y0,STATUS_Y1
14617         w STATUS_X0,STATUS_X1
14618
14619 ; Status-Code.
14620 :V5610      b $00
14621
14622 ; Texte für Status Datei/Disk.
14623 .V5611      b $00
14624         w T5644,V0470            ;Datei.
14625         w T09D4,V0483 +2        ;Disk.
14626
14627 ; Tabelle mit Statustexten.
14628 :V5612      w T561C ;Kopieren/mod.#08.
14629         w T562C ;Verschieben/mod.#00.
14630         w T5637 ;Löschen/mod.#00.
14631         w T532E ;Validate/mod.#01.
14632         w T563E ;Datei retten/mod.#01.
14633         w T7BEA ;DiskImage erstellen/mod.#08.
14634
14635 if LANG = LANG_DE
14636 :T561C      b "Kopiere"
14637 .V5623      b 0 ;0=Kopieren, $2F("/")=Kopiere/Ersetze.

```

```

        b "Ersetze",0                                14638
:T562C    b "Verschiebe"                            14639
:V5624    b 0 ;0=Versch., $2F("/")=Versch./Ersetze. 14640
        b "Ersetze",0                                14641
:T5637    b "Lösche",0                              14642
:T532E    b "Überprüfe",0                          14643
:T563E    b "Rette",0                              14644
:T7BEA    b "Erstelle",0                          14645
        b "Erstelle",0                              14646
:T5644    b " Datei: ",0                          14647
:T09D4    b " Disk: ",0                          14648
endif                                           14649
if LANG = LANG_EN                             14650
:T561C    b "Copy"                                14651
.V5623    b 0 ;0=Kopieren, $2F("/")=Copy/replace. 14652
        b "replace",0                              14653
:T562C    b "Move"                                14654
:V5624    b 0 ;0=Versch., $2F("/")=Move/replace. 14655
        b "replace",0                              14656
:T5637    b "Delete",0                            14657
:T532E    b "Validate",0                          14658
:T563E    b "Undelete",0                          14659
:T7BEA    b "Create",0                            14660
        b "Create",0                              14661
:T5644    b " file: ",0                          14662
:T09D4    b " disk: ",0                          14663
endif                                           14664
        b "disk",0                                14665
; Fenster platzieren.                        14666
; TD13: SetWindows                          14667
:MR5652    jsr    GotoFirstMenu                  14668
        b "GotoFirstMenu",0                      14669
:Z5655    lda    a0H                            14670
        pha                                14671
        lda    a0L                            14672
        pha                                14673
        b "a0L",0                              14674
        lda    #$ff                            14675
        sta    a2L                            14676
        b "a2L",0                              14677
        ldy    #4 -1                          14678
::1        lda    C0448,y                      ;Fenster geöffnet? 14679
        beq    :2                            ; => Nein, weiter... 14680
        inc    a2L                            14681
::2        dey                                14682
        bpl    :1                            14683
        b "a2L",0                              14684
        lda    a2L                            ;Fenster offen? 14685
        bpl    :3                            ; => Ja, weiter... 14686
        b "a2L",0                              14687
        pla                                14688
        sta    a0L                            14689
        pla                                14690
        sta    a0H                            14691
        rts                                14692

```

```

14693
14694 ::3          asl
14695             tay
14696             lda    :windata +0,y
14697             sta    a1L
14698             lda    :windata +1,y
14699             sta    a1H
14700
14701             ldy    #4 -1
14702 ::4          tya
14703             pha                    ;Fens.-Nr. speichern.
14704
14705             lda    C042A,y          ;Fenster-Nr. holen.
14706             tax
14707             lda    C0448,x          ;Fenster geöffnet?
14708             beq    :6              ; => Nein, weiter...
14709
14710             txa
14711             jsr    S11A9            ;a0=Fensterdaten.
14712
14713             ldy    #6 -1
14714 ::5          lda    (a1),y
14715             sta    (a0),y
14716             dey
14717             bpl    :5
14718
14719             lda    a1L
14720             clc
14721             adc    #6
14722             sta    a1L
14723             bcc    :6
14724             inc    a1H
14725
14726 ::6          pla                    ;Fens.-Nr. einlesen.
14727             tay
14728             dey                    ;Noch ein Fenster?
14729             bpl    :4              ; => Ja, weiter...
14730
14731             pla
14732             sta    a0L
14733             pla
14734             sta    a0H
14735
14736             jmp    S157E            ;Alle Fens. anzeigen.
14737
14738 ; Tabellenzeiger auf Fensterdaten.
14739 ; TD13: Gentab
14740 ::windata    w :win1
14741             w :win2
14742             w :win3
14743             w :win4
14744
14745 ; Fensterpositionen.
14746 ::win1       b DESK_Y0 + $00,DESK_Y1 ;Full
14747             w DESK_X0      ,DESK_X1

```



```

::win2      b DESK_Y0 + $60,DESK_Y1      ;Bottom      14748
            w DESK_X0      ,DESK_X1      14749
            b DESK_Y0 + $00,DESK_Y1 -$60 ;Top          14750
            w DESK_X0      ,DESK_X1      14751
::win3      b DESK_Y0 + $00,DESK_Y1 -$60 ;Right        14752
            w DESK_X0 +$0090,DESK_X1      14753
            b DESK_Y0 + $00,DESK_Y1 -$60 ;Left         14754
            w DESK_X0      ,DESK_X1 -$88  14755
            b DESK_Y0 + $60,DESK_Y1      ;Bottom       14756
            w DESK_X0      ,DESK_X1      14757
::win4      b DESK_Y0 + $00,DESK_Y1 -$60 ;Top/left     14758
            w DESK_X0      ,DESK_X1 -$88  14759
            b DESK_Y0 + $00,DESK_Y1 -$60 ;Top/right    14760
            w DESK_X0 +$0090,DESK_X1      14761
            b DESK_Y0 + $60,DESK_Y1      ;Bottom/left   14762
            w DESK_X0      ,DESK_X1 -$88  14763
            b DESK_Y0 + $60,DESK_Y1      ;Bottom/right  14764
            w DESK_X0 +$0090,DESK_X1      14765
            14766
; Position von Dateien tauschen.      14767
; TD13: SwapFile                      14768
:Z570E      jsr S5C89                  ;Hole Disk/Fenster. 14769
            tax                      14770
            beq :1                    14771
            14772
            jsr S5245                  ;Dateien invertieren. 14773
            cpx #CANCEL_ERR           14774
            beq :exit                 14775
            14776
            jsr S5CF3                  ;Fehler ausgeben.   14777
            14778
::exit      rts                      14779
            14780
::1         lda a1L                   14781
            jsr Z42EC                  ;Dateiname einlesen. 14782
            jsr S1412                  ;Datei suchen.      14783
            14784
            lda r1H                    ;Zeiger auf         14785
            pha                        ;Verzeichniseintrag  14786
            lda r1L                    ;zwischenspeichern.  14787
            pha                        14788
            lda r5L                    14789
            pha                        14790
            14791
            ldy #0                     14792
::2         lda diskBlkBuf,y           14793
            sta fileTrScTab,y         14794
            iny                       14795
            bne :2                     14796
            14797
            jsr S5262                  ;Suche mark.Datei.   14798
            jsr S46EB                  ;Name mark.Datei.   14799
            jsr S1412                  ;Datei suchen.      14800
            14801
            pla                        ;Zeiger auf         14802

```

```

14803          sta    r6L          ;Verzeichniseintrag
14804          pla          ;einlesen.
14805          sta    a7L
14806          pla
14807          sta    a7H
14808
14809          lda    r1H          ;Sektor Datei#1 =
14810          cmp    a7H          ;Sektor Datei#2?
14811          bne    :3
14812          lda    r1L
14813          cmp    a7L
14814  :::3        bne    :4          ; => Nein, weiter...
14815
14816          lda    #> diskBlkBuf
14817          b $2c
14818  :::4        lda    #> fileTrScTab
14819          sta    r6H
14820
14821          ldy    #30 -1          ;Eintrag tauschen.
14822  :::5        lda    (r5),y
14823          pha
14824          lda    (r6),y
14825          sta    (r5),y
14826          pla
14827          sta    (r6),y
14828          dey
14829          bpl    :5
14830
14831  ; Sektor #2 speichern.
14832  ;          lda    #< diskBlkBuf          ;Zeiger auf
14833  ;          sta    r4L          ;Zwischenspeicher.
14834  ;          lda    #> diskBlkBuf
14835  ;          sta    r4H
14836          jsr    PutBlock_dskBuf
14837
14838          lda    r1H
14839          cmp    a7H
14840          bne    :6
14841          lda    r1L
14842          cmp    a7L
14843  :::6        beq    x579C
14844
14845  ; Sektor #1 speichern.
14846          lda    a7L
14847          sta    r1L
14848          lda    a7H
14849          sta    r1H
14850
14851          jsr    S09C4          ;fileTrScTab > r4.
14852          jsr    PutBlock
14853
14854  ; Neuladen von Dateien und Icons.
14855  ; TD13: ReloadActiveWindow
14856  :x579C      ldx    C042A          ;Oberstes Fenster.
14857          jsr    S599A          ;Zeiger auf DIRBUF.

```

```

14858
14859      lda    r1H                ;Zeiger auf DIRBUF
14860      pha                    ;zwischenspeichern.
14861      lda    r1L
14862      pha
14863
14864      ;      lda    #< SIZE_DIRBUF    ;Speicher für
14865      ;      sta    r0L                ;Dateinamen und
14866      ;      lda    #> SIZE_DIRBUF    ;Icons löschen.
14867      ;      sta    r0H
14868      ;      jsr    ClearRam
14869      ;      jsr    Z4570                ;Speicher löschen.
14870
14871      ; Hinweis:
14872      ; Hier wird das Laufwerk geöffnet inkl.
14873      ; OpenDisk und ClearRam.
14874      ; Dabei wird dann auch r0 zerstört und
14875      ; muss hier nicht eingelesen werden.
14876      if FALSE
14877          pla                    ;Adresse DIRBUF.
14878          sta    r0L
14879          pla
14880          sta    r0H
14881
14882          lda    r0H
14883          pha
14884          lda    r0L
14885          pha
14886      endif
14887
14888      ldx    C042A                ;Oberstes Fenster.
14889      jsr    Z4587                ;Dateien einlesen.
14890
14891      pla                    ;Adresse DIRBUF.
14892      sta    r0L
14893      pla
14894      sta    r0H
14895
14896      inc    V18AD                ;Fens.-Aufbau-Status.
14897
14898      jsr    A2BA6                ;Anzeige Icons/Text
14899      txa                    ;initialisieren und
14900      beq    :1                ;Inhalt anzeigen.
14901
14902      jsr    Z5CEB                ;Fehler ausgeben und
14903      ;Fenster schließen.
14904      ::1      rts
14905
14906      ; Prüfen ob Datei markiert ist.
14907      ; Übergabe: A = Nummer der Dat. in Darstellung (0-15)
14908      ;      X = Fenster-Nr.
14909      ; Rückgabe: C = SEC, nicht markiert
14910      ;      CLC, Datei markiert
14911      ;      A = Datei-Nr. (0-254)
14912      ; TD13: CheckDispMark

```

```

14913 :Z57D8          pha
14914
14915          ldy    C04B2          ;Icon-/Text-Modus.
14916          beq    :1            ; => Icon-Modus.
14917          dey
14918          beq    :1            ; => Text-Modus.
14919
14920 ; Sortierte Anzeige.
14921          pla
14922          stx    V57EE
14923
14924          jsr    A2BBE          ;mark.Datei suchen.
14925
14926          ldx    V57EE
14927          jmp    Z57F4          ;Datei markiert?
14928
14929 ::1          pla            ;Datei-Nr. (0-15)
14930          clc
14931          adc    C050B,x        ;Fensterposition.
14932
14933 ; Prüfen ob Datei markiert ist.
14934 ; Übergabe: A = Nummer der Dat. im Fenster (0-254)
14935 ; Rückgabe: C = SEC, nicht markiert
14936 ;          CLC, Datei markiert
14937 ; TD13: CheckMark
14938 :Z57F4          ldy    #MAXFILES
14939 ::1          cmp    V634F -1,y ;Tab. mark.Dateien.
14940          beq    :2
14941          dey
14942          bne    :1
14943
14944          sec
14945          rts
14946
14947 ::2          clc
14948          rts
14949
14950 :V57EE          b $00 ;Zwischenspeicher Fenster-Nr.
14951
14952 ; Anzahl markierte Dateien ausgeben.
14953 ; Hinweis: X bleibt unverändert!
14954 ; TD13: DispMultiCount
14955 .S5802          txa            ;Evtl. Fehlerstatus.
14956          pha
14957
14958          ldy    #6 -1
14959 ::1          lda    windowTop,y
14960          pha
14961          dey
14962          bpl    :1
14963
14964          jsr    Z13D3          ;Textgrenzen Desktop.
14965
14966          lda    #0
14967          jsr    SetPattern

```

	jsr i_Rectangle	14968
	b SLCT_Y0 +1,SLCT_Y1 -1	14969
	w SLCT_X0 +1,SLCT_X1 -1	14970
		14971
	lda V04B7 ;Anzahl mark.Dateien.	14972
	sta r0L	14973
	lda #\$00 ;Highbyte löschen.	14974
	sta r0H	14975
		14976
	lda #< SLCT_X0 +2	14977
	sta r11L	14978
	lda #> SLCT_X0 +2	14979
	sta r11H	14980
		14981
	lda #SLCT_Y1 -6	14982
	sta r1H	14983
		14984
	lda #SET_LEFTJUST ! SET_SUPRESS	14985
	jsr PutDecimal	14986
		14987
	ldy #0	14988
::2	pla	14989
	sta windowTop,y	14990
	iny	14991
	cpy #6	14992
	bne :2	14993
		14994
	pla	14995
	tax ;Evtl. Fehlerstatus.	14996
		14997
	rts	14998
		14999
	; Markierte Dateien im Fenster anzeigen.	15000
	; Übergabe: WIN042F = Fenster-Nr.	15001
	; TD13: DispMarking	15002
.S5842	jsr Z4214 ;X/Y für erstes Icon.	15003
		15004
	lda #\$00	15005
::1	pha	15006
		15007
	ldx WIN042F ;Aktives Fenster.	15008
	jsr Z57D8 ;Datei markiert?	15009
	bcs :2 ; => Nein...	15010
		15011
	ldx WIN042F ;Aktives Fenster.	15012
	pla	15013
	cmp V050F,x ;Anz.Dat. im Fenster.	15014
	bcs :3	15015
		15016
	pha	15017
	jsr A2BB2 ;Rechteck f.Eintrag.	15018
	bcs :2 ; => Kein Eintrag...	15019
		15020
	jsr InvertRectangle	15021
		15022

```

15023 ::2          pla
15024
15025 ::3          clc
15026          adc    #1
15027          cmp    #MAXWINFILES      ;Max.Anz.Dat./Fens.
15028          bne    :1
15029          rts
15030
15031 ; Bereich für die Inhaltsausgabe definieren.
15032 ; TD13: MyDCFileSub
15033 .S5875          ldx    WIN042F      ;Aktives Fenster.
15034          jsr    S1E09      ;Arbeitsber.Fenster.
15035
15036          lda    r3L      ;Linken Rand
15037          sta    a5L      ;zwischenspeichern.
15038          lda    r3H
15039          sta    a5H
15040
15041          lda    r2L
15042          sec
15043          sbc    #ICON_SKIP_Y
15044          sta    r2L
15045
15046          lda    #< ICON_SKIP_X
15047          clc
15048          adc    r3L
15049          sta    r3L
15050          lda    #> ICON_SKIP_X
15051          adc    r3H
15052          sta    r3H
15053          rts
15054
15055 ; Position des ersten Icon links/oben.
15056 :Z4214          jsr    S5875      ;Ber. Fens.-Inhalt.
15057
15058          lda    r2L
15059          sta    a2L
15060
15061          lda    a5L
15062          sta    a3L
15063          lda    a5H
15064          sta    a3H
15065          rts
15066
15067 ; Alle Dateien wählen.
15068 ; TD13: SelectAll
15069 :MR589D          lda    #1      ;Alle Dateien wählen.
15070          b $2c
15071
15072 ; Dateien auf Seite wählen.
15073 ; TD13: SelectPage
15074 :MR58A0          lda    #0      ;Seite wählen.
15075
15076          dec    V04AE      ;Klick auf Gadget?
15077          beq    :1      ; => Ja, weiter...

```

			15078
	pha		15079
	jsr	GotoFirstMenu ;Hauptmenü.	15080
	pla	;Seite/Alle wählen.	15081
			15082
	inc	V04AE ;Klick zurücksetzen.	15083
			15084
::1	b	\$2c	15085
			15086
		; Erste Zeile wählen.	15087
		; TD13: n.v.	15088
:KB58A3	lda	#2 ;Erste Zeile wählen.	15089
	sta	V5909 ;Markierungsmodus.	15090
			15091
	jsr	A2BB5 ;Anz.Ze/Sp und Icons.	15092
			15093
	ldx	C042A ;Oberstes Fenster.	15094
	lda	C0448,x ;Fenster geöffnet?	15095
	bne	:1 ; => Ja, weiter...	15096
	rts		15097
			15098
::1	stx	WIN042F ;Aktives Fenster.	15099
	jsr	Z4214 ;X/Y für erstes Icon.	15100
	jsr	S5842 ;mark.Dat. anzeigen.	15101
			15102
	lda	#0 ;Dateizähler.	15103
::loop	pha		15104
			15105
	ldx	C042A ;Oberstes Fenster.	15106
			15107
	ldy	V5909 ;Markierungsmodus.	15108
	beq	:2 ; => Seite wählen.	15109
	cpy	#2	15110
	beq	:3 ; => Abwählen.	15111
			15112
		; Alle Dateien wählen.	15113
	cmp	V0513,x ;Anz.Dat. auf Disk.	15114
	bcs	:next	15115
			15116
	jsr	Z57F4 ;Datei markiert?	15117
	bcc	:next ; => Ja...	15118
	bcs	:5 ; => Nein...	15119
			15120
		; Dateien auf Seite wählen.	15121
::2	cmp	V050F,x ;Anz.Dat. im Fenster.	15122
	bcs	:next ; => Überspringen.	15123
	bcc	:4 ; => Markieren.	15124
			15125
		; Erste Zeile wählen.	15126
::3	cmp	V4CC7,x ;Anz.Dat. in Zeile.	15127
	bcs	:next ; => Überspringen.	15128
			15129
::4	jsr	Z57D8 ;Datei markiert?	15130
	bcc	:next ; => Ja, weiter...	15131
			15132

```

15133 ; Datei markieren.
15134 ::5          tax
15135          jsr   Z5216          ;mark.Datei in Tab.
15136
15137 ::next       pla
15138          clc
15139          adc   #1          ;Nächste Datei.
15140
15141          ldy   V5909        ;Markierungsmodus.
15142          beq   :6          ; => Seite/Zeile.
15143
15144          cmp   #MAXFILES    ;Alle Dat. getestet?
15145          bne   :loop        ; => Nein, weiter...
15146          beq   :done        ; => Ja, Ende...
15147
15148 ::6          ldx   C042A      ;Oberstes Fenster.
15149          cmp   V4CCB,x      ;Anz.Dat. im Fenster.
15150          bne   :loop        ; => Weiter...
15151
15152 ::done       jmp   S5842      ;mark.Dat. anzeigen.
15153
15154 :V5909       b $00 ;Markierungsmodus.
15155
15156 ; Zahl links-/rechtsbündig ausgeben.
15157 ; Entspricht PutDecimal, aber "einfacher".
15158 ; Übergabe: A = Modus:
15159 ;           Bit%7=1 linksbündig
15160 ;           Bit%7=0 rechtsbündig
15161 ;           Bit%6  Nicht verwendet
15162 ;           Bit%0-5 Breite des Textfeldes
15163 ;           r0 = Zahl
15164 ; TD13: NewPutDecimal
15165 .S590A      tax
15166          lda   r1H
15167          pha
15168
15169          txa
15170          pha
15171
15172          lda   #NULL        ;String-Ende.
15173          pha
15174
15175          lda   #< 10
15176          sta   r1L
15177          lda   #> 10
15178          sta   r1H
15179
15180 ::1          ldx   #r0
15181          ldy   #r1
15182          jsr   Ddiv
15183
15184          lda   r8L
15185          clc
15186          adc   #"0"
15187          pha

```



			15188
	lda	r0L	15189
	ora	r0H	15190
	bne	:1	15191
			15192
	ldy	#\$ff	15193
::2	iny		15194
	pla		15195
	sta	V5973,y	15196
	bne	:2	15197
			15198
	pla		15199
	bpl	:3	15200
			15201
	lda	#< V5973	15202
	sta	r0L	15203
	lda	#> V5973	15204
	sta	r0H	15205
	bne	:5	15206
			15207
::3	pha		15208
			15209
	lda	#< V5973	15210
	sta	r0L	15211
	lda	#> V5973	15212
	sta	r0H	15213
	jsr	S244C ;Hole Stringlänge.	15214
			15215
	pla		15216
	and	##00111111	15217
	clc		15218
	adc	r11L	15219
	sta	r11L	15220
	bcc	:4	15221
	inc	r11H	15222
			15223
::4	lda	r11L	15224
	sec		15225
	sbc	r1L	15226
	sta	r11L	15227
	lda	r11H	15228
	sbc	r1H	15229
	sta	r11H	15230
			15231
::5	pla		15232
	sta	r1H	15233
			15234
	jmp	S14C8 ;String ausgeben.	15235
			15236
:V5973	s	6 ;Zwischensp. Zahl im Textformat.	15237
			15238
	; Ermittlung des Fenstertitels.		15239
	; Übergabe: A = Fenster-Nr.		15240
	; Rückgabe: r1 = Zeiger auf String		15241
	; Y = Zeiger auf letztes Zeichen.		15242

```

15243 ; TD13: GetWinName
15244 .S5979          asl
15245                  tay
15246                  lda    V053B +0,y
15247                  sta    r1L
15248                  lda    V053B +1,y
15249                  sta    r1H
15250
15251                  ldy    #$00
15252 ::1              lda    (r1),y
15253                  beq    :2
15254                  iny
15255                  bne    :1
15256 ::2              rts
15257
15258 ; Ermitteln der SubDirXList.
15259 ; Übergabe: X = Fenster-Nr.
15260 ; Rückgabe: r0 = Adresse SubDirXList
15261 ; TD13: GetSubDirXList
15262 .S598F          lda    V0683,x
15263                  sta    r0L
15264                  lda    V0687,x
15265                  sta    r0H
15266                  rts
15267
15268 ; Zeiger auf DIRBUF1/DIRBUF2 setzen.
15269 ; Übergabe: X = Fenster-Nr.
15270 ; Wenn REU vorhanden:
15271 ; Daten für Fenster#1/#2 oder #3/#4
15272 ; aus REU einlesen.
15273 ; TD13: GetWinTabAdr
15274 .S599A          txa
15275                  pha
15276
15277                  ldy    C0429          ;System in REU?
15278                  beq    :4              ; => Nein, weiter...
15279
15280                  cmp    V59EF          ;Fenster in DIRBUF1?
15281                  beq    :2              ; => Ja, weiter...
15282                  cmp    V59F0          ;Fenster in DIRBUF2?
15283                  beq    :2              ; => Ja, weiter...
15284
15285                  ldx    #(r2H - r0L)    ;Daten für DIRBUF
15286 ::1              lda    V59E8,x        ;in REU setzen.
15287                  sta    r0,x
15288                  dex
15289                  bpl    :1
15290
15291                  lda    C0427          ;64K-Bank RAMDesk.
15292                  sta    r3L
15293
15294                  jsr    SwapRAM        ;DIRBUF/REU tauschen.
15295
15296 ::2              pla
15297                  pha

```

			15298
	sec	;Fenster-Nr -2 < 0?	15299
	sbc #2		15300
	bmi :3		15301
			15302
	ldy #2	;DIRBUF = Fens. 2/3.	15303
	b \$2c		15304
::3	ldy #0	;DIRBUF = Fens. 0/1.	15305
	sty V59EF	;Fenster 0/2.	15306
	iny		15307
	sty V59F0	;Fenster 1/3.	15308
			15309
::4	pla		15310
	tax		15311
			15312
	asl	;Zeiger auf DIRBUF1/	15313
	tay	;DIRBUF2 setzen.	15314
	lda V04E8,y		15315
	sta r1L		15316
	iny		15317
	lda V04E8,y		15318
	sta r1H		15319
			15320
	txa		15321
	rts		15322
			15323
:V59E8	w DIRBUF1		15324
	w DACC_DIRBUF		15325
	w SIZE_DIRBUF *2		15326
			15327
	; Hinweis:		15328
	; Ohne REU geht nur 0/1, mit REU wird		15329
	; DIRBUF mit der REU getauscht und es		15330
	; ist 0/1 oder 2/3.		15331
:V59EF	b \$00	;Vorgabe: Fenster-Nr. #0 im RAM.	15332
:V59F0	b \$01	;Vorgabe: Fenster-Nr. #1 im RAM.	15333
			15334
	; Diskname für aktuelles Fenster ermitteln.		15335
	; TD13: GetAktlWinDisk		15336
.S59F1	ldx C042A	;Oberstes Fenster.	15337
			15338
	; Diskname aus Pfad für Fenster ermitteln.		15339
	; Übergabe: X = Fenster-Nr.		15340
	; Rückgabe: V0483 = Diskname.		15341
	; r2L/A = 1, Quelle/Ziel unterschiedlich.		15342
	; TD13: GetWinDisk		15343
:Z59F4	txa	;Fenster-Nr.	15344
	asl		15345
	tay		15346
	lda V053B +0,y	;Zeiger auf Diskname.	15347
	sta r0L		15348
	lda V053B +1,y		15349
	sta r0H		15350
			15351
	lda C042A	;Oberstes Fenster.	15352

```

15353          asl
15354          tay
15355          lda    V053B +0,y          ;Zeiger auf Diskname.
15356          sta    r1L
15357          lda    V053B +1,y
15358          sta    r1H
15359
15360          ldy    #0                  ;Vorgabe:
15361          sty    r2L                  ;Quelldisk=Zieldisk.
15362
15363 ::1          lda    (r0),y
15364          cmp    (r1),y
15365          beq    :2
15366          cpy    #18                  ;"x:" + Diskname
15367          beq    :2
15368
15369          ldx    #1                  ;Quelldisk<>Zieldisk.
15370          stx    r2L
15371
15372 ::2          tax
15373          beq    :3
15374          cmp    #TDSPLIT              ;Trenner für Ordner?
15375          beq    :3                  ; => Ja, weiter...
15376          cmp    #"/" ! $80            ;Ende Diskname?
15377          beq    :3                  ; => Ja, weiter...
15378
15379          sta    V0483,y
15380          iny
15381          bne    :1
15382
15383 ::3          lda    #NULL
15384          sta    V0483,y              ;Ende Diskname.
15385
15386          lda    r2L                  ;Status Quelle/Ziel.
15387          rts
15388
15389 ; Ermittlung der Fensternummer, deren Pfadname genau
15390 ; mit dem Pfad eines bestimmten Fensters übereinstimmt.
15391 ; Übergabe: X      = Fenster-Nr.
15392 ; Rückgabe: a6/a7 = 4 Byte, $01=gleich, $00=ungleich.
15393 ; TD13: GetEqualWindows
15394 .S5A3B      ldy    #(a5H - a2L)
15395 ::1          lda    V053B,y          ;Zeiger auf Diskname
15396          sta    a2,y                ;nach a2-a5 kopieren.
15397          dey
15398          bpl    :1
15399
15400          txa                        ;a2 <> a2,x
15401          pha                        ;Fens.-Nr. speichern.
15402
15403          asl
15404          tay
15405
15406 ; Hinweis:
15407 ; Doppelter TXA-Befehl entfernt.

```

```

;          txa          ;Fens.-Nr. speichern. 15408
;          pha          15409
;          15410
;          lda  a2L,y    15411
;          ldx  a2L      15412
;          stx  a2L,y    15413
;          sta  a2L      15414
;          15415
;          lda  a2H,y    15416
;          ldx  a2H      15417
;          stx  a2H,y    15418
;          sta  a2H      15419
;          15420
; Zeiger auf Source: in a2 15421
; a3 = Fenster #2, wenn Source <> 2 sonst Fenster #1. 15422
; a4 = Fenster #3, wenn Source <> 3 sonst Fenster #1. 15423
; a5 = Fenster #4, wenn Source <> 4 sonst Fenster #1. 15424
;          lda  #1      15425
;          sta  a6H      15426
;          sta  a7L      15427
;          sta  a7H      15428
;          15429
;          ldx  #0      15430
;          ldy  #$00     15431
::2      lda  (a2),y    ;Source-Zeichen. 15432
;          cmp  (a3),y    ;Vergl. mit Fens.#1. 15433
;          beq  :3        15434
;          stx  a6H      ;Fens.#1 ungleich. 15435
;          15436
::3      cmp  (a4),y    ;Vergl. mit Fens.#2. 15437
;          beq  :4        15438
;          stx  a7L      ;Fens.#2 ungleich. 15439
;          15440
::4      cmp  (a5),y    ;Vergl. mit Fens.#3. 15441
;          beq  :5        15442
;          stx  a7H      ;Fens.#3 ungleich. 15443
;          15444
::5      cmp  #NULL     15445
;          beq  :6        15446
;          iny          15447
;          bne  :2        15448
;          15449
::6      pla          15450
;          tax          15451
;          lda  a6,x      ;Ordnung in a6-a7 15452
;          sta  a6L      ;wieder herstellen. 15453
;          ldy  #0        15454
;          sty  a6,x      15455
;          15456
; Jetzt stehen in a6-a7 4 Bytes der jeweiligen Fenster. 15457
; $01 : Fenster = Source, Source selbst ist $00. 15458
;          ldy  #4 -1     15459
::7      lda  C0448,y    ;Fenster geöffnet? 15460
;          bne  :8        ; => Ja, weiter... 15461
;          sta  a6,y      15462

```

```

15463 ::8          dey
15464          bpl      :7
15465
15466          rts
15467
15468 ;*** Icon-Routine: Trash-Icon.
15469 ;IR5A9C          lda    V0423          ;DnD aktiv?
15470          bne    MR5AB6          ; => Ja, weiter...
15471
15472          jsr    S524C          ;Dat.-Ausw. aufheben.
15473
15474          ldx    C042A          ;Oberstes Fenster.
15475          lda    C0448,x        ;Fenster geöffnet?
15476          bne    :1          ; => Ja, weiter...
15477          rts
15478
15479 ::1          lda    C04F8,x        ;Lfwk. für Fenster.
15480          jsr    S06AB          ;Laufwerk öffnen.
15481          jmp    S3997          ;Dateien retten.
15482
15483 ;*** Menü-Routine: Trash-Icon.
15484 ; TD13: TrashService/DeskDelete
15485 ;
15486 ; Übergabe: V04B7 = Anzahl Dateien.
15487 ;
15488 ; Hinweis:
15489 ; Bei V04B7=255 sind alle 255 Dateien
15490 ; im Fenster markiert. Dadurch versagt
15491 ; aber die BubbleSort-Routine, da diese
15492 ; den Wert $ff als Ende-Kennung in der
15493 ; Tabelle der mark.Dateien erwartet.
15494 ; Der "Workaround" funktioniert aber
15495 ; nur beim löschen über das GEOS-Menü.
15496 ; Am Ende der Tabelle wurde jetzt ein
15497 ; $ff-Byte ergänzt, damit kann hier der
15498 ; "Workaround" entfallen.
15499 :DISBALE_DEL255 = TRUE
15500 :MR5AB6          jsr    GotoFirstMenu      ;Datei/löschen.
15501          jsr    S52AD          ;Maustaste loslassen.
15502
15503          ldx    V04B7          ;Anzahl mark.Dateien.
15504          bne    :init
15505          rts
15506
15507 if DISBALE_DEL255 = FALSE
15508 ;
15509 ; Hinweis:
15510 ; Nach dem löschen der ersten Datei
15511 ; mit dem Rest weitermachen.
15512 ::loop          lda    #$00          ;Status: Beenden.
15513          pha
15514          beq    :2
15515 endif
15516
15517 ::init          lda    #$00

```

```

sta    V5C5A                ;Kein Fehler.                15518
                                                    15519
if DISBALE_DEL255 = FALSE 15520
;
; Hinweis: 15521
; Bei 255 markierten Dateien ist kein 15522
; Ende-Flag mehr für BubbleSort in der 15523
; Tabelle: Workaround: 15524
; Die erste Datei löschen, danach das 15525
; Löschen von Dateien fortsetzen. 15526
; Der Code funktioniert aber nicht bei 15527
; löschen von Dateien über DnD. 15528
; 15529
; lda    #$00                ;Status: Beenden. 15530
; cpx    #MAXFILES           ;255 Dateien? 15531
; bne    :1                  ; => Nein, weiter... 15532
;                               15533
; lda    #$01                ;Status: Fortsetzen. 15534
::1    pha                  15535
; cpx    #MAXFILES           ;255 Dateien? 15536
; bne    :sort               ; => Nein, weiter... 15537
;                               15538
::2    lda    #0              ;Erste Datei. 15539
; jsr    Z449B               ;Datei markieren. 15540
endif 15541
;                               15542
::sort jsr    Z5C50           ;BubbleSort. 15543
;                               15544
::del  jsr    Z525E           ;Mark.Datei suchen. 15545
; bcs    :done               ; => Ende... 15546
;                               15547
; jsr    Z5B35               ;Datei löschen. 15548
; txa                      15549
; bne    :err                ; => Fehler... 15550
; bcc    :del                ;Nächste Datei... 15551
;                               15552
::err  jsr    S5CF3           ;Fehler ausgeben. 15553
;                               15554
::done ldx    C042A           ;Oberstes Fenster. 15555
; lda    C050B,x             ;Fensterposition. 15556
; pha                      15557
; lda    #$00                15558
; sta    C050B,x             ;Fensterposition. 15559
;                               15560
; jsr    Z4576               ;Dat. neu einlesen. 15561
;                               15562
; ldx    C042A               ;Oberstes Fenster. 15563
; pla                      15564
; sta    C050B,x             ;Fensterposition. 15565
;                               15566
; lda    V0513,x             ;Anz.Dat. auf Disk. 15567
; cmp    C050B,x             ;Fensterposition. 15568
; beq    :3                  ; => Ungültig... 15569
; bcs    :5                  ; => Gültig... 15570
;                               15571
::3    jsr    A2BB5           ;Anz.Ze/Sp und Icons. 15572

```

```

15573
15574             ldx    C042A                ;Oberstes Fenster.
15575             lda    V0513,x              ;Anz.Dat. auf Disk.
15576             cmp    V4CCB,x              ;Anz.Dat. auf Seite.
15577             bcc     :4
15578             sec
15579             sbc     V4CCB,x
15580
15581             b $2c
15582 ::4          lda    #$00
15583             sta     C050B,x              ;Fensterposition.
15584
15585 ::5
15586
15587 if DISBALE_DEL255 = FALSE
15588 ;
15589 ; Hinweis:
15590 ; Nach dem löschen der ersten Datei
15591 ; mit dem Rest weitermachen.
15592             pla
15593             bne     :loop
15594 endif
15595
15596             jsr     S4296                ;Update akt.Fenster.
15597             jmp     S54E3                ;Hauptmenü.
15598
15599 ; Eintrag löschen.
15600 ; Übergabe: A = Datei-Nr. (0-254)
15601 ; Rückgabe: C = SEC, nicht mehr vorhanden oder Diskfehler.
15602 ; TD13: DeleteSub
15603 :Z5B35       jsr     S46EB                ;Name mark.Datei.
15604             bcs     :exit                ; => Keine Dateien...
15605
15606             jsr     S5B38                ;Datei löschen.
15607             bcs     :err                 ; => Nicht gefunden.
15608
15609 ; Fehlercode zwischenspeichern:
15610 ; Wird beim ersetzen der Ziel-Datei
15611 ; verwendet um den Vorgang abubrechen
15612 ; wenn Eintrag nicht gelöscht wurde.
15613 ; Ordner: FILE_EXIST = Nicht gelöscht.
15614 ; Verz. : DIR_NOT_EMPTY = Nicht gelöscht.
15615             txa
15616             beq     :ok
15617
15618 ; Hinweis:
15619 ; Fehler WR_PR_ON gilt für Disketten
15620 ; mit einem Schreibschutz.
15621 ; Fehler ERR_WR_PROT gilt für Dateien
15622 ; mit einem Schreibschutz.
15623 ;             cpx     #WR_PR_ON            ;Schreibschutz aktiv.
15624             cpx     #ERR_WR_PROT         ;Datei/Schreibsch.?
15625             beq     :skip
15626
15627             sta     V5C5A

```



```

15628
15629      b $2c
15630
15631 ; Hinweis:
15632 ; Beim löschen eines Ordners Zeiger
15633 ; auf nächste Datei setzen, da die
15634 ; aktuelle Datei nicht gelöscht wurde.
15635 ::skip      ldx    #NO_ERROR
15636
15637      inc    V5C5B      ;Zeiger nächste Dat.
15638
15639 ; Bei Ordner/Verzeichnis löschen:
15640 ; Löschen fortsetzen, auch wenn der
15641 ; Eintrag nicht gelöscht wurde.
15642 ::ok      clc      ;Datei gelöscht.
15643      rts
15644
15645 ; Fehler.
15646 ::err      stx    V5C5A      ;Fehler speichern.
15647      sec      ;Nicht gelöscht.
15648 ::exit     rts
15649
15650 ; Eintrag löschen.
15651 ; Übergabe: V0470 = Dateiname.
15652 ; Rückgabe: X = Fehler.
15653 ;      C = CLC, Datei gelöscht.
15654 ;      C = SEC, nicht mehr vorhanden oder Diskfehler.
15655 .S5B38      lda    #< V0470
15656      ldx    #> V0470
15657 :Z5B3A      jsr    S1414      ;Datei suchen.
15658      txa
15659      bne    :err
15660
15661      ldx    #STAT_DEL      ;Status "Lösche".
15662      jsr    S55B4      ;Status anzeigen.
15663
15664      lda    dirEntryBuf
15665      and    #ST_WR_PR      ;Schreibschutz?
15666      beq    :test      ; => Nein, weiter...
15667
15668      lda    #< D5BF2      ;Datei mit
15669      ldx    #> D5BF2      ;Schreibschutz.
15670      jsr    S2080      ;DBox, k. Maustaste.
15671
15672 ; Hinweis:
15673 ; Fehler WR_PR_ON gilt für Disketten
15674 ; mit einem Schreibschutz.
15675 ; Fehler ERR_WR_PROT gilt für Dateien
15676 ; mit einem Schreibschutz.
15677 ;      ldx    #WR_PR_ON      ;Schreibschutz aktiv.
15678 ;      ldx    #ERR_WR_PROT      ;Schreibschutz aktiv.
15679 ;      lda    r0L
15680 ;      cmp    #NO      ;Datei löschen?
15681 ;      beq    :exit      ; => Nein, Ende...
15682

```

```

15683 ::test          jsr    Z10F0          ;Ordner/Verz./Datei?
15684                txa
15685                beq    :del            ; => Datei.
15686                bpl    :folder         ; => Ordner.
15687
15688 ; mod.#09: Verzeichnis löschen.
15689 ::dir            jsr    Z39BC          ;Verzeichnis löschen.
15690                txa
15691                beq    :exit            ; => OK, Ende...
15692
15693 ::err            sec
15694                rts                    ;Fehler.
15695
15696 ; Ordner-Inhalt löschen
15697 ::folder          jsr    S09C7          ;Infoblock einlesen.
15698
15699                ldy    #16 -1          ;Dateinamen für
15700 ::1              lda    V0470,y        ;Ordner retten.
15701                pha
15702                dey
15703                bpl    :1
15704
15705                lda    V5C5B
15706                pha
15707
15708                lda    V5C5A
15709                pha
15710                lda    #$00
15711                sta    V5C5A
15712
15713                lda    diskBlkBuf +OFF_DIR_NUM
15714                jsr    Z5C38            ;0.-Inhalt löschen.
15715
15716                lda    V5C5A
15717                beq    :2
15718                tax
15719 ::2              pla
15720                sta    V5C5A
15721
15722                lda    V5C5B            ;=0 wenn alle
15723                sta    r0L              ;Dateien gelöscht.
15724                pla
15725                sta    V5C5B
15726
15727                ldy    #0              ;Dateinamen für
15728 ::3              pla                  ;Ordner wieder
15729                sta    V0470,y        ;zurücksetzen.
15730                iny
15731                cpy    #16
15732                bcc    :3
15733
15734                txa                    ;Fehler?
15735                bne    :err            ; => Ja, Abbruch...
15736
15737                ldx    #DIR_NOT_EMPTY

```

```

        lda    r0L                ;Alle gelöscht?          15738
        bne    :exit              ; => Nein, weiter...      15739
                                           15740
; Datei löschen.                                           15741
;                                                         15742
; Hinweis:                                                 15743
; Name nicht neu einlesen, da sich                         15744
; durch das löschen von Dateien im                        15745
; Ordner ggf. die Reihenfolge der                          15746
; Dateien geändert hat!                                    15747
::del      lda    #$00            ;Datei in Ordner          15748
          sta    r10L            ;"0" verschieben.         15749
          jsr    S2832           ;Datei/Ordner setzen.      15750
                                           15751
          jsr    S0B30           ;Datei löschen.           15752
;          txa                                     15753
;          bne    :err           ; => Fehler...             15754
                                           15755
::exit     clc                    ;OK.                      15756
          rts                                     15757
                                           15758
; Inhalt TopDesk-Ordner löschen.                          15759
; Übergabe: A = Ordner-Nr.                                15760
; TD13: DeleteDir                                         15761
:Z5C38     sta    a1L             15762
                                           15763
          ldx    C042A           ;Oberstes Fenster.       15764
          lda    C051B,x         ;Aktive Ordner-Nr.        15765
          pha                                     15766
                                           15767
          lda    a1L             15768
          sta    C051B,x         ;Aktive Ordner-Nr.        15769
                                           15770
          lda    #$00           ;Zeiger auf erste         15771
          sta    V5C5B          ;Datei setzen.            15772
                                           15773
::1        lda    V5C5B          ;Datei-Nr.               15774
          jsr    Z5B35           ;Eintrag löschen.         15775
          txa                                     15776
          bne    :2              ; => Fehler...            15777
          bcc    :1              ; => Nächste Datei.       15778
                                           15779
::2        ldy    C042A           ;Oberstes Fenster.       15780
          pla                                     15781
          sta    C051B,y         ;Aktive Ordner-Nr.        15782
                                           15783
;          ldx    #$ff           ;Fehler-Status in X!      15784
          rts                                     15785
                                           15786
:V5C5B     b $00    ;Zeiger auf Datei in Ordner.          15787
:V5C5A     b $00    ;>0 = Fehler beim löschen.           15788
                                           15789
; BubbleSort.                                             15790
; Sortieren von Byte-Werten in absteigender Reihenfolge. 15791
:Z5C50     lda    #< V634F       ;Zeiger Tabelle mit      15792

```

```

15793          sta    r2L          ;markierten Dateien.
15794          lda    #> V634F
15795          sta    r2H
15796
15797 ; Sortieren von Byte-Werten in absteigender Reihenfolge.
15798 ; Übergabe: r2 = Zeiger auf Bytewerte, Ende mit $FF.
15799 ; TD13: BubbleSort
15800 :Z5C5C          lda    #$00          ;Keine Werte
15801          sta    a2L          ;getauscht.
15802
15803          ldy    #$01
15804 ::1          lda    (r2),y
15805          cmp    #$ff          ;Ende erreicht?
15806          beq    :4          ; => Ja, Ende...
15807          dey
15808          cmp    (r2),y
15809          bcc    :2
15810
15811          lda    (r2),y          ;Werte tauschen.
15812          pha
15813          iny
15814          lda    (r2),y
15815          dey
15816          sta    (r2),y
15817          iny
15818          pla
15819          sta    (r2),y
15820
15821          lda    #$01          ;Werte wurden
15822          sta    a2L          ;getauscht.
15823          bne    :3
15824
15825 ::2          iny
15826 ::3          iny
15827          bne    :1
15828
15829 ::4          lda    a2L          ;Werte getauscht?
15830          bne    Z5C5C          ; => Neu sortieren.
15831          rts
15832
15833 ; Diskette für Fenster öffnen.
15834 ; Rückgabe: curDrive = Laufwerk.
15835 ;          X = Fenster-Nr.
15836 ;          A = Fehler, $00 = C=0, sonst C=1
15837 ; TD13: GetAktlDisk
15838 .S5C89          lda    r0H
15839          pha
15840          lda    r0L
15841          pha
15842
15843          lda    r6H
15844          pha
15845          lda    r6L
15846          pha
15847

```

	ldx	C042A	;Oberstes Fenster.	15848
	jsr	Z5CB3	;Diskette öffnen.	15849
	stx	a9H	;Fehlerstatus.	15850
				15851
	pla			15852
	sta	r6L		15853
	pla			15854
	sta	r6H		15855
				15856
	pla			15857
	sta	r0L		15858
	pla			15859
	sta	r0H		15860
				15861
	ldx	C042A	;Oberstes Fenster.	15862
				15863
	clc		; => Kein Fehler.	15864
				15865
	lda	a9H	;Fehler?	15866
	beq	:1	; => Nein, Ende...	15867
				15868
	sec		; => Fehler...	15869
				15870
::1	rts			15871
				15872
; Diskette öffnen.				15873
; Übergabe: X = Fenster-Nr.				15874
; Rückgabe: X = Fehler.				15875
; C = 0: Kein Fehler.				15876
; TD13: GetDisk				15877
:Z5CB3	txa			15878
	pha			15879
				15880
	lda	C042A	;Oberstes Fenster.	15881
	pha			15882
	stx	C042A	;Oberstes Fenster.	15883
	lda	C04F8,x	;Lfwk. für Fenster.	15884
	jsr	S06AB	;Laufwerk öffnen.	15885
	pla			15886
	sta	C042A	;Oberstes Fenster.	15887
				15888
	pla		;Fenster-Nr.	15889
	tay			15890
	lda	C051B,y	;Aktive Ordner-Nr.	15891
	bne	:1	; >0, weiter...	15892
				15893
; Kein Ordner aktiv, Diskname einlesen.				15894
	tya		;Fenster-Nr. im	15895
	tax		;X-Reg. übergeben.	15896
	jsr	Z4675	;Diskname einlesen.	15897
				15898
; Laufwerk- oder Ordner-Fehler?				15899
::1	clc		; => Kein Fehler.	15900
				15901
	txa		;Fehler?	15902

```

15903          beq    :2          ; => Nein, Ende...
15904
15905          sec                  ; => Fehler.
15906
15907 ::2          rts
15908
15909 ; Fehler "Nicht gefunden A-D" ausgeben.
15910 ; Übergabe: X = Fehler-Nr.
15911 .S5CD0          ldy    numDrives
15912                dey
15913                beq    :1
15914
15915                txa
15916                pha
15917
15918                lda    #8
15919                jsr    SetDevice
15920
15921                lda    #" "          ;Fehlertext anpassen:
15922                sta    V5DD4          ;"Laufwerk A - D"
15923
15924                lda    numDrives
15925                clc
15926                adc    #"A" -1
15927                sta    V5DD5
15928
15929                pla
15930                tax
15931
15932 ::1          jsr    S5CE1          ;Fehler ohne Refresh.
15933
15934                lda    #NULL
15935                sta    V5DD4
15936                rts
15937
15938 ; Diskfehler ausgeben, Fenster schließen.
15939 ; Übergabe: X = Fehler
15940 ;          WIN042F = Fenster-Nr.
15941 ; TD13: FehlerAusgabe2
15942 :Z5CEB
15943
15944 ; Hinweis:
15945 ; Wird nur das Fenster geschlossen ist
15946 ; danach kein Fenster "aktiv".
15947 ; Daher Fenster über den WindowHandler
15948 ; nach der Fehlermeldung schließen.
15949 if FALSE
15950                ldy    WIN042F          ;Aktives Fenster.
15951                lda    #$00
15952                sta    C0448,y          ;Fenster schließen.
15953 else
15954                jsr    S5CE1          ;Fehler ohne Refresh.
15955
15956 :Z5CF0          lda    #WN_CLOSE          ;Aktuelles Fenster
15957                sta    WIN042E          ;schließen.

```

	jmp	Z48CC	;Message-Handler.	15958
endif				15959
				15960
; Diskfehler ausgeben.				15961
; Übergabe: X = Fehler-Nr.				15962
; TD13: FehlerAusgabe				15963
.S5CF3	lda	#\$00	;DBox/Bildschirm neu.	15964
	b	\$2c		15965
				15966
; Fehler ausgeben, ohne Screen-Refresh.				15967
; Übergabe: X = Fehler-Nr.				15968
.S5CE1	lda	#\$ff		15969
	sta	:refresh	;Kein Screen-Refresh.	15970
				15971
	cpx	#NO_ERROR	;Kein Fehler?	15972
	bne	:err	; => Nein, weiter...	15973
	rts			15974
				15975
::err	ldy	curDrive		15976
	lda	RealDrvMode -8,y		15977
	and	#SET_MODE_SD2IEC		15978
	beq	:1		15979
				15980
; Hinweis:				15981
; Das SD2IEC meldet beim Versuch einen				15982
; Block über das TurboDOS auf eine				15983
; schreibgeschützte Karte zu schreiben				15984
; einen JobCode-Error \$02 an Stelle von				15985
; \$08 (Write protected).				15986
; Der Treiber addiert #30 zum JobCode-				15987
; Error -> \$02+30=\$20 Header error.				15988
; Fehlercode anpassen.				15989
; Bei MP3/KernalDriver meldet SD2IEC				15990
; ein "72, DISK FULL" beim Versuch auf				15991
; die Disk zu schreiben.				15992
	cpx	#\$72	;Disk full?	15993
	beq	:0	; => Ja, weiter...	15994
	cpx	#\$20	;Header not found?	15995
	bne	:1	; => Nein, weiter...	15996
				15997
::0	ldx	#\$26	; => Schreibschutz.	15998
				15999
::1	lda	curDrive	;Laufwerk speichern.	16000
	clc			16001
	adc	#"A" -8		16002
	sta	V5DD8		16003
				16004
	txa		;Fehler nach HEX	16005
	and	##%11110000	;wandeln.	16006
	lsr			16007
	lsr			16008
	lsr			16009
	lsr			16010
	clc			16011
	adc	#"0"		16012

```

16013      cmp    #"9" +1
16014      bcc    :2
16015      adc    #$06
16016  ::2    sta    V5DC6 +0
16017
16018      txa
16019      and    #%00001111
16020      clc
16021      adc    #"0"
16022      cmp    #"9" +1
16023      bcc    :3
16024      adc    #$06
16025  ::3    sta    V5DC6 +1
16026
16027      txa                                ;Fehlercode
16028      pha                                ;zwischenspeichern.
16029
16030      ldy    #MAXERR -1                  ;Fehlercode suchen.
16031  ::4    cmp    V5DE5,y
16032      beq    :5
16033      dey
16034      bpl    :4
16035
16036      ldy    #< T609F                    ;Fehler nicht
16037      lda    #> T609F                    ;gefunden.
16038      bne    :6
16039
16040  ::5    tya
16041      asl
16042      tax
16043
16044      ldy    V5E00 +0,x                  ;Zeiger auf
16045      lda    V5E00 +1,x                  ;Fehlertext.
16046
16047  ::6    sty    a2L
16048      sta    a2H
16049
16050      lda    #$04
16051      jsr    S24C0                        ;Fehlerton.
16052
16053  ; Hinweis:
16054  ; Fehlerkanal auslesen um ein evtl.
16055  ; blinken der Fehler-LED am SD2IEC
16056  ; zurückzusetzen.
16057  ; Das Problem ist aber die Firmware im
16058  ; SD2IEC, die ein Fehler-Blinken der
16059  ; LED selbstständig abschalten sollte.
16060  if FALSE
16061      jsr    PurgeTurbo
16062      jsr    InitForIO
16063      lda    #0
16064      jsr    $ffbd                        ;SETNAM
16065      lda    #15
16066      ldx    curDrive
16067      ldy    #15

```



```

        jsr    $ffba                ;SETLFS                16068
        jsr    $ffc0                ;OPENCHN              16069
        ldx    #15                  16070
        jsr    $ffc6                ;CHKIN                16071
::w1      jsr    $ffb7                ;READST              16072
        bne    :w2                  16073
        jsr    $ffcf                ;CHRIN                16074
        jmp    :w1                  16075
::w2      lda    #15                  16076
        jsr    $ffc3                ;CLOSE                16077
        jsr    $ffcc                ;CLRCHN               16078
        jsr    DoneWithIO           16079
endif                                           16080
                                           16081
        lda    COL043C              ;Farbe Dialogbox      16082
        pha                                ;zwischenspeichern. 16083
                                           16084
        lda    COL043E              ;Farbe Fehlerbox.     16085
        sta    COL043C              16086
                                           16087
        lda    #< D5D74             ;Diskfehler.         16088
        ldx    #> D5D74             16089
        jsr    S2090                ;DBox ohne Recover.   16090
                                           16091
        pla                                16092
        sta    COL043C              ;Farbe Dialogbox.     16093
                                           16094
        bit    :refresh              ;DBox/Bildschirm neu? 16095
        bmi    :7                   ; => Nein, weiter...    16096
                                           16097
        jsr    S524C                ;Dat.-Ausw. aufheben.  16098
;        jsr    S14F8                ;Bildschirm löschen.  16099
        jsr    S157E                ;Alle Fens. anzeigen. 16100
                                           16101
::7        pla                                16102
        tax                                16103
::exit     rts                        16104
                                           16105
; Fehlerbox: DeskTop neu aufbauen.             16106
::refresh  b $00    ;$FF=Nach DBox Bildschirm nicht neu. 16107
                                           16108
; Fehlerbox.                                   16109
; Übergabe: a2 = Zeiger auf Fehlertext.         16110
:D5D74     b %10000001              16111
                                           16112
        b OK,$11,$48                16113
                                           16114
if FALSE                                           16115
        b DBTXTSTR,$10,$0e          16116
        w T29CF                     16117
                                           16118
        b DBTXTSTR,$10,$1c          16119
        w T5D91                     16120
        b DBTXTSTR,$10,$2a          16121
        w T5DB0                     16122

```

```

16123
16124          b DBVARSTR,$10,$38
16125          b a2L
16126
16127          b DBTXTSTR,$10,$46
16128          w T5DCA
16129 endif
16130 if TRUE
16131          b DBTXTSTR,$10,$10
16132          w T29CF
16133
16134          b DBTXTSTR,$10,$20
16135          w T5DB1
16136
16137          b DBVARSTR,$10,$2c
16138          b a2L
16139
16140          b DBTXTSTR,$10,$3c
16141          w T5DCA
16142 endif
16143
16144          b NULL
16145
16146 if LANG = LANG_DE
16147 ;:T5D91          b "Vorgang abgebrochen!",0
16148 ;:T5DB0          b "Fehler $"
16149 :T5DB1          b "Fehler: $"
16150 :V5DC6          b ".."
16151                b BOLDON,0
16152 :T5DCA          b PLAINTEXT
16153                b "Laufwerk "
16154 endif
16155 if LANG = LANG_EN
16156 ;:T5D91          b "Operation cancelled!",0
16157 ;:T5DB0          b "Error $"
16158 :T5DB1          b "Error: $"
16159 :V5DC6          b ".."
16160                b BOLDON,0
16161 :T5DCA          b PLAINTEXT
16162                b "Drive "
16163 endif
16164
16165 ; Fehler-Laufwerk oder
16166 ; erstes Laufwerk im Bereich A-D.
16167 :V5DD8          b "x:"      ;Wird verändert.
16168 :V5DD4          b 0        ;Wird verändert.
16169
16170 ;                b "-"
16171 if LANG = LANG_DE
16172                b "bis "
16173 endif
16174 if LANG = LANG_EN
16175                b "to "
16176 endif
16177

```

```

; Letztes Laufwerk im Bereich A-D. 16178
;V5DD5      b "x:",0  ;Wird verändert. 16179
16180
; Fehlercodes. 16181
;V5DE5      b NO_BLOCKS           ;$01 16182
           b INV_TRACK            ;$02 16183
           b $03                  ;INSUFF_SPACE 16184
           b FULL_DIRECTORY       ;$04 16185
           b FILE_NOT_FOUND       ;$05 16186
           b BAD_BAM              ;$06 16187
           b $07                  ;UNOPENED_VLIR 16188
           b INV_RECORD           ;$08 16189
           b OUT_OF_RECORDS       ;$09 16190
           b STRUCT_MISMAT        ;$0A 16191
           b BFR_OVERFLOW         ;$0B 16192
           b CANCEL_ERR           ;$0C 16193
           b DEV_NOT_FOUND        ;$0D 16194
;           b INCOMPATIBLE        ; => Nicht verwendet. 16195
           b $20                  ;HDR_NOT_THERE 16196
           b NO_SYNC              ;$21 16197
           b $22                  ;DBLKNOT_THERE 16198
           b $23                  ;DAT_CHKSUM_ERR 16199
           b $25                  ;WR_VER_ERROR 16200
           b WR_PR_ON             ;$26 16201
           b $27                  ;HDR_CHKSUM_ERR 16202
           b $29                  ;DSK_ID_MISMAT 16203
           b $2e                  ;BYTE_DEC_ERR 16204
           b $34                  ;NO_DIRECTORY 16205
           b DOS_MISMATCH         ;$73 16206
;           b FILE_EXISTS         ; => Nicht verwendet. 16207
           b PART_ERR             ;$64 16208
           b ERR_41DBLSIDE        ;$80 16209
16210
; TODO 16211
; Fehlende Fehlermeldungen ergänzen. 16212
;           b TDLIST_FULL         ;$40 16213
;           b TDSUB_EXISTS        ;$42 16214
;           b ERR_UNKNOWN         ;$45 16215
;           b ERR_NO_DOC          ;$46 16216
16217
           b ERR_NO_APPL          ;$47 16218
           b DIR_NOT_EMPTY        ;$48 16219
           b ERR_NO_WINDOW        ;$49 16220
:ENDERRTAB 16221
:MAXERR     = ENDERRTAB - V5DE5 16222
16223
; Zeiger auf Fehlertexte. 16224
;V5E00      w :01 16225
           w :02 16226
           w :01 16227
           w :04 16228
           w :05 16229
           w :06 16230
           w :07 16231
           w :08 16232

```

```

16233          w :09
16234          w :0A
16235          w :0B
16236          w :0C
16237          w :0D
16238 ;          w :0E ;INCOMPATIBLE, nicht verwendet.
16239          w :20
16240          w :21
16241          w :22
16242          w :23
16243          w :25
16244          w :26
16245          w :27
16246          w :29
16247          w :2E
16248          w :34
16249          w :73
16250 ;          w :41 ;FILE_EXISTS, nicht verwendet.
16251          w :64
16252          w :80
16253
16254 ;          w :40 ;TDLIST_FULL, kein Fehler.
16255 ;          w :42
16256 ;          w :45
16257 ;          w :46
16258
16259          w :47
16260          w :48
16261          w :4A
16262
16263 ; Fehlertexte.
16264 if LANG = LANG_DE
16265 ::01          b "Diskette voll",0
16266 ::02          b "Ungültiger Track",0
16267 ::04          b "Verzeichnis voll",0
16268 ::05          b "Datei nicht gefunden",0
16269 ::06          b "BAM fehlerhaft",0
16270 ::07          b "VLIR-Datei nicht geöffnet",0
16271 ::08          b "Ungültiger Datensatz",0
16272 ::09          b "Zu viele Datensätze",0
16273 ::0A          b "Falsche Dateistruktur",0
16274 ::0B          b "Datei-/Datenspeicher voll",0
16275 ::0C          b "Abbruchfehler",0
16276 ::0D          b "Gerät nicht vorhanden",0
16277 ;::0E          b "Falscher Grafikmodus",0
16278 ::20          b "Header nicht gefunden",0
16279 ::21          b "Keine formatierte Disk",0
16280 ::22          b "Datenblock nicht vorhanden",0
16281 ::23          b "Daten-Prüfsummenfehler",0
16282 ::25          b "Fehler beim Schreiben",0
16283 ::26          b "Schreibschutz auf Disk",0
16284 ::27          b "Header-Prüfsummenfehler",0
16285 ::29          b "Falsche Disk-ID",0
16286 ::2E          b "Byte-Decodierungsfehler",0
16287 ::34          b "Kein Verzeichnis",0

```

```

:::73      b "Disk mit falscher DOS-Marke",0      16288
;:::41     b "Name bereits vergeben",0           16289
:::64      b "Partition-Fehler",0                 16290
:::80      b "Doppelseitige Disk in 1541",0       16291
           16292
;:::40     b "Zu viele Ordner geöffnet",0         16293
;:::42     b "Ordner bereits geöffnet",0          16294
;:::45     b "Unbekannter Fehler",0               16295
;:::46     b "Kein Dokument",0                   16296
           16297
:::47      b "Anwendung nicht gefunden",0         16298
:::48      b "Verzeichnis nicht leer",0           16299
:::4A      b "Kein Fenster frei",0                16300
endif                                             16301
if LANG = LANG_EN                               16302
:::01      b "Disk full",0                        16303
:::02      b "Invalid track",0                    16304
:::04      b "Full directory",0                   16305
:::05      b "File not found",0                   16306
:::06      b "Bad BAM",0                          16307
:::07      b "VLIR file not open",0               16308
:::08      b "Record invalid",0                   16309
:::09      b "Too many records",0                 16310
:::0A      b "Wrong file structure",0             16311
:::0B      b "Not enough memory",0                16312
:::0C      b "Cancel error",0                     16313
:::0D      b "Drive not ready",0                  16314
;:::0E     b "Wrong graphic mode",0               16315
:::20      b "File header not found",0            16316
:::21      b "Disk not formatted",0               16317
:::22      b "Data block not found",0             16318
:::23      b "Data checksum error",0              16319
:::25      b "Write error",0                      16320
:::26      b "Disk write protected",0             16321
:::27      b "Header checksum error",0            16322
:::29      b "Wrong disk ID",0                   16323
:::2E      b "Byte decode error",0                16324
:::34      b "No directory",0                     16325
:::73      b "Invalid DOS version",0              16326
;:::41     b "File already exist",0               16327
:::64      b "Partition error",0                  16328
:::80      b "Double-sided disk in 1541",0        16329
           16330
;:::40     b "Too many open folders",0            16331
;:::42     b "Folder already opened",0            16332
;:::45     b "Unknown error",0                    16333
;:::46     b "Not a document file",0              16334
           16335
:::47      b "Application not found",0            16336
:::48      b "Directory not empty",0              16337
:::4A      b "No free window",0                   16338
endif                                             16339
           16340
; String für unbekannten Fehler.                  16341
:T609F     b NULL ;Unbekannter Fehlercode.        16342

```

```

16343
16344 ; Hinweis:
16345 ; Wird nur in mod.#02 als Speicher für
16346 ; den Verzeichnisheader verwendet.
16347 ;
16348 ; TODO:
16349 ; Wird nur in mod.#02 verwendet, muss
16350 ; im Hauptmodul verbleiben, da beim
16351 ; duplizieren zwischen mod.#09 und
16352 ; mod.#02 gewechselt wird.
16353 .V6056          b $00    ;Tr/Verzeichnisheader.
16354 .V6057          b $00    ;Se/Verzeichnisheader.
16355
16356 ; Uhrzeit aktualisieren.
16357 ; TD13: ShowClock
16358 ;
16359 ; Hinweis:
16360 ; Aufruf aus mod.#12 oder durch
16361 ; die GEOS-Prozess-Routine.
16362 ;
16363 ; Wird V61E7 auf $ff gesetzt, dann
16364 ; soll die Uhrzeit komplett angezeigt
16365 ; werden, siehe mod.#00b/MAININIT.
16366 ;
16367 :ENABLE_CLKWIN = TRUE
16368 .S60DF          lda     #ST_WR_FORE
16369                sta     dispBufferOn
16370
16371                php
16372                sei
16373
16374                ldx     #$00                ;Nicht aktualisieren.
16375                lda     seconds
16376                cmp     V61E8
16377                beq     :1
16378
16379                sta     V61E8
16380                ldx     #$ff                ;Zeit aktualisieren.
16381
16382 ; YY/MM/DD und hh/mm vergleichen.
16383 ::1            ldy     #3 +2 -1            ;Datum + Uhrzeit.
16384 ::2            lda     year,y
16385                cmp     V61E3,y
16386                beq     :3
16387                ldx     #$01                ;Alles aktualisieren.
16388                sta     V61E3,y
16389 ::3            dey
16390                bpl     :2
16391
16392                lda     millenium
16393                sta     V61E2
16394
16395                plp
16396
16397                txa

```

```

        bmi    :4                ;Nur Sekunden akt.      16398
        bne    S6118             ;Alles aktualisieren.   16399
                                           16400
        rts                      ;Nicht aktualisieren.   16401
                                           16402
::4      lda    #$01             ;Nur Sekunden akt.      16403
        b $2c                    16404
                                           16405
; Datum/Uhrzeit anzeigen. 16406
.S6118   lda    #$00             ;Alles aktualisieren.   16407
        pha                      16408
        tax                      16409
        bne    :1                16410
                                           16411
        ldx    V61E5             ;Tag.                  16412
        lda    #OFF_DAY          16413
        jsr    Z61B8             ;Zahl DEZ -> ASCII.     16414
        ldx    V61E4             ;Monat.                16415
        lda    #OFF_MONTH        16416
        jsr    Z61B8             ;Zahl DEZ -> ASCII.     16417
        ldx    V61E2             ;millenium.            16418
        lda    #OFF_MILLENIUM    16419
        jsr    Z61B8             ;Zahl DEZ -> ASCII.     16420
        ldx    V61E3             ;Jahr.                  16421
        lda    #OFF_YEAR         16422
        jsr    Z61B8             ;Zahl DEZ -> ASCII.     16423
        ldx    V61E6             ;Stunde.                16424
        lda    #OFF_HOUR         16425
        jsr    Z61B8             ;Zahl DEZ -> ASCII.     16426
        ldx    V61E7             ;Minute.                16427
        lda    #OFF_MINUTES      16428
        jsr    Z61B8             ;Zahl DEZ -> ASCII.     16429
                                           16430
::1      ldx    V61E8             ;Sekunde.              16431
        lda    #OFF_SECONDS      16432
        jsr    Z61B8             ;Zahl DEZ -> ASCII.     16433
                                           16434
; Textgrenzen für Datum/Uhrzeit setzen. 16435
if ENABLE_CLKWIN = TRUE 16436
        ldx    #2 +4 -1          ;2 Byte + 2 Word.      16437
::4      lda    :coord,x          16438
        sta    windowTop,x       16439
        dex                      16440
        bpl    :4                16441
endif 16442
                                           16443
; Hinweis: 16444
; Linken/rechten Rand für Textausgabe 16445
; setzen, oben/unten kann hier aber 16446
; ausserhalb der Zeit-Anzeige liegen! 16447
; In dem Fall würden dann nur Sekunden 16448
; angezeigt bis sich der Minuten-Wert 16449
; das nächste mal ändert. 16450
if ENABLE_CLKWIN = FALSE 16451
        lda    #< TIME_X0 +1     ;Textgrenzen Uhrzeit. 16452

```

```

16453          sta    leftMargin +0
16454          lda    #> TIME_X0 +1
16455          sta    leftMargin +1
16456
16457          lda    #< TIME_X1 -1
16458          sta    rightMargin +0
16459          lda    #> TIME_X1 -1
16460          sta    rightMargin +1
16461 endif
16462
16463          pla
16464          bne     :3
16465
16466          jsr     Z6160                ;Datum/Uhrzeit.
16467
16468          lda     r11L                ;Pos. für Sekunden
16469          sec
16470          sbc     #$05                ;berechnen.
16471          sta     W61A4 +0
16472          bcs     :2
16473          dec     r11H
16474
16475 ::2          lda     r11H
16476          sta     W61A4 +1
16477
16478 ::3          jsr     Z61A1                ;Sekunden ausgeben.
16479
16480 ; Grenzen für Textausgabe zurücksetzen.
16481 ::_61AF      jmp     Z13D3                ;Textgrenzen Desktop.
16482
16483 ; Textgrenzen für Datum/Uhrzeit.
16484 if ENABLE_CLKWIN = TRUE
16485 ::coord      b TIME_Y0 +1, TIME_Y1 -1
16486              w TIME_X0 +1, TIME_X1 -1
16487 endif
16488
16489 ; Datum/Uhrzeit ausgeben.
16490 :Z6160        jsr     i_PutString
16491              w TIME_X0 +2
16492              b TIME_Y1 -5
16493
16494 .V6167        b PLAINTXT                ;Tag.
16495 .V6168        b "0"
16496              b PLAINTXT
16497              b "0"
16498
16499              b PLAINTXT
16500              b "."
16501
16502              b PLAINTXT                ;Monat.
16503 :V616E        b "0"
16504              b PLAINTXT
16505              b "0"
16506
16507              b PLAINTXT

```



	b "."	16508
		16509
:V6174	b PLAINTEXT ;Jahr (4st.)	16510
	b "0"	16511
	b PLAINTEXT	16512
	b "0"	16513
:V6178	b PLAINTEXT	16514
	b "0"	16515
	b PLAINTEXT	16516
	b "0"	16517
		16518
	b PLAINTEXT ;Cleanup.	16519
	b " "	16520
		16521
	b GOTOXY	16522
	w TIME_X0 +46	16523
	b TIME_Y1 -5	16524
		16525
:V6183	b PLAINTEXT ;Stunde.	16526
	b "0"	16527
	b PLAINTEXT	16528
	b "0"	16529
		16530
	b PLAINTEXT	16531
	b ":"	16532
		16533
:V6189	b PLAINTEXT ;Minute.	16534
	b "0"	16535
	b PLAINTEXT	16536
	b "0"	16537
		16538
	b PLAINTEXT ;Cleanup.	16539
	b " "	16540
	b NULL	16541
		16542
	rts	16543
		16544
	; Sekunden ausgeben.	16545
:Z61A1	jsr i_PutString	16546
:W61A4	w \$0150 ;Wird berechnet!	16547
	b TIME_Y1 -5	16548
		16549
	b ":"	16550
:V61A8	b "0"	16551
	b PLAINTEXT	16552
	b "0 "	16553
	b PLAINTEXT	16554
	b NULL	16555
		16556
	rts	16557
		16558
.OFF_DAY	= V6168 - V6168	16559
.OFF_MONTH	= V616E - V6168	16560
.OFF_MILLENIUM	= V6174 - V6168	16561
.OFF_YEAR	= V6178 - V6168	16562

```

16563 .OFF_HOUR          = V6183 - V6168
16564 .OFF_MINUTES     = V6189 - V6168
16565 .OFF_SECONDS      = V61A8 - V6168
16566
16567 ; Zahl von DEZ nach ASCII wandeln.
16568 ; Übergabe: A = Offset.
16569 ;           X = Zahl 0-99.
16570 ; TD13: DivnSet
16571 :Z61B8              pha
16572
16573                     stx    r0L
16574                     lda    #$00                ;Highbyte = 0.
16575                     sta    r0H
16576                     ldx    #r0
16577
16578                     ldy    #10
16579                     sty    r1L
16580 ;                     lda    #$00                ;Highbyte = 0.
16581                     sta    r1H
16582                     ldy    #r1
16583
16584                     jsr    Ddiv
16585
16586                     pla
16587                     tax
16588
16589                     lda    r0L                ;Zahl in Uhrzeit-
16590                     clc                        ;String eintragen.
16591                     adc    #"0"
16592                     sta    V6168,x
16593                     inx                        ;Nächstes Zeichen.
16594                     inx                        ;PLAINTEXT überspr.
16595                     lda    r8L
16596                     adc    #"0"
16597                     sta    V6168,x
16598                     rts
16599
16600 :V61E2              b $00 ;millenium
16601 :V61E3              b $00 ;year
16602 :V61E4              b $00 ;month
16603 :V61E5              b $00 ;day
16604 :V61E6              b $00 ;hour
16605 :V61E7              b $00 ;minutes
16606 .V61E8             b $00 ;seconds
16607
16608 ; Zeiger auf Tabelle mit Fensterfarben.
16609 ;Übergabe: X = Fenster-Nr.
16610 .S61E9             txa
16611                     pha
16612
16613 ; Hinweis:
16614 ; TopDesk verwendet die Farbtabelle für
16615 ; Fenster#1 für alle Fenster, wenn die
16616 ; Option "Verschiedene Farben" im Menü
16617 ; "Fenster" nicht aktiviert ist.

```

```

; Ist die Option aber aktiviert, dann                                16618
; überschreibt eine PadColorPref auf                                16619
; einer Disk dauerhaft die Farbtabelle                              16620
; für das betroffene Fenster.                                       16621
if ENABLE_DIFFICOL = TRUE                                           16622
    ldy    MT3F58                                                    ;Versch. Fens.farben?    16623
    cpy    #""                                                       16624
    beq    :2                                                         ; => Ja, weiter...      16625
    ldx    #$00                                                       ;Nur Farbe Fens. #1.    16626
::2    ldy    V621B,x                                                  16627
    sty    V621A                                                      ;Aktive Fensterfarbe.   16628
                                           16629
;    pla                                           16630
;    pha                                           16631
                                           16632
    ldy    MT3F59                                                    ;Versch. Iconfarben?    16633
    cpy    #""                                                       16634
    beq    :1                                                         ; => Ja, weiter...      16635
    lda    #$00                                                       ;Nur Farbe Fens. #1.    16636
endif                                                                16637
if ENABLE_DIFFICOL = FALSE                                           16638
    ldy    MT3F58                                                    ;Verschiedene Farben?   16639
    cpy    #""                                                       16640
    beq    :1                                                         ; => Ja, weiter...      16641
    lda    #$00                                                       ;Nur Farbe Fens. #1.    16642
endif                                                                16643
                                           16644
::1    asl                                           16645
    tax                                           16646
    lda    V6203 +0,x                                                  16647
    sta    a9L                                                         16648
    lda    V6203 +1,x                                                  16649
    sta    a9H                                                         16650
                                           16651
    pla                                           16652
    tax                                           16653
    rts                                           16654
                                           16655
if ENABLE_DIFFICOL = TRUE                                           16656
.V621A    b $0f ;Farbe aktives Fenster.                             16657
.V621B    b $0f ;Farbe Fenster #1 bis #4.                             16658
.V622C    b $0d                                                         16659
.V623D    b $0a                                                         16660
.V624E    b $0e                                                         16661
endif                                                                16662
                                           16663
; Zeiger auf Fensterfarben.                                           16664
:V6203    w V620B                                                         16665
    w V621C                                                         16666
    w V622D                                                         16667
    w V623E                                                         16668
                                           16669
; Hinweis:                                                            16670
; Die Originalfarben haben je nach                                    16671
; Fensterfarbe andere Iconfarben, wenn                                16672

```

```

16673 ; die Kombination Icon- / Fensterfarbe
16674 ; nicht gut erkennbar ist.
16675
16676 ; Farben Fenster #1 oder alle Fenster.
16677 ; Vorgabe Fensterfarbe: Hellgrau.
16678 if ENABLE_NEWICOL = FALSE
16679 .V620B          b $10 ;weiß      : Nicht-GEOS
16680                b $10 ;weiß      : BASIC
16681                b $30 ;cyan       : Assembler
16682                b $40 ;lila       : Daten
16683                b $50 ;grün       : System
16684                b $80 ;orange     : DeskAccessory
16685                b $20 ;rot        : Anwendung
16686                b $e0 ;hellblau   : Dokument
16687                b $60 ;blau       : Zeichensatz
16688                b $b0 ;dunkelgrau: Druckertreiber
16689                b $a0 ;hellrot    : Eingabetreiber/64
16690                b $70 ;gelb       : Laufwerkstreiber/Ordner
16691                b $d0 ;hellgrün   : Startprogramm
16692                b $10 ;weiß      : Temporär
16693                b $c0 ;grau       : Selbszausführend
16694                b $90 ;braun      : Eingabetreiber/128
16695 endif
16696 if ENABLE_NEWICOL = TRUE
16697 .V620B          b $b0 ;dunkelgrau: Nicht-GEOS
16698                b $b0 ;dunkelgrau: BASIC
16699                b $b0 ;dunkelgrau: Assembler
16700                b $b0 ;dunkelgrau: Daten
16701                b $00 ;schwarz     : System
16702                b $20 ;rot         : DeskAccessory
16703                b $60 ;blau       : Anwendung
16704                b $50 ;grün       : Dokument
16705                b $90 ;braun      : Zeichensatz
16706                b $00 ;schwarz     : Druckertreiber
16707                b $00 ;schwarz     : Eingabetreiber/64
16708                b $10 ;weiß      : Laufwerkstreiber/Ordner
16709                b $00 ;schwarz     : Startprogramm
16710                b $b0 ;dunkelgrau: Temporär
16711                b $40 ;lila       : Selbszausführend
16712                b $00 ;schwarz     : Eingabetreiber/128
16713 endif
16714
16715 if ENABLE_DIFFICOL = FALSE
16716 .V621B          b $0f
16717 endif
16718
16719 ; Farben Fenster #2.
16720 ; Vorgabe Fensterfarbe: Hellgrün.
16721 if ENABLE_NEWICOL = FALSE
16722 .V621C          b $10 ;weiß      : Nicht-GEOS
16723                b $10 ;weiß      : BASIC
16724                b $30 ;cyan       : Assembler
16725                b $40 ;lila       : Daten
16726                b $50 ;grün       : System
16727                b $80 ;orange     : DeskAccessory

```

```

        b $20 ;rot      : Anwendung          16728
        b $e0 ;hellblau : Dokument           16729
        b $60 ;blau     : Zeichensatz        16730
        b $b0 ;dunkelgrau: Druckertreiber    16731
        b $a0 ;hellrot  : Eingabetreiber/64   16732
::var3    b $90 ;braun   : Laufwerkstreiber/Ordner 16733
::var5    b $e0 ;hellblau : Startprogramm      16734
        b $10 ;weiß    : Temporär           16735
        b $c0 ;grau    : Selbszausführend    16736
        b $90 ;braun   : Eingabetreiber/128   16737
endif                                           16738
if ENABLE_NEWICOL = TRUE                      16739
.V621C    b $b0 ;dunkelgrau: Nicht-GEOS      16740
        b $b0 ;dunkelgrau: BASIC            16741
        b $b0 ;dunkelgrau: Assembler        16742
        b $b0 ;dunkelgrau: Daten            16743
        b $00 ;schwarz  : System             16744
        b $20 ;rot      : DeskAccessory      16745
        b $60 ;blau     : Anwendung          16746
        b $50 ;grün     : Dokument           16747
        b $90 ;braun   : Zeichensatz        16748
        b $00 ;schwarz  : Druckertreiber    16749
        b $00 ;schwarz  : Eingabetreiber/64   16750
        b $10 ;weiß    : Laufwerkstreiber/Ordner 16751
        b $00 ;schwarz  : Startprogramm      16752
        b $b0 ;dunkelgrau: Temporär         16753
        b $40 ;lila    : Selbszausführend    16754
        b $00 ;schwarz  : Eingabetreiber/128   16755
endif                                           16756
                                           16757
if ENABLE_DIFFICOL = FALSE                   16758
.V622C    b $0d                                           16759
endif                                           16760
                                           16761
; Farben Fenster #3.                          16762
; Vorgabe Fensterfarbe: Hellrot.              16763
if ENABLE_NEWICOL = FALSE                    16764
.V622D    b $10 ;weiß    : Nicht-GEOS      16765
        b $10 ;weiß    : BASIC            16766
        b $30 ;cyan     : Assembler        16767
        b $40 ;lila     : Daten            16768
        b $50 ;grün     : System             16769
        b $80 ;orange   : DeskAccessory      16770
        b $20 ;rot      : Anwendung          16771
        b $e0 ;hellblau : Dokument           16772
        b $60 ;blau     : Zeichensatz        16773
        b $b0 ;dunkelgrau: Druckertreiber    16774
::var2    b $e0 ;hellblau : Eingabetreiber/64   16775
        b $70 ;gelb     : Laufwerkstreiber/Ordner 16776
        b $d0 ;hellgrün : Startprogramm      16777
        b $10 ;weiß    : Temporär           16778
        b $c0 ;grau    : Selbszausführend    16779
::var6    b $e0 ;hellblau : Eingabetreiber/128   16780
endif                                           16781
if ENABLE_NEWICOL = TRUE                      16782

```

```

16783 .V622D          b $b0 ;dunkelgrau: Nicht-GEOS
16784                b $b0 ;dunkelgrau: BASIC
16785                b $b0 ;dunkelgrau: Assembler
16786                b $b0 ;dunkelgrau: Daten
16787                b $00 ;schwarz  : System
16788                b $20 ;rot      : DeskAccessory
16789                b $60 ;blau     : Anwendung
16790                b $50 ;grün     : Dokument
16791                b $90 ;braun    : Zeichensatz
16792                b $00 ;schwarz  : Druckertreiber
16793                b $00 ;schwarz  : Eingabetreiber/64
16794                b $10 ;weiß     : Laufwerkstreiber/Ordner
16795                b $00 ;schwarz  : Startprogramm
16796                b $b0 ;dunkelgrau: Temporär
16797                b $40 ;lila     : Selbszausführend
16798                b $00 ;schwarz  : Eingabetreiber/128
16799 endif
16800
16801 if ENABLE_DIFFICOL = FALSE
16802 .V623D          b $0a
16803 endif
16804
16805 ; Farben Fenster #4.
16806 ; Vorgabe Fensterfarbe: Hellblau.
16807 if ENABLE_NEWICOL = FALSE
16808 .V623E          b $10 ;weiß     : Nicht-GEOS
16809                b $10 ;weiß     : BASIC
16810                b $30 ;cyan     : Assembler
16811                b $40 ;lila     : Daten
16812                b $50 ;grün     : System
16813 ::var1          b $90 ;braun    : DeskAccessory
16814                b $20 ;rot      : Anwendung
16815                b $e0 ;hellblau : Dokument
16816                b $60 ;blau     : Zeichensatz
16817                b $b0 ;dunkelgrau: Druckertreiber
16818                b $a0 ;hellrot  : Eingabetreiber/64
16819 ::var4          b $60 ;blau     : Laufwerkstreiber/Ordner
16820                b $d0 ;hellgrün : Startprogramm
16821                b $10 ;weiß     : Temporär
16822                b $c0 ;grau     : Selbszausführend
16823                b $90 ;braun    : Eingabetreiber/128
16824 endif
16825 if ENABLE_NEWICOL = TRUE
16826 .V623E          b $b0 ;dunkelgrau: Nicht-GEOS
16827                b $b0 ;dunkelgrau: BASIC
16828                b $b0 ;dunkelgrau: Assembler
16829                b $b0 ;dunkelgrau: Daten
16830                b $00 ;schwarz  : System
16831                b $20 ;rot      : DeskAccessory
16832                b $60 ;blau     : Anwendung
16833                b $50 ;grün     : Dokument
16834                b $90 ;braun    : Zeichensatz
16835                b $00 ;schwarz  : Druckertreiber
16836                b $00 ;schwarz  : Eingabetreiber/64
16837                b $10 ;weiß     : Laufwerkstreiber/Ordner

```

```

        b $00 ;schwarz    : Startprogramm          16838
        b $b0 ;dunkelgrau: Temporär                16839
        b $40 ;lila      : Selbszausführend        16840
        b $00 ;schwarz    : Eingabetreiber/128      16841
endif                                           16842
                                           16843
if ENABLE_DIFFICOL = FALSE                    16844
.V624E          b $0e                          16845
endif                                           16846
                                           16847
; Ordner-Liste Fenster #1-#4.                  16848
;--- TD.WIN                                    16849
.C624F          s MAXDIRSIZE                    16850
:C628F          s MAXDIRSIZE                    16851
:C62CF          s MAXDIRSIZE                    16852
:C630F          s MAXDIRSIZE                    16853
;---                                           16854
                                           16855
; Tabelle mit markierten Dateien.              16856
; -> $FF    = Nicht markiert.                 16857
; -> 0-254 = Nr. der Datei im Fenster.         16858
.V634F          s MAXFILES                      16859
                                           16860
; Hinweis:                                    16861
; BubbleSort beendet das sortieren,           16862
; wenn der Wert 255 in der Tabelle mit         16863
; den markierten Dateien gefunden wird.        16864
; Bei 255 markierten Dateien gibt es          16865
; den Wert in der Tabelle aber nicht.          16866
; Fix für BubbleSort:                         16867
;---                                           16868
        b $ff                                    16869
;---                                           16870
                                           16871
; Hinweis:                                    16872
; Ab hier liegen max. 16 Dateinamen            16873
; zu je 16+2 Bytes (max. Anzahl Dateien        16874
; in einem Laufwerksfenster + Tr/Se für       16875
; den Infoblock) und dazu 16 Icons zu         16876
; je 64Byte (inkl. Kopfbyte $bf).             16877
;                                               16878
; Puffer in DIRBUF umfassen 1312 Bytes.        16879
.DIRBUF1                                               16880
.DIRBUF2 = DIRBUF1 +SIZE_DIRBUF                  16881
:ENDDBUF = DIRBUF2 +SIZE_DIRBUF                  16882
                                           16883
;                                               16884
; BASE_MOD_B                                     16885
;                                               16886
; TODO:                                         16887
; Ab hier beginnt COPYMEM_LOW = $6F00.        16888
; Konstante in Label umwandeln.               16889
;                                               16890
;                                               16891

```

```

16892
16893 ;
16894 ; SourceCode: src.mod#01.s
16895 ;
16896
16897 ;
16898 ; Modul-Info:
16899 ; 0 - Dateien retten
16900 ; 1 - Disk aufräumen
16901 ; 2 - Disk löschen
16902 ;
16903 if .p
16904             t "TopSym"
16905             t "TopSym.MP3"
16906             t "TopSym.ROM"
16907             t "DTopDeskI18N.ext"
16908             t "src.DTopDesk.ext"
16909
16910 ; Größe Dialogbox:
16911 :DBOX_X0      = $0040
16912 :DBOX_XW      = $00c0
16913 :DBOX_X1      = DBOX_X0 +DBOX_XW -1
16914 :DBOX_Y0      = $20
16915 :DBOX_YH      = $70
16916 :DBOX_Y1      = DBOX_Y0 +DBOX_YH -1
16917
16918 ; Größe Dateifenster.
16919 :FBOX_X0      = DBOX_X0 +$08
16920 :FBOX_XW      = $0070
16921 :FBOX_X1      = FBOX_X0 +FBOX_XW -1
16922 :FBOX_Y0      = DBOX_Y0 +$08
16923 :FBOX_YH      = $48
16924 :FBOX_Y1      = FBOX_Y0 +FBOX_YH -1
16925 :ENTRY_YH     = 12
16926 :FBOX_FILES   = FBOX_YH / ENTRY_YH
16927
16928 ; Position Navigation:
16929 :POSNAV_X     = DBOX_X0+$10
16930 :POSNAV_Y     = DBOX_Y1 -$08 -$10 +1
16931 endif
16932
16933             o BASE_MOD_A
16934             n "obj.#01"
16935
16936 ::ModEntry_0   jmp    N7303 ;Dateien retten.
16937 ::ModEntry_1   jmp    N72EC ;Disk aufräumen.
16938 ::ModEntry_2   jmp    N6EB8 ;Disk löschen.
16939
16940 ; 2: Disk löschen.
16941 :N6EB8         jsr    S4E88             ;Hauptverzeichnis.
16942
16943             jsr    S5C89             ;Hole Disk/Fenster.
16944             tax
16945             bne    :err             ; => Fehler, Abbruch.
16946

```



```

        jsr    S59F1                ;Diskname akt.Fens.      16947
                                           16948
        lda    #< V0483 +2          16949
        sta    r6L                  16950
        lda    #> V0483 +2          16951
        sta    r6H                  16952
        jsr    z6EFD                ;Inhalt löschen.        16953
        txa                                ;Fehler/Abbruch?    16954
        beq    :1                    ; => Nein, weiter...      16955
                                           16956
        cmp    #CANCEL_ERR          16957
        bne    :err                  ; => Fehler...          16958
        rts                          ; => Abbruch.            16959
                                           16960
; Zu ROOT wechseln.                  16961
::1      ldx    C042A                ;Oberstes Fenster.      16962
        jsr    S598F                ;Zeiger Ordnerliste.      16963
        jsr    S28A7                ;Ordnerliste löschen.     16964
                                           16965
        ldx    C042A                ;Oberstes Fenster.      16966
        lda    #$00                 16967
        sta    C051B,x              ;Aktive Ordner-Nr.        16968
        sta    C050B,x              ;Fensterposition.        16969
                                           16970
::2      jsr    S46D4                ;Ordner verlassen.      16971
        bcc    :2                    ;Weiter bis ROOT.        16972
                                           16973
; Hinweis:                          16974
; Im Textmodus wird der Bereich für  16975
; VLIR-Module überschrieben!         16976
        jmp    S4296                ;Update akt.Fenster.     16977
                                           16978
::err    jmp    S5CF3                ;Fehler ausgeben.       16979
                                           16980
; Inhalt löschen.                    16981
:z6EFD   lda    #< d6F57            ;Inhalt löschen?        16982
        ldx    #> d6F57            16983
        jsr    S2090                ;DBox ohne Recover.      16984
                                           16985
        ldy    r0L                  16986
        cpy    #YES                 ;Löschen?                16987
        beq    :erase               ; => Ja, weiter...        16988
                                           16989
        ldx    #CANCEL_ERR          16990
        rts                          ;Abbruch...             16991
                                           16992
; BAM löschen/speichern.             16993
::erase  jsr    z75BC                ;BAM löschen.           16994
        jsr    PutDirHead           ;BAM speichern.          16995
        txa                          16996
        bne    :err                  ; => Fehler, Abbruch.    16997
                                           16998
        jsr    z794E                ;Anfang Verzeichnis.     16999
                                           17000
; Verzeichnis lesen/löschen/speichern. 17001

```

```

17002      jsr  EnterTurbo      ;TurboDOS ein.
17003      txa
17004      bne  :err2           ; => Fehler, Abbruch.
17005
17006      jsr  InitForIO        ;I/O aktivieren.
17007
17008      jsr  z74AF             ;r4 = ":diskBlkBuf".
17009  ::loop      jsr  ReadBlock      ;Block lesen.
17010      txa
17011      bne  :err2
17012
17013      lda  #2                ;Einträge löschen.
17014  ::1          tay
17015      cmp  #2                ;Erster Eintrag?
17016      bne  :2                ; => Nein, weiter...
17017
17018      clc                    ;Sonderbehandlung.
17019      adc  #TDFOLDER_POS0 -1
17020
17021  ::2          tax
17022      dex                    ;Zeiger Ordner-Nr.
17023      lda  #NULL
17024      sta  diskBlkBuf,x      ;Ordner-Nr. löschen.
17025      ldx  diskBlkBuf,y
17026      sta  diskBlkBuf,y      ;Dateityp löschen.
17027
17028      txa
17029      and  #FTYPE_MODES
17030      cmp  #FTYPE_DIR        ;Verzeichnis?
17031      bne  :4                ; => Nein, weiter...
17032
17033  ; Eintrag löschen. Damit kann ein
17034  ; Unterverzeichnis nicht wieder
17035  ; zurückgeholt werden (UNDELETE).
17036      tya                    ;Zeiger auf Eintrag.
17037      tax
17038
17039      lda  #30                ;Anzahl Bytes.
17040      sta  r0L
17041
17042      lda  #NULL
17043  ::3          sta  diskBlkBuf,x      ;Eintrag löschen.
17044      inx
17045      dec  r0L
17046      bne  :3
17047
17048  ::4          tya
17049      clc                    ;Nächster Eintrag.
17050      adc  #32                ;Block gelöscht?
17051      bcc  :1                ; => Nein, weiter...
17052
17053      jsr  WriteBlock        ;Block speichern.
17054      txa
17055      bne  :err2
17056

```

```

        lda    diskBlkBuf +1                17057
        sta    r1H                          17058
        lda    diskBlkBuf +0                17059
        sta    r1L                          ;Noch ein Sektor? 17060
        bne    :loop                        ; => Ja, weiter... 17061
                                           17062
::err2      jmp    DoneWithIO                ;I/O abschalten. 17063
                                           17064
; Dialogbox: Inhalt von Disk löschen? 17065
:d6F57      b    %10000001                  17066
                                           17067
        b    DBTXTSTR,$08,$10              17068
        w    T29CF                          17069
                                           17070
        b    DBTXTSTR,$08,$20              17071
        w    :t1                            17072
                                           17073
        b    DBTXTSTR,$08,$2c              17074
        w    T29D3                          17075
        b    DBVARSTR,$12,$2c              17076
        b    r6L                            17077
                                           17078
        b    YES      ,$01,$48              17079
        b    CANCEL   ,$11,$48              17080
                                           17081
        b    NULL                                17082
                                           17083
if LANG = LANG_DE 17084
::t1      b    "Alle Dateien auf Diskette löschen?",0 17085
endif      17086
if LANG = LANG_EN 17087
::t1      b    "Delete all files on disk?",0 17088
endif      17089
                                           17090
; 2: Dateien retten. 17091
:N7303     jsr    i_FillRam                  ;Puffer für 17092
        w    OS_VARS - v7B58                ;Dateinamen löschen. 17093
        w    v7B58                          17094
        b    $00                            17095
                                           17096
; Modul "Datei retten" neu laden/starten. 17097
        lda    #$00                        ;Kein VLIR-Modul mehr 17098
        sta    V0422                        ;im Speicher! 17099
                                           17100
; Vorgaben für Dateisuche setzen. 17101
;        lda    #$00                        17102
        sta    r10L                        ;Nur Ordner-Nr.=0! 17103
        sta    r11H                        ;0=Nichts überlesen. 17104
                                           17105
; Hinweis: 17106
; Auf 1581/Native können hiermit nur 17107
; gelöschte Dateien innerhalb der 17108
; ersten 144 Einträge gefunden werden! 17109
; 17110
; Hinweis: 17111

```

```

17112 ; Der Puffer bietet nur Platz für
17113 ; ca.100 Dateien.
17114         lda    #144                ;Nur 144 Dateien
17115         sta    r11L                ;testen.
17116
17117         lda    #%100000000         ;Gelöschte Dateien,
17118         sta    r12L                ;leere ignorieren.
17119 ;         lda    #%100000000         ;Alle Dateitypen.
17120         sta    r12H
17121
17122         lda    #17                  ;Länge Dateiname.
17123         sta    r13L
17124         lda    #4                  ;Bytes überlesen.
17125         sta    r13H
17126
17127         lda    #< v7B58            ;Puffer für
17128         sta    r3L                ;Dateinamen.
17129         lda    #> v7B58
17130         sta    r3H
17131
17132         jsr    S2603              ;Dateien suchen.
17133
17134         lda    #< v7B58
17135         sta    r3L
17136         lda    #> v7B58
17137         sta    r3H
17138
17139         lda    r14L
17140         sta    v72EB              ;Dateien vorhanden?
17141         bne    :loop              ; => Ja, weiter...
17142
17143         sta    x74C0              ;DBox anpassen.
17144         beq    :cont
17145
17146 ; Dateinamen bereinigen.
17147 ::loop      ldy    #0
17148 ::1         lda    (r3),y
17149         cmp    #$a0              ;Ende Dateiname?
17150         beq    :3                ; => Ja, weiter...
17151
17152         cmp    #$20              ;Zeichen gültig?
17153         bcc    :2
17154         cmp    #$7f
17155         bcc    :4                ; => Ja, weiter...
17156
17157 ::2         lda    #"?"          ;Ungültiges Zeichen.
17158         b $2c
17159 ::3         lda    #NULL          ;$A0 => NULL.
17160         sta    (r3),y
17161
17162 ::4         iny
17163         cpy    #16
17164         bne    :1
17165
17166         lda    #NULL

```

```

        sta    (r3),y                ;Ende Dateiname.          17167
                                                                    17168
        lda    r3L                    17169
        clc                                17170
        adc    #17                    17171
        sta    r3L                    17172
        bcc    :6                     17173
        inc    r3H                    17174
                                                                    17175
::6      dec    r14L                  17176
        bne    :loop                  17177
                                                                    17178
::cont   lda    v72EB                  ;Anzahl Dateien.        17179
        bne    :5                     ; => OK, weiter...      17180
                                                                    17181
        lda    V046E                  ;Datei retten aktiv?    17182
        beq    :init                  ; => Nein, weiter...      17183
                                                                    17184
        dec    V046E                  ;Eine/Alle retten.      17185
                                                                    17186
        jsr    S524C                  ;Dat.-Ausw. aufheben.    17187
                                                                    17188
; Menü starten um Status zu löschen.  17189
;          jsr    S54E3                ;Hauptmenü starten.    17190
                                                                    17191
; Hinweis:                            17192
; Beim kopieren/löschen von Dateien  17193
; wird der Status erst nach dem Update 17194
; des Fensters gelöscht.              17195
; Die Routine kann aber nicht nach dem 17196
; Update über JSR aufgerufen werden, da 17197
; im Textmodus der Modulspeicher zum    17198
; Teil überschrieben wird.            17199
        lda    #> S54E3 -1            ;Hauptmenü starten.    17200
        pha                                17201
        lda    #< S54E3 -1            17202
        pha                                17203
                                                                    17204
; Hinweis:                            17205
; Im Textmodus wird der Bereich für    17206
; VLIR-Module überschrieben!          17207
        jmp    S4296                  ;Update akt.Fenster.    17208
                                                                    17209
::5      lda    V046E                  ;Datei retten aktiv?    17210
        bne    :rescue_all            ; => Ja, fortsetzen.    17211
                                                                    17212
; Auswahlbox anzeigen.                17213
::init   lda    v72EB                  ;Anzahl Dateien.        17214
        sta    V04B7                  ;Anzahl mark.Dateien.   17215
        jsr    S5802                  ;Anzahl anzeigen.       17216
                                                                    17217
if ENABLE_ORIGCOL = FALSE              17218
        lda    COL043C                ;Systemfarbe für       17219
        pha                                ;Dateiauswahlbox    17220
        lda    C_FBoxBack              ;verwenden.           17221

```

```

17222          sta    COL043C
17223 endif
17224
17225          lda    #< d74BF
17226          ldx    #> d74BF
17227          jsr    S2090          ;DBox ohne Recover.
17228
17229 if ENABLE_ORIGCOL = FALSE
17230          pla
17231          sta    COL043C          ;Ffarbe zurücksetzen.
17232 endif
17233
17234          lda    v72EB          ;Anzahl Dateien.
17235          beq    :exit          ; => Keine Dateien.
17236
17237          ldx    r0L          ;Dialogbox-Status.
17238          cpx    #OK          ;Alle Dateien retten?
17239          beq    :rescue_all    ; => Ja, weiter...
17240          cpx    #YES          ;Einzelne Datei?
17241          beq    :rescue_one    ; => Ja, weiter...
17242
17243          lda    v707E          ;Doppelklick?
17244          bne    :rescue_one    ; => Ja, weiter...
17245
17246 ; Abbruch.
17247 ::exit          jsr    S524C          ;Dat.-Ausw. aufheben.
17248          jmp    S54E3          ;Hauptmenü.
17249
17250 ; Alle Dateien retten.
17251 ::rescue_all    ldy    #$00          ;Erste Datei =
17252          sty    v72E9          ;Markierte Datei.
17253
17254          iny
17255          ;Alle retten.
17256          dec    v72EB          ;Anzahl Dateien.
17257
17258          b $2c
17259
17260 ; Einzelne Datei retten.
17261 ::rescue_one    ldy    #$00          ;Eine Datei retten.
17262          sty    V046E          ;Status: Eine/Alle.
17263
17264 ; Zu rettende Datei suchen.
17265 :z73D6          lda    #< v7B58
17266          sta    r6L
17267          lda    #> v7B58
17268          sta    r6H
17269
17270          ldy    v72E9          ;Markierte Datei.
17271          beq    :2
17272
17273 ::1          lda    r6L
17274          clc
17275          adc    #17
17276          sta    r6L

```

	bcc	:0		17277
	inc	r6H		17278
				17279
::0	dey			17280
	bne	:1		17281
				17282
::2	ldy	#16	;Dateiname kopieren.	17283
::3	lda	(r6),y		17284
	sta	v7B0C,y		17285
	dey			17286
	bpl	:3		17287
				17288
	lda	v72EB	;Anzahl Dateien.	17289
	sta	V04B7	;Anzahl mark.Dateien.	17290
				17291
	jsr	S5802	;Anzahl anzeigen.	17292
				17293
; Zeiger auf Verzeichnis-Anfang.				17294
	jsr	Get1stDirEntry		17295
	txa			17296
	bne	:err		17297
				17298
; Dateiname suchen.				17299
::find	ldy	#\$00		17300
	lda	(r5),y	;Datei gelöscht?	17301
	bne	:next	; => Nein, weiter...	17302
				17303
	ldy	#3 +0	;Dateiname	17304
::4	lda	v7B0C -3,y	;vergleichen.	17305
	beq	:found		17306
	cmp	(r5),y		17307
	bne	:next		17308
	iny			17309
	cpy	#3 +16		17310
	bne	:4		17311
	beq	:found		17312
				17313
::next	jsr	GetNxtDirEntry		17314
	txa			17315
	beq	:find		17316
::err	rts			17317
				17318
; Datei im Verzeichnis retten.				17319
; Übergabe: r1 = Tr/Se Verzeichnisblock.				17320
; r5 = Zeiger auf Verzeichnis-Eintrag.				17321
::found	jsr	EnterTurbo	;TurboDOS ein.	17322
	txa			17323
	bne	:err	; => Fehler, Abbruch.	17324
				17325
	jsr	InitForIO	;I/O aktivieren.	17326
				17327
	lda	r1H	;Tr/Se Dir-Block.	17328
	pha			17329
	lda	r1L		17330
	pha			17331

```

17332
17333         lda    r5H                ;Zeiger auf Eintrag.
17334         pha
17335         lda    r5L
17336         pha
17337
17338 ; Hinweis:
17339 ; Hier wird r5 von diskBlkBuf auf
17340 ; fileHeader umgeleitet, da Validate
17341 ; hier den Verzeichnisblock erwartet.
17342         inc    r5H
17343
17344         ldy    #0                ;Dir-Block für
17345 ::11      lda    diskBlkBuf,y      ;Validate kopieren.
17346         sta    fileHeader,y
17347         iny
17348         bne    :11
17349
17350         sec                    ;Status: Rette...
17351         jsr    z7970            ;Datei retten.
17352
17353         ldy    #0                ;Dir-Block wieder
17354 ::12      lda    fileHeader,y      ;zurücksetzen.
17355         sta    diskBlkBuf,y
17356         iny
17357         bne    :12
17358
17359         pla
17360         sta    r5L
17361         pla
17362         sta    r5H                ;Zeiger auf Eintrag.
17363
17364         pla
17365         sta    r1L
17366         pla
17367         sta    r1H                ;Tr/Se Dir-Block.
17368
17369         txa                    ;Fehler?
17370         beq    :undel           ; => Nein, weiter...
17371
17372 ; Fehler bei "Datei retten".
17373         stx    v7140            ;Fehlercode.
17374
17375         ldy    #3
17376         lda    #NULL            ;Dateiname löschen.
17377         sta    (r5),y
17378
17379         jsr    z74AF            ;Dir-Block speichern.
17380         jsr    WriteBlock
17381
17382         jsr    DoneWithIO       ;I/O abschalten.
17383         jsr    PutDirHead
17384
17385 ; Bei "Alle retten" wird kein Fehler angezeigt.
17386         lda    V046E            ;Status: Eine/Alle?

```



```

                                bne    :restart                ; => Alle Dateien.      17387
                                                                17388
                                ldx     v7140                  ;Fehlercode.      17389
                                jsr     S5CF3                  ;Fehler ausgeben.  17390
                                                                17391
; Zurück zur "Datei retten"-Dialogbox.      17392
::reset                                jsr     S54F0            ;Status löschen.  17393
                                                                17394
; Hinweis:                                  17395
; Hier Anzahl an Dateien zurücksetzen,      17396
; da durch Status von "Datei retten"       17397
; die Anzahl verändert wurde.                17398
                                jsr     S524C                  ;Dat.-Ausw. aufheben. 17399
                                                                17400
; "Dateien retten" neu starten.              17401
                                lda     #> S3997 -1           17402
                                pha                                17403
                                lda     #< S3997 -1           17404
                                pha                                17405
;                                            17406
; Hinweis:                                  17407
; Im Textmodus wird der Bereich für        17408
; VLIR-Module überschrieben!               17409
                                jmp     S4296                  ;Update akt.Fenster. 17410
                                                                17411
; Dateityp festlegen.                      17412
::undel                                ldx     #%10000000 ! PRG  17413
                                ldy     #$15                  17414
                                lda     (r5),y                 ;VLIR-Datei?  17415
                                beq     :21                    ; => Nein, weiter... 17416
;                                            17417
                                ldx     #%10000000 ! USR      17418
                                inx                                17419
::21                                    txa                                17420
                                ldy     #$00                  17421
                                sta     (r5),y                 ;Dateityp setzen.  17422
                                                                17423
                                jsr     z74AF                  ;Dir-Block speichern. 17424
                                jsr     WriteBlock              17425
                                                                17426
                                jsr     DoneWithIO              ;I/O abschalten.  17427
                                jsr     PutDirHead              17428
                                                                17429
; Weitere Dateien retten?                  17430
                                lda     V046E                  ;Eine/Alle retten.  17431
                                beq     :reset                  ; => Dialogbox.  17432
                                                                17433
; Nächste Datei retten...                  17434
::restart                                jmp     S3997          ; => Ja, weiter...  17435
                                                                17436
; Zeiger in r4 auf ":diskBlkBuf".          17437
:z74AF                                lda     #< diskBlkBuf     17438
                                sta     r4L                    17439
                                lda     #> diskBlkBuf          17440
                                sta     r4H                    17441
                                rts                                17441

```

```

17442
17443 ; Dialogboxroutine gelöschte Dateien.
17444 :r6F7C      jsr    z72A5          ;Größe Dateifenster.
17445
17446 if ENABLE_ORIGCOL = TRUE
17447             lda    #DBFILE_COL    ;Farbe Dateifenster.
17448 else
17449             lda    C_FBoxFiles     ;Farbe Dateifenster.
17450 endif
17451             jsr    S1561           ;Screencolor setzen.
17452
17453             dec    r2L             ;Rahmen um Datei-
17454             inc    r2H             ;fenster zeichnen.
17455
17456             inc    r4L
17457             dec    r3L
17458
17459             lda    #%11111111
17460             jsr    FrameRectangle
17461
17462             jsr    i_BitmapUp      ;Navigationspfeile.
17463             w NAVICON
17464             b POSNAV_X/8, POSNAV_Y
17465             b NAVICON_X, NAVICON_Y
17466
17467             ldx    #(r4H - r2L)
17468 ::1          lda    :coord,x
17469             sta    r2,x
17470             dex
17471             bpl    :1
17472
17473 if ENABLE_ORIGCOL = TRUE
17474             lda    #NAVBAR_COL
17475 else
17476             lda    C_DBoxDIcon     ;Farbe Navigation.
17477 endif
17478             jsr    S1561           ;Screencolor setzen.
17479
17480             lda    #< r707F
17481             sta    otherPressVec +0
17482             lda    #> r707F
17483             sta    otherPressVec +1
17484
17485             lda    #$00
17486             sta    v72E9           ;Markierte Datei.
17487             sta    v707E           ;Doppelklick.
17488             sta    v72EA           ;Erste Datei / Liste.
17489
17490             jsr    z71F3           ;Update Dateiliste.
17491             jmp    z7292           ;Datei invertieren.
17492
17493 ::coord      b DBOX_Y1 -$08 -$10 +1
17494             b DBOX_Y1 -$08 -$10 +1 +NAVICON_Y -1
17495             w (DBOX_X0+$10)
17496             w (DBOX_X0+$10) +NAVICON_X*8 -1

```

			17497
		; Mausabfrage für Dateifenster.	17498
:r707F	bit	mouseData	17499
	bmi	:exit	17500
			17501
	jsr	z72A5 ;Größe Dateifenster.	17502
			17503
	jsr	IsMseInRegion ;Klick Dateifenster?	17504
	beq	z70E0 ; => Nein, weiter...	17505
			17506
	jsr	z7292 ;Datei invertieren.	17507
			17508
	jsr	z72A5 ;Größe Dateifenster.	17509
			17510
	lda	mouseYPos ;Zeiger auf Eintrag	17511
	sec	;in Liste berechnen.	17512
	sbc	r2L	17513
	sta	r0L	17514
	lda	#\$00	17515
	sta	r0H	17516
	sta	r1H	17517
			17518
	lda	#ENTRY_YH ;12 Pixel Höhe eines	17519
	sta	r1L ;Dateieintrags.	17520
			17521
	ldx	#r0	17522
	ldy	#r1	17523
	jsr	Ddiv	17524
			17525
	lda	r0L ;Datei angeklickt?	17526
	clc		17527
	adc	v72EA ;Erste Datei / Liste.	17528
	cmp	v72EB ;Anzahl Dateien.	17529
	bcc	:1 ; => Ja, weiter...	17530
			17531
	ldx	v72EB ;Anzahl Dateien -1	17532
	dex	; = Letzte Datei.	17533
	txa		17534
::1	sta	v72E9 ;Markierte Datei.	17535
			17536
	jsr	z7292 ;Datei invertieren.	17537
			17538
	lda	#10	17539
	sta	dblClickCount	17540
			17541
	lda	##10000000	17542
	sta	mouseData	17543
::wait	bit	mouseData	17544
	bpl	:dblclk	17545
	lda	dblClickCount	17546
	bne	:wait	17547
::exit	rts		17548
			17549
::dblclk	lda	v72EB ;Anzahl Dateien.	17550
	beq	:exit ; => Keine, Ende...	17551

```

17552
17553             inc    v707E                ;Doppelklick.
17554             jmp    RstrFrmDialogue
17555
17556 ; Klick auf Navbar auswerten.
17557 :z70E0        lda    #$00
17558             sta    v7146                ;Datei/Seite zurück.
17559             sta    v71F2                ;Dauerfunkt. löschen.
17560             sta    v7139                ;Zähler auf Anfang.
17561
17562             ldx    #(r4H - r2L)          ;Koordinaten für
17563 ::1           lda    :coord,x            ;Navbar auf Anfang
17564             sta    r2,x                  ;setzen.
17565             dex
17566             bpl    :1
17567
17568 ::loop        jsr    IsMseInRegion        ;Icon gewählt?
17569             bne    z7128                ; => Ja, weiter...
17570
17571             lda    r4L                    ;Koord. für das
17572             sta    r3L                    ;nächste Icon.
17573             clc
17574             adc    #< 16
17575             sta    r4L
17576             lda    r4H
17577             sta    r3H
17578             adc    #> 16
17579             sta    r4H
17580
17581             inc    r3L                    ;X-Links korrigieren.
17582             bne    :2
17583             inc    r3H
17584
17585 ::2           inc    v7139
17586             lda    v7139
17587             cmp    #6                    ;Alle Icons getestet?
17588             bne    :loop                ; => Nein, weiter...
17589
17590             rts
17591
17592 ::coord       b POSNAV_Y,POSNAV_Y +15
17593             w POSNAV_X,POSNAV_X +15
17594
17595 ; Navbar-Icon-Routine aufrufen.
17596 :z7128        lda    v7139
17597             asl
17598             tay
17599             lda    v713A +0,y
17600             ldx    v713A +1,y
17601             jmp    CallRoutine
17602
17603 ; Tabelle mit Icon-Funktionen.
17604 :v713A        w r719E ;Nächste Datei.
17605             w r7167 ;Vorherige Datei.
17606             w r7151 ;Zum Ende.

```

	w r7147	;Zum Anfang.	17607
	w r7189	;Nächste Seite.	17608
	w r7162	;Vorherige Seite.	17609
			17610
	; Zum Anfang.		17611
:r7147	lda v71F2	;Dauerfunktion?	17612
	bne navFileInvert	; => Ja, weiter...	17613
			17614
	ldx v72EA	;Erste Datei / Liste.	17615
	beq navExit		17616
			17617
	ldx #\$00	;Erste Datei.	17618
	jmp z71AD	;Neue Pos. anzeigen.	17619
			17620
	; Zum Ende.		17621
:r7151	lda v71F2	;Dauerfunktion?	17622
	bne navFileInvert	; => Ja, weiter...	17623
			17624
	lda v72EB	;Anzahl Dateien.	17625
	sec		17626
	sbc #FB0X_FILES		17627
	bcc r7147		17628
			17629
	cmp v72EA	;Erste Datei / Liste.	17630
	beq navExit		17631
			17632
	tax	;Letzte Datei.	17633
	jmp z71AD	;Neue Pos. anzeigen.	17634
			17635
	; Navigation beenden.		17636
:navFileInvert	jmp z7292	;Datei invertieren.	17637
:navExit	rts		17638
			17639
	; Vorherige Seite.		17640
:r7162	lda #\$01		17641
	sta v7146		17642
			17643
	; Vorherige Datei.		17644
:r7167	ldx v72EA	;Erste Datei / Liste.	17645
	bne :1		17646
			17647
	lda v71F2	;Dauerfunktion?	17648
	bne navFileInvert	; => Ja, weiter...	17649
	beq navExit	; => Nein, Ende...	17650
			17651
::1	dex		17652
			17653
	lda v7146	;Seite zurück?	17654
	beq :3	; => Nein, weiter...	17655
			17656
	lda v72EA	;Erste Datei / Liste.	17657
	sec		17658
	sbc #FB0X_FILES	;Neue Seite möglich?	17659
	bcs :2	; => Ja, weiter...	17660
	lda #\$00	;Zum Anfang.	17661

```

17662 ::2          tax
17663 ::3          jmp    z71AD          ;Neue Pos. anzeigen.
17664
17665 ; Nächste Seite.
17666 :r7189         lda    v72EA          ;Erste Datei / Liste.
17667              clc
17668              adc    #FB0X_FILES +FB0X_FILES +1
17669              cmp    v72EB          ;Anzahl Dateien.
17670              bcs    r7151
17671
17672              lda    v72EA          ;Erste Datei / Liste.
17673              clc
17674              adc    #FB0X_FILES
17675              tax
17676              jmp    z71AD          ;Neue Pos. anzeigen.
17677
17678 ; Nächste Datei.
17679 :r719E         ldx    v72EA          ;Erste Datei / Liste.
17680              inx
17681              lda    v72EB          ;Anzahl Dateien.
17682              sec
17683              sbc    v72EA          ;Erste Datei / Liste.
17684              cmp    #FB0X_FILES +1
17685              bcs    z71AD
17686
17687              lda    v71F2          ;Dauerfunktion?
17688              bne    navFileInvert    ; => Ja, weiter...
17689              rts
17690
17691 ; Neue Position anzeigen.
17692 :z71AD         txa
17693              pha
17694
17695              ldy    v71F2          ;Dauerfunktion?
17696              bne    :1              ; => Ja, weiter...
17697              jsr    z7292          ;Datei invertieren.
17698
17699 ::1           pla
17700              sta    v72EA          ;Erste Datei / Liste.
17701              cmp    v72E9          ;Markierte Datei.
17702              bcc    :2
17703              sta    v72E9          ;Markierte Datei.
17704
17705 ::2           clc
17706              adc    #FB0X_FILES -1
17707              cmp    v72E9          ;Markierte Datei.
17708              bcs    :3
17709              sta    v72E9          ;Markierte Datei.
17710
17711 ::3           jsr    z71F3          ;Update Dateiliste.
17712
17713              ldx    #3
17714 ::4           jsr    SCPU_Pause
17715              dex
17716              bne    :4

```

			17717
	lda	mouseData ;Taste gedrückt?	17718
	bmi	:5 ; => Nein, Ende...	17719
			17720
	inc	v71F2 ;Dauerfunktion?	17721
	jmp	z7128 ;Nav. wiederholen.	17722
			17723
::5	jmp	z7292 ;Datei invertieren.	17724
			17725
		; Dateiliste ausgeben.	17726
:z71F3	lda	v72EB ;Dateien gefunden?	17727
	bne	:1 ; => Ja, weiter...	17728
			17729
	lda	#< x74C1 ;Hinweis ausgeben und	17730
	sta	r0L ;Ausgabe beenden.	17731
	lda	#> x74C1	17732
	sta	r0H	17733
	jmp	PutString	17734
			17735
::1	lda	rightMargin +1	17736
	pha		17737
	lda	rightMargin +0	17738
	pha		17739
			17740
	jsr	z72A5 ;Größe Dateifenster.	17741
			17742
	lda	#\$00 ;Füllmuster.	17743
	jsr	S140C ;Rechteck zeichnen.	17744
			17745
	lda	#\$00	17746
	sta	r15L	17747
			17748
	jsr	z72BE ;Koord. mark. Datei.	17749
			17750
	lda	r4L	17751
	sta	rightMargin +0	17752
	lda	r4H	17753
	sta	rightMargin +1	17754
			17755
	ldx	#FBOX_FILES	17756
	lda	#FBOX_FILES	17757
	sec		17758
	sbc	v72EB ;Anzahl Dateien.	17759
	bcc	:2 ; => Mehr Dateien.	17760
	ldx	v72EB ;Anzahl Dateien.	17761
::2	stx	r15H	17762
			17763
	lda	v72EA ;Erste Datei / Liste.	17764
	sta	r0L	17765
			17766
	lda	#17 ;Größe Dateieintrag.	17767
	sta	r1L	17768
			17769
	ldy	#r0	17770
	ldx	#r1	17771

```

17772      jsr    BBMult
17773
17774      lda    r1L      ;Zeiger auf Datei-
17775      clc          ;Liste berechnen.
17776      adc    #< v7B58
17777      sta    r14L
17778      lda    r1H
17779      adc    #> v7B58
17780      sta    r14H
17781
17782  ::loop      lda    r15L
17783      jsr    z72BE      ;Koord. mark. Datei.
17784
17785      lda    r3H      ;X-Koordinate für
17786      sta    r11H      ;Dateiname.
17787      lda    r3L
17788      sta    r11L
17789
17790      lda    r2L      ;Y-Koordinate für
17791      clc          ;Dateiname.
17792      adc    #$07
17793      sta    r1H
17794
17795      lda    r14H      ;Dateiname ausgeben.
17796      sta    r0H
17797      lda    r14L
17798      sta    r0L
17799      jsr    PutString
17800
17801      lda    #$00      ;Highbyte immer 0.
17802      sta    r0H
17803
17804      lda    r15L      ;Datei-Nr.
17805      sec          ;Erste Datei = Nr.1!
17806      adc    v72EA      ;Erste Datei / Liste.
17807      sta    r0L
17808
17809      lda    #< FBOX_X1 -1 -24
17810      sta    r11L
17811      lda    #> FBOX_X1 -1 -24
17812      sta    r11H
17813
17814      lda    #24 ! SET_RIGHTJUST!SET_SUPRESS
17815      jsr    PutDecimal
17816
17817      lda    #17
17818      clc
17819      adc    r14L
17820      sta    r14L
17821      bcc    :3
17822      inc    r14H
17823
17824  ::3      inc    r15L
17825      lda    r15L
17826      cmp    r15H      ;Liste voll?

```



	bne	:loop	; => Nein, weiter...	17827
				17828
::done	pla			17829
	sta	rightMargin +0		17830
	pla			17831
	sta	rightMargin +1		17832
	rts			17833
				17834
; Dateieintrag invertieren.				17835
:z7292	lda	v72EB	;Anzahl Dateien.	17836
	bne	:1	; >0, weiter...	17837
	rts			17838
				17839
::1	lda	v72E9	;Markierte Datei.	17840
	sec			17841
	sbc	v72EA	;Erste Datei / Liste.	17842
	jsr	z72BE	;Koord. mark. Datei.	17843
	jmp	InvertRectangle		17844
				17845
; Größe für Dateifenster festlegen.				17846
:z72A5	ldx	#(r4H - r2L)		17847
::1	lda	:coord,x		17848
	sta	r2,x		17849
	dex			17850
	bpl	:1		17851
	rts			17852
				17853
:coord	b	FBOX_Y0, FBOX_Y1		17854
	w	FBOX_X0, FBOX_X1		17855
				17856
; Koordinaten für markierte Datei berechnen.				17857
:z72BE	sta	r0L	;Eintrag im Fenster.	17858
				17859
	lda	#ENTRY_YH	;12 Pixel Höhe für	17860
	sta	r1L	;Dateieintrag.	17861
				17862
	ldy	#r1		17863
	ldx	#r0		17864
	jsr	BBMult		17865
				17866
	jsr	z72A5	;Größe Dateifenster.	17867
				17868
	lda	r0L		17869
	clc			17870
	adc	r2L		17871
	sta	r2L		17872
	clc			17873
	adc	#ENTRY_YH		17874
	sta	r2H		17875
				17876
	inc	r2L		17877
	dec	r2H		17878
	dec	r2H		17879
				17880
	inc	r3L		17881

```

17882      bne    :1
17883      inc    r3H
17884
17885  ::1      ldx    #r4
17886      jmp    Ddec
17887
17888  :v72EB    b $00    ;Anzahl Dateien.
17889  :v72E9    b $00    ;Markierte Datei.
17890  :v72EA    b $00    ;Erste Datei / Liste.
17891  :v7146    b $00    ;Datei/Seite zurück.
17892  :v707E    b $00    ;Doppelklick-Status.
17893  :v71F2    b $00    ;Dauerfunktion für Navigation.
17894  :v7139    b $00    ;Nr. Navbar-Icon (0-5).
17895  :v7140    b $00    ;Fehlercode.
17896
17897  :v7B0C    s 17     ;Dateiname für "Datei retten".
17898
17899  ; Dialogbox: Auswahl gelöschte Dateien.
17900  :d74BF    b %00000001
17901
17902      b DBOX_Y0,DBOX_Y1
17903      w DBOX_X0,DBOX_X1
17904
17905      b DB_USR_ROUT
17906      w r6F7C
17907
17908      b CANCEL    , $11, $58
17909
17910  if LANG = LANG_DE
17911  :x74C0    b DBTXTSTR, $88, $0a
17912      w :1
17913      b DBTXTSTR, $88, $14
17914      w :3
17915
17916      b DBTXTSTR, $88, $32
17917      w :2
17918      b DBTXTSTR, $88, $3c
17919      w :3
17920  endif
17921  if LANG = LANG_EN
17922  :x74C0    b DBTXTSTR, $88, $0a
17923      w :1
17924      b DBTXTSTR, $88, $14
17925      w :2
17926
17927      b DBTXTSTR, $88, $32
17928      w :1
17929      b DBTXTSTR, $88, $3c
17930      w :3
17931  endif
17932
17933      b YES      , $11, $18
17934      b OK       , $11, $40
17935
17936      b NULL

```

```

17937
if LANG = LANG_DE 17938
::1 b PLAINTEXT 17939
      b "Diese Datei",0 17940
::2 b "Alle Dateien",0 17941
::3 b "retten?",0 17942
endif 17943
if LANG = LANG_EN 17944
::1 b PLAINTEXT 17945
      b "Undelete",0 17946
::2 b "this file?",0 17947
::3 b "all files?",0 17948
endif 17949
17950
:x74C1 b GOTOXY 17951
      w FBOX_X0 +8 17952
      b FBOX_Y0 +8 +8 17953
      b PLAINTEXT,BOLDON 17954
if LANG = LANG_DE 17955
      b "Keine Dateien!" 17956
endif 17957
if LANG = LANG_EN 17958
      b "No files!" 17959
endif 17960
      b PLAINTEXT,0 17961
17962
; Breite/Höhe Navbar-Icon. 17963
:NAVICON_X = 12 17964
:NAVICON_Y = 16 17965
17966
; Navbar-Icon (TopDesk-Version). 17967
if FALSE 17968
:NAVICON 17969
      b $0c 17970
      b $ff 17971
      b $db 17972
      b $80,$01,$00,$01,$00,$01,$00,$01,$00,$01,$00,$01 17973
      b $80,$01,$00,$01,$00,$01,$1f,$e1,$00,$01,$00,$01 17974
      b $83,$01,$03,$01,$03,$01,$00,$01,$06,$c1,$06,$c1 17975
      b $83,$01,$07,$81,$03,$01,$03,$01,$06,$c1,$0e,$e1 17976
      b $83,$01,$0f,$c1,$03,$01,$07,$81,$06,$c1,$1e,$f1 17977
      b $83,$01,$1b,$61,$03,$01,$0f,$c1,$06,$c1,$36,$d9 17978
      b $83,$01,$03,$01,$03,$01,$1b,$61,$06,$c1,$06,$c1 17979
      b $83,$01,$03,$01,$1b,$61,$03 17980
      b $cd 17981
      b $01,$06,$c1,$06,$c1 17982
      b $9b,$61,$03,$01,$0f,$c1,$03,$01,$36,$d9,$06,$c1 17983
      b $8f,$c1,$03,$01,$07,$81,$03,$01,$1e,$f1,$06,$c1 17984
      b $87,$81,$03,$01,$03,$01,$03,$01,$0e,$e1,$06,$c1 17985
      b $83,$01,$03,$01,$00,$01,$03,$01,$06,$c1,$06,$c1 17986
      b $80,$01,$00,$01,$1f,$e1,$00,$01,$00,$01,$00,$01 17987
      b $80,$01,$00,$01,$00,$01,$00,$01,$00,$01,$00,$01 17988
      b $0c 17989
      b $ff 17990
endif 17991

```

```

17992
17993 ; Navbar-Icon (DTopDesk-Version).
17994 :NAVICON
17995 ::nb_00          = %111111111 +%111111111 *256
17996 ::nb_01          = %100000000 +%000000001 *256
17997 ::nb_02          = %100000000 +%000000001 *256
17998 ::nb_03          = %100000000 +%000000001 *256
17999 ::nb_04          = %100000001 +%100000001 *256
18000 ::nb_05          = %100000001 +%100000001 *256
18001 ::nb_06          = %100000001 +%100000001 *256
18002 ::nb_07          = %100000001 +%100000001 *256
18003 ::nb_08          = %100011111 +%111110001 *256
18004 ::nb_09          = %100001111 +%111100001 *256
18005 ::nb_0A          = %100000111 +%111000001 *256
18006 ::nb_0B          = %100000001 +%100000001 *256
18007 ::nb_0C          = %100000000 +%000000001 *256
18008 ::nb_0D          = %100000000 +%000000001 *256
18009 ::nb_0E          = %100000000 +%000000001 *256
18010 ::nb_0F          = %111111111 +%111111111 *256
18011
18012 ::nb_10          = %111111111 +%111111111 *256
18013 ::nb_11          = %100000000 +%000000001 *256
18014 ::nb_12          = %100000000 +%000000001 *256
18015 ::nb_13          = %100000000 +%000000001 *256
18016 ::nb_14          = %100000001 +%100000001 *256
18017 ::nb_15          = %100000111 +%111000001 *256
18018 ::nb_16          = %100001111 +%111100001 *256
18019 ::nb_17          = %100011111 +%111110001 *256
18020 ::nb_18          = %100000001 +%100000001 *256
18021 ::nb_19          = %100000001 +%100000001 *256
18022 ::nb_1A          = %100000001 +%100000001 *256
18023 ::nb_1B          = %100000001 +%100000001 *256
18024 ::nb_1C          = %100000000 +%000000001 *256
18025 ::nb_1D          = %100000000 +%000000001 *256
18026 ::nb_1E          = %100000000 +%000000001 *256
18027 ::nb_1F          = %111111111 +%111111111 *256
18028
18029 ::nb_20          = %111111111 +%111111111 *256
18030 ::nb_21          = %100000000 +%000000001 *256
18031 ::nb_22          = %100000000 +%000000001 *256
18032 ::nb_23          = %100000001 +%100000001 *256
18033 ::nb_24          = %100000001 +%100000001 *256
18034 ::nb_25          = %100000001 +%100000001 *256
18035 ::nb_26          = %100000001 +%100000001 *256
18036 ::nb_27          = %100011111 +%111110001 *256
18037 ::nb_28          = %100001111 +%111100001 *256
18038 ::nb_29          = %100000111 +%111000001 *256
18039 ::nb_2A          = %100000001 +%100000001 *256
18040 ::nb_2B          = %100000000 +%000000001 *256
18041 ::nb_2C          = %100111111 +%111111001 *256
18042 ::nb_2D          = %100000000 +%000000001 *256
18043 ::nb_2E          = %100000000 +%000000001 *256
18044 ::nb_2F          = %111111111 +%111111111 *256
18045
18046 ::nb_30          = %111111111 +%111111111 *256

```

```

::nb_31      = %10000000 +%00000001 *256      18047
::nb_32      = %10000000 +%00000001 *256      18048
::nb_33      = %10011111 +%11111001 *256      18049
::nb_34      = %10000000 +%00000001 *256      18050
::nb_35      = %10000001 +%10000001 *256      18051
::nb_36      = %10000011 +%11000001 *256      18052
::nb_37      = %10000111 +%11100001 *256      18053
::nb_38      = %10001111 +%11110001 *256      18054
::nb_39      = %10000001 +%10000001 *256      18055
::nb_3A      = %10000001 +%10000001 *256      18056
::nb_3B      = %10000001 +%10000001 *256      18057
::nb_3C      = %10000001 +%10000001 *256      18058
::nb_3D      = %10000000 +%00000001 *256      18059
::nb_3E      = %10000000 +%00000001 *256      18060
::nb_3F      = %11111111 +%11111111 *256      18061
                                           18062
::nb_40      = %11111111 +%11111111 *256      18063
::nb_41      = %10000000 +%00000001 *256      18064
::nb_42      = %10000000 +%00000001 *256      18065
::nb_43      = %10011111 +%11111001 *256      18066
::nb_44      = %10010000 +%00001001 *256      18067
::nb_45      = %10010000 +%00001001 *256      18068
::nb_46      = %10010001 +%10001001 *256      18069
::nb_47      = %10010001 +%10001001 *256      18070
::nb_48      = %10010111 +%11101001 *256      18071
::nb_49      = %10010011 +%11001001 *256      18072
::nb_4A      = %10010001 +%10011001 *256      18073
::nb_4B      = %10010000 +%00101001 *256      18074
::nb_4C      = %10011111 +%11111001 *256      18075
::nb_4D      = %10000000 +%00000001 *256      18076
::nb_4E      = %10000000 +%00000001 *256      18077
::nb_4F      = %11111111 +%11111111 *256      18078
                                           18079
::nb_50      = %11111111 +%11111111 *256      18080
::nb_51      = %10000000 +%00000001 *256      18081
::nb_52      = %10000000 +%00000001 *256      18082
::nb_53      = %10011111 +%11111001 *256      18083
::nb_54      = %10010000 +%00001001 *256      18084
::nb_55      = %10010000 +%00001001 *256      18085
::nb_56      = %10010001 +%10001001 *256      18086
::nb_57      = %10010011 +%11001001 *256      18087
::nb_58      = %10010111 +%11101001 *256      18088
::nb_59      = %10010001 +%10001001 *256      18089
::nb_5A      = %10010001 +%10011001 *256      18090
::nb_5B      = %10010000 +%00101001 *256      18091
::nb_5C      = %10011111 +%11111001 *256      18092
::nb_5D      = %10000000 +%00000001 *256      18093
::nb_5E      = %10000000 +%00000001 *256      18094
::nb_5F      = %11111111 +%11111111 *256      18095
                                           18096
; Navbar-Icon-Definition.                  18097
      b $80 +6*2*2                        18098
      w :nb_00,:nb_10,:nb_20              ;Zeile #1. 18099
      w :nb_30,:nb_40,:nb_50              18100
      w :nb_01,:nb_11,:nb_21              ;Zeile #2. 18101

```

```

18102      w :nb_31,:nb_41,:nb_51
18103
18104      b $80 +6*2*2
18105      w :nb_02,:nb_12,:nb_22      ;Zeile #3.
18106      w :nb_32,:nb_42,:nb_52
18107      w :nb_03,:nb_13,:nb_23      ;Zeile #4.
18108      w :nb_33,:nb_43,:nb_53
18109
18110      b $80 +6*2*2
18111      w :nb_04,:nb_14,:nb_24      ;Zeile #5.
18112      w :nb_34,:nb_44,:nb_54
18113      w :nb_05,:nb_15,:nb_25      ;Zeile #6.
18114      w :nb_35,:nb_45,:nb_55
18115
18116      b $80 +6*2*2
18117      w :nb_06,:nb_16,:nb_26      ;Zeile #7.
18118      w :nb_36,:nb_46,:nb_56
18119      w :nb_07,:nb_17,:nb_27      ;Zeile #8.
18120      w :nb_37,:nb_47,:nb_57
18121
18122      b $80 +6*2*2
18123      w :nb_08,:nb_18,:nb_28      ;Zeile #9.
18124      w :nb_38,:nb_48,:nb_58
18125      w :nb_09,:nb_19,:nb_29      ;Zeile #10.
18126      w :nb_39,:nb_49,:nb_59
18127
18128      b $80 +6*2*2
18129      w :nb_0A,:nb_1A,:nb_2A      ;Zeile #11.
18130      w :nb_3A,:nb_4A,:nb_5A
18131      w :nb_0B,:nb_1B,:nb_2B      ;Zeile #12.
18132      w :nb_3B,:nb_4B,:nb_5B
18133
18134      b $80 +6*2*2
18135      w :nb_0C,:nb_1C,:nb_2C      ;Zeile #13.
18136      w :nb_3C,:nb_4C,:nb_5C
18137      w :nb_0D,:nb_1D,:nb_2D      ;Zeile #14.
18138      w :nb_3D,:nb_4D,:nb_5D
18139
18140      b $80 +6*2*2
18141      w :nb_0E,:nb_1E,:nb_2E      ;Zeile #15.
18142      w :nb_3E,:nb_4E,:nb_5E
18143      w :nb_0F,:nb_1F,:nb_2F      ;Zeile #16.
18144      w :nb_3F,:nb_4F,:nb_5F
18145
18146 ; 1: Disk aufräumen.
18147 ; TD13: _Validate
18148 :N72EC      jsr      S5245      ;Dateien invertieren.
18149
18150      jsr      S5C89      ;Hole Disk/Fenster.
18151      tax
18152      bne      :1      ; => Fehler, Abbruch.
18153
18154      jsr      z7537      ;Disk aufräumen.
18155      txa      ;Fehler?
18156      beq      :2      ; => Nein, weiter...

```

```

18157
::1          cpx    #CANCEL_ERR          ;Abbruch?          18158
              beq    :2                    ; => Ja, Ende...    18159
18160
              jsr    S5CF3                  ;Fehler ausgeben.    18161
18162
::2          jsr    S524C                  ;Dat.-Ausw. aufheben. 18163
18164
; Menü starten um Status zu löschen.    18165
;              jsr    S54F0                  ;Hauptmenü starten.    18166
18167
; Hinweis:                              18168
; Beim kopieren/löschen von Dateien      18169
; wird der Status erst nach dem Update    18170
; des Fensters gelöscht.                  18171
; Die Routine kann aber nicht nach dem    18172
; Update über JSR aufgerufen werden, da    18173
; im Textmodus der Modulspeicher zum      18174
; Teil überschrieben wird.                18175
              lda    #> S54F0 -1           ;Hauptmenü starten. 18176
              pha                                18177
              lda    #< S54F0 -1           18178
              pha                                18179
18180
; Hinweis:                              18181
; Im Textmodus wird der Bereich für        18182
; VLIR-Module überschrieben!              18183
              jmp    S4296                  ;Update akt.Fenster. 18184
18185
; Disk ausfräumen.                        18186
; TD13: Validate                          18187
:z7537       jsr    z7541                  ;Datenblöcke belegen. 18188
              txa                                18189
              bne    :err                  ; => Fehler, Abbruch. 18190
18191
              jsr    z7ACD                  ;Boot-Sek. belegen. 18192
;              txa                                18193
;              bne    :err                  ; => Fehler, Abbruch. 18194
18195
::err        rts                            18196
18197
; Datenblöcke belegen.                    18198
:z7541       ldx    curDrive                18199
              lda    RealDrvType -8,x       18200
              and    #ST_DM0DES            18201
              cmp    #DrvPCDOS             18202
              beq    :1                     18203
              cmp    #DrvNative            18204
              bne    :1                     18205
18206
              jsr    OpenRootDir            ;Hauptverzeichnis. 18207
18208
::1          jsr    z78A4                  ; => OpenDisk.      18209
              txa                                18210
              bne    :err                  18211

```

```

18212
18213         jsr    z75BC                ;BAM löschen.
18214         txa
18215         bne     :err
18216
18217 ; Verzeichnis überprüfen.
18218 ::init         lda    #$00                ;Modus:
18219         sta     :modeDiskBB            ;Verzeichnis testen.
18220
18221         jsr     EnterTurbo            ;TurboDOS ein.
18222         txa
18223         bne     :err                ; => Fehler, Abbruch.
18224
18225         jsr     InitForIO            ;I/O aktivieren.
18226
18227         jsr     z794E                ;Anfang Verzeichnis.
18228
18229 ::loop         jsr     z78D8            ;Dir-Block prüfen.
18230         txa
18231         bne     :exit
18232
18233         lda     fileHeader +1
18234         sta     r1H
18235         lda     fileHeader +0
18236         sta     r1L                ;Verzeichnis-Ende?
18237         bne     :loop                ; => Nein, weiter...
18238
18239         bit     :modeDiskBB            ;Border getestet?
18240         bmi     :done                ; => Ja, Ende...
18241
18242         dec     :modeDiskBB
18243
18244         lda     #< curDirHead
18245         sta     r5L
18246         lda     #> curDirHead
18247         sta     r5H
18248         jsr     ChkDkGEOS
18249
18250         bit     isGEOS                ;GEOS-Diskette?
18251         bpl     :done                ; => Nein, weiter...
18252
18253         lda     curDirHead +$ac
18254         sta     r1H
18255         lda     curDirHead +$ab
18256         sta     r1L                ;Borderblock?
18257         bne     :loop                ; => Ja, prüfen...
18258         beq     :done                ; => Nein, Ende...
18259
18260 ::exit         pha
18261         jsr     DoneWithIO            ;I/O abschalten.
18262         pla
18263         tax                ;Fehlerstatus.
18264 ::err         rts
18265
18266 ::done         jsr     DoneWithIO            ;I/O abschalten.

```



	jsr	PutDirHead	;BAM speichern.	18267
				18268
	ldy	curDrive	;Unterverzeichnisse?	18269
	lda	RealDrvMode -8,y		18270
	and	#SET_MODE_SUBDIR		18271
	cmp	#SET_MODE_SUBDIR		18272
	bne	:skip	; => Nein, weiter...	18273
				18274
	jsr	z77BD	;Validate SubDir.	18275
	txa			18276
::skip	rts			18277
				18278
::modeDiskBB	b	\$00		18279
				18280
			; BAM löschen.	18281
			; Löscht alle Blöcke in der BAM	18282
			; Übergabe: curDirHead	18283
			; Rückgabe: X = Fehlernummer.	18284
			; TD13: ClearBAM	18285
:z75BC	lda	curType		18286
	and	#ST_DMODES		18287
				18288
	cmp	#Drv1581		18289
	beq	:x1581		18290
	cmp	#DrvNative		18291
	beq	:native		18292
				18293
::x1541_71	jmp	z764E	;1541/71-BAM löschen.	18294
::native	jmp	z76E2	;Native-BAM löschen.	18295
				18296
			; 1581-BAM löschen.	18297
::x1581	ldy	#16		18298
::11	lda	#40		18299
	sta	dir2Head,y		18300
	sta	dir3Head,y		18301
	iny			18302
	ldx	#4		18303
	lda	#\$ff		18304
::12	sta	dir2Head,y		18305
	sta	dir3Head,y		18306
	iny			18307
	dex			18308
	bpl	:12		18309
	tya			18310
	bne	:11		18311
				18312
	lda	#\$28		18313
	sta	dir2Head +0		18314
	lda	#\$02		18315
	sta	dir2Head +1		18316
	lda	#\$00		18317
	sta	dir3Head +0		18318
	lda	#\$ff		18319
	sta	dir3Head +1		18320
				18321

```

18322          lda    #$25
18323          sta    dir2Head +$fa
18324          lda    #$f8
18325          sta    dir2Head +$fb
18326
18327 ; Verzeichnis in BAM belegen.
18328 :z760C      jsr    z794E          ;Anfang Verzeichnis.
18329
18330          lda    r1L
18331          sta    curDirHead +0
18332          lda    r1H
18333          sta    curDirHead +1
18334
18335          jsr    EnterTurbo        ;TurboDOS ein.
18336          txa
18337          bne    :err            ; => Fehler, Abbruch.
18338
18339          jsr    InitForIO         ;I/O aktivieren.
18340
18341          jsr    z79EB            ;Blöcke belegen.
18342          txa
18343          bne    :exit            ; => Fehler, Abbruch.
18344
18345          lda    #< curDirHead
18346          sta    r5L
18347          lda    #> curDirHead
18348          sta    r5H
18349          jsr    ChkDkGEOS
18350
18351          bit    isGEOS            ;GEOS-Diskette?
18352          bpl    :exit            ; => Nein, weiter...
18353
18354          lda    curDirHead +$ac
18355          sta    r6H
18356          lda    curDirHead +$ab
18357          sta    r6L              ;Borderblock?
18358          beq    :exit            ; => Nein, weiter...
18359
18360          jsr    z7A23            ;Block belegen.
18361
18362 ::exit      jsr    DoneWithIO     ;I/O abschalten.
18363 ::err      rts
18364
18365 ; 1541/71-BAM löschen.
18366 :z764E      pha
18367
18368          lda    #$01
18369          sta    r1L
18370
18371          ldy    #$04
18372 ::21       sty    r0L
18373          jsr    z7A31            ;max.Sek./Track.
18374
18375          ldy    r0L
18376          lda    r1H

```

	sta	curDirHead +0,y	18377
	lda	#\$ff	18378
	sta	curDirHead +1,y	18379
	sta	curDirHead +2,y	18380
			18381
	lda	r1H	18382
	sec		18383
	sbc	#\$10	18384
	tax		18385
	lda	v7AA9 -1,x	18386
	sta	curDirHead +3,y	18387
	iny		18388
	iny		18389
	iny		18390
	iny		18391
	inc	r1L	18392
	cpy	#\$90	18393
	bcc	:21	18394
			18395
	dec	curDirHead +\$48	18396
			18397
	lda	#\$fe	18398
	sta	curDirHead +\$49	18399
			18400
	pla		18401
	cmp	#\$02	18402
	bne	:23	18403
			18404
	lda	curDirHead +\$03	18405
	beq	:23 ; => 1541-BAM...	18406
			18407
	jsr	i_FillRam	18408
	w	105	18409
	w	dir2Head	18410
	b	\$ff	18411
			18412
	jsr	i_FillRam	18413
	w	256 -105	18414
	w	dir2Head +105	18415
	b	\$00	18416
			18417
	lda	#\$24	18418
	sta	r1L	18419
			18420
	lda	#\$02	18421
	sta	r0H	18422
			18423
	ldy	#\$dd	18424
::22	sty	r0L	18425
	jsr	z7A31 ;max.Sek./Track.	18426
			18427
	ldy	r0L	18428
	lda	r1H	18429
	sta	curDirHead,y	18430
			18431

```

18432          lda    r1H
18433          sec
18434          sbc    #$10
18435          tax
18436          lda    v7AA9 -1,x
18437          ldx    r0H
18438          sta    dir2Head,x
18439
18440          lda    r0H
18441          clc
18442          adc    #$03
18443          sta    r0H
18444          inc    r1L
18445          iny
18446          bne    :22
18447
18448          lda    #$00
18449          sta    curDirHead +$ee
18450          sta    dir2Head +$33
18451          sta    dir2Head +$34
18452          sta    dir2Head +$35
18453
18454 ::23          jmp    z760C          ;Verzeichnis belegen.
18455
18456 ; Native-BAM löschen.
18457 :z76E2        jsr    EnterTurbo      ;TurboDOS ein.
18458          txa
18459          bne    :err          ; => Fehler, Abbruch.
18460
18461          jsr    InitForIO          ;I/O aktivieren.
18462
18463 ; Inhalt Native-BAM-Block $01/$02 löschen.
18464          ldy    #$20
18465          lda    #$ff
18466 ::31          sta    dir2Head,y
18467          iny
18468          bne    :31
18469
18470          lda    #$01          ;Zeiger auf ersten
18471          sta    r6L          ;Native-Sektor.
18472          lda    #$00
18473          sta    r6H
18474
18475 ::32          jsr    z7A23          ;Block belegen.
18476
18477          inc    r6H
18478          lda    r6H
18479          cmp    #$20          ;BAM gelöscht?
18480          bcc    :32          ; => Nein, weiter...
18481
18482          jsr    DoneWithIO        ;I/O abschalten.
18483
18484          jsr    PutDirHead        ;BAM speichern.
18485          txa
18486          bne    :err

```

```

18487
;          jsr   EnterTurbo          ;TurboDOS ein.          18488
;          txa                      18489
;          bne   :err                ; => Fehler, Abbruch.    18490
18491
          jsr   InitForIO            ;I/O aktivieren.        18492
18493
; Inhalt Native-BAM-Block $01/$xx löschen.          18494
          ldy   #$00                  18495
          lda   #$ff                  18496
::33      sta   diskBlkBuf,y          18497
          iny                      18498
          bne   :33                  18499
18500
          lda   #$01                  ;Zweiter BAM-Block.    18501
          sta   r1L                  18502
          lda   #$03                  18503
          sta   r1H                  18504
18505
          jsr   z74AF                 ;r4 = ":diskBlkBuf".    18506
::34      jsr   WriteBlock            ;Block speichern.    18507
          txa                      18508
          beq   :35                  ; => OK, weiter...      18509
18510
; Fehler, Abbruch.                                18511
          pha                      18512
          jsr   DoneWithIO           ;I/O abschalten.      18513
          pla                      18514
          tax                      18515
          rts                      18516
18517
::35      inc   r1H                  18518
          lda   r1H                  18519
          cmp   #$22                 ;BAM gelöscht?          18520
          bcc   :34                 ; => Nein, weiter...      18521
18522
          jsr   DoneWithIO           ;I/O abschalten.      18523
18524
          jsr   GetDirHead           ;BAM einlesen.          18525
          txa                      18526
          bne   :err                ; => Fehler, Abbruch.    18527
18528
          lda   #$3f                 18529
          sta   dir2Head +$24        18530
18531
          jmp   z760C                ;Verzeichnis belegen.    18532
18533
; Unterverzeichnisse überprüfen.                    18534
;                                                    18535
; Code stammt aus GeoDOS64.                        18536
:z77BD    jsr   z794E                ;Anfang Verzeichnis.    18537
18538
          lda   #< fileHeader        18539
          sta   r4L                  18540
          lda   #> fileHeader        18541

```

```

18542          sta    r4H
18543
18544 ::loop      jsr    GetBlock          ;Block lesen.
18545          txa
18546          bne     :err
18547
18548          ldy     #2
18549 ::1         lda    fileHeader,y
18550          and     #FTYPE_MODES
18551          cmp     #FTYPE_DIR          ;CMD-SubDir?
18552          beq     :opensub            ; => Ja, weiter...
18553
18554 ::next      tya
18555          clc
18556          adc     #$20
18557          tay
18558          bcc     :1                  ; => Nächste Datei.
18559
18560          lda     fileHeader +1
18561          sta     r1H
18562          lda     fileHeader +0
18563          sta     r1L                  ;Verzeichnis-Ende?
18564          bne     :loop              ; => Nein, weiter..
18565
18566          ldx     curDirHead +$22
18567          ldy     curDirHead +$23
18568          cpx     #$00
18569          bne     :parent
18570          cpy     #$00                ;ROOT-Verzeichnis?
18571          bne     :parent            ; => Nein, weiter...
18572
18573          ldx     #NO_ERROR
18574 ::err       rts
18575
18576 ; Zurück zum übergeordneten Verzeichnis.
18577 ::parent    lda     curDirHead +36    ;Track.
18578          pha
18579          lda     curDirHead +37      ;Sektor.
18580          pha
18581          lda     curDirHead +38      ;Zeiger auf Eintrag.
18582          sta     r8L
18583
18584          txa
18585          pha
18586          tya
18587          pha
18588          jsr     PutDirHead          ;BAM speichern.
18589          pla
18590          sta     r1H
18591          pla
18592          sta     r1L
18593
18594          jsr     OpenSubDir          ;Unterverzeichnis.
18595
18596          pla

```

	sta	r1H		18597
	pla			18598
	sta	r1L		18599
				18600
	txa			18601
	bne	:err	; => Fehler, Abbruch.	18602
				18603
	lda	#< fileHeader		18604
	sta	r4L		18605
	lda	#> fileHeader		18606
	sta	r4H		18607
	jsr	GetBlock	;Parent-Header.	18608
	txa			18609
	bne	:err	; => Fehler, Abbruch.	18610
				18611
	ldy	r8L		18612
	jmp	:next	;Nächster Eintrag.	18613
				18614
; Native-SubDir verarbeiten.				18615
::opensub	tya			18616
	pha			18617
	jsr	PutDirHead		18618
	pla			18619
	tay			18620
				18621
	lda	fileHeader +1,y		18622
	sta	r1L		18623
	lda	fileHeader +2,y		18624
	sta	r1H		18625
				18626
	jsr	OpenSubDir	;Unterverzeichnis.	18627
	txa			18628
	bne	:err	; => Fehler, Abbruch.	18629
				18630
; Hinweis:				18631
; Zuerst das Unterverzeichnis selbst				18632
; überprüfen/validieren.				18633
	jsr	z785D	;Verzeichniszweig.	18634
	txa			18635
	bne	:err	; => Fehler, Abbruch.	18636
				18637
; Hinweis:				18638
; Danach alle Unterverzeichnisse im				18639
; Verzeichnis überprüfen/validieren.				18640
	jmp	z77BD	;Validate SubDir.	18641
				18642
; Verzeichniszweig überprüfen.				18643
:z785D	jsr	EnterTurbo	;TurboDOS ein.	18644
	txa			18645
	bne	:err	; => Fehler, Abbruch.	18646
				18647
	jsr	InitForIO	;I/O aktivieren.	18648
				18649
	jsr	z794E	;Anfang Verzeichnis.	18650
				18651

```

18652 ::loop          jsr    z78D8          ;Dir-Block prüfen.
18653                txa
18654                bne     :done
18655
18656                lda     fileHeader +1
18657                sta     r1H
18658                lda     fileHeader +0
18659                sta     r1L          ;Verzeichnis-Ende?
18660                bne     :loop        ; => Nein, weiter...
18661
18662 ::done          jsr    DoneWithIO        ;I/O abschalten.
18663 ::err          rts
18664
18665 ; Diskette öffnen und Diskname einlesen.
18666 ; Ersatz für OpenDisk ohne CalcBlksFree.
18667 :z78A4          jsr    NewDisk
18668                txa
18669                bne     :err
18670
18671                jsr    GetDirHead
18672                txa
18673                bne     :err
18674
18675                jsr    ChkDkGEOS
18676
18677                ldx     #r1
18678                jsr    GetPtrCurDkNm
18679
18680                lda     #< curDirHead +$90
18681                sta     r0L
18682                lda     #> curDirHead +$90
18683                sta     r0H
18684
18685                ldx     #r0
18686                ldy     #r1
18687                lda     #16
18688                jsr    CopyFString
18689
18690                ldx     #NO_ERROR
18691 ::err          rts
18692
18693 ; Verzeichnisblock überprüfen.
18694 :z78D8          lda     r1H
18695                pha
18696                lda     r1L
18697                pha
18698
18699                lda     #< fileHeader
18700                sta     r4L
18701                lda     #> fileHeader
18702                sta     r4H
18703                jsr    ReadBlock
18704                txa
18705                bne     :cancel
18706

```



	lda	r4L		18707
	clc			18708
	adc	#< 2		18709
	sta	r5L		18710
	lda	r4H		18711
	adc	#> 2		18712
	sta	r5H		18713
				18714
::loop	ldy	#0		18715
	lda	(r5),y		18716
	bmi	:1	; => Closed...	18717
	beq	:next	; => Dat.gelöscht...	18718
				18719
	ora	%%10000000	;Closed-Flag setzen.	18720
	sta	(r5),y		18721
				18722
::1	ldy	#\$16		18723
	lda	(r5),y		18724
	cmp	#TEMPORARY	;SWAPFILE?	18725
	bne	:2	; => Nein, weiter...	18726
				18727
	lda	#\$00	;Datei löschen.	18728
	tay			18729
	sta	(r5),y		18730
	beq	:next	; => Nächste Datei...	18731
				18732
::2	clc		;Status: Überprüfe...	18733
	jsr	z7970	;Datei überprüfen.	18734
	txa			18735
	beq	:next	; => OK, weiter...	18736
				18737
::cancel	pla		;Tr/Se Dir-Block.	18738
	sta	r1L		18739
	pla			18740
	sta	r1H		18741
::err	rts			18742
				18743
::next	lda	r5L	;Zeiger auf	18744
	clc		;nächste Datei...	18745
	adc	#\$20		18746
	sta	r5L	;Block bearbeitet?	18747
	bcc	:loop	; => Nächste Datei...	18748
				18749
	pla		;Tr/Se Dir-Block.	18750
	sta	r1L		18751
	pla			18752
	sta	r1H		18753
				18754
	lda	#< fileHeader		18755
	sta	r4L		18756
	lda	#> fileHeader		18757
	sta	r4H		18758
	jsr	WriteBlock	;Block speichern.	18759
	txa			18760
	bne	:err	; => Fehler, Abbruch.	18761

```

18762
18763             jmp    VerWriteBlock           ;VERIFY...
18764
18765 ; Zeiger auf ersten Verzeichnisblock.
18766 ; Nur Sektor setzen, nicht einlesen.
18767 :z794E             lda    curType
18768                 and     #ST_DMODES
18769                 cmp     #Drv1581
18770                 beq     :x1581
18771                 cmp     #DrvNative
18772                 beq     :native
18773
18774                 lda     #$12                 ;1541/1571.
18775                 ldy     #$01
18776                 bne     :set
18777
18778 ::x1581             lda     #$28                 ;1581.
18779                 ldy     #$03
18780                 bne     :set
18781
18782 ::native             lda     curDirHead +0         ;Native.
18783                 ldy     curDirHead +1
18784
18785 ::set               sta     r1L
18786                 sty     r1H
18787                 rts
18788
18789 ; Datei überprüfen.
18790 ; Übergabe: r5 = Zeiger auf Verzeichnis-Eintrag.
18791 ;             C = 0: Status "Überprüfe Datei"
18792 ;             1: Status "Rette Datei"
18793 ; Rückgabe: X = Fehlernumer.
18794 ; Hinweis : Benötigt EnterTurbo/InitForIO!
18795 ; TD13: ValidateFile
18796 :z7970             jsr     z7A6B                 ;Status anzeigen.
18797
18798                 lda     #$00
18799                 sta     r2L
18800                 sta     r2H
18801
18802                 ldy     #$16
18803                 lda     (r5),y                 ;GEOS-Datei?
18804                 beq     :seq                     ; => Nein, weiter...
18805
18806                 ldy     #$13                 ;Tr/Se Infoblock.
18807                 jsr     z79E1
18808
18809                 jsr     z79EB                 ;Infoblock belegen.
18810                 txa
18811                 bne     :err
18812
18813                 ldy     #$15
18814                 lda     (r5),y                 ;VLIR oder Seq.?
18815                 beq     :seq                     ; => Seq.Datei.
18816

```

```

; VLIR-Datei: 18817
    ldy    #$01                ;Tr/Se VLR-Header. 18818
    jsr    z79E1                18819
                                18820
    jsr    S09C4                ;fileTrScTab > r4. 18821
    jsr    ReadBlock            18822
    txa                                18823
    bne    :err                 18824
                                18825
    ldy    #2                   18826
::loop        lda    fileTrScTab,y 18827
    sta    r1L                  18828
    iny                                18829
    ldx    fileTrScTab,y        18830
    stx    r1H                  18831
                                18832
    cpy    #1                   ;Datensätze belegt? 18833
    beq    :seq                 ; => Ja, weiter... 18834
                                18835
    iny                         18836
                                18837
    lda    r1L                  ;Datensatz vorhanden? 18838
    bne    :1                   ; => Ja, weiter... 18839
    txa                                18840
    bne    :loop                ;Nächster Datensatz. 18841
    beq    :seq                 ;Letzter Datensatz. 18842
                                18843
::1           tya                18844
    pha                18845
    jsr    z79EB            ;Datensatz belegen. 18846
    pla                18847
    tay                18848
                                18849
    txa                18850
    bne    :err           18851
    beq    :loop          18852
                                18853
; Seq.Datei oder VLIR-Header. 18854
::seq        ldy    #$01                ;Tr/Se Daten. 18855
    jsr    z79E1                18856
                                18857
    jsr    z79EB            ;Datei belegen. 18858
    txa                18859
    bne    :err           18860
                                18861
    ldy    #$1c                ;Dateigröße 18862
    lda    r2L                ;aktualisieren. 18863
    sta    (r5),y             18864
    iny                        18865
    lda    r2H                18866
    sta    (r5),y             18867
                                18868
    ldx    #NO_ERROR          18869
::err        rts                18870
                                18871

```

```

18872 ; Tr/Se nach r1L/r1H einlesen.
18873 :z79E1      lda    (r5),y
18874          sta    r1L
18875          iny
18876          lda    (r5),y
18877          sta    r1H
18878          rts
18879
18880 ; Sektorkette in BAM belegen.
18881 :z79EB      lda    r1L
18882          beq     :err
18883
18884          jsr     z74AF          ;r4 = ":diskBlkBuf".
18885 ::loop      jsr     ReadLink
18886          txa
18887          bne     :err
18888
18889          lda    r1L
18890          sta    r6L
18891          lda    r1H
18892          sta    r6H
18893
18894          jsr     z7A23          ;Block belegen.
18895          txa
18896          bne     :err
18897
18898          inc    r2L          ;Blockzähler +1.
18899          bne     :next
18900          inc    r2H
18901
18902 ::next      lda    diskBlkBuf +1
18903          sta    r1H
18904          lda    diskBlkBuf +0
18905          sta    r1L
18906          bne     :loop
18907
18908 ::err      rts
18909
18910 ; Einzelnen Block in BAM belegen.
18911 :z7A23      jsr     DoneWithIO          ;I/O abschalten.
18912
18913          jsr     AllocateBlock
18914
18915          txa
18916          pha
18917          jsr     InitForIO          ;I/O aktivieren.
18918          pla
18919          tax
18920
18921          rts
18922
18923 ; Max.Sektoren/Track einlesen (nur 1541/71).
18924 ; Rückgabe: X = Fehler (wird nicht ausgewertet).
18925 :z7A31      lda    r1L
18926          beq     :err

```

```

18927
18928         lda    curType
18929         and    #ST_DM0DES           ;1541 oder 1571?
18930         cmp    #Drv1571
18931         beq    :x1571               ; => 1571, weiter...
18932
18933 ::x1541         lda    r1L
18934                 cmp    #$24
18935                 bcc    :21
18936
18937 ::err          ldx    #INV_TRACK
18938                 rts
18939
18940 ::x1571         lda    r1L
18941                 cmp    #$47
18942                 bcs    :err
18943
18944 ::21           ldy    #$07
18945 ::22           cmp    v7A9D,y
18946                 bcs    :ok
18947                 dey
18948                 bpl    :22
18949                 bmi    :err
18950
18951 ::ok           tya
18952                 and    #$03
18953                 tay
18954                 lda    v7AA5,y
18955                 sta    r1H
18956
18957                 ldx    #NO_ERROR
18958                 rts
18959
18960 ; 1541/71: Tracks:
18961 :v7A9D         b $01,$12,$19,$1f,$24,$35,$3c,$42
18962
18963 ; 1541/71: Sektoren:
18964 :v7AA5         b $15,$13,$12,$11
18965
18966 ; Tabelle zum belegen von Sektoren in der BAM.
18967 :v7AA9         b %00000001
18968                 b %00000011
18969                 b %00000111
18970                 b %00001111
18971                 b %00011111
18972
18973 ; Status "Überprüfe/Rette Datei" anzeigen.
18974 ; Übergabe: C = 0: Status "Überprüfe Datei"
18975 ;           1: Status "Rette Datei"
18976 :z7A6B         lda    r5H
18977                 pha
18978                 lda    r5L
18979                 pha
18980
18981                 php           ;Status retten.

```

```

18982
18983         jsr    DoneWithIO           ;I/O abschalten.
18984
18985         ldy    #3 +0                ;Dateiname kopieren.
18986 :::1         lda    (r5),y
18987         cmp    #$a0
18988         beq    :2
18989         sta    V0470 -3,y
18990         iny
18991         cpy    #3 +16
18992         bne    :1
18993
18994 :::2         lda    #NULL
18995         sta    V0470 -3,y
18996
18997         ldx    #STAT_VERIFY         ;Status "Überprüfe".
18998         plp
18999         bcc    :3
19000         ldx    #STAT_UNDEL         ;Status "Rette".
19001 :::3         jsr    S55B4           ;Status anzeigen.
19002
19003         jsr    InitForIO           ;I/O aktivieren.
19004
19005         pla
19006         sta    r5L
19007         pla
19008         sta    r5H
19009
19010         rts
19011
19012 ; Hinweis:
19013 ; Routine zum reservieren eines CBM-
19014 ; Boot-Sektors $01/$00.
19015 ;
19016 ; Hinweis:
19017 ; Bei NativeMode unnötig, da in der
19018 ; "leeren" BAM die ersten 32 Blocks
19019 ; als belegt markiert werden.
19020 ;
19021 :z7ACD         ldx    #NO_ERROR
19022
19023         ldy    curDrive
19024         lda    RealDrvType -8,y
19025         and    #ST_DMODES
19026         cmp    #DrvNative           ;NativeMode?
19027         beq    :err                 ; => Ja, Ende...
19028
19029         jsr    GetDirHead
19030         txa
19031         bne    :err                 ; => Fehler, Abbruch.
19032
19033         lda    #$01
19034         sta    r1L
19035         lda    #$00
19036         sta    r1H

```

```

19037
19038      jsr    GetBlock_dskBuf
19039      txa
19040      bne    :err
19041
19042      lda    diskBlkBuf +0          ;Disk mit Bootsek.?
19043      cmp    #"C"
19044      bne    :err
19045      lda    diskBlkBuf +1
19046      cmp    #"B"
19047      bne    :err
19048      lda    diskBlkBuf +2
19049      cmp    #"M"
19050      bne    :err
19051
19052      lda    r1L                    ;Ja, Boot-Sektor
19053      sta    r6L                    ;reservieren.
19054      lda    r1H
19055      sta    r6H
19056      jsr    AllocateBlock
19057      txa
19058      bne    :err
19059
19060      jsr    PutDirHead
19061      rts
19062
19063 ; Zwischenspeicher für Dateiliste.
19064 :v7B58      b $00
19065

```

```

19066
19067 ;
19068 ; SourceCode: src.mod#02.s
19069 ;
19070
19071 ;
19072 ; Modul-Info:
19073 ; 0 - TopDesk-Info
19074 ; 1 - Datei umbenennen
19075 ; 2 - Datei duplizieren
19076 ; 3 - Fortsetzung Datei duplizieren
19077 ;     Rücksprung aus mod.#09:
19078 ;     Verzeichnis duplizieren
19079 ; 4 - GEOS beenden
19080 ; 5 - TopDesk beenden
19081 ;
19082 if .p
19083             t "TopSym"
19084             t "TopSym.MP3"
19085             t "TopSym.ROM"
19086             t "DTopDeskI18N.ext"
19087             t "src.DTopDesk.ext"
19088
19089 ; Größe Exit-Dialogbox.
19090 :DBEXIT_X0      = $0048
19091 :DBEXIT_X1      = $00f7
19092 :DBEXIT_Y0      = $38
19093 :DBEXIT_Y1      = $6f
19094 endif
19095
19096             o BASE_MOD_A
19097             n "obj.#02"
19098
19099 ::ModEntry_0    jmp     N7086 ;TopDesk-Info.
19100 ::ModEntry_1    jmp     N71E0 ;Datei umbenennen.
19101 ::ModEntry_2    jmp     N71D1 ;Datei duplizieren.
19102 ::ModEntry_3    jmp     N7221 ;Fortsetzung: Verz. dupl.
19103 ::ModEntry_4    jmp     N6EBB ;GEOS beenden.
19104 ::ModEntry_5    jmp     N6EB0 ;TopDesk beenden.
19105
19106 ; 5: TopDesk beenden.
19107 :N6EB0
19108 if ENABLE_ORIGCOL = TRUE
19109             lda     COL043C           ;Farbe Dialogbox.
19110             pha
19111             lda     #TDEXIT_COL      ;Farbe für
19112             sta     COL043C           ;Hintergrund.
19113 endif
19114
19115             lda     #< d7010
19116             ldx     #> d7010
19117             jsr     S2090             ;DBox ohne Recover.
19118
19119 if ENABLE_ORIGCOL = TRUE
19120             pla

```



```

                                sta    COL043C                ;Farbe Dialogbox.      19121
endif                                                                    19122
                                                                    19123
                                lda    sysDBData              19124
                                cmp    #YES                   19125
                                beq    :1                     19126
                                rts                            19127
                                                                    19128
::1                            jmp    EnterDeskTop           19129
                                                                    19130
; Dialogbox: DTopDesk beenden.                                          19131
:d7010                          b %00000001                 19132
                                b DBEXIT_Y0,DBEXIT_Y1        19133
                                w DBEXIT_X0,DBEXIT_X1        19134
                                                                    19135
if ENABLE_ORIGCOL = FALSE                                             19136
                                b DB_USR_ROUT                19137
                                w r7069                      19138
endif                                                                    19139
                                                                    19140
                                b DBTXTSTR,$36,$0b           19141
                                w :1                          19142
                                b DBTXTSTR,$08,$19           19143
                                w :2                          19144
                                                                    19145
                                b YES    ,$02,$20            19146
                                b CANCEL,$0e,$20             19147
                                b NULL                        19148
                                                                    19149
if LANG = LANG_DE                                                     19150
::1                            b BOLDON,"Programm beenden",PLAINTEXT,0 19151
::2                            b "Zum System-Desktop zurückkehren?",0    19152
endif                                                                    19153
if LANG = LANG_EN                                                     19154
::1                            b BOLDON,"Exit program",PLAINTEXT,0      19155
::2                            b "Return to the system desktop?",0      19156
endif                                                                    19157
                                                                    19158
; 4: GEOS beenden:                                                    19159
; - Dialogbox öffnen.                                                 19160
; - Bildschirm mit Hinweis füllen.                                     19161
; - Tastaturabfrage über appMain.                                     19162
;                                                                      19163
; TODO:                                                                19164
; GEOS-Reset/appMain statt Dialogbox.                                19165
;                                                                      19166
:N6EBB                                                                  19167
if ENABLE_ORIGCOL = TRUE                                              19168
                                lda    COL043C                ;Farbe Dialogbox.      19169
                                pha                            19170
                                lda    #TDEXIT_COL            ;Farbe für           19171
                                sta    COL043C                ;Hintergrund.         19172
endif                                                                    19173
                                                                    19174
                                lda    #< d7018              19175

```

```

19176          ldx    #> d7018
19177          jsr     S2090                ;DBox ohne Recover.
19178
19179 if ENABLE_ORIGCOL = TRUE
19180          pla
19181          sta     COL043C                ;Farbe Dialogbox.
19182 endif
19183
19184          lda     sysDBData
19185          cmp     #OK
19186          beq     :1
19187          rts
19188
19189 ::1          jmp     z6EF0                ;Zum BASIC wechseln.
19190
19191 ; Dialogbox: GEOS beenden.
19192 :d7018      b %00000001
19193          b DBEXIT_Y0,DBEXIT_Y1
19194          w DBEXIT_X0,DBEXIT_X1
19195
19196 if ENABLE_ORIGCOL = FALSE
19197          b DB_USR_ROUT
19198          w r7069
19199 endif
19200
19201          b DBTXTSTR,$2e,$0b
19202          w :1
19203          b DBTXTSTR,$03,$19
19204          w :2
19205
19206          b OK,$02,$20
19207          b CANCEL,$0e,$20
19208          b NULL
19209
19210 if LANG = LANG_DE
19211 ::1          b BOLDON,"GEOS beenden",PLAINTEXT,0
19212 ::2          b "Hiermit beenden Sie die GEOS-Sitzung.",0
19213 endif
19214 if LANG = LANG_EN
19215 ::1          b BOLDON,"Shutdown GEOS",PLAINTEXT,0
19216 ::2          b "This will end your GEOS session.",0
19217 endif
19218
19219 if ENABLE_ORIGCOL = FALSE
19220 ; Titelzeile einfärben.
19221 :r7069      ldx     #(r4H - r2L)
19222 ::1          lda     :coord,x
19223          sta     r2,x
19224          dex
19225          bpl     :1
19226
19227          lda     #0
19228          jsr     S140C                ;Rechteck zeichnen.
19229
19230          lda     C_DBoxTitel          ;Farbe Infobox.

```

```

19231
19232         ldx    C0438                ;TopDesk in Farbe?
19233         bne    :4                  ; => Ja, weiter...
19234
19235         lda    screencolors        ;Bildschirmfarben
19236         ldy    #4                  ;invertieren.
19237 :::2      clc
19238         tax
19239         bpl    :3
19240         sec
19241 :::3      rol
19242         dey
19243         bne    :2
19244
19245 :::4      jmp    DirectColor        ;Screencolor setzen.
19246
19247 :::coord  b DBEXIT_Y0, DBEXIT_Y0 +15
19248         w DBEXIT_X0, DBEXIT_X1
19249 endif
19250
19251 ; Nach BASIC verlassen.
19252 if ENABLE_BASICINFO = FALSE
19253 :z6EF0    lda    #< :NoBASIC        ;Kein BASIC-Befehl.
19254         sta    r0L
19255         lda    #> :NoBASIC
19256         sta    r0H
19257
19258         lda    #$00                ;Kein Programm laden.
19259         sta    r5L
19260         sta    r5H
19261
19262         sta    $0800                ;Kein PRG starten.
19263         sta    $0801
19264         sta    $0802
19265         sta    $0803
19266
19267         lda    #< $0803            ;Programmende
19268         sta    r7L                  ;definieren.
19269         lda    #> $0803
19270         sta    r7H
19271         jmp    ToBasic              ;Nach BASIC beenden.
19272
19273 :::NoBASIC b NULL
19274 endif
19275
19276 ; BASIC-Meldung ausgeben.
19277 if ENABLE_BASICINFO = TRUE
19278 :z6EF0    lda    #< d6EE6
19279         ldx    #> d6EE6
19280         jmp    S2090                ;DBox ohne Recover.
19281
19282 ; Dialogbox: BASIC-Modus.
19283 :d6EE6    b %00000001
19284
19285         b $00,$05

```

```

19286          w $0000,$000a
19287
19288          b DB_USR_ROUT
19289          w r6EF1
19290
19291          b NULL
19292
19293 ; Bildschirm initialisieren.
19294 :r6EF1      jsr   InitForIO
19295
19296          lda   #END_GEOS_COL      ;Rahmenfarbe setzen.
19297          and   #%00001111
19298          sta   $d020
19299
19300          jsr   DoneWithIO
19301
19302          ldx   #(r4H - r2L)
19303 ::1        lda   V1963,x
19304          sta   windowTop,x
19305          sta   r2,x
19306          dex
19307          bpl   :1
19308
19309          lda   #END_GEOS_COL
19310          jsr   S1561              ;Screencolor setzen.
19311
19312          lda   #$00              ;Füllmuster.
19313          jsr   S140C              ;Rechteck zeichnen.
19314 endif
19315
19316 if ENABLE_BASICINFO!LANG = TRUE!LANG_DE
19317          jsr   i_PutString
19318          w $0018
19319          b $32
19320          b BOLDON,OUTLINEON
19321          b "Sie können den Computer ausschalten",0
19322          jsr   i_PutString
19323          w $003c
19324          b $64
19325          b PLAINTEXT
19326          b "oder drücken Sie die Space-Taste um in den",0
19327          jsr   i_PutString
19328          w $0041
19329          b $72
19330          b PLAINTEXT
19331          b "Basic-Modus Ihres Computers zu gelangen.",0
19332 endif
19333 if ENABLE_BASICINFO!LANG = TRUE!LANG_EN
19334          jsr   i_PutString
19335          w $0024
19336          b $32
19337          b BOLDON,OUTLINEON
19338          b "You can turn off the computer",0
19339          jsr   i_PutString
19340          w $0044

```

```

        b $64                                     19341
        b PLAINTEXT                               19342
        b "or press the space bar to enter the",0 19343
        jsr i_PutString                            19344
        w $0050                                    19345
        b $72                                     19346
        b PLAINTEXT                               19347
        b "BASIC mode of your computer.",0        19348
endif                                             19349
                                                19350
if ENABLE_BASICINFO = TRUE                       19351
    lda #< r6FB6                                  19352
    sta appMain +0                                19353
    lda #> r6FB6                                  19354
    sta appMain +1                                19355
    jmp ClearMouseMode                            19356
                                                19357
; Tastaturabfrage für RESET-Modus.                19358
:r6FB6     lda keyData                            19359
           cmp #" "                                19360
           beq :1                                  19361
           rts                                     19362
                                                19363
::1        lda curDrive                           19364
           jsr SetDevice                           19365
           jsr PurgeTurbo                          19366
                                                19367
           sei                                     19368
           lda CPU_DATA                            19369
           and #%11000000                          19370
           ora #KRNL_BAS_IO_IN                     19371
           sta CPU_DATA                            19372
                                                19373
           lda $e395                                19374
           sta R801C +1                             19375
           lda $e396                                19376
           sta R801C +2                             19377
                                                19378
           ldy #0                                    19379
::2        lda R8000,y                             19380
           sta $8000,y                             19381
           iny                                       19382
           bne :2                                    19383
                                                19384
           jmp ($fffc)                              19385
                                                19386
; RESET-Routine (ab $8000)                        19387
:CBMRESET                                     19388
; ---                                           19389
:R8000     w $8009                                  19390
           w $8009                                  19391
           b $c3,$c2,$cd,$38,$30                  19392
                                                19393
:R8009     sei                                     19394
           ldx #$ff                                19395

```

```

19396          stx    $d016
19397          jsr    IOINIT
19398          jsr    $fd50          ;RAM-Reset.
19399          jsr    $fd15          ;Standard-I/O.
19400          jsr    $ff5b          ;Bildschirm-Reset.
19401          cli
19402 :R801C      jsr    $e453          ;Standard-ROM.
19403 ;          jsr    $e4b7          ;JiffyDOS.
19404          jsr    $e3bf
19405          jsr    $e422
19406          ldx    #$fb
19407          txs
19408          stx    $8005
19409          jmp    $e386
19410 ;---
19411 endif
19412
19413 ; 0: TopDesk-Info.
19414 ;
19415 ; Koordinaten für Infobox:
19416 ;
19417 if PANEL_RIGHT = TRUE
19418 :INFO_X0 = $0008
19419 else
19420 :INFO_X0 = $0028
19421 endif
19422 :INFO_XW = $0100
19423 :INFO_X1 = INFO_X0 +INFO_XW -1
19424 :INFO_Y0 = $20
19425 :INFO_YH = $88
19426 :INFO_Y1 = INFO_Y0 +INFO_YH -1
19427
19428 :N7086
19429 if ENABLE_ORIGCOL = TRUE
19430          lda    COL043C          ;Systemfarbe für
19431          pha          ;Dateiauswahlbox
19432          lda    #TDINFO_COL      ;verwenden.
19433          sta    COL043C
19434 endif
19435
19436          lda    #< d7091
19437          ldx    #> d7091
19438          jsr    S2090          ;DBox ohne Recover.
19439
19440 if ENABLE_ORIGCOL = TRUE
19441          pla
19442          sta    COL043C          ;Farbe Dialogbox.
19443 endif
19444
19445          rts
19446
19447 ; Dialogbox: TopDesk-Info.
19448 :d7091      b %00000001
19449
19450          b INFO_Y0,INFO_Y1      ;Größe.

```

	w INFO_X0, INFO_X1	19451
		19452
	b DBTXTSTR, \$0a, \$10	19453
	w :1	19454
	b DBTXTSTR, \$0a, \$1c	19455
	w :2	19456
		19457
	b DBTXTSTR, \$0a, \$30	19458
	w :3	19459
	b DBTXTSTR, \$0a, \$3c	19460
	w :4	19461
		19462
	b DBTXTSTR, \$0a, \$50	19463
	w :5	19464
	b DBTXTSTR, \$0a, \$5a	19465
	w :6	19466
		19467
	b DBTXTSTR, \$0a, \$70	19468
	w :7	19469
	b DBTXTSTR, \$0a, \$7a	19470
	w :8	19471
		19472
	b DBSYSOPV	19473
	b NULL	19474
		19475
::1	b BOLDON	19476
	b OUTLINEON	19477
	b "D T o p D e s k 6 4"	19478
	b PLAINTEXT	19479
	b " "	19480
	t "opt.DT64.Date"	19481
	b 0	19482
		19483
::2	b "2024"	19484
	b GOTOX	19485
	w INFO_X0 +\$20	19486
	b "by: "	19487
	b BOLDON	19488
	b "Markus Kanet"	19489
	b PLAINTEXT	19490
	b GOTOX	19491
	w INFO_X0 +\$90	19492
if LANG = LANG_DE		19493
	b "(Fehlerkorrekturen)",0	19494
endif		19495
if LANG = LANG_EN		19496
	b "(Bug fixes)",0	19497
endif		19498
		19499
if LANG = LANG_DE		19500
::3	b "Basiert auf TopDesk64:",0	19501
endif		19502
if LANG = LANG_EN		19503
::3	b "Based on TopDesk64:",0	19504
endif		19505

```

19506
19507 ::4          b "2021"
19508          b GOTOX
19509          w INFO_X0 +$20
19510          b "by: "
19511          b BOLDON
19512          b "TD64 - Friends"
19513          b PLAINTEXT
19514          b GOTOX
19515          w INFO_X0 +$90
19516 if LANG = LANG_DE
19517          b "(Erweiterungen)",0
19518 endif
19519 if LANG = LANG_EN
19520          b "(New features)",0
19521 endif
19522
19523 if LANG = LANG_DE
19524 ::5          b "-Spezial Version"
19525          b ", ab MP3-64 3.3 r6 ! !",0
19526
19527 ::6          b "-Auflösung: 320x200"
19528          b ", Farbe mit 8x8 Pixel",0
19529 endif
19530 if LANG = LANG_EN
19531 ::5          b "-Special version"
19532          b ", requires MP3-64 3.3r6!",0
19533
19534 ::6          b "-Resolution: 320x200"
19535          b ", color with 8x8 pixel",0
19536 endif
19537
19538 ::7          b PLAINTEXT
19539          b "V1.2"
19540          b GOTOX
19541          w INFO_X0 +$20
19542          b "(c) 1991 by: "
19543          b "W. Knupe, H.J. Ciprina, V. Goehrke",0
19544
19545 ::8          b PLAINTEXT
19546          b "V4.1"
19547          b GOTOX
19548          w INFO_X0 +$20
19549          b "(c) 2000 by: "
19550          b "MegaCom Software",0
19551
19552 ; 2: Datei duplizieren.
19553 ;
19554 ; *FEHLER*
19555 ; Die Routine hat ein grundsätzliches
19556 ; Problem wenn zwischen zwei markierten
19557 ; Dateien ein gelöschter Dateieintrag
19558 ; liegt: Wird die erste markierte Datei
19559 ; dupliziert, dann wird im nächsten
19560 ; Durchlauf der neue Eintrag als zweite

```



```

; markierte Datei behandelt. 19561
; => Hier wird dann eine andere Datei 19562
; dupliziert als zuvor markiert. 19563
; 19564
; Fix:Nur einzelne Dateien duplizieren. 19565
; (Wird im Hauptmodul getestet) 19566
:N71D1 lda #$00 19567
      sta V04A9 ;Modus: Kopieren. 19568
      19569
; Hinweis: 19570
; SubDir-Info nur nach dem ersten 19571
; Programmstart löschen. 19572
; sta V5596 ;SubDir-Info löschen. 19573
      19574
      lda #$ff ;Modus: Duplizieren. 19575
      b $2c 19576
      19577
; 1: Datei umbenennen. 19578
:N71E0 lda #$00 ;Modus: Umbenennen. 19579
      19580
      ldx V04B7 ;Anzahl mark.Dateien. 19581
      bne :1 ; => OK, weiter... 19582
::exit rts 19583
      19584
::1 sta V71CF ;Umbenenn./Dupliz. 19585
      19586
      ldx C042A ;Oberstes Fenster. 19587
      lda V0513,x ;Anz.Dat. auf Disk. 19588
      sta V7213 ;Aktuelle Dateizahl. 19589
      19590
      lda C04F8,x ;Lfwk. für Fenster. 19591
      sta V0A72 ;Lfwk. Source-Disk. 19592
      sta V0A73 ;Lfwk. Target-Disk. 19593
      stx WIN042F ;Aktives Fenster. 19594
      19595
      jsr S5842 ;mark.Dat. anzeigen. 19596
      19597
      jsr S5C89 ;Hole Disk/Fenster. 19598
      tax 19599
      beq x7214 19600
      19601
      jsr S5245 ;Dateien invertieren. 19602
      cpx #CANCEL_ERR 19603
      beq :exit 19604
      19605
::err jmp S5CF3 ;Fehler ausgeben. 19606
      19607
:x7214 lda #< V634F ;Zeiger Tabelle mit 19608
      sta r2L ;markierten Dateien. 19609
      lda #> V634F 19610
      sta r2H 19611
      19612
      lda #$00 ;$00=Noch kein Name. 19613
      sta V71D0 19614
      19615

```

```

19616 ::loop          lda    V71CF          ;Umbenennen?
19617                beq     :1              ; => Ja, weiter...
19618
19619                ldx     #BFR_OVERFLOW
19620                lda     V7213            ;Dateien im Fenster.
19621                cmp     #MAXFILES        ;Speicher voll?
19622                beq     :err             ; => Ja, Fehler...
19623
19624                inc     V7213            ;Anzahl Dateien +1.
19625
19626 ::1                jsr     S5262          ;Suche mark.Datei.
19627                bcs     :2              ; => Keine Datei...
19628
19629                jsr     S46EB            ;Name mark.Datei.
19630                jsr     z7265            ;Umbenenn./Dupliz.
19631                bcc     :loop            ; => OK, weiter...
19632
19633 ::2                lda     V71D0          ;Name eingegeben?
19634                beq     :exit            ; => Nein, Abbruch...
19635                lda     V71CF          ;Umbenennen?
19636                beq     :3              ; => Ja, weiter...
19637
19638                lda     curType
19639                and     #ST_DM0DES
19640                cmp     #DrvNative        ;Laufwerk = Native?
19641                bne     :3              ; => Nein, weiter...
19642                jmp     S4321            ;Alle Fenster neu.
19643 ::3                jmp     S4296          ;Update akt.Fenster.
19644 ::err              jmp     S5CF3          ;Fehler ausgeben.
19645 ::exit            rts
19646
19647 ; Umbenennen/Duplizieren.
19648 :z7265            jsr     S1412          ;Datei suchen.
19649                txa
19650                beq     :found
19651
19652                jmp     z72DE            ; => Nicht gefunden.
19653
19654 ::found            lda     dirEntryBuf +1 ;Zeiger auf ersten
19655                sta     V6056            ;Datenblock.
19656                lda     dirEntryBuf +2
19657                sta     V6057
19658
19659                lda     V71CF          ;Umbenennen?
19660                beq     z7295            ; => Ja, weiter...
19661
19662 ; Test für "Verzeichnis/Ordner duplizieren".
19663                lda     dirEntryBuf
19664                and     #FTYPE_MODES
19665                cmp     #FTYPE_DIR        ;Verzeichnis?
19666                beq     :subdir          ; => Ja, weiter...
19667
19668                lda     dirEntryBuf +22
19669                cmp     #TDFOLDER        ;TopDesk-Ordner?
19670                bne     z7295            ; => Nein, weiter...

```

```

19671
; *TDFOLDERFIX* 19672
; 19673
; Der VLIR-Test funktioniert hier nicht 19674
; zuverlässig. Ein Ordner hat als Tr/Se 19675
; $00/$ff in Byte #1/2, da ein Ordner 19676
; keine Daten beinhaltet. 19677
        lda    dirEntryBuf +1        ;Daten vorhanden? 19678
        bne    z7295                ; => Kein Ordner. 19679
;        lda    dirEntryBuf +21        19680
;        bne    z7295                ; => VLIR, weiter... 19681
19682
; Ordner kann nicht dupliziert werden. 19683
        ldx    #STRUCT_MISMAT        ;Fehler für einen 19684
        bne    z72DE                ;TD-Ordner anzeigen. 19685
19686
; CMD-Verzeichnis duplizieren. 19687
; 19688
; TODO: 19689
; Wenn CMD-Verzeichnisse nicht kopiert 19690
; werden können sondern nur ein leeres 19691
; Verzeichnis erstellt wird, dann kann 19692
; Duplizieren direkt beendet werden: 19693
; Es ist einfacher Verzeichnisse nicht 19694
; zu kopieren und den Anwender neue 19695
; Verzeichnisse erstellen zu lassen. 19696
::subdir        jmp    S39C5        ;Verz. duplizieren. 19697
19698
; 3: Fortsetzung: Verzeichnis duplizieren. 19699
:N7221        lda    #$01        ;$01=Name eingegeben. 19700
        sta    V71D0        ;=SubDir dupliziert. 19701
19702
        lda    V04AB        19703
        beq    :1        19704
19705
        sec                ; => Abbruch. 19706
        rts                19707
19708
::1        clc                ; => Dupl.fortsetzen. 19709
        rts                19710
19711
; Datei umbenennen/duplizieren. 19712
:z7295        lda    #NULL        ;Dialogbox 19713
        sta    v73A0        ;vorbereiten. 19714
19715
        ldy    #$00        19716
::1        lda    V0470,y        ;Dateiname als 19717
        sta    V0496,y        ;Vorgabe kopieren. 19718
        beq    x72A7        19719
        iny                19720
        bne    :1        19721
19722
; Dialogbox: Name eingeben. 19723
:x72A7        lda    #< V0496        ;Neuer Dateiname. 19724
        sta    a1L        19725

```

```

19726          lda    #> V0496
19727          sta    a1H
19728
19729          lda    #< d7392
19730          ldx    #> d7392
19731          jsr    S2090          ;DBox ohne Recover.
19732
19733          lda    V0496          ;Name eingegeben?
19734          beq    x72A7          ; => Nein, nochmal...
19735
19736          lda    r0L
19737          cmp    #CANCEL          ;Abbruch?
19738          bne    :1          ; => Nein, weiter...
19739
19740 :::exit      clc
19741          rts
19742
19743 :::1         lda    V71CF          ;Umbenennen?
19744          bne    :3          ; => Nein, weiter...
19745
19746          ldy    #0
19747 :::2         lda    V0470,y          ;Dateiname Neu =
19748          cmp    V0496,y          ;Dateiname Alt?
19749          bne    :3          ; => Nein, weiter...
19750          tax
19751          beq    :exit          ; => Ja, Ende...
19752          iny
19753          bne    :2
19754
19755 :::3         lda    #$01          ;$01=Name eingegeben.
19756          sta    V71D0
19757
19758          lda    #< V0496          ;Neuer Dateiname.
19759          sta    r6L
19760          lda    #> V0496
19761          sta    r6H
19762          jsr    FindFile          ;Datei suchen.
19763          txa
19764          bne    :err
19765
19766          lda    #DBTXTSTR          ;Dialogbox anpassen.
19767          sta    v73A0
19768          bne    x72A7          ; => Name eingeben.
19769
19770 :::err      cpx    #FILE_NOT_FOUND
19771          beq    x72EA
19772
19773 :z72DE      jsr    S5CF3          ;Fehler ausgeben.
19774          sec
19775          rts
19776
19777 ; Umbenennen/Duplizieren ausführen.
19778 :x72EA      lda    V71CF          ;Umbenennen?
19779          beq    x732D          ; => Ja, weiter...
19780

```

```

        lda    #< V0470                ;Dateiname.                19781
        sta    r12L                    19782
        lda    #> V0470                19783
        sta    r12H                    19784
                                           19785
        lda    #< V0483 +2              ;Diskname ohne "x:".      19786
        sta    r10L                    19787
        lda    #> V0483 +2              19788
        sta    r10H                    19789
                                           19790
        lda    r10L                    19791
        sta    r11L                    19792
        lda    r10H                    19793
        sta    r11H                    19794
                                           19795
        lda    #< V0496                ;Neuer Dateiname.          19796
        sta    r13L                    19797
        lda    #> V0496                19798
        sta    r13H                    19799
                                           19800
        ldx    WIN042F                  ;Aktives Fenster.          19801
        lda    C051B,x                  ;Aktive Ordner-Nr.          19802
        sta    V0A71                    ;Nr. Ziel-Ordner.          19803
                                           19804
; Hinweis:                               19805
; Hier wird COPYMEM_LOW verändert, aber    19806
; die Routine zum einlesen von Daten      19807
; ändert die Adresse auf COPYMEM_MAX,     19808
; daher macht das hier keinen Sinn.       19809
;                                           19810
; Hier sollte der CopyBuffer begrenzt     19811
; werden um das aktive Modul nicht        19812
; zu überschreiben.                       19813
;                                           19814
; Allerdings sichert die CopyFile-        19815
; Routine das APP_RAM von $1200-$7fff     19816
; (COPYMEM_MAX - COPYMEM_HIGH) in der     19817
; REU -> Routine bleibt erhalten.         19818
:v7300 = $7300                           19819
                                           19820
;--- Kann entfallen:                     19821
;                                           19822
;         lda    MLB04AD                ;HB Anfang CopyBuf.      19822
;         pha                                           19823
;         lda    #> v7300                ;???                     19824
;         sta    MLB04AD                19825
;---                                     19826
                                           19827
        jsr    S0A74                    ;Datei kopieren.          19828
                                           19829
;--- Kann entfallen:                     19830
;                                           19831
;         pla                                           19831
;         sta    MLB04AD                ;HB Anfang CopyBuf.      19832
;---                                     19833
                                           19834
        txa                             19835

```

```

19836          bne    z72DE          ; => Fehler, Abbruch.
19837          clc
19838          rts
19839
19840 ; Datei/Verzeichnis umbenennen.
19841 :x732D      lda    #< V0496          ;Neuer Name.
19842          sta    r0L
19843          lda    #> V0496
19844          sta    r0H
19845
19846          lda    #< V0470          ;Alter Name.
19847          sta    r6L
19848          lda    #> V0470
19849          sta    r6H
19850
19851          jsr    RenameFile          ;Datei umbenennen.
19852          txa
19853          beq    :1
19854
19855 ::err      jmp    z72DE
19856
19857 ::1        lda    dirEntryBuf
19858          and    #FTYPE_MODES
19859          cmp    #FTYPE_DIR          ;Verzeichnis?
19860          bne    :exit          ; => Nein, Ende...
19861
19862          lda    V6056          ;Zeiger Tr/Se auf
19863          sta    r1L          ;Verzeichnisheader
19864          lda    V6057          ;setzen.
19865          sta    r1H
19866
19867 ; Verzeichnisheader einlesen.
19868 ;          lda    #< diskBlkBuf          ;Zeiger auf
19869 ;          sta    r4L          ;Zwischenspeicher.
19870 ;          lda    #> diskBlkBuf
19871 ;          sta    r4H
19872 ;          jsr    GetBlock_dskBuf
19873 ;          txa
19874 ;          bne    :err
19875
19876          ldy    #4          ;Nauen Namen als
19877          ldx    #0          ;Verzeichnisnamen
19878 ::2        lda    V0496,x          ;schreiben.
19879          beq    :3
19880          sta    (r4),y
19881          iny
19882          inx
19883          cpx    #16
19884          bcc    :3
19885          bcs    :5
19886
19887 ::3        lda    #$a0          ;Name mit $A0 auf
19888 ::4        sta    (r4),y          ;16Z. auffüllen.
19889          iny
19890          inx

```

```

                                cpx    #16                                19891
                                bcc     :4                                19892
                                                                    19893
::5                            lda     V6056                        ;Zeiger Tr/Se auf 19894
                                sta     r1L                        ;Verzeichnisheader 19895
                                lda     V6057                        ;setzen.        19896
                                sta     r1H                        19897
                                jsr     PutBlock                    ;dskBlkBuf speichern. 19898
                                txa                                     19899
                                bne     :err                        19900
                                                                    19901
::exit                        clc                                     19902
                                rts                                 19903
                                                                    19904
; Dialogbox: Dateiname eingeben. 19905
:d7392                        b %10000001                        19906
                                                                    19907
                                b CANCEL , $11, $48                19908
                                                                    19909
                                b DBTXTSTR, $08, $10                ;"Neuer Name". 19910
                                w t73A6                            19911
                                                                    19912
                                b DBTXTSTR, $08, $1c                ;">".        19913
                                w T29D5                            19914
                                b DBTXTSTR, $08, $3a                ;"?"..       19915
                                w T29D3                            19916
                                                                    19917
                                b DBTXTSTR, $08, $2e                ;"Alter Name". 19918
                                w t73BD                            19919
                                b DBTXTSTR, $10, $3a                ;Dateiname.   19920
                                w V0470                            19921
                                                                    19922
                                b DBGETSTRING                        ;Name eingeben. 19923
                                b $10, $16                        19924
                                b a1L, 16                          19925
                                                                    19926
; Wird verändert. 19927
; Ist die Datei bereits vorhanden, dann 19928
; wird hier DBTXTSTR eingesetzt und der 19929
; Text "bereits vorhanden" angezeigt. 19930
:v73A0                        b NULL , $08, $4a                    ;"Fehler".    19931
                                w T7464                            19932
                                b DBTXTSTR, $08, $54                ;"Bereits vorhanden". 19933
                                w t73C0                            19934
                                                                    19935
                                b NULL                              19936
                                                                    19937
if LANG = LANG_DE 19938
:t73A6                        b BOLDON                            19939
                                b "Neuen Namen eingeben:", 0      19940
                                                                    19941
:t73BD                        b BOLDON                            19942
                                b "Aktueller Name:"                19943
                                b PLAINTEXT, 0                    19944
:t73C0                        b "Name bereits vergeben!", 0      19945

```

```
19946 endif
19947 if LANG = LANG_EN
19948 :t73A6          b BOLDON
19949                b "Enter new name:",0
19950
19951 :t73BD          b BOLDON
19952                b "Current name:"
19953                b PLAINTEXT,0
19954 :t73C0          b "Name already exists!",0
19955 endif
19956
```



```

19957
; 19958
; SourceCode: src.mod#03.s 19959
; 19960
19961
; 19962
; Modul-Info: 19963
; 0 - DiskImage wechseln 19964
; 1 - Verzeichnis sortieren 19965
; 19966
if .p 19967
    t "TopSym" 19968
    t "TopSym.MP3" 19969
    t "TopSym.ROM" 19970
    t "DTopDeskI18N.ext" 19971
    t "src.DTopDesk.ext" 19972
19973
; Größe Zwischenspeicher für 19974
; DiskImage wechseln: 19975
:MAX_ENTRIES = 90 19976
:SIZE_DATABUF = MAX_ENTRIES *17 19977
endif 19978
19979
    o BASE_MOD_A 19980
    n "obj.#03" 19981
19982
::ModEntry_0 jmp N70DA ;Disk-Img. wechseln. 19983
;:ModEntry_1 jmp N6EB2 ;Verz. sortieren. 19984
19985
; 1: Verzeichnis sortieren. 19986
:N6EB2 lda #$00 ;Kein VLIR-Modul mehr 19987
sta V0422 ;im Speicher! 19988
19989
jsr z6EBD ;Dateien ordnen. 19990
19991
; Hinweis: 19992
; Im Textmodus wird der Bereich für 19993
; VLIR-Module überschrieben! 19994
jmp S4296 ;Update akt.Fenster. 19995
19996
; TD13: Ordnen 19997
;z6EBD jsr S5C89 ;Hole Disk/Fenster. 19998
tax ;Fehlercode -> XReg. 19999
bcs :1 ; => Fehler, Abbruch. 20000
20001
jsr z6ED9 ;Diskette testen. 20002
txa 20003
bne :1 ; => Fehler, Abbruch. 20004
20005
jsr z6EF3 ;Verz. einlesen. 20006
txa 20007
bne :1 ; => Fehler, Abbruch. 20008
20009
jsr z6F66 ;Dateien sortieren. 20010
20011

```

```

20012          ldx    r10H          ;Verz. geändert?
20013          beq    :1            ; => Nein, Ende...
20014
20015          jsr     z6FA5          ;Verz. speichern.
20016
20017 :::1          jsr     S5245      ;Dateien invertieren.
20018
20019          cpx     #CANCEL_ERR    ;Abbruch?
20020          beq     :err           ; => Ja, Ende...
20021
20022          jsr     S5CF3          ;Fehler ausgeben.
20023
20024 :::err        rts
20025
20026 ; Diskette testen.
20027 :z6ED9        ldx     #NO_ERROR
20028          lda     curDirHead +$bd
20029          cmp     #"P"          ;Hauptdiskette?
20030          beq     :1            ; => Ja, Fehler...
20031          cmp     #"B"          ;Systemdiskette?
20032          bne     :exit         ; => Nein, weiter...
20033
20034 :::1          lda     #< d7078  ;Nicht auf Haupt-
20035          ldx     #> d7078      ;oder Systemdiskette.
20036          jsr     S2090        ;DBox ohne Recover.
20037
20038          ldx     #CANCEL_ERR    ;Abbruch.
20039 :::exit        rts
20040
20041 ; Verzeichnis einlesen.
20042 :z6EF3        ldx     C042A      ;Oberstes Fenster.
20043          lda     C051B,x        ;Aktive Ordner-Nr.
20044          sta     v7009          ;Ordner-Nr.
20045
20046          lda     curDirHead +0  ;Erster Verzeichnis-
20047          sta     r1L            ;Track/Sektor.
20048          lda     curDirHead +1
20049          sta     r1H
20050
20051          lda     #< DIRBUF_S    ;Zwischenspeicher.
20052          sta     r4L
20053          lda     #> DIRBUF_S
20054          sta     r4H
20055
20056          lda     #0            ;Sektorzähler.
20057          sta     r10L
20058
20059 ;          lda     #$00          ;Modus:
20060          sta     modeDiskBB     ;Directory sortieren.
20061
20062 :::loop        jsr     z6F5E      ;Block einlesen.
20063          txa
20064          bne     :err
20065
20066          inc     r10L

```

	inc	r10L		20067
				20068
	ldy	r4H		20069
	iny			20070
	cpy	#> DIRBUF_E	;Speicher voll?	20071
	bcc	:1	; => Nein, weiter...	20072
				20073
	ldx	#BFR_OVERFLOW		20074
::err	rts			20075
				20076
::1	sty	r4H	;Pos. nächster Block.	20077
				20078
	lda	r1L	;Weiterer Dir-Block?	20079
	bne	:loop	; => Ja, weiter...	20080
				20081
	bit	modeDiskBB	;Directory/BorderB?	20082
	bmi	:done	; => BorderB, Ende...	20083
				20084
	bit	isGEOS	;GEOS-Diskette?	20085
	bpl	:done	; => Nein, Ende...	20086
				20087
	dec	modeDiskBB	;Modus: BorderBlock.	20088
				20089
	jsr	z6F5F	;Tr/Se BorderBlock.	20090
	clv			20091
	bvc	:loop	; => BorderBlock.	20092
				20093
::done	dec	r10L	;Kein BorderBlock.	20094
	dec	r10L		20095
				20096
	dec	r4H		20097
::exit	rts			20098
				20099
	; Block einlesen und Folgebblock speichern.			20100
:z6F5E	jsr	GetBlock		20101
	txa			20102
	bne	:err		20103
				20104
	ldx	r10L		20105
				20106
	ldy	#0	;Zeiger auf nächsten	20107
	lda	(r4),y	;Block speichern.	20108
	sta	fileTrScTab +0,x		20109
	sta	r1L		20110
	lda	#\$00		20111
	sta	(r4),y		20112
	iny			20113
	lda	(r4),y		20114
	sta	fileTrScTab +1,x		20115
	sta	r1H		20116
				20117
	jsr	z6F60	;Ordner-Nr. anpassen.	20118
				20119
	ldx	#NO_ERROR		20120
::err	rts			20121

```

20122
20123 ; Zeiger auf BorderBlock setzen.
20124 :z6F5F      lda    curDirHead +$ab
20125           sta    r1L
20126           lda    curDirHead +$ac
20127           sta    r1H
20128           rts
20129
20130 ; Ordner-Nr. Quelle anpassen.
20131 :z6F60      ldy    #TDFOLDER_POS0
20132           lda    (r4),y           ;Ordner-Nr. für
20133           tax                      ;ersten Eintrag.
20134           lda    #$00
20135           sta    (r4),y           ;Ordner-Nr. löschen.
20136
20137           ldy    #1               ;Ordner-Nr. erster
20138           txa                      ;Eintrag an temp.
20139           sta    (r4),y           ;Position schreiben.
20140           dey
20141           tya
20142           sta    (r4),y
20143
20144           rts
20145
20146 ; Dateien sortieren.
20147 :z6F66      ldy    #0
20148           sty    r10H
20149
20150 ;           ldy    #0
20151 ::1         lda    V634F,y         ;Tab. mark.Dateien.
20152           cmp    #$ff             ;Ende der Tabelle?
20153           beq    :2               ; => Ja, weiter...
20154           iny
20155           cpy    #MAXFILES
20156           bne    :1
20157
20158           ldy    #0
20159 ::2         cpy    #0               ;Mark.Dat. vorhanden?
20160           beq    :exit            ; => Nein, Ende...
20161
20162           dey
20163
20164 ::loop      lda    V634F,y         ;Tab. mark.Dateien.
20165           sta    r5H               ;Markierte Datei.
20166           lda    #$ff
20167           sta    V634F,y         ;Tab. mark.Dateien.
20168
20169           sty    r5L
20170
20171           lda    r5H               ;Markierte Datei.
20172           jsr    z700A             ;Dateien tauschen.
20173           inc    r10H
20174
20175           ldy    #0
20176 ::3         lda    V634F,y         ;Tab. mark.Dateien

```

	cmp	#\$ff	;korrigieren.	20177
	beq	:5		20178
	cmp	r5H		20179
	bcs	:4		20180
	clc			20181
	adc	#\$01		20182
	sta	V634F,y	;Tab. mark.Dateien.	20183
::4	iny			20184
	cpy	#MAXFILES		20185
	bne	:3		20186
				20187
::5	ldy	r5L		20188
	dey			20189
	cpy	#\$ff		20190
	bne	:loop		20191
				20192
::exit	rts			20193
				20194
				20195
				20195
; Verzeichnis zurückschreiben.				20195
:z6FA5	jsr	z7008	;Ordner-Nr. anpassen.	20196
				20197
	ldx	r10L	;Erster Block?	20198
	beq	:last	; => Ja, weiter...	20199
				20200
; Nächsten Dir-Block speichern.				20201
	jsr	z7007	;Folgeblock setzen.	20202
				20203
	dec	r10L	;Zeiger auf Adresse	20204
	dec	r10L	;Folgeblock.	20205
				20206
	jsr	z7006	;Tr/Se DirBlk setzen.	20207
				20208
	lda	r1L	;Anfang BorderB?	20209
	bne	:1	; => Nein, weiter...	20210
				20211
	jsr	z6F5F	;Tr/Se BorderBlock.	20212
				20213
::1	jsr	PutBlock	;Dir-Block speichern.	20214
	txa			20215
	bne	:err		20216
				20217
	dec	r4H		20218
	bne	z6FA5	; => Nächster Block.	20219
				20220
; Ersten Dir-Block speichern.				20221
::last	jsr	z7007	;Folgeblock setzen.	20222
				20223
	lda	curDirHead +0		20224
	sta	r1L		20225
	lda	curDirHead +1		20226
	sta	r1H		20227
				20228
	jsr	PutBlock		20229
	txa			20230
	bne	:err		20231

```

20232
20233             jsr    S524C                ;Dat.-Ausw. aufheben.
20234
20235 ;             ldx    #NO_ERROR
20236 ::err        rts
20237
20238 :v7009        b $00    ;Ordner-Nr.
20239
20240 ; Tr/Se für aktuellen Block setzen.
20241 :z7006        ldx    r10L
20242             lda    fileTrScTab +0,x
20243             sta    r1L
20244             lda    fileTrScTab +1,x
20245             sta    r1H
20246             rts
20247
20248 ; Folgeblock setzen.
20249 :z7007        ldx    r10L
20250             ldy    #0
20251             lda    fileTrScTab +0,x
20252             sta    (r4),y
20253             iny
20254             lda    fileTrScTab +1,x
20255             sta    (r4),y
20256             rts
20257
20258 ; Ordner-Nr. Ziel anpassen.
20259 :z7008        ldy    #1
20260             lda    (r4),y
20261             ldy    #TDFOLDER_POS0
20262             sta    (r4),y
20263             rts
20264
20265 ; Datei-Nr. A in Ordner sortieren.
20266 :z700A        tax
20267             inx
20268
20269             lda    #< DIRBUF_S
20270             sta    r0L
20271             lda    #> DIRBUF_S
20272             sta    r0H
20273
20274 ::1           ldy    #2
20275             lda    (r0),y                ;Datei gelöscht?
20276             bne    :2                    ; => Nein, weiter...
20277
20278             dey                            ;Datei gelöscht:
20279 ; Hinweis:
20280 ;Akku mit $00 laden, dadurch wir das
20281 ;Z-Flag gesetzt und nach :3 verzweigt.
20282             lda    #0                    ;Datei ohne Ordner.
20283             sta    (r0),y                ;Ordner-Nr. löschen.
20284             beq    :3
20285
20286 ::2           dey

```

	lda	(r0),y	;Datei in aktivem	20287
	cmp	v7009	;Ordner?	20288
	bne	:3		20289
				20290
	dex			20291
	beq	:swap		20292
				20293
::3	lda	#\$20		20294
	clc			20295
	adc	r0L		20296
	sta	r0L		20297
	bcc	:4		20298
	inc	r0H		20299
				20300
::4	clv			20301
	bvc	:1		20302
				20303
::swap	lda	#< diskBlkBuf		20304
	sta	r1L		20305
	lda	#> diskBlkBuf		20306
	sta	r1H		20307
				20308
	lda	#< 32		20309
	sta	r2L		20310
	lda	#> 32		20311
	sta	r2H		20312
				20313
	jsr	MoveData		20314
				20315
	lda	r0L		20316
	sec			20317
	sbc	#< DIRBUF_S		20318
	sta	r2L		20319
	lda	r0H		20320
	sbc	#> DIRBUF_S		20321
	sta	r2H		20322
				20323
	lda	#< DIRBUF_S		20324
	sta	r0L		20325
	clc			20326
	adc	#< 32		20327
	sta	r1L		20328
				20329
	lda	#> DIRBUF_S		20330
	sta	r0H		20331
	adc	#> 32		20332
	sta	r1H		20333
				20334
	jsr	MoveData		20335
				20336
	jsr	i_MoveData		20337
	w	diskBlkBuf		20338
	w	DIRBUF_S		20339
	w	32		20340
				20341

```

20342                rts
20343
20344 ; Modus: Verzeichnis oder BorderBlock/SysDir.
20345 :modeDiskBB      b $00
20346
20347 ; Verzeichnis sortieren überschreibt
20348 ; die Routine zum DiskImage wechseln.
20349 ; Damit können 17 Verzeichnis-Blöcke
20350 ; (136 Dateien) sortiert werden.
20351 :DIRBUF_S
20352 :DIRBUF_E        = diskBlkBuf
20353
20354 ; Dialogbox: System- oder Hauptdiskette.
20355 :d7078           b %10000001
20356
20357                b DBTXTSTR,$10,$10
20358                w :1
20359                b DBTXTSTR,$10,$20
20360                w :2
20361 if LANG = LANG_DE
20362                b DBTXTSTR,$10,$30
20363                w :3
20364 endif
20365
20366                b OK      , $11,$48
20367                b NULL
20368
20369 if LANG = LANG_DE
20370 ::1            b "Diese Operation kann nicht auf",NULL
20371 ::2            b "System- oder Hauptdisketten",NULL
20372 ::3            b "angewendet werden!",NULL
20373 endif
20374 if LANG = LANG_EN
20375 ::1            b "This function can not be used",NULL
20376 ::2            b "on system or boot disks!",NULL
20377 endif
20378
20379 ; 0: DiskImage wechseln.
20380 :N70DA         jsr    S524C                ;Dat.-Ausw. aufheben.
20381
20382                lda    #$00
20383                ldx    WIN042F              ;Aktives Fenster.
20384                sta    C050B,x              ;Fensterposition.
20385
20386                jsr    S55BE                ;Status löschen.
20387
20388                jsr    i_PutString
20389                w STATUS_X0 +2
20390                b STATUS_Y1 -6
20391                b PLAINTEXT
20392 if LANG = LANG_DE
20393                b "Dxx-Treiberwechsel mit ",0
20394 endif
20395 if LANG = LANG_EN
20396                b "To change DXX driver mode use ",0

```



```

endif                                                                    20397
                                                                           20398
        lda    #< v7410          ;"GEOS64.Editor"                      20399
        ldx    #> v7410          20400
                                                                           20401
        bit    sysVersion        ;MP64/GDOS64?                        20402
        bpl    :0                ; => MP64, weiter...                  20403
                                                                           20404
        lda    #< v7411          ;"GD.CONFIG"                          20405
        ldx    #> v7411          20406
                                                                           20407
:::0      sta    r0L              20408
          stx    r0H              20409
          jsr    PutString        ;Konfig. ausgeben.                    20410
                                                                           20411
; CMD-NativeMode?                                                         20412
        lda    curType           20413
        and    #ST_DMODES        20414
        cmp    #DrvNative        ;NativeMode?                        20415
        bne    :1                ; => Nein, weiter...                  20416
        lda    curDirHead +$22    20417
        beq    :1                ; => Kein SubDir.                    20418
                                                                           20419
        jsr    S09AC             ;Fens.dat. löschen.                  20420
                                                                           20421
; Hinweis:                                                                20422
; Im Textmodus wird der Bereich für                                       20423
; VLIR-Module überschrieben!                                             20424
;                                                                           20425
        jsr    S4296             ;Update akt.Fenster.                  20426
                                                                           20427
:::1      lda    #< v7475          20427
          ldx    #> v7475          20428
          jsr    z723D            ;Eine Ebene zurück.                  20429
                                                                           20430
; Auswahlbox anzeigen.                                                    20431
:z7143    lda    #$00              20432
          sta    v7495             ;Kein DiskImage.                    20433
          sta    v74A7             ;Verz.+ROOT+Parent.                 20434
          sta    v74A8             ;Anzahl Dateien.                    20435
                                                                           20436
          ldy    curDrive          ;Laufwerktyp.                       20437
          lda    driveType -8,y    20438
          and    #%0000 0111      20439
          asl                               20440
          tay                               20441
          lda    v744F +0,y        ;Typ DiskImage                     20442
          sta    v745F +5          ;festlegen:                         20443
          lda    v744F +1,y        ;D64/D71/D81/DNP.                  20444
          sta    v745F +6          20445
                                                                           20446
          lda    #< v745F          20447
          ldx    #> v745F          20448
          jsr    z733F            ;Verzeichnis lesen.                  20449
          cpx    #NO_ERROR         20450
          bne    :3                ; => Fehler...                      20451

```

```

20452
20453          sta    v74A8          ;Anzahl Dateien.
20454          cmp    #$00          ;Mind. 1 DiskImage?
20455          beq    :3            ; => Nein, weiter...
20456
20457          ldy    #0              ;Name des ersten
20458 :::1          lda    v74AA,y    ;DiskImage/Disk in
20459          beq    :2            ;CD-Befehl kopieren.
20460          sta    v7495,y
20461          iny
20462          cpy    #16
20463          bcc    :1
20464          lda    #NULL
20465 :::2          sta    v7495,y
20466          iny
20467          iny
20468          iny
20469          sty    v7490          ;Länge CD-Befehl.
20470
20471          jsr    i_MoveData
20472          w v74AA
20473          w v78C8
20474          w SIZE_DATABUF
20475
20476 :::3          lda    #< v7469
20477          ldx    #> v7469
20478          jsr    z733F          ;Verzeichnis lesen.
20479          cpx    #NO_ERROR
20480          bne    :err
20481
20482          sta    v74A9          ;Anz. Verzeichnisse.
20483          sta    v74A7          ;Verz.+ROOT+Parent.
20484          ldy    v74A8          ;Anzahl Dateien.
20485          beq    :9
20486          sty    r13H
20487
20488          ldx    #< v78C8
20489          stx    r14L
20490          ldx    #> v78C8
20491          stx    r14H
20492
20493          tax
20494 :::4          cpx    #MAX_ENTRIES
20495          bcs    :9
20496
20497          ldy    #16 -1
20498 :::5          lda    (r14),y
20499          sta    (r15),y
20500          dey
20501          bpl    :5
20502
20503          lda    r14L
20504          clc
20505          adc    #17
20506          sta    r14L

```

	bcc	:6		20507
	inc	r14H		20508
				20509
::6	lda	r15L		20510
	clc			20511
	adc	#17		20512
	sta	r15L		20513
	bcc	:7		20514
	inc	r15H		20515
				20516
::7	inx			20517
	dec	r13H		20518
	bne	:4		20519
				20520
::9	ldx	#NO_ERROR		20521
::err	txa			20522
	bne	:done		20523
				20524
	jsr	i_MoveData	;Platz für ROOT und	20525
	w	v74AA	;Zurück schaffen.	20526
	w	v74AA +17*2		20527
	w	SIZE_DATABUF		20528
				20529
	jsr	i_MoveData		20530
	w	v741E		20531
	w	v74AA		20532
	w	2*17		20533
				20534
	inc	v74A7	;Verz.+ROOT+Parent.	20535
	inc	v74A7		20536
				20537
	lda	#< v743E	;Zeiger auf Speicher	20538
	sta	r5L	;für gewählte Datei.	20539
	lda	#> v743E		20540
	sta	r5H		20541
				20542
	lda	#NULL	;Rückgabe-String	20543
	sta	v743E	;löschen.	20544
				20545
	lda	#< d7416	;Diaglogbox:	20546
	ldx	#> d7416	;DiskImage wählen.	20547
	jsr	S2090	;DBox ohne Recover.	20548
				20549
	lda	sysDBData		20550
	cmp	#OPEN	;Verzeichnis öffnen?	20551
	bne	:done	; => Nein, weiter...	20552
				20553
	lda	v743E	;DiskImage gewählt?	20554
	bne	z729C	; => Ja, öffnen.	20555
				20556
; Abbruch...				20557
::done	lda	v7495	;DiskImage vorhanden?	20558
	beq	:exit	; => Nein, Ende...	20559
				20560
	lda	#< v7490	;Erstes DiskImage	20561

```

20562          ldx    #> v7490          ;als Vorgabe.
20563          jsr     z723D
20564
20565 ::exit
20566
20567 ; Menü starten um Status zu löschen.
20568 ;          jsr     S54F0          ;Hauptmenü starten.
20569
20570 ; Hinweis:
20571 ; Beim kopieren/löschen von Dateien
20572 ; wird der Status erst nach dem Update
20573 ; des Fensters gelöscht.
20574 ; Die Routine kann aber nicht nach dem
20575 ; Update über JSR aufgerufen werden, da
20576 ; im Textmodus der Modulspeicher zum
20577 ; Teil überschrieben wird.
20578          lda     #> S54F0 -1        ;Hauptmenü starten.
20579          pha
20580          lda     #< S54F0 -1
20581          pha
20582
20583 ; Hinweis:
20584 ; Im Textmodus wird der Bereich für
20585 ; VLIR-Module überschrieben!
20586          jmp     S4296          ;Update akt.Fenster.
20587
20588 ; Befehl an Laufwerk senden.
20589 ; GDS: SendCom
20590 :z723D          sta     r15L
20591          stx     r15H
20592
20593          jsr     PurgeTurbo
20594          jsr     InitForIO
20595          jsr     z724D          ;Befehl senden.
20596          jmp     DoneWithIO
20597
20598 ; Befehl an Laufwerk senden.
20599 ; I/O muss bereits aktiv sein!
20600 :z724D          jsr     UNLSN
20601
20602          jsr     z740C          ;LISTEN senden.
20603          lda     #%111110000 !15    ;Befehlskanal #15.
20604          jsr     SECOND
20605
20606          lda     STATUS          ;Laufwerk vorhanden?
20607          bne     :err          ; => Nein, Abbruch...
20608
20609          ldy     #1          ;Zähler für Anzahl
20610          lda     (r15),y        ;Bytes einlesen.
20611          sta     r14H
20612          dey
20613          lda     (r15),y
20614          sta     r14L
20615
20616          lda     r15L          ;Zeiger auf

```

	clc		;Befehlsdaten.	20617
	adc	#2		20618
	sta	r15L		20619
	bcc	:1		20620
	inc	r15H		20621
	bcs	:1		20622
				20623
::loop	lda	(r15),y		20624
	jsr	CIOUT		20625
	iny			20626
	bne	:1		20627
	inc	r15H		20628
				20629
::1	lda	r14L	;Zähler Anzahl Bytes	20630
	sec		;korrigieren.	20631
	sbc	#< 1	;Bei \$00 Bytes keine	20632
	sta	r14L	;Daten senden.	20633
	lda	r14H		20634
	sbc	#> 1		20635
	sta	r14H		20636
	bcs	:loop		20637
				20638
::done	jsr	UNLSN	;Befehl ausführen.	20639
	ldx	#\$00	;Flag: "Kein Fehler".	20640
	rts			20641
				20642
::err	jsr	UNLSN		20643
	ldx	#\$ff	;Flag: "Fehler!".	20644
	rts			20645
				20646
; Eintrag aus Liste ausgewählt.				20647
:z729C	jsr	z72F9	;Eintrag suchen.	20648
	cpx	#NO_ERROR	;Eintrag gefunden?	20649
	bne	:err	; => Nein, Ende...	20650
				20651
	cmp	#\$00	;ROOT öffnen?	20652
	bne	:1	; => Nein, weiter...	20653
				20654
	lda	#< v7475		20655
	ldx	#> v7475		20656
	jsr	z723D	;DiskImage verlassen.	20657
				20658
	lda	#< v746F	;Zu ROOT wechseln.	20659
	ldx	#> v746F		20660
	bne	:2		20661
				20662
::1	cmp	#1	; "Zurück" gewählt?	20663
	bne	:3	; => Nein, weiter...	20664
				20665
	lda	#< v7475	;DiskImage verlassen.	20666
	ldx	#> v7475		20667
				20668
::2	jsr	z723D	;ROOT/Parent.	20669
::err	jmp	z7143	;Image-Auswahl.	20670
				20671

```

20672 ; Verzeichnis oder DiskImage gewählt.
20673 ::3          pha
20674
20675          ldy    #0          ;Verzeichnis oder
20676          ldx    #0 +3      ;DiskImage öffnen.
20677 ::4          lda    v743E,y  ;Datei-Eintrag in
20678          beq    :5          ;"CD"-Befehl
20679          sta    v747C,x      ;kopieren.
20680          inx
20681          iny
20682          cpy    #16
20683          bne    :4
20684          lda    #NULL
20685 ::5          sta    v747C,x
20686          stx    v747A        ;Länge "CD"-Befehl.
20687
20688          lda    #< v747A
20689          ldx    #> v747A
20690          jsr    z723D        ;"CD"-Befehl.
20691
20692          pla
20693          cmp    v74A7        ;Verzeichnis/Image?
20694          bcc    :err        ; => Verzeichnis.
20695
20696          jsr    OpenDisk     ;Disk öffnen.
20697
20698 ; Menü starten um Status zu löschen.
20699 ;          jsr    S54F0        ;Hauptmenü starten.
20700
20701 ; Hinweis:
20702 ; Beim kopieren/löschen von Dateien
20703 ; wird der Status erst nach dem Update
20704 ; des Fensters gelöscht.
20705 ; Die Routine kann aber nicht nach dem
20706 ; Update über JSR aufgerufen werden, da
20707 ; im Textmodus der Modulspeicher zum
20708 ; Teil überschrieben wird.
20709          lda    #> S54F0 -1   ;Hauptmenü starten.
20710          pha
20711          lda    #< S54F0 -1
20712          pha
20713
20714 ; Hinweis:
20715 ; Im Textmodus wird der Bereich für
20716 ; VLIR-Module überschrieben!
20717          jmp    S4296        ;Update akt.Fenster.
20718
20719 ; Gewählten Eintrag suchen.
20720 :z72F9        lda    v74A8        ;Anzahl Dateien.
20721          clc
20722          adc    v74A9        ;Anz. Verzeichnisse.
20723          clc
20724          adc    #2          ;Max. Anzahl an
20725          sta    r14L        ;Einträgen in Liste.
20726

```

	lda	#0	;Erster Eintrag.	20727
	sta	r14H		20728
				20729
	lda	#< v743E	;Gewählter Eintrag.	20730
	sta	r0L		20731
	lda	#> v743E		20732
	sta	r0H		20733
				20734
	lda	#< v74AA	;Zeiger auf	20735
	sta	r15L	;Dateiliste.	20736
	lda	#> v74AA		20737
	sta	r15H		20738
				20739
::loop	ldx	#r0		20740
	ldy	#r15		20741
	jsr	CmpString	;Eintrag gefunden?	20742
	beq	:exit	; => Ja, Ende...	20743
				20744
	lda	r15L	;Nächster Eintrag.	20745
	clc			20746
	adc	#17		20747
	sta	r15L		20748
	bcc	:1		20749
	inc	r15H		20750
				20751
::1	inc	r14H		20752
	lda	r14H		20753
	cmp	r14L	;Einträge geprüft?	20754
	bcc	:loop	; => Nein, weiter...	20755
				20756
	ldx	#\$ff	;Nicht gefunden.	20757
	rts			20758
				20759
::exit	lda	r14H	;Nr. Datei-Eintrag.	20760
	ldx	#\$00	;Eintrag gefunden.	20761
	rts			20762
				20763
				20764
				20765
				20766
; Aktuelles Verzeichnis einlesen.				20764
; Übergabe: A/X = Zeiger auf Befehl.				20765
; GDS: GetDir_BASIC				20766
:z733F	sta	r15L		20767
	stx	r15H		20768
				20769
	jsr	i_FillRam	;Zwischenspeicher	20770
	w	SIZE_DATABUF	;löschen.	20771
	w	v74AA		20772
	b	\$00		20773
				20774
	jsr	PurgeTurbo		20775
	jsr	InitForIO		20776
				20777
	jsr	z740C	;LISTEN senden.	20778
				20779
	bit	STATUS	;Laufwerk OK?	20780
	bpl	:1	; => Ja, weiter...	20781

```

20782
20783 ::dev_err      ldx  #DEV_NOT_FOUND
20784                jmp  z7401
20785
20786 ::1            lda  #%11110000 ! 0
20787                jsr  SECOND
20788
20789                bit  STATUS                ;Fehler?
20790                bmi  :dev_err              ; => Ja, Ende...
20791
20792                ldy  #0                    ;Befehl senden.
20793 ::2            lda  (r15),y              ;Ende erreicht?
20794                beq  :3                    ; => Ja, weiter...
20795                jsr  CIOUT
20796                iny
20797                bne  :2
20798
20799 ::3            jsr  UNLSN
20800
20801 ; TODO:
20802 ; PHA/PLA ist hier unnötig wenn die
20803 ; Sek.Adresse erst nach TALK in den
20804 ; AKKU geladen wird.
20805                lda  #%11110000 ! 0
20806                pha
20807
20808                lda  #NO_ERROR
20809                sta  STATUS
20810                lda  curDrive              ;Laufwerk auf
20811                jsr  TALK                  ;"Senden" umschalten.
20812
20813                pla
20814                jsr  TKSA
20815
20816                jsr  ACPTR                ;Byte einlesen.
20817
20818                bit  STATUS                ;Laufwerk OK?
20819                bpl  :11                  ; => Ja, weiter...
20820
20821 ; Dateifehler.
20822                ldx  #FILE_NOT_FOUND
20823                jmp  z7401
20824
20825 ; Verzeichnisdaten einlesen.
20826 ::11           ldy  #31                  ;Verzeichnis-Header
20827 ::12           jsr  ACPTR                ;überlesen.
20828                dey
20829                bne  :12
20830
20831                lda  #$00                  ;Anzahl Einträge=0.
20832                sta  v74A6
20833
20834                lda  #< v74AA             ;Zeiger auf
20835                sta  r15L                  ;Datenspeicher.
20836                lda  #> v74AA

```



	sta	r15H		20837
				20838
::loop	jsr	ACPTR		20839
	cmp	#\$00	;Ende erreicht?	20840
	beq	:done	; => Ja, weiter...	20841
				20842
	jsr	ACPTR	;Link-Byte und	20843
	jsr	ACPTR	;Zeilennummer	20844
	jsr	ACPTR	;überlesen.	20845
				20846
::21	jsr	ACPTR		20847
	cmp	#\$00	;Ende Dateiname?	20848
	beq	:25	; => Ja, weiter...	20849
	cmp	#\$22	;Ende Dateiname?	20850
	bne	:21	; => Ja, weiter...	20851
				20852
	ldy	#0		20853
::22	jsr	ACPTR	;Dateiname einlesen.	20854
	cmp	#\$22	;Ende Dateiname?	20855
	beq	:23	; => Ja, weiter...	20856
	sta	(r15),y		20857
	iny			20858
	bne	:22		20859
				20860
::23	lda	r15L	;Zeiger auf nächsten	20861
	clc		;Dateinamen setzen.	20862
	adc	#17		20863
	sta	r15L		20864
	bcc	:24		20865
	inc	r15H		20866
				20867
::24	inc	v74A6	;Anzahl Einträge +1.	20868
	lda	v74A6	;Max. Einträge?	20869
	cmp	#MAX_ENTRIES		20870
	beq	:done	; => Nein, weiter...	20871
				20872
::25	jsr	ACPTR	;Ende der Zeile	20873
	cmp	#\$00	;überlesen.	20874
	bne	:25		20875
	beq	:loop	;Nächster Eintrag...	20876
				20877
; Verzeichnis-Ende.				20878
::done	jsr	UNTALK	;Laufwerk abschalten.	20879
	jsr	z740C	;LISTEN senden.	20880
				20881
	lda	##11100000 ! 0		20882
	jsr	SECOND		20883
	jsr	UNLSN	;Laufwerk abschalten.	20884
				20885
	ldx	#NO_ERROR	;Flag: "Kein Fehler".	20886
				20887
; Verzeichnis abschließen.				20888
:z7401	txa			20889
	pha			20890
	jsr	DoneWithIO		20891

```

20892          pla
20893          tax                      ;Fehlerstatus.
20894
20895          lda    v74A6              ;Anzahl Einträge.
20896          rts
20897
20898 ; LISTEN an aktuelles Laufwerk senden.
20899 :z740C          lda    #$00
20900          sta    STATUS
20901          lda    curDrive
20902          jmp    LISTEN
20903
20904 ; Dialogbox: Partition/DiskImage wechseln.
20905 :d7416          b %10000001
20906
20907          b DBUSRFILES
20908          w v74AA
20909
20910          b OPEN,$00,$00
20911          b CANCEL,$00,$00
20912          b NULL
20913
20914 :v7410          b "GEOS64.Editor!",0
20915 :v7411          b "GD.CONFIG!",0
20916
20917 ; Einträge für Verzeichnis-Navigation.
20918 if LANG = LANG_DE
20919 :v741E          b "<=      (ROOT)",0
20920 :v742E          b "..      (ZURÜCK)",0
20921 endif
20922 if LANG = LANG_EN
20923 :v741E          b "<=      (ROOT)",0
20924 :v742E          b "..      (PARENT)",0
20925 endif
20926
20927 ; Gewählter Eintrag.
20928 :v743E          s 17
20929
20930 ; Befehle für DiskImage-Wechsel.
20931 :v744F          b "??647181NP?????"
20932 :v745F          b "$:*.D??=P",0
20933 :v7469          b "$:*=B",0
20934
20935 ; Befehl: Zu "ROOT" wechseln.
20936 :v746F          w $0004
20937          b "CD//"
20938
20939 ; Befehl: Eine Ebene zurück.
20940 :v7475          w $0003
20941          b "CD",$5f
20942
20943 ; Befehl: Verzeichnis für SD2IEC wechseln.
20944 :v747A          w $0000
20945 :v747C          b "CD:"
20946          s 17

```

		20947
; Befehl: Erstes DiskImage öffnen.		20948
:v7490	w \$0000	20949
	b "CD:"	20950
:v7495	s 17	20951
		20952
:v74A6	b \$00 ;Anzahl Einträge.	20953
:v74A7	b \$00 ;Verzeichnisse+ROOT.	20954
:v74A8	b \$00 ;Anzahl Dateien.	20955
:v74A9	b \$00 ;Anz. Verzeichnisse.	20956
		20957
; Zwischenspeicher.		20958
:v74AA		20959
:v78C8	= v74AA +SIZE_DATABUF	20960
		20961
; Endadresse testen.		20962
	g OS_VARS -2*SIZE_DATABUF	20963
		20964

```

20965
20966 ;
20967 ; SourceCode: src.mod#04.s
20968 ;
20969
20970 ;
20971 ; Modul-Info:
20972 ; 0 - Datei-Info
20973 ; 1 - Disk umbenennen
20974 ;
20975 if .p
20976             t "TopSym"
20977             t "TopSym.MP3"
20978             t "TopSym.ROM"
20979             t "DTopDeskI18N.ext"
20980             t "src.DTopDesk.ext"
20981
20982 ; Autor überprüfen:
20983 ; TRUE  = Autor entspr. GEOS-Dateityp anzeigen.
20984 ; FALSE = Autor anzeigen wenn gültig.
20985 :CHK_AUTHOR      = FALSE
20986
20987 ; Größe Infobox:
20988 :IBOX_Y0          = $18
20989 :IBOX_YH          = $a0
20990 :IBOX_Y1          = IBOX_Y0 +IBOX_YH -1
20991 :IBOX_X0          = $0048
20992 :IBOX_XW          = $00b8
20993 :IBOX_X1          = IBOX_X0 +IBOX_XW -1
20994
20995 ; Größe Titelzeile:
20996 :IBOXT_Y0          = IBOX_Y0
20997 :IBOXT_YH          = $10
20998 :IBOXT_Y1          = IBOXT_Y0 +IBOXT_YH -1
20999 :IBOXT_X0          = IBOX_X0
21000 :IBOXT_XW          = IBOX_XW -16
21001 :IBOXT_X1          = IBOXT_X0 +IBOXT_XW -1
21002 :IBOXT_CX          = IBOXT_XW /2 -4
21003
21004 ; Position CLOSE:
21005 :CICON_XW          = 2
21006 :CICON_X0          = IBOX_XW/8 -CICON_XW
21007 :CICON_YH          = 16
21008 :CICON_Y0          = IBOXT_Y0 -IBOX_Y0
21009
21010 ; X-Position für Datenausgabe:
21011 :IB_DATA_T0          = $0009
21012 :IB_DATA_T1          = $003d
21013 :IB_DATA_X0          = IBOX_X0 +IB_DATA_T1
21014 ; Y-Position für Titelzeile:
21015 :IB_DATA_Y0          = (IBOXT_Y0 -IBOX_Y0) +11
21016 ; Y-Position für Datenausgabe:
21017 :IB_DATA_Y1          = $19      ;GEOS-Klasse.
21018 :IB_DATA_Y2          = $23      ;GEOS-Dateityp.
21019 :IB_DATA_Y3          = $2d      ;Autor.

```

```

:IB_DATA_Y4      = $37      ;Dateistruktur.      21020
:IB_DATA_Y5      = $41      ;Datei/Datum.        21021
:IB_DATA_Y6      = $4b      ;Datei/Größe.        21022
                                           21023
:WPROT_Y0        = IB0X_Y0 +$50                    21024
:WPROT_YH        = $08                                21025
:WPROT_Y1        = WPROT_Y0 +WPROT_YH -1          21026
:WPROT_X0        = $0078                            21027
:WPROT_XW        = $0008                            21028
:WPROT_X1        = WPROT_X0 +WPROT_XW -1          21029
                                           21030
:ITEXT_Y0        = IB0X_Y0 +$60                    21031
:ITEXT_YH        = $38                                21032
:ITEXT_Y1        = ITEXT_Y0 +ITEXT_YH -1          21033
:ITEXT_X0        = $0050                            21034
:ITEXT_XW        = $00a8                            21035
:ITEXT_X1        = ITEXT_X0 +ITEXT_XW -1          21036
:ITEXT_OFF       = 2                                21037
                                           21038
:INFOTEXT        = fileHeader +160                 21039
:MAXLENGTH       = $60                                21040
:LINEHEIGHT      = 10                                21041
endif                                                    21042
                                           21043
o BASE_MOD_A                                           21044
n "obj.#04"                                           21045
                                           21046
::ModEntry_0      jmp      N6EB5      ;Datei-Info.    21047
;:ModEntry_1      jmp      N7773      ;Disk umbenennen. 21048
                                           21049
; 1: Disk umbenennen.                                21050
:N7773            jsr      S4E88                ;Hauptverzeichnis. 21051
                                           21052
                        jsr      S46EB                ;Name mark.Datei. 21053
                                           21054
                        ldx      C042A                ;Oberstes Fenster. 21055
                        jsr      S5A3B                ;Gleiche Fens.suchen. 21056
                                           21057
                        ldx      C042A                ;Oberstes Fenster. 21058
                        lda      #1                    21059
                        sta      a6,x                21060
                                           21061
                        jsr      S59F1                ;Diskname akt.Fens. 21062
                                           21063
                        lda      #< V0483 +2          21064
                        sta      r6L                21065
                        lda      #> V0483 +2          21066
                        sta      r6H                21067
                        jsr      z77E5                ;Diskname eingeben. 21068
                        txa                            21069
                        beq      :1                    ; => OK, weiter... 21070
                                           21071
                        jmp      S5CF3                ;Fehler ausgeben. 21072
                                           21073
::1              sta      a5H                    21074

```

```

21075
21076         ldx    #4 -1
21077 ::2       stx    a5L
21078         lda    a6,x           ;Fenster gefunden?
21079         beq    :7           ; => Nein, weiter...
21080         inc    a5H
21081         txa
21082         jsr    S5979         ;Zeiger Fenstertitel.
21083
21084         lda    r1L           ;Zeiger hinter
21085         clc                ;Laufwerk "X:".
21086         adc    #2
21087         sta    r1L
21088         bcc    :4
21089         inc    r1H
21090
21091 ::4       ldy    #0
21092 ::5       lda    (r6),y
21093         beq    :6
21094         sta    (r1),y
21095         iny
21096         bne    :5
21097
21098 ::6       cpy    #16
21099         beq    :7
21100         lda    #TDSPLIT
21101         sta    (r1),y
21102         iny
21103         bne    :6
21104
21105 ::7       ldx    a5L
21106         dex
21107         bpl    :2
21108
21109         ldx    a5H
21110         dex
21111         bne    :8
21112
21113         ldx    C042A         ;Oberstes Fenster.
21114         inc    V18AD         ;Fens.-Aufbau-Status.
21115         jmp    S141D         ;Fenster anzeigen.
21116
21117 ::8       jmp    S157E         ;Alle Fens. anzeigen.
21118
21119 ; Neuen Disknamen eingeben.
21120 :z77E5    ldy    #$00
21121 ::1       lda    (r6),y
21122         sta    dataDiskName,y
21123         beq    :2
21124         iny
21125         bne    :1
21126
21127 ::2       lda    r6H
21128         sta    a1H
21129         pha

```

	lda	r6L	21130
	sta	a1L	21131
	pha		21132
			21133
	lda	#< dataDiskName	21134
	sta	r5L	21135
	lda	#> dataDiskName	21136
	sta	r5H	21137
			21138
	lda	#< d784F	21139
	ldx	#> d784F	21140
	jsr	S2090 ;DBox ohne Recover.	21141
			21142
	pla		21143
	sta	r6L	21144
	pla		21145
	sta	r6H	21146
			21147
	ldy	r0L	21148
	cpy	#CANCEL	21149
	beq	:cancel	21150
			21151
	lda	dataDiskName ;Name eingegeben?	21152
	bne	:11 ; => Ja, weiter...	21153
			21154
::cancel	ldx	#CANCEL_ERR ;Abbruch...	21155
::err	rts		21156
			21157
::11	jsr	GetDirHead	21158
	txa		21159
	bne	:err	21160
			21161
	ldy	#16 -1	21162
::12	lda	#\$a0	21163
	sta	curDirHead +\$90,y	21164
	dey		21165
	bpl	:12	21166
			21167
	ldy	#0	21168
::13	lda	dataDiskName,y	21169
	sta	(r6),y	21170
	beq	:14	21171
	sta	curDirHead +\$90,y	21172
	iny		21173
	cpy	#16	21174
	bne	:13	21175
			21176
	lda	#NULL	21177
	sta	(r6),y	21178
			21179
::14	jsr	PutDirHead	21180
	txa		21181
	bne	:err	21182
			21183
;	lda	curDrive ;???	21184

```

21185 ;                jsr    SetDevice
21186
21187                jmp     OpenDisk                ;BAM aktualisieren.
21188
21189 ; Dialogbox: Diskname.
21190 :d784F                b    %10000001
21191
21192                b    CANCEL    , $11, $48
21193
21194                b    DBTXTSTR, $08, $10                ;"Neuer Name".
21195                w    :1
21196
21197                b    DBTXTSTR, $08, $1c                ;">"
21198                w    T29D5
21199                b    DBTXTSTR, $08, $3a                ;">"
21200                w    T29D3
21201
21202                b    DBTXTSTR, $08, $2e                ;"Alter Name".
21203                w    :3
21204                b    DBVARSTR, $10, $3a                ;Diskname.
21205                b    a1L
21206
21207                b    DBGETSTRING                ;Name eingeben.
21208                b    $10, $16
21209                b    r5L, 16
21210
21211                b    NULL
21212
21213 if LANG = LANG_DE
21214 ::1                b    BOLDON, "Neuer Name für Diskette:", 0
21215 ::3                b    BOLDON, "Aktueller Name:"
21216                b    PLAINTEXT, 0
21217 endif
21218 if LANG = LANG_EN
21219 ::1                b    BOLDON, "New disk name:", 0
21220 ::3                b    BOLDON, "Current disk name:"
21221                b    PLAINTEXT, 0
21222 endif
21223
21224 ; 0: Datei-Information für markierte Dateien.
21225 :N6EB5                lda    C042A                ;Oberstes Fenster.
21226                sta    WIN042F                ;Aktives Fenster.
21227
21228                jsr    S5842                ;mark.Dat. anzeigen.
21229
21230                jsr    S5C89                ;Hole Disk/Fenster.
21231                tax
21232                beq    :1
21233
21234                jsr    S5245                ;Dateien invertieren.
21235                cpx    #CANCEL_ERR                ;Abbruch?
21236                beq    :exit                ; => Ja, Ende...
21237                bne    :err                ;Fehler ausgeben.
21238
21239 ::1                ldx    V04B7                ;Anzahl mark.Dateien.

```



	beq	:exit	; => Keine, Ende...	21240
				21241
	lda	#< V634F	;Zeiger Tabelle mit	21242
	sta	r2L	;markierten Dateien.	21243
	lda	#> V634F		21244
	sta	r2H		21245
				21246
::2	jsr	S5262	;Suche mark.Datei.	21247
	bcs	:exit		21248
				21249
	jsr	S46EB	;Name mark.Datei.	21250
				21251
	jsr	z6EF7	;Info anzeigen.	21252
	txa		;Fehler?	21253
	bne	:err	; => Ja, Abbruch...	21254
				21255
	lda	keyData		21256
	cmp	#KEY_STOP	;RUN/STOP?	21257
	beq	:3	; => Ja, Abbruch...	21258
				21259
	jsr	S48B6	;Langer Mausklick?	21260
	bcs	:2	; => Nein, weiter...	21261
				21262
::3	jsr	S524C	;Dat.-Ausw. aufheben.	21263
				21264
::exit	rts			21265
				21266
::err	jmp	S5CF3	;Fehler ausgeben.	21267
				21268
; Infobox schließen.				21269
:r71F4	jmp	RstrFrmDialogue		21270
				21271
; Informationen für aktuelle Datei anzeigen.				21272
:z6EF7	jsr	S1412	;Datei suchen.	21273
	txa			21274
	bne	:err	; => Fehler, Abbruch.	21275
				21276
	lda	r1H	;Zeiger auf Tr/Se	21277
	pha		;Verzeichnisblock.	21278
	lda	r1L		21279
	pha			21280
				21281
	lda	r5H	;Zeiger innerhalb	21282
	pha		;Verzeichnisblock.	21283
	lda	r5L		21284
	pha			21285
				21286
	jsr	i_FillRam	;Infoblock löschen.	21287
	w	255		21288
	w	fileHeader		21289
	b	\$00		21290
				21291
	lda	dirEntryBuf +22		21292
	beq	:ok	; => Nicht GEOS.	21293
				21294

```

21295      jsr    S09C0      ;Infoblock > r1L/r1H.
21296
21297      lda    #< fileHeader
21298      sta    r4L
21299      lda    #> fileHeader
21300      sta    r4H
21301      jsr    GetBlock      ;Infoblock einlesen.
21302      txa
21303      beq    :ok
21304
21305      pla
21306      sta    r5L
21307      pla      ;Zeiger innerhalb
21308      sta    r5H      ;Verzeichnisblock.
21309
21310      pla
21311      sta    r1L
21312      pla      ;Zeiger auf Tr/Se
21313      sta    r1H      ;Verzeichnisblock.
21314
21315 ::err      rts
21316
21317 ::ok      lda    #SET_BOLD
21318      sta    currentMode
21319
21320      lda    #< V0470      ;Zeiger auf
21321      sta    r0L      ;Dateiname.
21322      lda    #> V0470
21323      sta    r0H
21324      jsr    S244C      ;Hole Stringlänge.
21325
21326      lda    #SET_PLAINTEXT
21327      sta    currentMode
21328
21329      lsr    r1L      ;Stringlänge/2.
21330
21331      lda    #IBOXT_CX
21332      sec
21333      sbc    r1L
21334      sta    v7020 +1
21335
21336 ; Angaben zur Datei vorbereiten:
21337 ; a1 = GEOS-Klasse.
21338 ; a2 = GEOS-Dateityp.
21339 ; a3 = Autor.
21340 ; a4 = Seq./VLIR.
21341
21342 ; Zeiger auf GEOS-Klasse.
21343      lda    #< fileHeader +77
21344      sta    a1L
21345      lda    #> fileHeader +77
21346      sta    a1H      ;GEOS-Klasse.
21347
21348 ; Zeiger auf Autor.
21349      lda    #< fileHeader +97

```

```

        sta    a3L                                21350
        lda    #> fileHeader +97                  21351
        sta    a3H                                ;Autor.          21352
                                                21353
        ldx    #16                                21354
        lda    dirEntryBuf +0                      21355
        and    #FTYPE_MODES                        21356
        cmp    #FTYPE_DIR                          ;Verzeichnis?  21357
        beq    :13                                ; => Kein Autor...  21358
                                                21359
        ldx    dirEntryBuf +22                      21360
if CHK_AUTHOR = FALSE                          21361
        beq    :13                                ; => Nicht-GEOS...  21362
; Hinweis:                                     21363
; Bei Font-Dateien liegen hier Daten           21364
; zu den einzelnen Punktgrößen.                21365
        cpx    #FONT                              ;Autor => Font-Daten.  21366
        beq    :13                                ; => Kein Autor...  21367
        cpx    #DATA                              ;Datendatei?      21368
        beq    :13                                ; => Kein Autor...  21369
        cpx    #TEMPORARY                        ;Temporär?      21370
        beq    :13                                ; => Kein Autor...  21371
endif                                           21372
        cpx    #TDFOLDER                          ;TopDesk-Ordner?  21373
        bne    :11                                ; => Nein, weiter...  21374
                                                21375
; *TDFOLDERFIX*                               21376
;                                               21377
; Der VLIR-Test funktioniert hier nicht        21378
; zuverlässig. Ein Ordner hat als Tr/Se         21379
; $00/$ff in Byte #1/2, da ein Ordner          21380
; keine Daten beinhaltet.                      21381
        lda    dirEntryBuf +1                      ;Daten vorhanden?  21382
        bne    :11                                ; => Ja, weiter...  21383
        ldx    #TDFOLDER_EXT                      21384
                                                21385
;                                               21386
; ldy    dirEntryBuf +21                        21387
; cpy    #VLIR                                  ;VLIR-Datei?      21387
; beq    :13                                    ; => Nein, Ordner...  21388
                                                21389
::11      ldy    #0                              ;Autor überprüfen.  21390
::12      lda    fileHeader +97,y                21391
        beq    :14                                21392
        cmp    #$20                              21393
        bcc    :13                                ; => Ungültig...  21394
        cmp    #$7f                              21395
        bcs    :13                                ; => Ungültig...  21396
        iny                                       21397
        cpy    #20 -1                            21398
        bcc    :12                              21399
        bcs    :14                              21400
                                                21401
::13      lda    #< t7013                        ;Autor ungültig,  21402
        sta    a3L                                ; => kein Text.    21403
        lda    #> t7013                          21404

```

```

21405          sta    a3H
21406
21407 ; Dateityp auswerten.
21408 ::14          cpx    #GTYPESMAX
21409          bcc     :15
21410          ldx     #GTYPE_ERR
21411 ::15          txa
21412
21413 ; Hinweis:
21414 ; TopDesk testet Dateityp und überprüft
21415 ; den GEOS-Autor an Hand einer Tabelle
21416 ; und des String-Inhalts auf einen
21417 ; gültigen Autor.
21418 if CHK_AUTHOR = TRUE
21419          tay
21420          ldx     v7612,y          ;Autor vorhanden?
21421          bne     :21          ; => Ja, weiter...
21422
21423          ldx     #< t7013
21424          stx     a3L
21425          ldx     #> t7013
21426          stx     a3H
21427 endif
21428
21429 ; GEOS-Dateityp.
21430 ::21          asl
21431          tay
21432          lda     v7622 +0,y
21433          sta     a2L
21434          lda     v7622 +1,y
21435          sta     a2H
21436
21437 ; Datei-Struktur.
21438          lda     #< t7003          ;"Sequentiell".
21439          ldy     #> t7003
21440
21441          ldx     fileHeader +70
21442          beq     :22
21443
21444          lda     #< t700F          ;"VLIR".
21445          ldy     #> t700F
21446
21447 ::22          sta     a4L
21448          sty     a4H
21449
21450 ; Infobox anzeigen.
21451          lda     #$00
21452          sta     v6EF6          ;Daten unverändert.
21453
21454          lda     COL043C          ;Farbe Dialogbox
21455          pha          ;zwischenspeichern.
21456
21457          lda     COL043F          ;Farbe Infobox.
21458          sta     COL043C
21459

```

	lda	#< d7014	21460
	ldx	#> d7014	21461
	jsr	S2090 ;DBox ohne Recover.	21462
			21463
	pla		21464
	sta	C0L043C ;Farbe Dialogbox.	21465
			21466
	pla		21467
	sta	r5L	21468
	pla	;Zeiger innerhalb	21469
	sta	r5H ;Verzeichnisblock.	21470
			21471
	pla		21472
	sta	r1L	21473
	pla	;Zeiger auf Tr/Se	21474
	sta	r1H ;Verzeichnisblock.	21475
			21476
	ldx	v6EF6 ;Daten geändert?	21477
	beq	:done ; => Nein, Ende...	21478
			21479
	ldy	#\$00	21480
	lda	dirEntryBuf ;Dateityp mit	21481
	sta	(r5),y ;Schreibschutz.	21482
	sta	fileHeader +68 ;Infoblock.	21483
			21484
	; Verzeichnisblock speichern.		21485
	lda	#< diskBlkBuf ;Zeiger auf	21486
	sta	r4L ;Zwischenspeicher.	21487
	lda	#> diskBlkBuf	21488
	sta	r4H	21489
	jsr	PutBlock_dskBuf	21490
	txa		21491
	bne	:done ; => Fehler, Abbruch.	21492
			21493
	; GEOS-Datei: Infoblock schreiben.		21494
	lda	dirEntryBuf +22	21495
	beq	:done ;Keine GEOS-Datei.	21496
			21497
	jsr	S09C0 ;Infoblock > r1L/r1H.	21498
			21499
	lda	#< fileHeader	21500
	sta	r4L	21501
	lda	#> fileHeader	21502
	sta	r4H	21503
	jsr	PutBlock ;Infoblock.	21504
			21505
::done	rts		21506
			21507
:v6EF6	b	\$00 ;>0 = Daten geändert.	21508
			21509
	; Dialogbox: Datei-Info.		21510
:d7014	b	%00000001	21511
	b	IBOX_Y0,IBOX_Y1	21512
	w	IBOX_X0,IBOX_X1	21513
			21514

```

21515 ; Dialogbox-Oberfläche.
21516         b DB_USR_ROUT
21517         w r7893
21518
21519 ; Fettschrift für Titelzeile ein.
21520         b DBTXTSTR,$32,$0a
21521         w t706D
21522
21523 ; Titel = Dateiname.
21524 :v7020         b DBTXTSTR,$00,IB_DATA_Y0
21525         w V0470
21526
21527 ; GEOS-Klasse.
21528         b DBTXTSTR
21529         b IB_DATA_T0,IB_DATA_Y1
21530         w t706F
21531
21532 ; GEOS-Dateityp.
21533         b DBTXTSTR
21534         b IB_DATA_T0,IB_DATA_Y2
21535         w t7078
21536
21537 ; Autor (wenn verfügbar).
21538         b DBTXTSTR
21539         b IB_DATA_T0,IB_DATA_Y3
21540         w t7081
21541
21542 ; Dateistruktur.
21543         b DBTXTSTR
21544         b IB_DATA_T0,IB_DATA_Y4
21545         w t70A6
21546
21547 ; Datei/Datum.
21548         b DBTXTSTR
21549         b IB_DATA_T0,IB_DATA_Y5
21550         w t7088
21551
21552 ; Datei/Größe.
21553         b DBTXTSTR
21554         b IB_DATA_T0,IB_DATA_Y6
21555         w t708F
21556
21557 ; Schreibschutz.
21558         b DBTXTSTR
21559         b (WPROT_X0 -IBOX_X0 +WPROT_XW +4)
21560         b (WPROT_Y0 -IBOX_Y0 +6)
21561         w t7097
21562
21563 ; GEOS-Klasse.
21564         b DBVARSTR,IB_DATA_T1,IB_DATA_Y1
21565         b a1L
21566
21567 ; GEOS-Dateityp.
21568         b DBVARSTR,IB_DATA_T1,IB_DATA_Y2
21569         b a2L

```

		21570
<b>; Autor (wenn verfügbar).</b>		21571
	<b>b DBVARSTR, IB_DATA_T1, IB_DATA_Y3</b>	21572
	<b>b a3L</b>	21573
		21574
<b>; Dateistruktur.</b>		21575
	<b>b DBVARSTR, IB_DATA_T1, IB_DATA_Y4</b>	21576
	<b>b a4L</b>	21577
		21578
<b>; Datei/Größe.</b>		21579
	<b>b DB_USR_ROUT</b>	21580
	<b>w r70B0</b>	21581
		21582
<b>; Datei/Datum.</b>		21583
	<b>b DB_USR_ROUT</b>	21584
	<b>w r7117</b>	21585
		21586
<b>; Schreibschutz.</b>		21587
	<b>b DB_USR_ROUT</b>	21588
	<b>w r7197</b>	21589
		21590
<b>; Schreibschutz ändern.</b>		21591
	<b>b DBOPVEC</b>	21592
	<b>w r71C0</b>	21593
		21594
<b>; Texteditor.</b>		21595
	<b>b DB_USR_ROUT</b>	21596
	<b>w r71F7</b>	21597
		21598
<b>; CLOSE-Icon.</b>		21599
	<b>b DBUSRICON, CICON_X0, CICON_Y0</b>	21600
	<b>w v776B</b>	21601
	<b>b NULL</b>	21602
		21603
<b>; "CLOSE"-Icon.</b>		21604
<b>:GR774D</b>	<b>b \$80 +2*16</b>	21605
		21606
	<b>b %00000000,%00000000</b>	21607
	<b>b %01111111,%11111110</b>	21608
	<b>b %01000000,%00000010</b>	21609
	<b>b %01000000,%00000010</b>	21610
	<b>b %01000000,%00000010</b>	21611
	<b>b %01000000,%00000010</b>	21612
	<b>b %01000111,%11100010</b>	21613
	<b>b %01000111,%11100010</b>	21614
		21615
	<b>b %01000111,%11100010</b>	21616
	<b>b %01000111,%11100010</b>	21617
	<b>b %01000000,%00000010</b>	21618
	<b>b %01000000,%00000010</b>	21619
	<b>b %01000000,%00000010</b>	21620
	<b>b %01000000,%00000010</b>	21621
	<b>b %01111111,%11111110</b>	21622
	<b>b %00000000,%00000000</b>	21623
		21624

```

21625 :GR774D_x          = 2
21626 :GR774D_y          = 16
21627
21628 :v776B              w GR774D
21629                   b $00,$00,GR774D_x,GR774D_y
21630                   w r71F4
21631
21632 ; Texte für Infobox.
21633 :t706D               b PLAINTEXT,BOLDON
21634 :t7013              b NULL ;Leerstring.
21635
21636 if LANG = LANG_DE
21637 :t706F               b "Klasse:",0
21638 :t7078              b "Dateityp:",0
21639 :t7081              b "Autor:",0
21640 :t70A6              b "Struktur:",0
21641 :t7088              b "Datum:",0
21642 :t708F              b "Größe: ",0
21643 :t7097              b "Schreibschutz"
21644                   b PLAINTEXT,0
21645
21646 :t7003               b "Sequentiell",0
21647 :t700F              b "VLIR",0
21648 endif
21649 if LANG = LANG_EN
21650 :t706F               b "Class:",0
21651 :t7078              b "File type:",0
21652 :t7081              b "Author:",0
21653 :t70A6              b "Structure:",0
21654 :t7088              b "Date:",0
21655 :t708F              b "Size: ",0
21656 :t7097              b "Write protected"
21657                   b PLAINTEXT,0
21658
21659 :t7003               b "Sequential",0
21660 :t700F              b "VLIR",0
21661 endif
21662
21663 :t7105               b " Blocks",0
21664 :t710D              b " KByte",0
21665
21666 if CHK_AUTHOR = TRUE
21667 ; GEOS-Info: 1=Autor anzeigen.
21668 :v7612              b $00 ;$00 = Nicht-GEOS.
21669                   b $01 ;$01 = BASIC.
21670                   b $01 ;$02 = Assembler.
21671                   b $00 ;$03 = Daten.
21672                   b $01 ;$04 = Systemdatei.
21673                   b $01 ;$05 = Hilfsmittel.
21674                   b $01 ;$06 = Anwendung.
21675                   b $00 ;$07 = Dokument.
21676                   b $00 ;$08 = Zeichensatz.
21677                   b $01 ;$09 = Druckertreiber.
21678                   b $01 ;$0A = Eingabetreiber.
21679                   b $00 ;$0B = Laufwerkstreiber/Ordner.

```



```

        b $01 ;$0C = Startprogramm.                21680
        b $00 ;$0D = Temporär.                    21681
        b $01 ;$0E = Selbstausführend.            21682
        b $01 ;$0F = Eingabetreiber (C128).        21683
                                                    21684
; Hinweis:                                         21685
; In TopDesk v4/v5 ist die Tabelle auf             21686
; die Dateitypen 0-15 begrenzt. Für die            21687
; Dateitypen 16-22 waren die folgenden            21688
; Bytes <=>$00 = Autor anzeigen.                  21689
; => Liste gem. v4/v5 erweitert.                  21690
        b $01 ;$10 = gateWay-Verzeichnis.          21691
        b $01 ;$11 = gateWay-Dokument.            21692
        b $01 ;$12 = Unbekannt.                   21693
        b $01 ;$13 = Unbekannt.                   21694
        b $01 ;$14 = Unbekannt.                   21695
        b $01 ;$15 = GeoShell-Befehl.             21696
        b $01 ;$16 = GeoFax-Drucker.              21697
                                                    21698
; Hinweis:                                         21699
; Der TopDesk-Ordner beinhaltet im                 21700
; Infoblock einen Autorenhinweis, bei             21701
; Datei/Info wurde dieser ignoriert.              21702
        b $00 ;$17 = TD-Ordner/TDFOLDER_EXT.      21703
endif                                              21704
                                                    21705
; TODO:                                           21706
; Die Liste mit Verzeichnis/Text-Modus            21707
; zusammenlegen, Ausnahme für TopDesk-            21708
; Ordner hinzufügen.                             21709
:v7622      w t7650 ;$00 = Nicht-GEOS.             21710
            w t765B ;$01 = BASIC.                  21711
            w t7661 ;$02 = Assembler.              21712
            w t766B ;$03 = Daten.                  21713
            w t7670 ;$04 = Systemdatei.            21714
            w t767C ;$05 = Hilfsmittel.            21715
            w t768A ;$06 = Anwendung.              21716
            w t7694 ;$07 = Dokument.               21717
            w t769D ;$08 = Zeichensatz.            21718
            w t76AE ;$09 = Druckertreiber.          21719
            w t76BD ;$0A = Eingabetreiber.          21720
            w t76D1 ;$0B = Laufwerkstreiber/Ordner. 21721
            w t76DB ;$0C = Startprogramm.            21722
            w t76E9 ;$0D = Temporär.               21723
            w t76F2 ;$0E = Selbstausführend.        21724
            w t7703 ;$0F = Eingabetreiber (C128).    21725
            w t7718 ;$10 = gateWay-Verzeichnis.      21726
            w t7729 ;$11 = gateWay-Dokument.         21727
:GTERR      w t7749 ;$12 = Unbekannt.               21728
            w t7749 ;$13 = Unbekannt.               21729
            w t7749 ;$14 = Unbekannt.               21730
            w t7731 ;$15 = GeoShell-Befehl.          21731
            w t773A ;$16 = GeoFax-Drucker.           21732
            w t7740 ;$17 = TD-Ordner/TDFOLDER_EXT.  21733
:GTYPESEND                                     21734

```

```

21735
21736 ; Max. Anzahl Dateitypen.
21737 :GTYPESMAX          = (GTYPESEND - v7622)/2
21738
21739 ; Ausnahme: Dateityp unbekannt.
21740 :GTYPE_ERR          = (GTERR - v7622)/2
21741
21742 if LANG = LANG_DE
21743 :t7650              b "Nicht-GEOS",0
21744 :t765B              b "BASIC",0
21745 :t7661              b "Assembler",0
21746 :t766B              b "Daten",0
21747 :t7670              b "Systemdatei",0
21748 :t767C              b "Hilfsprogramm",0
21749 :t768A              b "Anwendung",0
21750 :t7694              b "Dokument",0
21751 :t769D              b "Zeichensatz",0
21752 :t76AE              b "Druckertreiber",0
21753 :t76BD              b "Eingabetreiber (64)",0
21754 :t76D1              b "Laufwerkstreiber",0
21755 :t76DB              b "Startprogramm",0
21756 :t76E9              b "Temporär",0
21757 :t76F2              b "Selbstausführend",0
21758 :t7703              b "Eingabetreiber (128)",0
21759 :t7718              b "Unterverzeichnis",0
21760 :t7729              b "gateWay",0
21761 :t7731              b "geoShell",0
21762 :t773A              b "geoFax-Drucker",0
21763 :t7740              b "TopDesk-Ordner",0
21764 endif
21765 if LANG = LANG_EN
21766 :t7650              b "Not GEOS",0
21767 :t765B              b "BASIC",0
21768 :t7661              b "Assembler",0
21769 :t766B              b "Data",0
21770 :t7670              b "System file",0
21771 :t767C              b "Desk accessory",0
21772 :t768A              b "Application",0
21773 :t7694              b "Document",0
21774 :t769D              b "Font",0
21775 :t76AE              b "Printer driver",0
21776 :t76BD              b "Input driver (64)",0
21777 :t76D1              b "Disk driver",0
21778 :t76DB              b "Boot file",0
21779 :t76E9              b "Temporary",0
21780 :t76F2              b "Auto exec",0
21781 :t7703              b "Input driver (128)",0
21782 :t7718              b "Directory",0
21783 :t7729              b "gateWay",0
21784 :t7731              b "geoShell",0
21785 :t773A              b "geoFax printer",0
21786 :t7740              b "TopDesk folder",0
21787 endif
21788
21789 :t7749              b "???",0

```

		21790
<b>; Farbe Titelzeile.</b>		21791
<b>:r7893</b>	<b>ldx</b> <b> #(r4H - r2L)</b>	21792
<b>::1</b>	<b>lda</b> <b> :coord,x</b>	21793
	<b>sta</b> <b> r2,x</b>	21794
	<b>dex</b>	21795
	<b>bpl</b> <b> :1</b>	21796
		21797
	<b>lda</b> <b> #0</b>	21798
	<b>jsr</b> <b> S140C</b> <b> ;Rechteck zeichnen.</b>	21799
		21800
<b>; Farbe für Titelzeile berechnen.</b>		21801
	<b>lda</b> <b> C_DBoxTitel</b>	21802
		21803
	<b>ldx</b> <b> C0438</b> <b> ;TopDesk in Farbe?</b>	21804
	<b>bne</b> <b> :4</b> <b> ; =&gt; Ja, weiter...</b>	21805
		21806
	<b>lda</b> <b> screencolors</b>	21807
	<b>ldy</b> <b> #4</b>	21808
<b>::2</b>	<b>clc</b>	21809
	<b>tax</b>	21810
	<b>bpl</b> <b> :3</b>	21811
	<b>sec</b>	21812
<b>::3</b>	<b>rol</b>	21813
	<b>dey</b>	21814
	<b>bne</b> <b> :2</b>	21815
		21816
<b>::4</b>	<b>jmp</b> <b> DirectColor</b> <b> ;Screencolor setzen.</b>	21817
		21818
<b>::coord</b>	<b>b</b> <b> IB0XT_Y0,IB0XT_Y1</b>	21819
	<b>w</b> <b> IB0X_X0,IB0X_X1</b>	21820
		21821
<b>; Dateigröße ausgeben.</b>		21822
<b>:r70B0</b>	<b>lda</b> <b> dirEntryBuf +28</b>	21823
	<b>sta</b> <b> r0L</b>	21824
	<b>lda</b> <b> dirEntryBuf +29</b>	21825
	<b>sta</b> <b> r0H</b>	21826
		21827
	<b>lda</b> <b> #&lt; IB_DATA_X0</b>	21828
	<b>sta</b> <b> r11L</b>	21829
	<b>lda</b> <b> #&gt; IB_DATA_X0</b>	21830
	<b>sta</b> <b> r11H</b>	21831
		21832
	<b>lda</b> <b> #IB0X_Y0 +IB_DATA_Y6</b>	21833
	<b>sta</b> <b> r1H</b>	21834
		21835
	<b>lda</b> <b> MT3E11</b> <b> ;Anzeige in Blocks</b>	21836
	<b>cmp</b> <b> #""</b> <b> ;oder KByte?</b>	21837
	<b>beq</b> <b> :2</b> <b> ; =&gt; Blocks...</b>	21838
		21839
	<b>lda</b> <b> r0L</b>	21840
	<b>pha</b>	21841
		21842
	<b>lsr</b> <b> r0H</b>	21843
	<b>ror</b> <b> r0L</b>	21844

```

21845      lsr    r0H
21846      ror    r0L
21847
21848      pla
21849      and    #%00000011
21850      beq     :1
21851
21852      inc    r0L
21853      bne     :1
21854      inc    r0H
21855
21856  ::1      lda    #SET_LEFTJUST!SET_SUPRESS
21857          jsr    PutDecimal
21858
21859          lda    #< t710D
21860          sta    r0L
21861          lda    #> t710D
21862          sta    r0H
21863          jmp    PutString
21864
21865  ::2      lda    #"s"
21866          ldx    r0H
21867          bne     :3
21868          ldx    r0L
21869          dex
21870          bne     :3
21871          txa
21872  ::3      sta    t7105 +6
21873
21874          lda    #SET_LEFTJUST!SET_SUPRESS
21875          jsr    PutDecimal
21876
21877          lda    #< t7105
21878          sta    r0L
21879          lda    #> t7105
21880          sta    r0H
21881          jmp    PutString
21882
21883 ; Datum/Uhrzeit ausgeben.
21884 :r7117    lda    #< IB_DATA_X0
21885          sta    r11L
21886          lda    #> IB_DATA_X0
21887          sta    r11H
21888
21889          lda    #IBOX_Y0 +IB_DATA_Y5
21890          sta    r1H
21891
21892          ldy    #0
21893          sty    a4L
21894  ::loop   lda    v7188,y
21895          tay
21896          lda    dirEntryBuf,y
21897          sta    r0L
21898          lda    #$00
21899          sta    r0H

```

			21900
	ldy	a4L	21901
	lda	v7192,y	21902
	beq	:1	21903
			21904
	lda	r0L	21905
	cmp	#10	21906
	bcs	:1	21907
			21908
	lda	"0"	21909
	jsr	PutChar	21910
			21911
::1	lda	#SET_LEFTJUST!SET_SUPRESS	21912
	jsr	PutDecimal	21913
			21914
	ldy	a4L	21915
	lda	v718D,y	21916
	beq	:exit	21917
			21918
	jsr	PutChar	21919
			21920
			21921
			21922
			21923
			21924
			21925
			21926
			21927
			21928
			21929
			21930
			21931
			21932
			21933
			21934
			21935
			21936
			21937
::2	ldx	"0"	21938
	b	\$2c	21939
::3	ldx	"2"	21940
	lda	"0"	21941
			21942
::4	pha		21943
	txa		21944
	jsr	SmallPutChar	21945
	pla		21946
	jsr	SmallPutChar	21947
			21948
::5	ldy	a4L	21949
	bne	:loop	21950
			21951
::exit	lda	#PLAINTEXT	21952
	jmp	PutChar	21953
			21954

```

21955 ; Position in dirEntryBuf.
21956 :v7188          b $19  ;Tag
21957              b $18  ;Monat
21958              b $17  ;Jahr
21959              b $1a  ;Stunde
21960              b $1b  ;Minute
21961
21962 ; Trenner für Datum/Uhrzeit-Werte.
21963 :v718D          b "."
21964              b "."
21965              b " "
21966              b ":"
21967              b NULL
21968
21969 ; Ein-/zweistellige Ausgabe.
21970 :v7192          b $00  ;Einstellig : Tag.
21971              b $00  ;Einstellig : Monat.
21972              b $01  ;Zweistellig: Jahr.
21973              b $00  ;Einstellig : Stunde.
21974              b $01  ;Zweistellig: Minute.
21975
21976 ; Schreibschutz anzeigen.
21977 :r7197          jsr   i_FrameRectangle
21978              b WPROT_Y0,WPROT_Y1
21979              w WPROT_X0,WPROT_X1
21980              b %11111111
21981
21982              lda    C_InputField
21983              jsr    S1561                ;Screencolor setzen.
21984
21985              lda    dirEntryBuf
21986              and    #ST_WR_PR
21987              beq    :1
21988
21989              jsr    InvertRectangle
21990
21991 ::1            rts
21992
21993 ; Schreibschutz ändern.
21994 :r71C0          bit    mouseData
21995              bmi    :exit
21996
21997              ldx    #(r4H - r2L)
21998 ::1            lda    :coord,x
21999              sta    r2,x
22000              dex
22001              bpl    :1
22002
22003              jsr    IsMseInRegion
22004              beq    :exit
22005
22006              lda    #$ff
22007              sta    v6EF6                ;Daten geändert.
22008
22009              lda    dirEntryBuf

```

	eor	#ST_WR_PR	22010
	sta	dirEntryBuf	22011
			22012
	jsr	InvertRectangle	22013
			22014
::exit	rts		22015
			22016
::coord	b	WPROT_Y0, WPROT_Y1	22017
	w	WPROT_X0, WPROT_X1	22018
			22019
; Texteditor.			22020
:r71F7	lda	dirEntryBuf +22	22021
	bne	:1	22022
	rts		22023
			22024
::1	lda	otherPressVec +0	22025
	sta	v7267 +0	22026
	lda	otherPressVec +1	22027
	sta	v7267 +1	22028
			22029
	lda	#< r734E	22030
	sta	otherPressVec +0	22031
	lda	#> r734E	22032
	sta	otherPressVec +1	22033
			22034
	lda	#< r7269 ;Tastaturabfrage.	22035
	sta	keyVector +0	22036
	lda	#> r7269	22037
	sta	keyVector +1	22038
			22039
	lda	curHeight ;Textcursor ein.	22040
	jsr	InitTextPrompt	22041
			22042
	ldy	#MAXLENGTH -1 ;Infotext bereinigen.	22043
::2	lda	INFOTEXT,y	22044
	beq	:4	22045
	cmp	#CR	22046
	beq	:4	22047
	cmp	#" "	22048
	bcc	:3	22049
	cmp	#\$7f	22050
	bcc	:4	22051
::3	lda	#"*"	22052
	sta	INFOTEXT,y	22053
::4	dey		22054
	bpl	:2	22055
			22056
	ldy	#(r4H - r2L)	22057
::5	lda	v7600,y ;Grenzen für	22058
	sta	r2,y ;Textfenster setzen.	22059
	lda	v7603,y ;Grenzen für	22060
	sta	windowTop,y ;Textausgabe setzen.	22061
	dey		22062
	bpl	:5	22063
			22064

```

22065          lda    C_InputField          ;Farbe Infotext.
22066          jsr    S1561                  ;Screencolor setzen.
22067
22068          lda    #%11111111
22069          jsr    FrameRectangle
22070
22071          lda    #$00                      ;Position auf Anfang.
22072          sta    v760A
22073
22074          jmp     z739E                    ;Text anzeigen.
22075
22076 :v7267          w $0000 ;otherPressVec.
22077
22078 ; Tastaturabfrage.
22079 :r7269          lda    keyData
22080          cmp     #$7f
22081          bcc     :1
22082
22083 ; Auf C= + LeftArrow testen (Ende).
22084 ::cbmkey        cmp     #%10000000 ! KEY_LARROW
22085          bne     :exit
22086
22087 ::close         jmp     RstrFrmDialogue
22088
22089 ; Tastencodes auswerten.
22090 ::1             cmp     #" "              ;Sichtb. Zeichen?
22091          bcs     :keys                    ; => Ja, weiter...
22092
22093          cmp     #KEY_STOP                ;RUN/STOP?
22094          beq     :close                    ; => Ja, Ende...
22095
22096 ; Cursor-Tasten auswerten.
22097          cmp     #KEY_LEFT
22098          bne     :2
22099
22100          lda     v760A                    ;Aktuelle Position.
22101          beq     :exit
22102          dec     v760A                    ;Neue Position.
22103          jmp     z73C4                    ;Cursor setzen.
22104
22105 ::2             cmp     #KEY_RIGHT
22106          bne     :3
22107
22108          ldy     v760A                    ;Aktuelle Position.
22109          iny
22110          cpy     v7609
22111          bcs     :exit
22112          inc     v760A                    ;Neue Position.
22113          jmp     z73C4                    ;Cursor setzen.
22114
22115 ::3             cmp     #KEY_DOWN
22116          bne     :4
22117
22118          lda     stringY
22119          clc

```



	adc	#LINEHEIGHT		22120
	jsr	z7348	;r1H/r11 setzen.	22121
	jmp	z7378	;Textpos. setzen.	22122
				22123
::4	cmp	#KEY_UP		22124
	bne	:5		22125
				22126
	ldy	v760B	;Aktuelle Zeile.	22127
	beq	:exit		22128
	lda	stringY		22129
	sec			22130
	sbc	#LINEHEIGHT		22131
	jsr	z7348	;r1H/r11 setzen.	22132
	jmp	z7378	;Textpos. setzen.	22133
				22134
	; Auswertung Editor-Tasten.			22135
::5	ldx	#\$ff		22136
	stx	v6EF6	;Daten geändert.	22137
				22138
	cmp	#KEY_DELETE		22139
	bne	:6		22140
				22141
	ldy	v760A	;Aktuelle Position.	22142
	beq	:exit		22143
				22144
	jsr	z75F3	;Zeichen löschen.	22145
				22146
	dec	v760A	;Neue Position.	22147
	jmp	z739E	;Text anzeigen.	22148
				22149
::6	cmp	#CR		22150
	beq	:keys		22151
				22152
::exit	rts		;Sonderzeichen.	22153
				22154
	; Auswertung Zeichen-Tasten.			22155
::keys	ldx	#\$ff		22156
	stx	v6EF6	;Daten geändert.	22157
				22158
	ldy	v7609		22159
	cpy	#MAXLENGTH		22160
	bcs	:exit		22161
				22162
	jsr	z75DA	;Zeichen einfügen.	22163
				22164
	inc	v760A	;Neue Position.	22165
	ldy	v760A	;Aktuelle Position.	22166
	cpy	v7609		22167
	bcs	:12		22168
				22169
	jmp	z739E	;Text anzeigen.	22170
				22171
::12	jsr	z7435	;Text überprüfen.	22172
				22173
	lda	stringY		22174

```

22175      clc
22176      adc    #$07
22177      jsr    z7348          ;r1H/r11 setzen.
22178      stx    r0L
22179      sty    r0H
22180
22181      lda    keyData
22182      jsr    GetCharWidth
22183      beq    :15
22184
22185      clc
22186      adc    r0L
22187      sta    r0L
22188      bcc    :13
22189      inc    r0H
22190
22191  ::13      lda    r0H          ;Innerhalb der
22192            cmp    v7603 +5      ;Zeile?
22193            bne    :14
22194            lda    r0L
22195            cmp    v7603 +4
22196  ::14      bcc    :16          ; => Ja, weiter...
22197
22198  ::15      inc    v760B          ;Nächste Zeile.
22199            jmp    z739E          ;Text anzeigen.
22200
22201  ::16      lda    keyData          ;Zeichen ausgeben.
22202            jsr    PutChar
22203
22204            jmp    z73C4          ;Cursor setzen.
22205
22206 ; Neue X/Y-Position setzen.
22207 ; Übergabe: A = Y-Position.
22208  :z7348      sta    r1H
22209            ldx    stringX +0
22210            stx    r11L
22211            ldy    stringX +1
22212            sty    r11H
22213            rts
22214
22215 ; Mausabfrage.
22216  :r734E      lda    mouseData
22217            bpl    :1
22218
22219  ::exit      lda    v7267 +0
22220            ldx    v7267 +1
22221            jmp    CallRoutine
22222
22223  ::1         ldy    #(r4H - r2L)
22224  ::2         lda    v7603,y
22225            sta    r2,y
22226            dey
22227            bpl    :2
22228
22229            jsr    IsMseInRegion

```



```

22285          dec    v760B          ;Aktuelle Zeile.
22286
22287 ::1          jsr    z7563          ;Zeile neu ausgeben.
22288
22289          inc    v760B          ;Zeile +1.
22290          lda    v760B          ;Aktuelle Zeile.
22291          cmp    #MAXLINES
22292          bcc    :1
22293
22294          pla
22295          sta    v760B          ;Aktuelle Zeile.
22296          jmp    z73C4          ;Cursor setzen.
22297
22298 ; Cursor aktualisieren.
22299 :z73C4          jsr    z754D          ;Akt.Zei.Pos. testen.
22300
22301          lda    v760B          ;Aktuelle Zeile.
22302          asl
22303          sta    r1H
22304          asl
22305          asl
22306          clc
22307          adc    r1H
22308 ;          clc
22309          adc    v7603 +0
22310          bcs    :1
22311
22312          cmp    v7603 +1
22313          bcc    :2
22314
22315 ::1          lda    v7603 +1
22316          clc
22317          adc    #$03
22318 ::2          sta    r1H
22319
22320          lda    v7603 +2
22321          sta    r11L
22322          lda    v7603 +3
22323          sta    r11H
22324
22325          ldy    v760B          ;Aktuelle Zeile.
22326          ldx    v760C,y
22327 ::3          cpx    v760A          ;Aktuelle Position.
22328          beq    :5
22329
22330          lda    INFOTEXT,x
22331          jsr    GetCharWidth
22332
22333          clc
22334          adc    r11L
22335          sta    r11L
22336          bcc    :4
22337          inc    r11H
22338
22339 ::4          inx

```

	<b>bne</b>	<b>:3</b>	22340
			22341
<b>::5</b>	<b>lda</b>	<b>r11H</b>	22342
	<b>cmp</b>	<b>v7603 +5</b>	22343
	<b>bne</b>	<b>:6</b>	22344
	<b>lda</b>	<b>r11L</b>	22345
	<b>cmp</b>	<b>v7603 +4</b>	22346
<b>::6</b>	<b>bcc</b>	<b>:7</b>	22347
			22348
	<b>lda</b>	<b>v7603 +4</b>	22349
	<b>sta</b>	<b>r11L</b>	22350
	<b>lda</b>	<b>v7603 +5</b>	22351
	<b>sta</b>	<b>r11H</b>	22352
			22353
<b>::7</b>	<b>lda</b>	<b>r11L ;Neue Position</b>	22354
	<b>sta</b>	<b>stringX +0 ;für Textcursor.</b>	22355
	<b>lda</b>	<b>r11H</b>	22356
	<b>sta</b>	<b>stringX +1</b>	22357
			22358
	<b>lda</b>	<b>r1H</b>	22359
	<b>sta</b>	<b>stringY</b>	22360
			22361
	<b>jmp</b>	<b>PromptOn ;Textcursor ein.</b>	22362
			22363
<b>; Text überprüfen.</b>			22364
<b>:z7435</b>	<b>ldy</b>	<b>#\$00</b>	22365
	<b>sty</b>	<b>r2L</b>	22366
	<b>sty</b>	<b>v760C</b>	22367
			22368
<b>:z743C</b>	<b>lda</b>	<b>#\$00</b>	22369
	<b>sta</b>	<b>r2H</b>	22370
			22371
	<b>lda</b>	<b>v7603 +2</b>	22372
	<b>sta</b>	<b>r0L</b>	22373
	<b>lda</b>	<b>v7603 +3</b>	22374
	<b>sta</b>	<b>r0H</b>	22375
			22376
	<b>ldy</b>	<b>r2L</b>	22377
	<b>ldx</b>	<b>v760C,y</b>	22378
	<b>dex</b>		22379
<b>::1</b>	<b>inx</b>		22380
	<b>lda</b>	<b>INFOTEXT,x</b>	22381
	<b>beq</b>	<b>:8</b>	22382
	<b>cmp</b>	<b>#CR</b>	22383
	<b>beq</b>	<b>:5</b>	22384
	<b>cmp</b>	<b>#" "</b>	22385
	<b>bne</b>	<b>:2</b>	22386
	<b>stx</b>	<b>r1L</b>	22387
	<b>dec</b>	<b>r2H</b>	22388
<b>::2</b>	<b>jsr</b>	<b>GetCharWidth</b>	22389
			22390
	<b>clc</b>		22391
	<b>adc</b>	<b>r0L</b>	22392
	<b>sta</b>	<b>r0L</b>	22393
	<b>bcc</b>	<b>:3</b>	22394

```

22395         inc    r0H
22396
22397 ::3        lda    r0H
22398         cmp    v7603 +5
22399         bne    :4
22400         lda    r0L
22401         cmp    v7603 +4
22402 ::4        bcc    :1
22403
22404         lda    r2H
22405         beq    :6
22406
22407         ldx    r1L
22408 ::5        inx
22409 ::6        ldy    r2L
22410         cpy    #$04
22411         bcs    :7
22412
22413         inc    r2L
22414         txa
22415         sta    v760C +1,y
22416         jmp    z743C
22417
22418 ::7        lda    INFOTEXT,x
22419         beq    :8
22420         inx
22421         bne    :7
22422
22423 ::8        inc    r2L
22424         ldy    r2L
22425         inx
22426         txa
22427         sta    v7609
22428 ::9        sta    v760C,y
22429         iny
22430         cpy    #$05
22431         bcc    :9
22432         rts
22433
22434 ; Mausposition in Textfenster suchen.
22435 :z74AC     lda    v7603 +2
22436         sta    r0L
22437         lda    v7603 +3
22438         sta    r0H
22439
22440         lda    r11L
22441         sta    r2L
22442         lda    r11H
22443         sta    r2H
22444
22445         lda    r2L
22446         sec
22447         sbc    r0L
22448         sta    r2L
22449         lda    r2H

```

	sbc	r0H	22450
	sta	r2H	22451
			22452
	lda	r1H	22453
	sec		22454
	sbc	v7603 +0	22455
			22456
	ldy	#\$00	22457
::1	sec		22458
	sbc	#LINEHEIGHT	22459
	bcc	:2	22460
	iny		22461
	bne	:1	22462
::2	sty	r3H	22463
			22464
	ldx	v760C,y	22465
	cpx	v7609	22466
	bcs	:9	22467
			22468
	stx	r3L	22469
::3	lda	INFOTEXT,x	22470
	beq	:8	22471
			22472
	cmp	#CR	22473
	bne	:4	22474
			22475
	lda	#"M"	22476
::4	jsr	GetCharWidth	22477
			22478
	clc		22479
	adc	r0L	22480
	sta	r0L	22481
	bcc	:5	22482
	inc	r0H	22483
			22484
::5	lda	r0L	22485
	sta	r1L	22486
	lda	r0H	22487
	sta	r1H	22488
			22489
	lda	r1L	22490
	sec		22491
	sbc	r11L	22492
	sta	r1L	22493
	lda	r1H	22494
	sbc	r11H	22495
	sta	r1H	22496
			22497
	txa		22498
	pha		22499
	ldx	#r1	22500
	jsr	Dabs	22501
	pla		22502
	tax		22503
			22504

```

22505          inx
22506
22507          lda   r2H
22508          cmp   r1H
22509          bne   :6
22510          lda   r2L
22511          cmp   r1L
22512 ::6       bcc   :7
22513
22514          lda   r1L
22515          sta   r2L
22516          lda   r1H
22517          sta   r2H
22518
22519          stx   r3L
22520 ::7       ldy   r3H
22521          inx
22522          txa
22523          dex
22524          cmp   v760C +1,y
22525          bcc   :3
22526
22527 ::8       lda   r3L
22528          sta   v760A           ;Neue Position.
22529
22530          clc
22531          rts
22532
22533 ::9       ldx   v7609
22534          dex
22535          stx   v760A           ;Neue Position.
22536
22537          clc
22538          rts
22539
22540 ; Aktuelle Zeichenposition überprüfen.
22541 :z754D     ldy   #0
22542 ::1       lda   v760C,y
22543          cmp   v760A           ;Aktuelle Position.
22544          beq   :3
22545          bcs   :2
22546
22547          iny
22548          cpy   #MAXLINES
22549          bcc   :1
22550
22551 ::2       dey
22552 ::3       sty   v760B           ;Aktuelle Zeile.
22553          rts
22554
22555 ; Aktuelle Zeile neu ausgeben.
22556 :z7563     lda   v760B           ;Aktuelle Zeile.
22557          asl
22558          sta   r2L
22559          asl

```



	asl		22560
	clc		22561
	adc	r2L	22562
	clc		22563
	adc	v7603 +0	22564
	sta	r2L	22565
	bcs	:exit	22566
			22567
	cmp	v7603 +1	22568
	bcs	:exit	22569
			22570
	clc		22571
	adc	#\$09	22572
	cmp	v7603 +1	22573
	bcc	:1	22574
			22575
	lda	v7603 +1	22576
			22577
::1	sta	r2H	22578
			22579
	lda	v7603 +2	22580
	sta	r3L	22581
	lda	v7603 +3	22582
	sta	r3H	22583
			22584
	lda	v7603 +4	22585
	sta	r4L	22586
	lda	v7603 +5	22587
	sta	r4H	22588
			22589
	lda	#\$00	22590
	jsr	S140C ;Rechteck zeichnen.	22591
			22592
	lda	r2L	22593
	clc		22594
	adc	#\$07	22595
	sta	r1H	22596
			22597
	lda	v7603 +2	22598
	sta	r11L	22599
	lda	v7603 +3	22600
	sta	r11H	22601
			22602
	ldy	v760B ;Aktuelle Zeile.	22603
	lda	v760C,y	22604
	cmp	v7609	22605
	bcs	:exit	22606
			22607
	sta	r15L	22608
	lda	v760C +1,y	22609
	sta	r15H	22610
			22611
::2	ldy	r15L	22612
	lda	INFOTEXT,y	22613
	beq	:3	22614

```

22615          jsr    PutChar
22616 ::3          inc    r15L
22617          lda    r15L
22618          cmp    r15H
22619          bcc    :2
22620
22621 ::exit        rts
22622
22623 ; Zeichen einfügen.
22624 :z75DA        ldy    v7609
22625          iny
22626 ::1          dey
22627          lda    INFOTEXT +0,y
22628          sta    INFOTEXT +1,y
22629          cpy    v760A          ;Aktuelle Position.
22630          beq    :2
22631          bcs    :1
22632
22633 ::2          lda    keyData
22634          sta    INFOTEXT,y
22635          rts
22636
22637 ; Zeichen löschen.
22638 :z75F3        ldy    v760A          ;Aktuelle Position.
22639 ::1          lda    INFOTEXT +0,y
22640          sta    INFOTEXT -1,y
22641          iny
22642          cpy    v7609
22643          bcc    :1
22644          rts
22645
22646 ; Eingabefenster Infotext.
22647 :v7600        b ITEXT_Y0    ;y-Oben
22648          b ITEXT_Y1    ;y-unten
22649          w ITEXT_X0    ;x-Links
22650          w ITEXT_X1    ;x-Rechts
22651
22652 ; Textgrenzen Eingabefenster Infotext.
22653 :v7603        b ITEXT_Y0 +ITEXT_OFF    ;y-Oben
22654          b ITEXT_Y1 -ITEXT_OFF    ;y-unten
22655          w ITEXT_X0 +ITEXT_OFF    ;x-Links
22656          w ITEXT_X1 -ITEXT_OFF    ;x-Rechts
22657
22658 :v7609        b $00    ;Länge Text.
22659 :v760A        b $00    ;Aktuelle Position.
22660 :v760B        b $00    ;Aktuelle Zeile.
22661
22662 ; Zeiger auf Zeilenanfang.
22663 :MAXLINES     = 5
22664 :v760C        s MAXLINES +1
22665

```

```

; 22666
; 22667
; SourceCode: src.mod#05.s 22668
; 22669
; 22670
; 22671
; Modul-Info: 22672
; 0 - Anzeige: Icon-Modus 22673
; 22674
if .p 22675
    t "TopSym" 22676
    t "TopSym.MP3" 22677
    t "TopSym.ROM" 22678
    t "DTopDeskI18N.ext" 22679
    t "src.DTopDesk.ext" 22680
; 22681
; Position für Datei-Nr: 22682
:FNUM_XOFF = 20 22683
:FNUM_YOFF = 10 22684
:FNUM_XOFF2 = 31 22685
; 22686
; Anzeigebereich für Icons: 22687
:ICON_OFF_XL = 8 22688
:ICON_OFF_XR = 8 22689
:ICON_OFF_YO = 8 22690
:ICON_OFF_YU = 8 22691
; 22692
:NAME_BASELINE = 6 22693
endif 22694
; 22695
o BASE_MOD_B 22696
n "obj.#05" 22697
; 22698
::ModEntry_0 jmp M2BB5 ;Init, Inhalt anzeigen. 22699
::ModEntry_1 jmp M2BB8 ;Inhalt anzeigen. 22700
::ModEntry_2 jmp M2EEE ;Init, Mausklick auswerten. 22701
::ModEntry_3 jmp M2EF6 ;Mausklick auswerten. 22702
::ModEntry_4 jmp M2F20 ;Rechteck für Eintrag. 22703
::ModEntry_5 jmp M2FB8 ;Anz. Icons/Seite berechnen. 22704
::ModEntry_6 jmp M2FC0 ;Zeiger auf DnD-icon. 22705
; 22706
; 0: Initialisierung, Inhalt anzeigen. 22707
; TD13: _MyDispFiles 22708
:M2BB5 jsr S5875 ;Ber. Fens.-Inhalt. 22709
; 22710
; 1: Inhalt anzeigen. 22711
; Übergabe: r0 = Zeiger auf DIRBUF. 22712
:M2BB8 lda V18AD ;Fens.-Aufbau-Status. 22713
beq :1 ; => Kein Aufbau. 22714
; 22715
ldx WIN042F ;Aktives Fenster. 22716
lda #$00 ;Anzahl Icons/Zeile 22717
sta v2EC0,x ;zurücksetzen. 22718
; 22719
::1 sta v2E3F 22720

```

```

22721
22722             lda    #4 +1                ;Zuerst 5 Icons
22723             sta    v2E3E                ;versuchen auf dem
22724                                     ;Bildschirm zeigen.
22725                                     ;Dabei werden die
22726                                     ;Icons gezählt, die
22727                                     ;in eine Icon-Zeile
22728                                     ;passen (1-4).
22729
22730             jsr     z2BD6                ;Icons ausgeben.
22731
22732             lda    #$00
22733             sta    V18AD                ;Fens.-Aufbau-Status.
22734             rts
22735
22736 ; Darstellung von Dateieinträgen.
22737 ; TD13: _DispFiles
22738 :z2BD6             ldx    WIN042F                ;Aktives Fenster.
22739             lda    V0537,x                ;Fens.Mod. Icon/Text.
22740             cmp    C04B2                ;Richtiger Modus?
22741             beq    :1                ; => Ja, weiter...
22742
22743             ldx    #$00                ;Ende.
22744             rts
22745
22746 ::1             lda    C04F8,x                ;Lfwk. für Fenster.
22747             jsr     SetDevice
22748
22749             lda    r2L                ;Y-Koordinate Icon
22750             sta    a3H                ;oben/links im Fens.
22751
22752             lda    #0                ;Füllmuster.
22753             jsr     SetPattern
22754
22755 ; Hinweis:
22756 ; Diese Routine löscht den inneren
22757 ; Fensterbereich. Dabei werden die
22758 ; folgenden Routinen ausgeführt:
22759 ;             jsr    S11D0                ;Textgrenzen r5-r7.
22760 ;             jsr    S1D1C                ;Schnittfläche?
22761 ;             bcs    z2C2E                ; => Nicht vorhanden.
22762 ;             jsr    Rectangle
22763 ; Der Code ähnelt daher dem Code hinter
22764 ; dem JMP-Befehl, nur wird der Bereich
22765 ; nicht um 8 Pixel reduziert.
22766             jsr    S14EB                ;Rechteck zeichnen.
22767
22768 if FALSE
22769             jmp     z2C2E
22770
22771 ; TODO:
22772 ; Dieser Code wird aktuell nicht mehr
22773 ; ausgeführt, evtl. Debugging-Code:
22774 ; Wird der Code ausgeführt, dann wird
22775 ; der innere Fensterbereich mit einem

```

```

; Füllmuster gefüllt. 22776
;--- 22777
::x2BF9      jsr    S11D0      ;Textgrenzen r5-r7. 22778
              jsr    S1D1C      ;Schnittfläche? 22779
              bcs    z2C2E      ; => Nicht vorhanden. 22780
              22781
              lda    r2L 22782
              clc 22783
              adc    #ICON_OFF_Y0 22784
              sta    r2L 22785
              22786
              sec 22787
              lda    r2H 22788
              sbc    #ICON_OFF_YU 22789
              sta    r2H 22790
              22791
              lda    r4L 22792
              sec 22793
              sbc    #< ICON_OFF_XR 22794
              sta    r4L 22795
              lda    r4H 22796
              sbc    #> ICON_OFF_XR 22797
              sta    r4H 22798
              22799
              lda    #< ICON_OFF_XL 22800
              clc 22801
              adc    r3L 22802
              sta    r3L 22803
              lda    #> ICON_OFF_XL 22804
              adc    r3H 22805
              sta    r3H 22806
              22807
              lda    #0      ;Füllmuster. 22808
;          lda    #2      ;Testmuster. 22809
              jsr    S140C      ;Rechteck zeichnen. 22810
              22811
; Hinweis: 22812
; r2 bis r4 sind jetzt verändert! 22813
; (Textausgabe nur im mark.Bereich) 22814
;--- 22815
endif 22816
              22817
:z2C2E      ldx    WIN042F      ;Aktives Fenster. 22818
              jsr    S61E9      ;a9=Fensterfarben. 22819
              22820
if ENABLE_DIFFICOL = FALSE 22821
::4          ldy    #$10 22822
              lda    (a9),y      ;Fensterfarbe. 22823
endif 22824
if ENABLE_DIFFICOL = TRUE 22825
::4          lda    V621A      ;Fensterfarbe. 22826
endif 22827
              22828
              jsr    S1561      ;Screencolor setzen. 22829
              22830

```

```

22831          jsr    S13E5          ;Backup/Set TextWin.
22832
22833          lda    #$00
22834          sta    a3L
22835 ; Hinweis:
22836 ; Siehe Z1E36, die Y-Koordinate wird
22837 ; mehrfach korrigiert bis die obere
22838 ; Y-Koordinate auf CARD+1Pixel zeigt.
22839          lda    a3H
22840          clc
22841          adc    #8 +ICON_SKIP_Y
22842          sta    a3H
22843
22844          lda    r0H          ;Adresse DIRBUF.
22845          pha
22846          lda    r0L
22847          pha
22848          jsr    z2E4E          ;Datei-Nr. ausgeben.
22849          pla
22850          sta    r0L
22851          pla
22852          sta    r0H          ;Adresse DIRBUF.
22853
22854          lda    r0L          ;Zeiger auf Namen/a1,
22855          sta    a1L          ;Icons/a2 berechnen.
22856          clc
22857          adc    #< SIZE_DIRNAM
22858          sta    a2L
22859          lda    r0H
22860          sta    a1H
22861          adc    #> SIZE_DIRNAM
22862          sta    a2H
22863
22864 ; Nächste Icon-Zeile anzeigen.
22865 :z2C69      lda    #$00
22866          sta    v2EB8
22867
22868          ldy    #$00
22869          lda    (a1),y          ;Datei vorhanden?
22870          bne    :1              ; => Ja, weiter...
22871
22872          jmp    z2E11
22873
22874 ; Hinweis:
22875 ; In Byte#0 des Datei-Icons steht der
22876 ; GEOS-Dateityp. Bei "Nicht-GEOS" ist
22877 ; der Wert = $00, daher wird im GEOS-
22878 ; Dateityp ein zus. Bit gesetzt wenn
22879 ; das Icon bereits eingelesen wurde.
22880 ::1         lda    (a2),y          ;Icon vorhanden?
22881          bne    x2C89          ;Bit%7=1, weiter...
22882
22883          jsr    z2E5F          ;Icon einlesen.
22884          txa          ;Fehler?
22885          beq    x2C89          ; => Nein, weiter...

```

			22886
;	txa		22887
	pha		22888
	jsr	S524C ;Dat.-Ausw. aufheben.	22889
	pla		22890
	tax		22891
	rts		22892
			22893
;	Ausgabe Datei-Icon.		22894
:x2C89	lda	a4L	22895
	sta	r3L	22896
	clc		22897
	adc	#< 24	22898
	sta	r4L	22899
	lda	a4H	22900
	sta	r3H	22901
	adc	#> 24	22902
	sta	r4H	22903
			22904
	lda	a3H	22905
	sta	r2L	22906
	clc		22907
	adc	#21	22908
	sta	r2H	22909
			22910
	jsr	S11D0 ;Textgrenzen r5-r7.	22911
			22912
;	Hinweis:		22913
;	Hier nur der rechte Rand für das		22914
;	Fenster benötigt.		22915
	lda	r4L ;Nur rechten Rand	22916
	sta	v2EB9 +0 ;zwischenspeichern.	22917
	lda	r4H	22918
	sta	v2EB9 +1	22919
			22920
	jsr	S1D1C ;Schnittfläche?	22921
	bcc	:2 ; => Ja, weiter...	22922
			22923
	lda	V18AD ;Fens.-Aufbau-Status.	22924
	bne	:4 ; => Fenster-Aufbau.	22925
	inc	v2EB8 ;Geschl.Fenster +1.	22926
			22927
::2	lda	a0H	22928
	pha		22929
	lda	a0L	22930
	pha		22931
	ldx	WIN042F ;Aktives Fenster.	22932
	jsr	S11A9 ;a0=Fensterdaten.	22933
	jsr	S11C0 ;Fens.daten > r5-r7.	22934
	pla		22935
	sta	a0L	22936
	pla		22937
	sta	a0H	22938
			22939
	lda	r7L ;Rechten Rand	22940

```

22941          sec                      ;korrigieren.
22942          sbc    #< ICON_OFF_XR +1
22943          sta    r7L
22944          lda    r7H
22945          sbc    #> ICON_OFF_XR +1
22946          sta    r7H
22947
22948          lda    v2EB9 +1              ;Highbyte rechter
22949          cmp    r7H                  ;Fensterrand.
22950          beq    :3
22951          bcs    :4
22952          bcc    :8
22953
22954 ::3          lda    v2EB9 +0          ;Lowbyte rechter
22955          cmp    r7L                  ;Fensterrand.
22956          bcc    :8
22957
22958 ::4          lda    #4*3
22959          sta    v2E3E
22960
22961          lda    r5H
22962          ldx    V18AD                ;Fens.-Aufbau-Status.
22963          beq    :5                  ; => Kein Aufbau.
22964
22965          sec
22966          sbc    #ICON_YOFF +2        ;Ist eine weitere
22967                                     ;Icon-Zeile möglich?
22968 ::5          cmp    r2L
22969          bcs    :7                  ; => Ja, weiter...
22970
22971 ::6          jmp    z2E40            ; => Nein, Ende...
22972
22973 ::7          lda    v2E3F
22974          cmp    #$03                ;Letzte Zeile?
22975          beq    :6                  ; => Ja, Ende...
22976
22977          inc    v2E3F                ;Zeile +1.
22978
22979          lda    a3H
22980          clc
22981          adc    #ICON_YOFF
22982          sta    a3H
22983
22984          jsr    z2E4E                ;Datei-Nr. ausgeben.
22985          jmp    z2C69
22986
22987 ::8          dec    v2E3E            ;Noch ein Icon?
22988          beq    :4                  ; => Nein, Ende...
22989
22990          lda    v2EB8                ;Geschl.Fenster?
22991          bne    z2D86                ; => Nein, weiter...
22992
22993          lda    V18AD                ;Fens.-Aufbau-Status.
22994          beq    :9                  ; => Kein Aufbau.
22995

```



```

        ldx  WIN042F          ;Aktives Fenster.      22996
        lda  v2E3F           ;Erste Zeile?          22997
        bne  :9              ; => Nein, weiter...     22998
                                           22999
        inc  v2EC0,x         ;Icons/Zeile +1.        23000
                                           23001
::9      lda  a2L             23002
        sta  r0L             23003
        lda  a2H             23004
        sta  r0H             23005
                                           23006
        lda  a4L             23007
        sta  r10L            23008
        lda  a4H             23009
        sta  r10H            23010
                                           23011
        lda  #3              ;Breite immer 3Card.    23012
        sta  r13L            23013
        lda  #21             ;Höhe 21Pixel.         23014
        sta  r13H            23015
                                           23016
        lda  r4L             23017
        bne  :a              23018
        dec  r4H             23019
::a      dec  r4L            23020
                                           23021
; Hinweis:                               23022
; In der Tabelle der Fensterfarben ist   23023
; in Byte $10 im Lownibble der Wert für  23024
; die Hintergrundfarbe gespeichert, in   23025
; den Bytes $00-$0F im Highnibble der   23026
; Wert für die Vordergrundfarbe des     23027
; Datei-Icons.                          23028
if ENABLE_DIFFICOL = FALSE              23029
        ldy  #$10            23030
        lda  (a9),y          ;Fensterfarbe.        23031
endif                                   23032
if ENABLE_DIFFICOL = TRUE                23033
        lda  V621A           ;Fensterfarbe.        23034
endif                                   23035
        and  #%00001111      ;Lownibble isolieren. 23036
        tax                                   23037
                                           23038
; Hinweis:                               23039
; Der GEOS-Dateityp ist hier evtl. mit   23040
; Bit%7=1 verknüpft = Icon vorhanden.    23041
        ldy  #$00            23042
        lda  (a2),y          ;Dateityp einlesen.   23043
        pha                                   23044
        and  #%01111111      ;Bit%7 löschen.       23045
        cmp  #16             ;Max. 16 Farben.       23046
        bcc  :10             23047
        lda  #0              ;Unbek. Dateityp.      23048
::10     tay                                   23049
        txa                                   23050

```

```

23051          ora    (a9),y          ;Farbe für Icon.
23052          jsr    S1561          ;Screencolor setzen.
23053
23054 ; Hinweis:
23055 ; An der Pos. des Kopfbytes für das
23056 ; Datei-Icon steht der GEOS-Dateityp.
23057          lda    #$bf          ;Kopfbyte für
23058          ldy    #$00          ;Icon setzen.
23059          sta    (a2),y
23060
23061          ldx    a3H
23062          jsr    S1E6F          ;Icon ausgeben.
23063
23064          pla
23065          ldy    #$00
23066          sta    (a2),y
23067
23068 ; Ausgabe Dateiname.
23069 ; Übergabe: a1 = Zeiger Dateiname.
23070 ;          a4 = X-Koordinate.
23071 ;          a3 = Y-Koordinate.
23072 :z2D86          lda    a4L
23073          clc
23074          adc    #< 24/2          ;Icon 24 Pix. breit.
23075          sta    r11L          ;Ausgabe Name mittig
23076          lda    a4H          ;unter Icon.
23077          adc    #> 24/2
23078          sta    r11H
23079
23080          lda    a1L          ;Zeiger auf
23081          sta    r0L          ;Dateiname setzen.
23082          lda    a1H
23083          sta    r0H
23084
23085          ldy    #$10
23086          lda    (r0),y          ;Tr/Se Infoblock
23087          pha          ;zwischenspeichern.
23088          lda    #NULL
23089          sta    (r0),y
23090
23091          jsr    S244C          ;Hole Stringlänge.
23092
23093          lda    r1L
23094          sta    r4L
23095          lda    r1H
23096          sta    r4H
23097
23098          lsr    r1H          ;Länge / 2.
23099          ror    r1L
23100
23101          lda    r11L          ;X-Koordinate =
23102          sec          ;X - Länge / 2.
23103          sbc    r1L
23104          sta    r11L
23105          lda    r11H

```

	sbc	r1H		23106
	sta	r11H		23107
				23108
	lda	a3H	;Y-Koordinate =	23109
	clc		;Y + Höhe +6.	23110
	adc	#21 +NAME_BASELINE		23111
	sta	r1H		23112
				23113
	lda	r11L	;Prüfen ob Name	23114
	sta	r3L	;sichtbar ist.	23115
	clc			23116
	adc	r4L		23117
	sta	r4L		23118
	lda	r11H		23119
	sta	r3H		23120
	adc	r4H		23121
	sta	r4H		23122
				23123
	lda	r1H		23124
	clc			23125
	adc	#1		23126
	sta	r2H		23127
	sec			23128
	sbc	#7		23129
	sta	r2L		23130
				23131
	jsr	S11D0	;Textgrenzen r5-r7.	23132
	jsr	S1D1C	;Schnittfläche?	23133
	bcs	:3	; => Nicht vorhanden.	23134
				23135
; Name sichtbar, ausgeben.				23136
	ldy	#\$00		23137
::1	lda	(r0),y		23138
	beq	:3		23139
	cmp	#\$a0		23140
	beq	:3		23141
	sty	r15L		23142
	jsr	SmallPutChar		23143
	ldy	r15L		23144
	iny			23145
	bne	:1		23146
				23147
::3	ldy	#\$10		23148
	pla		;Tr/Se Infoblock	23149
	sta	(a1),y	;zurücksetzen.	23150
				23151
; Nächste Icon-Zeile.				23152
:z2E11	ldx	a3L		23153
	inx			23154
	stx	a3L		23155
	cpx	#4*4	;Max. 16 Icons?	23156
	beq	z2E40	; => Ja, Ende...	23157
				23158
	lda	#< ICON_XOFF	;Position für	23159
	clc		;nächstes Icon/Zeile.	23160

```

23161          adc    a4L
23162          sta    a4L
23163          bcc    :1
23164          inc    a4H
23165
23166 ::1          lda    #18                      ;Nächste Datei.
23167          clc
23168          adc    a1L
23169          sta    a1L
23170          bcc    :2
23171          inc    a1H
23172
23173 ::2          lda    #64                      ;Nächstes Icon.
23174          clc
23175          adc    a2L
23176          sta    a2L
23177          bcc    :3
23178          inc    a2H
23179
23180 ::3          jmp    z2C69
23181
23182 :v2E3E      b $0a    ;Max.Icons ausgeben (4/12).
23183 :v2E3F      b $00    ;Zeile 0-3.
23184
23185 ; Ausgewählte Icons markieren.
23186 :z2E40      lda    WIN042F                    ;Aktives Fenster.
23187          cmp    C042A                        ;Oberstes Fenster.
23188          bne    :1                          ; => Nein, weiter...
23189
23190 ; Nur im obersten Fenster markieren.
23191          jsr    S5842                        ;mark.Dat. anzeigen.
23192
23193 ::1          ldx    #$00
23194          rts
23195
23196 ; Zeiger auf Icon, ggf. aus Infoblock einlesen.
23197 :z2E5F      ldy    #16
23198          lda    (a1),y                      ;Infoblock?
23199          beq    :2                          ; => Nein, weiter...
23200          sta    r1L
23201          iny
23202          lda    (a1),y
23203          sta    r1H                        ;Zeiger Infoblock.
23204
23205 ;          lda    #< diskBlkBuf                ;Zeiger auf
23206 ;          sta    r4L                        ;Zwischenspeicher.
23207 ;          lda    #> diskBlkBuf
23208 ;          sta    r4H
23209          jsr    GetBlock_dskBuf
23210          txa
23211          bne    :err
23212
23213          ldy    #64 -1
23214 ::1          lda    diskBlkBuf +4,y
23215          sta    (a2),y

```



```

23271 ; Rechter Rand für Fenster.
23272 :v2EB9          w $0000
23273
23274 :v2EC0          s 4 ;Anzahl Icons in Zeile für Fens.#1-#4.
23275
23276 ; Datei-Nr. ausgeben.
23277 ; Übergabe: a5 = X-Koordinate Arbeitsbereich.
23278 :z2E4E          lda  a5L
23279                clc
23280                adc  #< FNUM_XOFF2
23281                sta  a4L
23282                lda  a5H
23283                adc  #> FNUM_XOFF2
23284                sta  a4H
23285
23286 ; Datei-Nr. ausgeben.
23287 ; TD13: DispNumber
23288 :z2EC4          lda  a3L ;Eintrag-Nr.
23289                ldx  WIN042F ;Aktives Fenster.
23290                clc
23291                adc  C050B,x ;Fensterposition.
23292                sta  r0L
23293                inc  r0L
23294                lda  #$00 ;Highbyte = 0.
23295                sta  r0H
23296
23297                lda  a4L ;X-Koordinate.
23298                sec
23299                sbc  #< FNUM_XOFF
23300                sta  r11L
23301                lda  a4H
23302                sbc  #> FNUM_XOFF
23303                sta  r11H
23304
23305                lda  a3H ;Y-Koordinate.
23306                clc
23307                adc  #FNUM_YOFF
23308                sta  r1H
23309
23310                lda  #SET_LEFTJUST!SET_SUPRESS
23311                jmp  PutDecimal
23312
23313 ; 2: Initialisierung, Mausklick auswerten.
23314 ; TD13: _MyCheckFiles
23315 :M2EEE          jsr  M2FB8 ;Anz.Ze/Sp und Icons.
23316
23317                pha  ;Anz. Icons/Seite.
23318                jsr  S5875 ;Ber. Fens.-Inhalt.
23319                pla
23320
23321 ; 3: Mausklick auswerten.
23322 ; Übergabe: A = Anzahl Icons/Seite.
23323 ; TD13: _CheckFiles
23324 :M2EF6          sta  a8H
23325

```

	lda	r2L		23326
	sta	a2L		23327
				23328
	lda	a5L		23329
	sta	a3L		23330
	lda	a5H		23331
	sta	a3H		23332
				23333
	lda	#0		23334
::1	jsr	A2BB2	;Rechteck f.Eintrag.	23335
	bcs	:2	; => Kein Eintrag...	23336
	pha			23337
	jsr	IsMseInRegion		23338
	bne	:3		23339
	pla			23340
::2	clc			23341
	adc	#1		23342
	cmp	a8H	;Wird verändert.	23343
	bne	:1		23344
				23345
	ldx	#\$ff	;Nicht gefunden.	23346
	rts			23347
				23348
::3	pla			23349
	tax		;Icon 0-(MAXFILES-1).	23350
	rts			23351
				23352
				23353
				23354
				23355
				23356
				23357
				23358
				23359
				23360
				23361
	ldy	WIN042F	;Aktives Fenster.	23362
	lda	v2EC0,y	;Anzahl Icons/Zeile.	23363
	sta	r4L		23364
	lda	#\$00		23365
	sta	r4H		23366
				23367
	ldx	#r3		23368
	ldy	#r4		23369
	jsr	Ddiv		23370
				23371
	lda	r3L		23372
	sta	r2L		23373
	lda	r3H		23374
	sta	r2H		23375
				23376
	lda	r8L		23377
	sta	r3L		23378
	lda	r8H		23379
	sta	r3H		23380

```

23381
23382         lda    #< ICON_XOFF
23383         sta    r4L
23384         lda    #> ICON_XOFF
23385         sta    r4H
23386
23387         ldx    #r3
23388         ldy    #r4
23389         jsr    DMult
23390
23391         lda    #31
23392         clc
23393         adc    r3L
23394         sta    r3L
23395         bcc    :1
23396         inc    r3H                ;X-Koordinate links.
23397
23398 ::1         lda    #< ICON_YOFF
23399         sta    r4L
23400         lda    #> ICON_YOFF
23401         sta    r4H
23402
23403         ldx    #r2
23404         ldy    #r4
23405         jsr    DMult
23406
23407         lda    r2L
23408         clc
23409         adc    #13
23410         sta    r2L                ;Y-Koordinate oben.
23411
23412         clc
23413         lda    a2L
23414         adc    r2L
23415         sta    r2L
23416         bcs    :2                ;Y-Koord. zu groß!
23417
23418         clc
23419         adc    #21 -1
23420         sta    r2H                ;Y-Koordinate unten.
23421
23422         lda    r3L
23423         clc
23424         adc    #< 24 -1
23425         sta    r4L
23426         lda    r3H
23427         adc    #> 24 -1
23428         sta    r4H                ;X-Koordinate rechts.
23429
23430         lda    a3L
23431         clc
23432         adc    r3L
23433         sta    r3L
23434         lda    a3H
23435         adc    r3H

```



	sta	r3H		23436
				23437
	lda	a3L		23438
	clc			23439
	adc	r4L		23440
	sta	r4L		23441
	lda	a3H		23442
	adc	r4H		23443
	sta	r4H		23444
				23445
	jsr	S11D0	;Textgrenzen r5-r7.	23446
	jsr	S1D1C	;Schnittfläche?	23447
				23448
	pla			23449
	rts			23450
				23451
::2	pla			23452
	sec		;Rechteck ungültig.	23453
	rts			23454
				23455
	; 5: Anzahl Icons/Seite berechnen.			23456
:M2FB8	ldx	WIN042F	;Aktives Fenster.	23457
	jsr	S1E09	;Arbeitsber.Fenster.	23458
				23459
	; Anzahl Zeilen berechnen.			23460
	lda	r2H		23461
	sec			23462
	sbc	r2L		23463
	sbc	#9	;Abzug Rand oben.	23464
				23465
	ldy	#1	;Mind. 1 Zeile hoch.	23466
	sec		;Bereits gesetzt.	23467
::1	sbc	#ICON_YOFF	;Icon-Höhe = \$18	23468
	bcc	:2	;Icon-Titel = \$08	23469
	iny			23470
	cpy	#4	;max. 4 Zeilen.	23471
	bne	:1		23472
				23473
::2	ldx	WIN042F	;Aktives Fenster.	23474
	tya			23475
	sta	V4CC3,x		23476
				23477
	; Anzahl Spalten berechnen.			23478
	lda	r4L		23479
	sec			23480
	sbc	r3L		23481
	sta	r4L		23482
	lda	r4H		23483
	sbc	r3H		23484
	sta	r4H		23485
				23486
	ldy	#1		23487
				23488
	lda	r4L		23489
	sec			23490

```

23491          sbc    #< ICON_XOFF
23492          sta    r4L
23493          lda    r4H
23494          sbc    #> ICON_XOFF
23495          sta    r4H
23496          bcc    :4
23497
23498 ::3          lda    r4L
23499          sec
23500          sbc    #< ICON_XOFF
23501          sta    r4L
23502          lda    r4H
23503          sbc    #> ICON_XOFF
23504          sta    r4H
23505          bcc    :4
23506
23507          iny
23508          cpy    #4                      ;Max 4 Icons/Zeile.
23509          bne    :3
23510
23511 ::4          ldx    WIN042F              ;Aktives Fenster.
23512          tya
23513          sta    V4CC7,x
23514
23515 ; Anzahl Icons/Seite berechnen.
23516          ldy    V4CC3,x
23517          lda    #$00
23518 ::5          clc
23519          adc    V4CC7,x
23520          dey
23521          bne    :5
23522
23523          cmp    #MAXWINFILES
23524          bcc    :6
23525          lda    #MAXWINFILES
23526 ::6          sta    V4CCB,x
23527          rts
23528
23529 ; 6: Zeiger auf DnD-Icon.
23530 :M2FC0      lda    V04B7                  ;Anzahl mark.Dateien.
23531          cmp    #1                      ;Nur eine Datei?
23532          beq    :1                      ; => Ja, weiter...
23533
23534 ; DnD-Sprite: Mehrfachauswahl.
23535          ldx    #< GR4DDD                  ;MultiFile-Modus.
23536          ldy    #> GR4DDD
23537          rts
23538
23539 ; DnD-Sprite: Einzelne Datei.
23540 ::1          lda    a1L
23541          sta    a2L
23542          lda    #$00                      ;Highbyte immer 0.
23543          sta    a2H
23544
23545          lda    #< 64                      ;Icon-Größe:

```

sta	a3L	; \$bf + 63 Byte Daten.	23546
lda	#> 64		23547
sta	a3H		23548
			23549
ldx	#a2		23550
ldy	#a3		23551
jsr	DMult		23552
			23553
ldx	C042A	; Oberstes Fenster.	23554
jsr	S599A	; Zeiger auf DIRBUF.	23555
			23556
lda	#< 16*18 +1	; Zeiger auf erstes	23557
clc		; Byte hinter dem	23558
adc	r1L	; Icon-Kopfbyte \$bf.	23559
sta	r1L		23560
lda	#> 16*18 +1		23561
adc	r1H		23562
sta	r1H		23563
			23564
lda	r1L	; Zeiger auf Datei-	23565
clc		; Icon berechnen.	23566
adc	a2L		23567
tax			23568
lda	r1H		23569
adc	a2H		23570
tay			23571
rts			23572
			23573
; Commodore-Icon (Nicht-GEOS-Dateien).			23574
:GR2FBA	b \$bf		23575
	b %00000000,%00000000,%11111100		23576
	b %00000000,%00000001,%00000010		23577
	b %11111111,%11111110,%00000001		23578
	b %10000000,%00000000,%00000001		23579
	b %10000000,%11111100,%00000001		23580
	b %10000001,%11111100,%00000001		23581
	b %10000011,%10000011,%11100001		23582
	b %10000011,%10000011,%11000001		23583
	b %10000011,%10000000,%00000001		23584
	b %10000011,%10000011,%11000001		23585
	b %10000011,%10000011,%11100001		23586
	b %10000001,%11111100,%00000001		23587
	b %10000000,%11111100,%00000001		23588
	b %10000000,%00000000,%00000001		23589
	b %10000001,%11000001,%00000001		23590
	b %10000010,%00000011,%00000001		23591
	b %10000011,%11000101,%00000001		23592
	b %10000010,%00101111,%10000001		23593
	b %10000001,%11000001,%00000001		23594
	b %10000000,%00000000,%00000001		23595
	b %11111111,%11111111,%11111111		23596
			23597
; Verzeichnis-Icon.			23598
:GR2FFA	b \$bf		23599
	b %00000000,%00000001,%11111110		23600

```

23601      b %00000000,%00000010,%00000001
23602      b %00111111,%11111110,%00000001
23603      b %01000000,%00000000,%00000001
23604      b %01000000,%00000000,%00000001
23605      b %01111111,%11111111,%11111101
23606      b %10000000,%00000000,%00000011
23607      b %10000000,%00000000,%00000011
23608      b %10000000,%00000000,%00000011
23609      b %10111111,%00111011,%11110011
23610      b %10111011,%10111011,%10111011
23611      b %10111011,%10111011,%10111011
23612      b %10111011,%10111011,%10111011
23613      b %10111011,%10111011,%11110011
23614      b %10111011,%10111011,%11110011
23615      b %10111111,%00111011,%10111011
23616      b %10000000,%00000000,%00000011
23617      b %10000000,%00000000,%00000011
23618      b %10000000,%00000000,%00000011
23619      b %10000000,%00000000,%00000010
23620      b %01111111,%11111111,%11111100
23621
23622 ; MultiFile-Icon.
23623 :GR4DDD      b %11111111,%11111111,%11111111
23624      b %10000000,%00000000,%00000001
23625      b %10000000,%00000000,%00000001
23626      b %10100000,%10000001,%00001001
23627      b %10110001,%10000001,%01000001
23628      b %10101010,%10100101,%01101001
23629      b %10100100,%10100101,%01001001
23630      b %10100000,%10100101,%01001001
23631      b %10100000,%10100101,%01001001
23632      b %10100000,%10011101,%00101001
23633      b %10000000,%00000000,%00000001
23634      b %10000111,%01010000,%00000001
23635      b %10000100,%00010000,%00000001
23636      b %10000100,%01010011,%00000001
23637      b %10000111,%01010100,%10000001
23638      b %10000100,%01010111,%10000001
23639      b %10000100,%01010100,%00000001
23640      b %10000100,%01010011,%10000001
23641      b %10000000,%00000000,%00000001
23642      b %10000000,%00000000,%00000001
23643      b %11111111,%11111111,%11111111
23644
23645 ; Endadresse testen:
23646      g BASE_MOD_B + SIZE_FSERVICE
23647

```

```

; 23648
; 23649
; SourceCode: src.mod#06.s 23650
; 23651
; 23652
; 23653
; Modul-Info: 23654
; - Anzeige: Text-Modus 23655
; 23656
if .p 23657
    t "TopSym" 23658
    t "TopSym.MP3" 23659
    t "TopSym.ROM" 23660
    t "DTopDeskI18N.ext" 23661
    t "src.DTopDesk.ext" 23662
; 23663
; Spaltenpositionen: 23664
:TAB_NAME = $000a 23665
:TAB_SIZE = $005a 23666
:TAB_GTYPE = $0073 23667
:TAB_DATE = $00b4 23668
:TAB_TIME = $00df 23669
:ALTNR = 2 23670
endif 23671
; 23672
o BASE_MOD_B 23673
n "obj.#06" 23674
; 23675
::ModEntry_0 jmp M2BBB ;Init, Inhalt anzeigen. 23676
::ModEntry_1 jmp M2BBE ;Inhalt anzeigen. 23677
::ModEntry_2 jmp M2EA4 ;Init, Mausklick auswerten. 23678
::ModEntry_3 jmp M2EA7 ;Mausklick auswerten. 23679
::ModEntry_4 jmp M2ECE ;Rechteck für Eintrag. 23680
::ModEntry_5 jmp M2F00 ;Anz. Zeilen/Seite berechnen. 23681
::ModEntry_6 jmp M2F08 ;Zeiger auf DnD-Icon. 23682
::ModEntry_7 jmp M2F18 ;Dateien sortieren. 23683
::ModEntry_8 jmp M30DA ;mark.Datei suchen. 23684
; 23685
; 0: Initialisierung, Inhalt anzeigen. 23686
; TD13: __MyDispFiles 23687
:M2BBB jsr S5875 ;Ber. Fens.-Inhalt. 23688
; 23689
; 1: Inhalt anzeigen. 23690
; TD13: __DispFiles 23691
:M2BBE ldx WIN042F ;Aktives Fenster. 23692
lda V0537,x ;Fens.Mod. Icon/Text. 23693
cmp C04B2 ;Richtiger Modus? 23694
beq :1 ; => Ja, weiter... 23695
rts 23696
; 23697
::1 lda r0L 23698
clc 23699
adc #< SIZE_DIRNAM 23700
sta a6L 23701
lda r0H 23702

```

```

23703          adc    #> SIZE_DIRNAM
23704          sta    a6H
23705
23706          lda    r2L
23707          clc
23708          adc    #2
23709          sta    r1H
23710
23711          lda    a5L          ;X-Koordinate für
23712          sta    a3L          ;Zeilenanfang.
23713          lda    a5H
23714          sta    a3H
23715
23716          lda    #$00
23717          jsr    SetPattern
23718          jsr    S14EB        ;Rechteck zeichnen.
23719          jsr    S13E5        ;Backup/Set TextWin.
23720
23721 ; Max. 16 Zeilen ausgeben.
23722          ldx    #MAXWINFILES
23723 ::loop      stx    a2L
23724
23725          lda    r1H
23726          pha
23727          clc
23728          adc    #TEXT_YOFF
23729          sta    r1H
23730
23731          pla
23732          cmp    windowBottom ;Eintrag im Fenster?
23733          bcs    :next        ; => Nein, weiter...
23734
23735          lda    r1H
23736          cmp    windowTop    ;Eintrag im Fenster?
23737          bcc    :next        ; => Nein, weiter...
23738
23739          lda    #< TAB_NAME
23740          ldx    #> TAB_NAME
23741          jsr    z2D50        ;X-Koord. Name.
23742
23743          ldy    #0
23744          lda    (r0),y        ;Weitere Datei?
23745          beq    :done        ; => Nein, Ende...
23746
23747 ::11        lda    (r0),y
23748          cmp    #$a0          ;Ende Dateiname?
23749          beq    :12          ; => Ja, weiter...
23750          sty    a2H
23751          jsr    SmallPutChar
23752          ldy    a2H
23753          iny
23754          cpy    #16
23755          bne    :11
23756
23757 ::12        jsr    z2C6D        ;Datei-Info.

```

				23758
::next	lda	#18	;Zeiger auf den	23759
	jsr	z307B	;nächsten Dateinamen.	23760
				23761
	lda	#64	;Zeiger auf das	23762
	jsr	z3078	;nächste Icon.	23763
				23764
	ldx	a2L	;max. Anzahl an	23765
	dex		;Dateien ausgegeben?	23766
	bne	:loop	; => Nein, weiter...	23767
				23768
::done	lda	WIN042F	;Aktives Fenster.	23769
	cmp	C042A	;Oberstes Fenster.	23770
	bne	:exit	; => Nein, Ende...	23771
				23772
	lda	leftMargin +0	;Linken Rand für	23773
	sec		;Markierung setzen.	23774
	sbc	#< 6		23775
	sta	leftMargin +0		23776
	lda	leftMargin +1		23777
	sbc	#> 6		23778
	sta	leftMargin +1		23779
				23780
	jsr	S5842	;mark.Dat. anzeigen.	23781
				23782
::exit	ldx	#\$00		23783
	rts			23784
				23785
			; GEOS-Informationen ausgeben.	23786
			; Ab der eigentlichen Iconposition ist	23787
			; eingetragen: CBM-Typ, Track, AltNr,	23788
			; Dateistruktur, GEOS-Dateityp, Jahr,	23789
			; Monat, Tag, Stunde, Minute, Größe.	23790
			; AltNr ist die Position (0-x), welche	23791
			; die Datei auf der Diskette hat.	23792
			; TD13: :sub	23793
:z2C6D	lda	r0H		23794
	pha			23795
	lda	r0L		23796
	pha			23797
				23798
			; GEOS-Dateityp ausgeben.	23799
	lda	#< TAB_GTYPE		23800
	ldx	#> TAB_GTYPE		23801
	jsr	z2D50	;X-Koord. GEOS-Typ.	23802
				23803
	ldy	#0		23804
	lda	(a6),y		23805
	and	#FTYPE_MODES		23806
	cmp	#FTYPE_DIR		23807
	beq	:0		23808
				23809
	ldy	#4		23810
	lda	(a6),y		23811
	cmp	#TDFOLDER	;TopDesk-Ordner?	23812

```

23813          bne    :1          ; => Nein, weiter...
23814
23815 ; *TDFOLDERFIX*
23816 ;
23817 ; TopDesk behandelt den Typ #11 immer
23818 ; als Ordner, zusätzlich Daten-Track
23819 ; auswerten. Ein Ordner hat als Tr/Se
23820 ; $00/$ff in Byte #1/2, da ein Ordner
23821 ; keine Daten beinhaltet.
23822          pha
23823
23824          ldy    #1
23825          lda    (a6),y
23826          tax
23827
23828          pla
23829
23830          cpx    #$00          ;Track vorhanden?
23831          bne    :1          ; => Ja, Datei...
23832
23833          lda    #TDFOLDER_EXT ;TopDesk-Ordner.
23834          b $2c
23835 ::0          lda    #GTYPE_DIR ;Unterverzeichnis.
23836
23837 ::1          cmp    #GTYPESMAX
23838          bcc    :2
23839
23840          lda    #GTYPE_ERR    ;Unbekannt.
23841
23842 ::2          asl
23843          tax
23844          lda    v2D79 +0,x
23845          sta    r0L
23846          lda    v2D79 +1,x
23847          sta    r0H
23848          jsr    S14C8          ;String ausgeben.
23849
23850 ; Datum ausgeben.
23851          lda    #< TAB_DATE
23852          ldx    #> TAB_DATE
23853          jsr    z2D50          ;X-Koord. Datum.
23854
23855          ldy    #7          ;Tag.
23856          lda    (a6),y
23857          jsr    z2D60          ;Zahl 0-9 ausgeben.
23858          lda    #"."
23859          jsr    SmallPutChar
23860
23861          ldy    #6          ;Monat.
23862          lda    (a6),y
23863          pha
23864          jsr    z2D60          ;Zahl 0-9 ausgeben.
23865
23866          lda    #"."
23867          jsr    SmallPutChar

```



	pla		23868
	beq	:3	23869
			23870
; TODO:			23871
; Alternativ zusätzlich :millenium			23872
; überprüfen?			23873
	ldy	#5	23874
	lda	(a6),y ;Jahr.	23875
	pha		23876
	cmp	#80 ;Jahrtausend	23877
	bcc	:4 ;berechnen.	23878
			23879
	ldx	#"1"	23880
	lda	#"9"	23881
	bne	:5	23882
			23883
::3	pha		23884
			23885
	ldx	#"0"	23886
	b	\$2c	23887
::4	ldx	#"2"	23888
	lda	#"0"	23889
			23890
::5	pha		23891
	txa		23892
	jsr	SmallPutChar	23893
	pla		23894
	jsr	SmallPutChar	23895
	pla		23896
	jsr	z2D6B	23897
			23898
; Uhrzeit ausgeben.			23899
	lda	#< TAB_TIME	23900
	ldx	#> TAB_TIME	23901
	jsr	z2D50 ;X-Koord. Zeit.	23902
			23903
	ldy	#8	23904
	lda	(a6),y ;Stunde.	23905
	and	##01111111	23906
	sta	r0L	23907
	lda	#\$00	23908
	sta	r0H	23909
			23910
	lda	#10!SET_RIGHTJUST!SET_SUPPRESS	23911
	jsr	S590A ;Zahl ausgeben.	23912
	lda	#" : "	23913
	jsr	SmallPutChar	23914
			23915
	ldy	#9	23916
	lda	(a6),y ;Minute.	23917
	jsr	z2D6B	23918
			23919
; Dateigröße ausgeben.			23920
	lda	#< TAB_SIZE	23921
	ldx	#> TAB_SIZE	23922

```

23923      jsr    z2D50      ;X-Koord. Dateigröße.
23924
23925      ldy    #10
23926      lda    (a6),y      ;Dateigröße.
23927      sta    r0L
23928      iny
23929      lda    (a6),y
23930      sta    r0H
23931
23932      lda    MT3E11
23933      cmp    #"*"      ;Blocks?
23934      beq    :6      ; => Ja, weiter...
23935
23936      lda    r0L      ;Blocks in KByte
23937      pha      ;umrechnen.
23938      lsr    r0H
23939      ror    r0L
23940      lsr    r0H
23941      ror    r0L
23942      pla
23943      and    #%00000011
23944      beq    :6
23945
23946      inc    r0L
23947      lda    r0L      ;???
23948      bne    :6
23949      inc    r0H
23950
23951  ::6      lda    #20!SET_RIGHTJUST!SET_SUPRESS
23952      jsr    S590A      ;Zahl ausgeben.
23953
23954      pla
23955      sta    r0L
23956      pla
23957      sta    r0H
23958      rts
23959
23960  ; Tabstop setzen.
23961  :z2D50      clc
23962      adc    a3L
23963      sta    r11L
23964      txa
23965      adc    a3H
23966      sta    r11H
23967      rts
23968
23969  ; Zahl 00-99 ausgeben.
23970  ; TD13: :num2
23971  :z2D6B      cmp    #10
23972      bcs    z2D60      ;Zahl 00-99 ausgeben.
23973      pha
23974      lda    #"0"      ;10er ausgeben.
23975      jsr    PutChar
23976      pla
23977  ;      jmp    z2D60      ;Zahl 0-9 ausgeben.

```

```

; Zahl 0-9 linksbündig ausgeben.
; TD13: :num
:z2D60          sta    r0L
                lda    #$00
                sta    r0H
                lda    #SET_LEFTJUST!SET_SUPRESS
                jmp     S590A          ;Zahl ausgeben.

; TODO:
; Die Liste mit Datei-Information
; zusammenlegen, Ausnahme für TopDesk-
; Ordner hinzufügen.
; TD13: TypTab
:v2D79          w t2DA7  ;$00 = Nicht-GEOS.
                w t2DB2  ;$01 = BASIC.
                w t2DB8  ;$02 = Assembler.
                w t2DC2  ;$03 = Daten.
                w t2DC7  ;$04 = Systemdatei.
                w t2DD3  ;$05 = Hilfsmittel.
                w t2DE1  ;$06 = Anwendung.
                w t2DEB  ;$07 = Dokument.
                w t2DF4  ;$08 = Zeichensatz.
                w t2E05  ;$09 = Druckertreiber.
                w t2E14  ;$0A = Eingabetreiber.
                w t2E28  ;$0B = Laufwerkstreiber/Ordner.
                w t2E32  ;$0C = Startprogramm.
                w t2E40  ;$0D = Temporär.
                w t2E49  ;$0E = Selbstausführend.
                w t2E5A  ;$0F = Eingabetreiber (C128).
:GTDIR          w t2E6F  ;$10 = Unterverzeichnis.
                w t2E80  ;$11 = gateWay-Dokument.
                w t2EA0  ;$12 = Unbekannt.
                w t2EA0  ;$13 = Unbekannt.
:GTERR          w t2EA0  ;$14 = Unbekannt.
                w t2E88  ;$15 = GeoShell-Befehl.
                w t2E91  ;$16 = GeoFax-Drucker.
                w t2E9A  ;$17 = TD-Ordner / TDFOLDER_EXT.

:GTYPESEND      = (GTYPESEND - v2D79)/2
:GTYPESEMAX     = (GTYPESEND - v2D79)/2
:GTYPE_ERR      = (GTERR - v2D79)/2
:GTYPE_DIR      = (GTDIR - v2D79)/2

if LANG = LANG_DE
:t2DA7          b "Nicht-GEOS",0
:t2DB2          b "BASIC",0
:t2DB8          b "Assembler",0
:t2DC2          b "Daten",0
:t2DC7          b "Systemdatei",0
:t2DD3          b "Hilfsprogramm",0
:t2DE1          b "Anwendung",0
:t2DEB          b "Dokument",0
:t2DF4          b "Zeichensatz",0
:t2E05          b "Druckertreiber",0
:t2E14          b "Eingabetreiber (64)",0

```

```

24033 :t2E28          b "Laufwerkstreiber",0
24034 :t2E32          b "Startprogramm",0
24035 :t2E40          b "Temporär",0
24036 :t2E49          b "Selbstausführend",0
24037 :t2E5A          b "Eingabetreiber (128)",0
24038 :t2E6F          b "Unterverzeichnis",0
24039 :t2E80          b "gateWay",0
24040 :t2E88          b "geoShell",0
24041 :t2E91          b "geoFax-Drucker",0
24042 :t2E9A          b "TopDesk-Ordner",0
24043 endif
24044 if LANG = LANG_EN
24045 :t2DA7          b "Not GEOS",0
24046 :t2DB2          b "BASIC",0
24047 :t2DB8          b "Assembler",0
24048 :t2DC2          b "Data",0
24049 :t2DC7          b "System file",0
24050 :t2DD3          b "Desk accessory",0
24051 :t2DE1          b "Application",0
24052 :t2DEB          b "Document",0
24053 :t2DF4          b "Font",0
24054 :t2E05          b "Printer driver",0
24055 :t2E14          b "Input driver (64)",0
24056 :t2E28          b "Disk driver",0
24057 :t2E32          b "Start file",0
24058 :t2E40          b "Temporary",0
24059 :t2E49          b "Auto exec",0
24060 :t2E5A          b "Input driver (128)",0
24061 :t2E6F          b "Directory",0
24062 :t2E80          b "gateWay",0
24063 :t2E88          b "geoShell",0
24064 :t2E91          b "geoFax printer",0
24065 :t2E9A          b "TopDesk folder",0
24066 endif
24067
24068 :t2EA0          b "???",0
24069
24070 ; 2: Initialisierung, Mausklick auswerten.
24071 ; TD13: __MyCheckFiles
24072 :M2EA4          jsr   S5875          ;Ber. Fens.-Inhalt.
24073
24074 ; 3: Mausklick auswerten.
24075 ; TD13: __CheckFiles
24076 :M2EA7          lda   r2L
24077              sta   a2L
24078
24079              lda   r3L
24080              sta   a3L
24081              lda   r3H
24082              sta   a3H
24083
24084              lda   #$00
24085 ::1            jsr   M2ECE          ;Schnittfläche?
24086              bcs   :2              ; => Nein, weiter...
24087

```

	pha		24088
	jsr	IsMseInRegion ;Maus auf Zeile?	24089
	bne	:3 ; => Ja, weiter...	24090
	pla		24091
			24092
::2	clc		24093
	adc	#1	24094
	cmp	#MAXWINFILES ;Max. 16 Zeilen?	24095
	bne	:1 ; => Nein, weiter...	24096
			24097
	ldx	#\$ff	24098
	rts		24099
			24100
::3	pla		24101
	tax		24102
	rts		24103
			24104
		; 4: Rechteck / Dateieintrag ermitteln.	24105
		; Ermittlung des Iconrechtecks einer	24106
		; Datei einer DispFile-Darstellung in	24107
		; Bezug auf das Textfenster.	24108
		; Übergabe: Textfenster	24109
		; -> windowTop / rightMargin	24110
		; Rückgabe: r2-r4: Rechteck-Koordinaten.	24111
		; TD13: __GetFileRect	24112
:M2ECE	pha		24113
			24114
		; Hinweis:	24115
		; Berechnung: Zeile x Zeilenhöhe	24116
		; => BBMULT zu zeitaufwändig.	24117
		; => ASL/ADC zu unflexibel für	24118
		; andere Zeilenhöhen.	24119
	tax		24120
			24121
	lda	#0 ;Pos. x Zeilenhöhe.	24122
	beq	:2	24123
::1	clc		24124
	adc	#TEXT_YOFF	24125
::2	dex		24126
	bpl	:1	24127
			24128
	clc		24129
	adc	windowTop ; + obere Grenze.	24130
	sta	r2L	24131
	adc	#TEXT_YOFF -1	24132
	sta	r2H ;Untere Grenze.	24133
			24134
	lda	a3L	24135
	clc		24136
	adc	#< 8	24137
	sta	r3L	24138
	lda	a3H	24139
	adc	#> 8	24140
	sta	r3H ;Linke Grenze.	24141
			24142

```

24143      lda    rightMargin +0
24144      sec
24145      sbc    #< 3
24146      sta    r4L
24147      lda    rightMargin +1
24148      sbc    #> 3
24149      sta    r4H          ;Rechte Grenze.
24150
24151      jsr    S11D0          ;Textgrenzen r5-r7.
24152
24153      lda    #8
24154      jsr    z307D
24155
24156      jsr    S1D1C          ;Schnittfläche?
24157
24158      pla
24159      rts
24160
24161 ; 5: Anzahl Zeilen/Seite berechnen.
24162 :M2F00      ldx    WIN042F          ;Aktives Fenster.
24163      lda    #1
24164      sta    V4CC7,x          ;Nur 1 Spalte.
24165      jsr    S1E09          ;Arbeitsber.Fenster.
24166
24167      lda    r2H
24168      sec
24169      sbc    r2L
24170      sta    r4L
24171      lda    #$00          ;Highbyte immer 0.
24172      sta    r4H
24173
24174      lda    #< TEXT_YOFF          ;Zeilenhöhe.
24175      sta    r3L
24176      lda    #> TEXT_YOFF
24177      sta    r3H
24178
24179      ldx    #r4
24180      ldy    #r3
24181      jsr    Ddiv
24182
24183      lda    r8L          ;Rest der Division.
24184      cmp    #4          ;Eintrag sichtbar?
24185      bcc    :1          ; => Nein, weiter...
24186      inc    r4L          ; + Halber Eintrag.
24187
24188 ::1      lda    r4L
24189      ldx    WIN042F          ;Aktives Fenster.
24190      sta    V4CCB,x          ;Anzahl Zeilen.
24191      rts
24192
24193 ; 6: Zeiger auf DnD-Icon.
24194 :M2F08      ldx    #< GR5181          ;DnD-Sprite für
24195      ldy    #> GR5181          ;Dateien tauschen.
24196      rts
24197

```

```

; 7: Dateien sortieren. 24198
; Ab VLIR-Ladeadresse ist der Speicher 24199
; wie folgt belegt: 24200
; $0000 = Word, NULL 24201
; 30-Byte-Verzeichniseinträge < 150. 24202
; Ab der eigentlichen Iconposition ist 24203
; eingetragen: CBM-Typ, Track, AltNr, 24204
; Dateistruktur, GEOS-Dateityp, Jahr, 24205
; Monat, Tag, Stunde, Minute, Größe. 24206
; AltNr ist die Position (0-x), welche 24207
; die Datei auf der Diskette hat. 24208
; TD13: __SortFileBuffer 24209
:M2F18          sta    a5L          ;Offset. 24210
                stx    a6L          ;Fenster-Nr. 24211
                24212
                jsr    S599A        ;Zeiger auf DIRBUF. 24213
                24214
; Hinweis: 24215
; Das sortieren von Dateien nach Name 24216
; funktioniert bis ca.150 Dateien, da 24217
; der Speicher sonst den Bereich ab 24218
; $8000 überschreiben würde. 24219
                ldx    #MAXFILES -1 ;Zähler 1-255. 24220
                24221
; Zeiger auf letzten Eintrag. 24222
:ENDFILESBUF    = BASE_MOD_A +2 +(MAXFILES -1)*30 24223
                lda    #< ENDFILESBUF 24224
                sta    a4L 24225
                lda    #> ENDFILESBUF 24226
                sta    a4H 24227
                24228
; *TDFOLDERFIX* 24229
; 24230
; TopDesk behandelt den Typ #11 immer 24231
; als Ordner, zusätzlich Daten-Track 24232
; auswerten. Dazu muss die Original- 24233
; Datei-Nr als AltNr in Byte #2 des 24234
; Eintrags (Sektor) und nicht in das 24235
; Byte #1 (Track) gespeichert werden. 24236
                ldy    #ALTNR        ;Dateien innerhalb 24237
::1             lda    a4H          ;CopyMem nummerieren. 24238
                cmp    MHB04AC      ;HB Ende CopyBuf. 24239
                bcs    :2 24240
                24241
                txa                ;AltNr setzen = 24242
                sta    (a4),y        ;Position 0-254. 24243
                beq    z2F4A        ; => Ende erreicht. 24244
                24245
::2             lda    a4L          ;Zeiger auf nächsten 24246
                sec                ;Dateieintrag. 24247
                sbc    #< 30 24248
                sta    a4L 24249
                lda    a4H 24250
                sbc    #> 30 24251
                sta    a4H 24252

```

```

24253
24254             dex                      ;Alle nummeriert?
24255             jmp      :1              ; => Nein, weiter...
24256
24257 ; a1 = Zeiger auf ersten Eintrag.
24258 ; a2 = Zeiger auf zweiten Eintrag.
24259 :z2F4A      lda      #< BASE_MOD_A +2
24260             sta      a1L
24261             lda      #> BASE_MOD_A +2
24262             sta      a1H
24263
24264             lda      #< BASE_MOD_A +2 +30
24265             sta      a3L
24266             lda      #> BASE_MOD_A +2 +30
24267             sta      a3H
24268
24269 ::loop      ldy      #3              ;Zeiger auf Name.
24270             lda      (a3),y          ;Name = 0 ?
24271             beq      :test           ; => Ja, Ende...
24272
24273             ldx      C04B2           ;Icon-/Text-Modus.
24274             dex
24275             beq      :test           ; => Text-Modus.
24276             dex
24277             beq      :name           ; => Nach Name.
24278
24279             jsr      z3083           ;Sortieren.
24280             bcc      :next           ; 1 < 2
24281             bcs      :swap           ; 1 > 2
24282
24283 ::name      lda      (a3),y
24284             jsr      z2FAB           ;Großbuchs. wechseln.
24285             sta      a8L            ;Zeichen Eintrag #1.
24286
24287             lda      (a1),y
24288             jsr      z2FAB           ;Großbuchs. wechseln.
24289             cmp      #$a0           ;Ende Name?
24290             bne      :1              ; => Nein, weiter...
24291             lda      #" "
24292
24293 ::1         cmp      a8L
24294             bcc      :next           ; 1 < 2
24295             beq      :2              ; 1 = 2
24296
24297 ::swap      jsr      z310F           ;Eintrag tauschen.
24298
24299             inc      v3082           ;Geänd.Einträge +1.
24300
24301 ::next      lda      #30             ;Zeiger auf den
24302             jsr      z307A           ;nächsten Eintrag.
24303
24304             lda      #30
24305             jsr      z3079
24306
24307             jmp      :loop

```



			24308
::2	iny		24309
	cpy	#19 ;Name verglichen?	24310
	bne	:name ; => Nein, weiter...	24311
	beq	:next ; => Nächste Datei.	24312
			24313
::test	lda	v3082 ;Geänd.Einträge >0 ?	24314
	beq	z2FCA ; => Nein, weiter...	24315
			24316
	lda	#\$00	24317
	sta	v3082 ;Geänd.Einträge =0.	24318
			24319
	jmp	z2F4A ;Neu sortieren.	24320
			24321
:v3082	b	\$00 ;Anz. Einträge geändert.	24322
			24323
; Einträge kopieren.			24324
:z2FCA	lda	r1L	24325
	clc		24326
	adc	#< SIZE_DIRNAM	24327
	sta	r0L	24328
	lda	r1H	24329
	adc	#> SIZE_DIRNAM	24330
	sta	r0H	24331
			24332
	lda	#< BASE_MOD_A +2	24333
	sta	a1L	24334
	lda	#> BASE_MOD_A +2	24335
	sta	a1H	24336
			24337
	lda	#\$00 ;Highbyte Offset	24338
	sta	a5H ;löschen.	24339
			24340
	ldx	C04B2 ;Icon-/Text-Modus.	24341
	dex	;Text-Modus?	24342
	bne	:1 ; => Nein, weiter...	24343
			24344
	sta	a5L ;Offset auf 0 setzen.	24345
			24346
::1	lda	#< 30	24347
	sta	r2L	24348
	lda	#> 30	24349
	sta	r2H	24350
			24351
	ldx	#r2 ;Offset-Files	24352
	ldy	#a5 ;überspringen.	24353
	jsr	DMult	24354
			24355
	lda	r2L	24356
	clc		24357
	adc	a1L	24358
	sta	a1L	24359
	lda	r2H	24360
	adc	a1H	24361
	sta	a1H	24362

```

24363
24364         lda    #$00                ;Zähler
24365         sta    a5L                ;initialisieren.
24366
24367 ::loop      ldy    #0
24368         lda    (a1),y              ;Datei vorhanden?
24369         beq    :done              ; => Nein, weiter...
24370
24371         ldy    #3 -1                ;CBM-Dateityp und
24372 ::11        lda    (a1),y              ;Tr/Se Datenblock
24373         sta    (r0),y              ;kopieren.
24374         dey
24375         bpl    :11
24376
24377         lda    #3
24378         jsr    z307A                ;a1 +3.
24379
24380         lda    #3
24381         jsr    z307B                ;r0 +3.
24382
24383         ldy    #16 +2 -1            ;Name und Tr/Se des
24384 ::12        lda    (a1),y              ;Infoblock kopieren.
24385         sta    (r1),y
24386         dey
24387         bpl    :12
24388
24389         lda    #16 +2
24390         jsr    z307A                ;a1 +16 +2.
24391
24392         lda    #16 +2
24393         jsr    z307C                ;r1 +16 +2.
24394
24395         ldy    #9 -1                ;GEOS-Typ, Struktur,
24396 ::13        lda    (a1),y              ;Datum, Zeit, Größe
24397         sta    (r0),y              ;kopieren.
24398         dey
24399         bpl    :13
24400
24401 ; Zeiger auf nächsten Eintrag.
24402         lda    #9
24403         jsr    z307A                ;a1 +9.
24404
24405         lda    #61
24406         jsr    z307B                ;r0 +61.
24407
24408         inc    a5L                ;Anz. Einträge +1.
24409
24410         ldx    #MAXWINFILES
24411         cpx    a5L                ;Max. 16 Dateien?
24412         bne    :loop              ; => Nein, weiter...
24413
24414 ::done      lda    a5L
24415         ldx    a6L
24416         sta    V050F,x            ;Anz.Dat. im Fenster.
24417         clc

```

	rts		24418
			24419
	; Großbuchstaben in Kleinbuchstaben wechseln.		24420
	; TD13: :grglkl10		24421
:z2FAB	cmp	#"A"	24422
	bcc	:1	24423
	cmp	#"Z" +1	24424
	bcs	:1	24425
	adc	#\$20	24426
::1	rts		24427
			24428
	; Zeropage-Register korrigieren.		24429
	; Übergabe: A = Korrekturwert.		24430
:z3078	ldx	#a6L	24431
	b	\$2c	24432
:z3079	ldx	#a3L	24433
	b	\$2c	24434
:z307A	ldx	#a1L	24435
	b	\$2c	24436
:z307B	ldx	#r0L	24437
	b	\$2c	24438
:z307C	ldx	#r1L	24439
	b	\$2c	24440
:z307D	ldx	#r6L	24441
	clc		24442
	adc	zpage +0,x	24443
	sta	zpage +0,x	24444
	bcc	:1	24445
	inc	zpage +1,x	24446
::1	rts		24447
			24448
	; Eintrag sortieren.		24449
:z3083	dex		24450
	beq	:date ; => Datum.	24451
	dex		24452
	beq	:size ; => Größe	24453
			24454
	; GEOS-Dateityp.		24455
::type	ldy	#22	24456
	lda	(a1),y	24457
	cmp	(a3),y	24458
	bne	:cmp1	24459
			24460
	clc	;Nicht tauschen.	24461
::cmp1	rts	;Evtl. tauschen.	24462
			24463
	; Dateigröße.		24464
::size	ldy	#28	24465
	sec		24466
	lda	(a1),y	24467
	sbc	(a3),y	24468
	iny		24469
	lda	(a1),y	24470
	sbc	(a3),y	24471
	bmi	:11	24472

```

24473          clc                      ;Nicht tauschen.
24474          rts
24475 ::11      sec                      ;Tauschen.
24476          rts
24477
24478 ; Datum.
24479 ::date      ldy    #26
24480          lda    (a3),y
24481          and    #%01111111
24482          sta    (a3),y
24483          lda    (a1),y
24484          and    #%01111111
24485          sta    (a1),y
24486
24487          ldy    #23
24488
24489 ::21      lda    (a1),y
24490          cpy    #23
24491          bne    :22
24492          sec
24493          sbc    #20
24494
24495 ::22      sta    a8L
24496
24497          lda    (a3),y
24498          cpy    #23
24499          bne    :23
24500          sec
24501          sbc    #20
24502
24503 ::23      cmp    a8L
24504          bne    :cmp2
24505
24506          iny
24507          cpy    #28                      ;Jahr/Monat/Tag
24508          bne    :21                      ;vergleichen.
24509
24510          clc                      ;Nicht tauschen.
24511 ::cmp2      rts                      ;Evtl. tauschen.
24512
24513 ; 8: Dateipos. im Verzeichnis einlesen.
24514 ; Übergabe: A = Dateinummer (0-15)
24515 ;           X = Fenster-Nr.
24516 ; Rückgabe: A = Dateinummer (0-255)
24517 ; TD13: __GetRealPos
24518 :M30DA      pha
24519
24520          jsr    S599A                      ;Zeiger auf DIRBUF.
24521
24522          lda    #< SIZE_DIRNAM
24523          clc
24524          adc    r1L
24525          sta    r1L
24526          lda    #> SIZE_DIRNAM
24527          adc    r1H

```

	sta	r1H	24528
			24529
	pla		24530
	sta	r0L	24531
			24532
	lda	#< 64	24533
	sta	r2L	24534
	lda	#> 64	24535
	sta	r2H	24536
			24537
	ldx	#r2	24538
	ldy	#r0	24539
	jsr	BMult	24540
			24541
	lda	r2L	24542
	clc		24543
	adc	r1L	24544
	sta	r1L	24545
	lda	r2H	24546
	adc	r1H	24547
	sta	r1H	24548
			24549
	ldy	#ALNR	24550
	lda	(r1),y	24551
	rts		24552
			24553
	; Zwei Dateieinträge tauschen.		24554
	; Übergabe: a1 = Zeiger auf 2x30-Byte-Eintrag.		24555
	; TD13: SwapEntry		24556
:z310F	lda	a1L	24557
	clc		24558
	adc	#< 30	24559
	sta	a2L	24560
	lda	a1H	24561
	adc	#> 30	24562
	sta	a2H	24563
			24564
	ldy	#30 -1	24565
::1	lda	(a1),y	24566
	pha		24567
	lda	(a2),y	24568
	sta	(a1),y	24569
	pla		24570
	sta	(a2),y	24571
	dey		24572
	bpl	:1	24573
			24574
	rts		24575
			24576
	; DnD-Sprite für Textmodus und Dateien tauschen.		24577
	; TD13: TextSprite		24578
:GR5181	b	%00000000,%00000000,%00000000	24579
	b	%00000000,%00000000,%00000000	24580
	b	%00000000,%00000000,%00000000	24581
	b	%00000000,%00000000,%00000000	24582

24583	b %00000000,%00000000,%00000000
24584	b %00000000,%00000000,%00000000
24585	b %00000000,%00000000,%00000000
24586	b %11111110,%01100110,%01111111
24587	b %10000000,%00000000,%00000001
24588	b %10000000,%00000000,%00000001
24589	b %10000000,%00000000,%00000001
24590	b %10000000,%00000000,%00000001
24591	b %10000000,%00000000,%00000001
24592	b %10000000,%00000000,%00000001
24593	b %11111110,%01100110,%01111111
24594	b %00000000,%00000000,%00000000
24595	b %00000000,%00000000,%00000000
24596	b %00000000,%00000000,%00000000
24597	b %00000000,%00000000,%00000000
24598	b %00000000,%00000000,%00000000
24599	b %00000000,%00000000,%00000000
24600	
24601	; Endadresse testen:
24602	g BASE_MOD_B + SIZE_FSERVICE
24603	

```

24604
; 24605
; SourceCode: src.mod#07.s 24606
; 24607
24608
; 24609
; Modul-Info: 24610
; 0 - Laufwerk-Icons definieren 24611
; 1 - TopDesk initialisieren 24612
; 2 - Drucker-/Eingabetreiber laden 24613
; 3 - Laufwerke tauschen 24614
; 24615
if .p 24616
    t "TopSym" 24617
    t "TopSym.MP3" 24618
    t "TopSym.ROM" 24619
    t "DTopDeskI18N.ext" 24620
    t "src.DTopDesk.ext" 24621
24622
; Alle Laufwerk-Icons anzeigen: 24623
:EN_ICONVIEW = FALSE 24624
endif 24625
24626
    o BASE_MOD_A 24627
    n "obj.#07" 24628
24629
::ModEntry_0    jmp    N715F    ;Lfwk.-Icons testen. 24630
::ModEntry_1    jmp    N7074    ;TopDesk initialisieren. 24631
::ModEntry_2    jmp    N6F30    ;Treiber wechseln. 24632
;:ModEntry_3    jmp    N6EB8    ;Laufwerke tauschen. 24633
24634
; 3: Laufwerke tauschen. 24635
:N6EB8          lda     r15H      24636
                pha      24637
                lda     r15L      24638
                pha      24639
24640
                jsr     z7561      ;Laufwerke tauschen. 24641
                bcs     :exit      ; => Abbruch... 24642
24643
                jsr     N715F      ;Lfwk.-Icons testen. 24644
24645
                jsr     z6EDE      ;TD-Info tauschen. 24646
24647
                ldx     C042A      ;Oberstes Fenster. 24648
                lda     C0448,x    ;Fenster geöffnet? 24649
                beq     :exit      ; => Nein, weiter... 24650
24651
                lda     C04F8,x    ;Lfwk. für Fenster. 24652
                jsr     S06AB      ;Laufwerk öffnen. 24653
24654
:::exit         pla      24655
                sta     r15L      24656
                pla      24657
                sta     r15H      24658

```

```

24659      rts
24660
24661 ; TopDesk-Info für Quell-/Ziel-Laufwerk tauschen.
24662 :z6EDE      ldx    #$03
24663 ::loop      lda    C0448,x          ;Fenster geöffnet?
24664            beq     :next          ; => Nein, weiter...
24665
24666            lda    C04F8,x          ;Lfwk. für Fenster.
24667            cmp     v6F2E          ;Quell-Laufwerk?
24668            bne     :1
24669
24670            lda    v6F2F          ;Ziel-Laufwerk.
24671            bne     :2
24672
24673 ::1          cmp     v6F2F          ;Ziel-Laufwerk?
24674            bne     :next          ; => Nein, weiter...
24675
24676            lda    v6F2E          ;Quell-Laufwerk.
24677
24678 ::2          sta    C04F8,x          ;Lfwk. für Fenster.
24679            pha
24680            txa
24681            pha
24682            jsr     S5979          ;Zeiger Fenstertitel.
24683            pla
24684            tax
24685            pla
24686            clc
24687            adc     #"A" -8          ;Laufwerk in
24688            ldy     #$00          ;Titelzeile anpassen.
24689            sta     (r1),y
24690
24691 ::next       dex
24692            bpl     :loop
24693
24694            lda    v6F2F          ;Ziel-Laufwerk =
24695            cmp     C0424          ;TopDesk/Laufwerk?
24696            bne     :3          ; => Nein, weiter...
24697
24698            lda    v6F2E          ;Quell-Laufwerk.
24699            bne     :4
24700
24701 ::3          lda    v6F2E          ;Quell-Laufwerk =
24702            cmp     C0424          ;TopDesk/Laufwerk?
24703            bne     :5          ; => Nein, weiter...
24704
24705            lda    v6F2F          ;Ziel-Laufwerk.
24706 ::4          sta    C0424          ;TopDesk/Laufwerk.
24707 ::5          rts
24708
24709 :v6F2E       b $00 ;Quell-Laufwerk.
24710 :v6F2F       b $00 ;Ziel-Laufwerk.
24711
24712 ; Laufwerke tauschen.
24713 :z7561       lda    V3A1A          ;Quell-Laufwerk.

```



	sta	v6F2E		24714
	ldx	V3A1B	;Ziel-Laufwerk.	24715
	stx	v6F2F		24716
	cmp	V3A1B	;Quelle < Ziel?	24717
	bcc	:1	; => Nein, weiter...	24718
				24719
	tax			24720
	lda	V3A1B	;Quelle und Ziel	24721
	sta	V3A1A	;tauschen.	24722
	stx	V3A1B		24723
				24724
::1	ldx	V3A1A	;Laufwerk#1.	24725
	lda	V04B8 -8,x	;Geräteinfo einlesen.	24726
	bmi	:2	; => 64Net	24727
				24728
	ldx	V3A1B	;Laufwerk#2.	24729
	lda	V04B8 -8,x	;Geräteinfo einlesen.	24730
	bpl	:3	; => Kein 64Net	24731
				24732
::2	sec		;Nicht möglich.	24733
	rts			24734
				24735
::3	lda	V3A1A	;Laufwerk#1.	24736
	jsr	SetDevice		24737
	jsr	PurgeTurbo	;TurboDOS abschalten.	24738
				24739
	jsr	InitForIO		24740
				24741
	ldx	V3A1A	;Laufwerk#1.	24742
	jsr	z7600	;Adresse ändern?	24743
	bcc	:5	; => Nein, weiter...	24744
::4	lda	#30		24745
	ldx	V3A1A	;Laufwerk#1 auf	24746
	jsr	z762D	;Adr. #30 tauschen.	24747
				24748
::5	ldx	V3A1B	;Laufwerk#2.	24749
	jsr	z7600	;Adresse ändern?	24750
	bcc	:7	; => Nein, weiter...	24751
::6	lda	V3A1A	;Laufwerk#1.	24752
	ldx	V3A1B	;Laufwerk#2 auf	24753
	jsr	z762D	;Lfwk. #1 tauschen.	24754
				24755
::7	ldx	V3A1A	;Laufwerk#1.	24756
	jsr	z7600	;Adresse ändern?	24757
	bcc	:9	; => Nein, weiter...	24758
::8	ldx	#30		24759
	lda	V3A1B	;Adr. #30 auf	24760
	jsr	z762D	;Laufwerk#2 tauschen.	24761
				24762
::9	jsr	DoneWithIO		24763
				24764
	ldx	V3A1B	;Laufwerk#2.	24765
	stx	curDrive	;Laufwerk setzen.	24766
	jsr	z760C	;Treiber tauschen.	24767
	ldx	V3A1A	;Laufwerk#1.	24768

24769	jsr	z760C	;Treiber tauschen.
24770			
24771	ldx	V3A1A	;Laufwerk#1.
24772	ldy	V3A1B	;Laufwerk#2.
24773			
24774	lda	ramBase -8,x	;Daten tauschen.
24775	pha		
24776	lda	ramBase -8,y	
24777	sta	ramBase -8,x	
24778	pla		
24779	sta	ramBase -8,y	
24780			
24781	lda	driveType -8,x	
24782	pha		
24783	lda	driveType -8,y	
24784	sta	driveType -8,x	
24785	pla		
24786	sta	driveType -8,y	
24787			
24788	lda	driveData -8,x	
24789	pha		
24790	lda	driveData -8,y	
24791	sta	driveData -8,x	
24792	pla		
24793	sta	driveData -8,y	
24794			
24795	lda	turboFlags -8,x	
24796	pha		
24797	lda	turboFlags -8,y	
24798	sta	turboFlags -8,x	
24799	pla		
24800	sta	turboFlags -8,y	
24801			
24802	lda	RealDrvType -8,x	
24803	pha		
24804	lda	RealDrvType -8,y	
24805	sta	RealDrvType -8,x	
24806	pla		
24807	sta	RealDrvType -8,y	
24808			
24809	lda	RealDrvMode -8,x	
24810	pha		
24811	lda	RealDrvMode -8,y	
24812	sta	RealDrvMode -8,x	
24813	pla		
24814	sta	RealDrvMode -8,y	
24815			
24816	lda	drivePartData -8,x	
24817	pha		
24818	lda	drivePartData -8,y	
24819	sta	drivePartData -8,x	
24820	pla		
24821	sta	drivePartData -8,y	
24822			
24823	lda	doubleSideFlg -8,x	

	pha		24824
	lda	doubleSideFlg -8,y	24825
	sta	doubleSideFlg -8,x	24826
	pla		24827
	sta	doubleSideFlg -8,y	24828
			24829
	clc	; Laufwerke getauscht.	24830
	rts		24831
			24832
; Testen ob Adr. getauscht werden muss.			24833
:z7600	lda	RealDrvType -8,x	24834
	bmi	:0 ; => RAMDisk...	24835
	lda	V04B8 -8,x	24836
	and	##00000011 ; RAMLink/RAMDrive?	24837
	beq	:1 ; => Nein, weiter...	24838
			24839
	cpx	C0509 ; RAMLink-Adresse?	24840
	beq	:1 ; => Ja, ändern...	24841
			24842
::0	jsr	z7437 ; Laufwerk testen.	24843
	txa	; Laufwerk vorhanden?	24844
	bne	:2 ; => Nein, Ende...	24845
			24846
::1	sec	; Adresse ändern.	24847
	rts		24848
			24849
::2	clc	; Nicht ändern.	24850
	rts		24851
			24852
; Laufwerk testen.			24853
:z7437	lda	curDevice	24854
	pha		24855
	stx	curDevice	24856
			24857
	lda	#\$00	24858
	sta	STATUS	24859
			24860
	jsr	UNLSN	24861
			24862
	lda	curDevice	24863
	jsr	LISTEN	24864
	lda	#\$ff	24865
	jsr	SECOND	24866
	jsr	UNLSN	24867
			24868
	ldx	STATUS	24869
	beq	:done	24870
			24871
	ldx	#\$ff	24872
			24873
::done	pla		24874
	sta	curDevice	24875
	rts		24876
			24877
; Treiber tauschen.			24878

```

24879 ; Übergabe: X = Laufwerk 8-11.
24880 :z760C      lda    #< DISK_BASE          ;Adresse: C64.
24881          sta    r0L
24882          lda    #> DISK_BASE
24883          sta    r0H
24884
24885          lda    DskDrvBaseL -8,x
24886          sta    r1L
24887          lda    DskDrvBaseH -8,x
24888          sta    r1H
24889
24890          lda    #< DISK_SIZE          ;Größe.
24891          sta    r2L
24892          lda    #> DISK_SIZE
24893          sta    r2H
24894
24895          lda    #$00                  ;Bank #0.
24896          sta    r3L
24897
24898          jmp     SwapRAM
24899
24900 ; Geräteadresse ändern.
24901 ; Übergabe: X = Aktuelle Adresse.
24902 ;           A = Neue Adresse.
24903 :z762D      stx    r10L
24904
24905          clc
24906          adc    #%00100000
24907          sta    v7664
24908          adc    #%00100000
24909          sta    v7665
24910
24911          lda    #$00
24912          sta    STATUS
24913
24914          jsr    UNLSN
24915
24916          lda    r10L
24917          jsr    LISTEN
24918          lda    #%11110000 ! 15
24919          jsr    SECOND
24920
24921          ldy    #0
24922 ::1        lda    v765E,y
24923          jsr    CIOUT
24924          iny
24925          cpy    #8
24926          bne    :1
24927
24928          jmp     UNLSN
24929
24930 ; Befehl zum ändern der Geräteadresse.
24931 :v765E      b "M-W"
24932          w $0077
24933          b $02

```



```

24989 ::skip      lda    V25D0          ;Dateityp für öffnen.
24990             cmp    #PRINTER      ;Drucker wechseln?
24991             beq    z6FB2          ; => Ja, weiter...
24992
24993             lda    r15L            ;Zeiger Dateiname.
24994             sta    r6L
24995             lda    r15H
24996             sta    r6H
24997
24998             lda    #$00
24999             sta    r0L            ;Nur Datei laden.
25000             sta    r10L          ;DARecoverFlag = 0.
25001
25002             jsr    GetFile          ;Eingabetr. laden.
25003             txa                    ;Fehler?
25004             bne    :exit          ; => Ja, Abbruch...
25005
25006             jsr    z6FB3            ;Name übernehmen.
25007
25008             jsr    InitMouse        ;Maus initialisieren.
25009
25010 ::exit      rts
25011
25012 ; Neuen Drucker-/Eingabetreiber setzen/anzeigen.
25013 :z6FB2      lda    #< PrntFilename
25014             ldx    #> PrntFilename
25015             bne    z6FB4
25016
25017 :z6FB3      lda    #< inputDevName
25018             ldx    #> inputDevName
25019
25020 :z6FB4      sta    r1L            ;Zeiger auf Name für
25021             stx    r1H            ;Drucker/Eingabe.
25022
25023             lda    r15L            ;Zeiger Dateiname.
25024             sta    r0L
25025             lda    r15H
25026             sta    r0H
25027
25028             lda    #< 16           ;Max. Länge.
25029             sta    r2L
25030             lda    #> 16
25031             sta    r2H
25032
25033             jsr    MoveData        ;Name kopieren.
25034
25035             jsr    S4840           ;Lfwk/Prnt/Inpt-Info.
25036
25037 ; Inhalt REU aktualisieren.
25038 :z702A      lda    sysRAMFlg
25039             and    #%00100000     ;REU: $8400-$88FF?
25040             beq    :exit          ; => Ja, weiter...
25041
25042             ldx    #(r3L - r0L)
25043 ::1         lda    :geos,x         ;GEOS-Register in

```

```

        sta    r0,x                ;REU aktualisieren.      25044
        dex                                25045
        bpl    :1                    25046
                                25047
        jsr    StashRAM              25048
                                25049
        ldx    #(r3L - r0L)          25050
::2      lda    :mouse,x             ;Eingabetreiber in      25051
        sta    r0,x                 ;REU aktualisieren.      25052
        dex                                25053
        bpl    :2                    25054
                                25055
        jsr    StashRAM              25056
                                25057
::exit   rts                        25058
                                25059
; GEOS-Register in REU/Bank#0.      25060
::geos   w $8400                    25061
        w $7900                    25062
        w $0500                    25063
        b $00                      25064
                                25065
; Eingabetreiber in REU/Bank#0.      25066
::mouse  w MOUSE_BASE               25067
        w $fac0                    25068
        w $0180                    25069
        b $00                      25070
                                25071
; Dialogbox: Neuen Treiber installieren? 25072
:d6FCB   b %10000001                25073
                                25074
        b DBVARSTR,$08,$10          25075
        b r14L                     25076
                                25077
        b DBTXTSTR,$08,$1c          25078
        w T29D3                    25079
        b DBVARSTR,$12,$1c          25080
        b r15L                     25081
                                25082
        b DBTXTSTR,$08,$30          25083
        w :1                        25084
                                25085
        b YES      , $01,$4c        25086
        b NO       , $11,$4c        25087
        b NULL                                25088
                                25089
if LANG = LANG_DE                    25090
::1      b PLAINTEXT                25091
        b "Neues Gerät installieren ?",0 25092
                                25093
:t6FED   b BOLDON,"Neuer Druckertreiber:",0 25094
:t6FFC   b BOLDON,"Neuer Eingabetreiber:",0 25095
endif                                25096
if LANG = LANG_EN                    25097
::1      b PLAINTEXT                25098

```

```

25099      b "Install new device ?",0
25100
25101 :t6FED      b BOLDON,"New printer driver:",0
25102 :t6FFC      b BOLDON,"New input driver:",0
25103 endif
25104
25105 ; 1: TopDesk initialisieren.
25106 :N7074      lda    #$00
25107           sta    iconSelFlag
25108
25109 if EN_ICONVIEW = TRUE
25110           jsr    _DrawIcons           ;Debug: icons zeigen.
25111 endif
25112
25113 ; Hinweis:
25114 ; Ist RamTopDesk aktiv, dann muss hier
25115 ; zumindest das Laufwerk zurückgesetzt
25116 ; werden, damit PadColorPref vorrangig vom
25117 ; vom Startlaufwerk geladen wird.
25118 ; Damit sind Iconfarben beim ersten
25119 ; Start und bei der Rückkehr über die
25120 ; Routine ":EnterDeskTop" von einem
25121 ; CBM-Laufwerk gleich.
25122           lda    C0424                ;TopDesk/Laufwerk
25123           jsr    SetDevice            ;aktivieren.
25124
25125 ; Hinweis:
25126 ; TD64 lädt zu Beginn die PadColorPref
25127 ; auf dem Startlaufwerk und ersetzt die
25128 ; Standardfarben für die Datei-Icons.
25129 ;
25130 ; Variante#1:
25131 ; Nur laden wenn "Verschiedene Farben"
25132 ; deaktiviert ist. Die Farben werden
25133 ; beim öffnen der Fenster automatisch
25134 ; angepasst, wenn eine PadColorPref
25135 ; gefunden wird.
25136 ;--- Var1:
25137 if ENABLE_DIFFICOL = FALSE
25138           lda    MT3F58                ;Versch. Farben?
25139           cmp    #""
25140           beq    :diffcol              ; => Ja, weiter...
25141 endif
25142 ;if ENABLE_DIFFICOL = TRUE
25143 ;           lda    MT3F59                ;Versch. Iconfarben?
25144 ;           cmp    #""
25145 ;           beq    :diffcol              ; => Ja, weiter...
25146 ;endif
25147 ;---
25148
25149 ; Einheitliche Fensterfarben:
25150 ;--- Var1+Var2:
25151 ::samecol      ldx    #0                ;Immr Fenster #0.
25152 ;           ldx    C042A                ;Oberstes Fenster.
25153           jsr    S3280                ;PadColorPref laden.

```



```

;---
; Variante#2:
; PadColorPref als Vorgabe auf alle
; Fenster anwenden, aber nur wenn die
; Option "Verschiedene Fensterfarben"
; deaktiviert ist. Es besteht das
; Risiko das einige Iconfarben gleich
; sind wie die Fensterfarben.
;--- Var2:
if ENABLE_DIFFICOL = TRUE
    lda    MT3F58                ;Versch. Fens.Farben?
    cmp    #""
    beq    :diffcol              ; => Ja, weiter...
    lda    V04AF                ;Neue PadColorPref?
    beq    :diffcol              ; => Nein, weiter...
    ldy    #15                  ;Farben auf alle
::copy    lda    V620B,y          ;Fenster anwenden.
    sta    V621C,y
    sta    V622D,y
    sta    V623E,y
    dey
    bpl    :copy
endif
;---
; Verschiedene Fensterfarben:
;--- Var1+Var2:
::diffcol    jsr    S327F          ;Prefs. laden.
;---
::cont        ldx    #%00000000
    lda    C0438                ;TopDesk in Farbe?
    beq    :1                    ; => Nein, weiter...
    ldx    #%10000000          ;Dialogbox (System)
::1          stx    Flag_SetColor ;in Farbe anzeigen.
; Hinweis:
; Systemfarben nicht verändern.
;
;    lda    COL043C              ;Farbe Dialogbox.
;    sta    C_DBoxBack
;
;    lda    COL0439              ;GEOS-Hintergrund.
;    sta    C_GEOS_BACK
;    sta    screencolors
;
;    ldy    #< ResetScreen
;    ldx    #> ResetScreen
;
;    lda    C0438                ;TopDesk in Farbe?
;    beq    :2                    ; => Nein, weiter...
;
;    lda    sysRAMFlg

```

```

25209          and    #%00001000          ;Hintergrundbild?
25210          beq     :2                  ; => Nein, weiter...
25211
25212          ldy     #< GetBackScreen
25213          ldx     #> GetBackScreen
25214
25215 ::2          tya
25216          jsr     CallRoutine          ;Hintergrund löschen.
25217
25218          jsr     S151A                ;Panel zeichnen.
25219
25220          lda     #< C04BC              ;Zeiger Fensterdaten
25221          sta     WIN0446 +0           ;auf Fenstertabelle.
25222          lda     #> C04BC
25223          sta     WIN0446 +1
25224
25225 ; Fenster aktivieren.
25226 ; TD13: DoWindows
25227          lda     #$00                ;Kein Fenster aktiv.
25228          sta     C0448 +0
25229          sta     C0448 +1
25230          sta     C0448 +2
25231          sta     C0448 +3
25232
25233 ; Neue GEOS/appMain-Routine setzen.
25234          lda     appMain +0
25235          sta     V044C +0
25236          lda     appMain +1
25237          sta     V044C +1            ;GEOS/appMain.
25238
25239          lda     #< S1160              ;":MyAppMain"
25240          sta     appMain +0
25241          lda     #> S1160
25242          sta     appMain +1
25243
25244 ; Neue GEOS/mouseVector-Routine setzen (doppelt, s.u.).
25245 if FALSE
25246          lda     mouseVector +0
25247          sta     V045B +0
25248          lda     mouseVector +1
25249          sta     V045B +1            ;GEOS/mouseVector.
25250
25251          lda     #< S18EC              ;":NewMouseService"
25252          sta     mouseVector +0
25253          lda     #> S18EC
25254          sta     mouseVector +1
25255 endif
25256
25257 ; Neue GEOS/RecoverVector-Routine setzen.
25258          lda     RecoverVector +0
25259          sta     V045E +0
25260          lda     RecoverVector +1
25261          sta     V045E +1            ;GEOS/RecoverVector.
25262
25263          lda     #< S2185              ;":MyRecoverService".

```

sta	RecoverVector +0	25264
lda	#> S2185	25265
sta	RecoverVector +1	25266
		25267
jsr	S437D ;TopDesk-Oberfläche.	25268
		25269
lda	#< R43DA ;Tastaturabfrage.	25270
sta	keyVector +0	25271
lda	#> R43DA	25272
sta	keyVector +1	25273
		25274
jsr	N715F ;Lfwk.-Icons testen.	25275
		25276
lda	#< DI3616 ;Zeiger auf Tabelle	25277
sta	r0L ;für DoIcons.	25278
lda	#> DI3616	25279
sta	r0H	25280
		25281
	; Icon-Menü aktivieren.	25282
	; Übergabe: r0 = Zeiger/Icon-Tabelle.	25283
	; TD13: NewDoIcons	25284
if FALSE		25285
lda	V045B +0 ;GEOS/mouseVector.	25286
sta	mouseVector +0	25287
lda	V045B +1	25288
sta	mouseVector +1	25289
endif		25290
		25291
lda	r0L ;Zeiger auf Icon-	25292
sta	V044E +0 ;Tabelle speichern.	25293
lda	r0H	25294
sta	V044E +1	25295
		25296
jsr	DoIcons	25297
		25298
lda	WIN0446 +0 ;Zeiger Fensterdaten.	25299
sta	a0L	25300
lda	WIN0446 +1	25301
sta	a0H	25302
		25303
	; Neue GEOS/mouseVector-Routine setzen.	25304
lda	mouseVector +0	25305
sta	V045B +0	25306
lda	mouseVector +1	25307
sta	V045B +1 ;GEOS/mouseVector.	25308
		25309
lda	#< S18EC ;":NewMouseService"	25310
sta	mouseVector +0	25311
lda	#> S18EC	25312
sta	mouseVector +1	25313
		25314
jsr	S157E ;Alle Fens. anzeigen.	25315
		25316
lda	#< R3B2A ;DnD-Sprite-Anzeige.	25317
sta	V044C +0	25318

```

25319          lda    #> R3B2A
25320          sta    V044C +1
25321
25322          lda    #< R3B53          ;Mausabfrage.
25323          sta    otherPressVec +0
25324          lda    #> R3B53
25325          sta    otherPressVec +1
25326
25327          jsr    S524C          ;Dat.-Ausw. aufheben.
25328
25329 ; Hinweis:
25330 ; Laufwerk erneut aktivieren, da beim
25331 ; einlesen der Laufwerk-Icons auch das
25332 ; aktuelle Laufwerk verändert wird.
25333          lda    C0424          ;TopDesk/Laufwerk.
25334          jsr    SetDevice
25335          jsr    OpenDisk
25336
25337 ; Hinweis:
25338 ; Auf Diskettenfehler testen, da bei
25339 ; einem Fehler auch kein Drucker- oder
25340 ; Eingabetreiber geladen werden kann.
25341          txa          ;Diskettenfehler?
25342          bne     :12          ; => Ja, weiter...
25343
25344          lda    #$00          ;GEOS-Bootup für
25345          sta    firstBoot      ;autom. Treiberwahl.
25346
25347          lda    inputDevName    ;Input definiert?
25348          bne     :11          ; => Ja, weiter...
25349
25350          lda    #INPUT_DEVICE
25351          jsr    z7111          ;Eingabetreiber.
25352
25353 ::11          lda    PrntFileName    ;Drucker definiert?
25354          bne     :12          ; => Ja, weiter...
25355
25356          lda    #PRINTER
25357          jsr    z7111          ;Druckertreiber.
25358
25359 ::12          lda    #$ff
25360          sta    firstBoot
25361
25362          jsr    S31F7          ;Anfang CopyMem.
25363          jsr    S4840          ;Lfwk/Prnt/Inpt-Info.
25364
25365          jmp    S3186          ;Initialisierung #2.
25366
25367 ; Drucker-/Eingabetreiber beim Systemstart laden.
25368 :z7111        sta    r7L          ;GEOS-Dateityp.
25369
25370          ldy    #1          ;Nur eine Datei.
25371          sty    r7H
25372          dey
25373          sty    V0470          ;Dateiname löschen.

```

		25374
	lda #< V0470	25375
	sta r6L	25376
	lda #> V0470	25377
	sta r6H	25378
	jsr FindFTypes ;Datei suchen.	25379
		25380
	lda V0470 ;Datei gefunden?	25381
	beq :exit ; => Nein, Ende...	25382
		25383
	lda #\$00	25384
	sta r1L ;PrintFlag löschen.	25385
		25386
	lda #< V0470	25387
	sta r6L	25388
	lda #> V0470	25389
	sta r6H	25390
	jsr S2931 ;Datei öffnen.	25391
		25392
::exit	rts	25393
		25394
if EN_ICONVIEW = TRUE		25395
:_DrawIcons	lda #0	25396
	jsr SetPattern	25397
	jsr i_Rectangle	25398
	b 0,199	25399
	w 0,319	25400
		25401
	lda #8 ;Speicher für	25402
	sta r10L ;Laufwerk-Icon.	25403
		25404
	lda #1	25405
	sta r11L ;X-Koordinate.	25406
	lda #8	25407
	sta r11H ;Y-Koordinate.	25408
		25409
	lda #0	25410
::loop	pha	25411
	jsr z7170 ;Icon erzeugen.	25412
		25413
	lda V72DD +0	25414
	sta r0L	25415
	lda V72DD +1	25416
	sta r0H	25417
		25418
	lda r11L	25419
	sta r1L	25420
	lda r11H	25421
	sta r1H	25422
		25423
	lda #DRVICON_X	25424
	sta r2L	25425
	lda #DRVICON_Y	25426
	sta r2H	25427
		25428

```

25429      jsr    BitmapUp
25430
25431      lda    r11L
25432      clc
25433      adc    #4
25434      cmp    #1 +8*4
25435      bcc    :1
25436
25437      lda    r11H
25438      clc
25439      adc    #24
25440      sta    r11H
25441
25442      lda    #1
25443  ::1      sta    r11L
25444
25445      pla
25446      clc
25447      adc    #1
25448      cmp    #MAX_ICONS
25449      bcc    :loop
25450
25451      ldx    #20
25452  ::wait   jsr    SCPU_Pause
25453      dex
25454      bne    :wait
25455
25456      rts
25457  endif
25458
25459  ; Hinweis:
25460  ; Ab hier nur Daten die beim laden des
25461  ; Druckertreibers überschrieben werden
25462  ; können. Ansonsten die Nr. des aktiven
25463  ; VLIR-Moduls in V0422 zurücksetzen.
25464      g PRINTBASE
25465
25466  ; 0: Laufwerk-Icons definieren.
25467  :N715F      lda    #$00
25468      sta    V04B8 +0      ;Geräteinfo A:
25469      sta    V04B8 +1      ;Geräteinfo B:
25470      sta    V04B8 +2      ;Geräteinfo C:
25471      sta    V04B8 +3      ;Geräteinfo D:
25472      sta    C0509      ;RAMLink-Adresse.
25473      sta    C0508      ;Keine RL installiert.
25474
25475      jsr    z72FD      ;RAMLink-Adr. suchen.
25476
25477      ldx    #8
25478  ::loop      stx    r10L
25479      lda    driveType -8,x      ;Laufwerk vorhanden?
25480      beq    :end      ; => Nein, Ende...
25481
25482      txa
25483      jsr    SetDevice      ;Lfwk. aktivieren.

```

	jsr	PurgeTurbo	;TurboDOS löschen.	25484
				25485
	jsr	z7160	;64Net?	25486
	bcs	:next	; => Ja, weiter...	25487
				25488
	jsr	z7162	;CBM/SD?	25489
	bcs	:next	; => Ja, weiter...	25490
				25491
	jsr	z7164	;CMD-FD?	25492
	bcs	:next	; => Ja, weiter...	25493
				25494
	jsr	z7166	;CMD-HD?	25495
	bcs	:next	; => Ja, weiter...	25496
				25497
	jsr	z7168	;RAMDisk?	25498
	bcs	:next	; => Ja, weiter...	25499
				25500
	lda	C0508	;RL/RD vorhanden?	25501
	beq	:next	; => Nein, weiter...	25502
				25503
	jsr	z716A	;RAMLink/RAMDrive?	25504
	bcs	:next	; => Ja, weiter...	25505
				25506
	lda	#OFF_UNKNOWN	;Lfwk. unbekannt.	25507
	jsr	z7170	;Icon erzeugen.	25508
				25509
::next	ldx	r10L		25510
	inx			25511
	cpx	#12		25512
	bcc	:loop		25513
				25514
			; TODO:	25515
			; Wozu ist das noch erforderlich?	25516
			;	25517
			; Unter GEOS/MP3 ist immer eine REU	25518
			; vorhanden und somit sind immer bis	25519
			; zu vier Laufwerke möglich.	25520
			;	25521
::end	jsr	S28DB	;Anz. Lfwk. setzen.	25522
				25523
	lda	numDrives	;Anzahl Icons in	25524
	clc		;DoIcons-Tabelle	25525
	adc	#MENU_ICONS	;festlegen.	25526
	sta	DI3616		25527
				25528
	rts			25529
				25530
			; RAMLink mit Adresse 8-30 suchen.	25531
:z72FD	jsr	PurgeTurbo	;TurboDOS löschen.	25532
				25533
	lda	curDrive	;Akt. Lfwk. sichern.	25534
	pha			25535
				25536
	lda	#8	;Suche ab Lfwk. #8.	25537
	sta	curDrive		25538

```

25539
25540 ::loop          jsr    z7440          ;ROM-Info einlesen.
25541                  txa                    ;Fehler?
25542                  bne    :next          ; => Nächstes Lfwk.
25543
25544                  lda    #< v7540      ;"CMD RL"
25545                  sta    r1L
25546                  lda    #> v7540
25547                  sta    r1H
25548
25549 ; Hinweis:
25550 ; Hier wird nur eine RAMLink erkannt.
25551 ; Ein RAMDrive ("CMD RD") kann hier
25552 ; nicht entdeckt werden!
25553 ;          lda    #6                  ;6Z. testen (CMD RL).
25554 ; Nur 5Z. vergleichen.
25555          lda    #5                  ;5Z. testen (CMD R)?
25556          ldx    #r0
25557          ldy    #r1
25558          jsr    CmpFString          ;Strings vergleichen.
25559          beq    :found              ; => RL/RD, weiter...
25560
25561 ::next          inc    curDrive
25562          lda    curDrive
25563          cmp    #30                  ;Alle Laufwerke?
25564          bne    :loop              ; => Nein, weiter...
25565
25566          lda    #$00                  ;Keine RL/RD.
25567          beq    :exit
25568
25569 ::found          lda    v752C +5
25570          cmp    #"L"                  ;RL oder RD?
25571          beq    :1                  ; => RL, weiter...
25572
25573          lda    #%01000000          ;RAMDrive.
25574          b $2c
25575 ::1              lda    #%10000000          ;RAMLink.
25576          sta    C0508
25577
25578          lda    curDrive
25579 ::exit          sta    C0509          ;RL/RD-Adresse.
25580
25581          pla
25582          sta    curDrive              ;Akt. Lfwk. setzen.
25583          rts
25584
25585 ; Auf 64Net testen.
25586 :z7160          ldy    #5 -1
25587 ::1              lda    :64net,y          ;64Net-Kennung.
25588          cmp    DiskDrvVersion,y
25589          bne    :err                ; => Kein 64Net...
25590          dey
25591          bpl    :1
25592
25593          jsr    z7163                ;Laufwerksmodus.

```



	bcs	:err	; => Ungültig...	25594
if ENABLE_DRVMODE = FALSE				25595
	cmp	#Drv1541		25596
	beq	:41		25597
	cmp	#Drv1571		25598
	beq	:71		25599
endif				25600
	cmp	#Drv1581		25601
if ENABLE_DRVMODE = TRUE				25602
	bcc	:41		25603
endif				25604
	bne	:err	; => Ungültig...	25605
				25606
::81	lda	#OFF_64NET81		25607
	b	\$2c		25608
if ENABLE_DRVMODE = FALSE				25609
::71	lda	#OFF_64NET71		25610
	b	\$2c		25611
endif				25612
::41	lda	#OFF_64NET41		25613
				25614
	jsr	z7170	;Icon erzeugen.	25615
				25616
	sec		;Laufwerk erkannt.	25617
	rts			25618
				25619
::err	clc		;Nicht erkannt.	25620
	rts			25621
				25622
::64net	b	"64NET"		25623
				25624
; Auf 1541-1581/SD2IEC testen.				25625
:z7162	ldx	curDrive		25626
	lda	RealDrvType -8,x		25627
	bmi	:err	; => RAMDisk...	25628
				25629
;	ldx	curDrive		25630
	lda	RealDrvMode -8,x		25631
	bmi	:err	; => CMD...	25632
	and	#SET_MODE_SD2IEC		25633
	bne	:sd	; => SD2IEC...	25634
				25635
	jsr	z7163	;Laufwerksmodus.	25636
	bcs	:err	; => Ungültig...	25637
if ENABLE_DRVMODE = FALSE				25638
	cmp	#Drv1541		25639
	beq	:41		25640
	cmp	#Drv1571		25641
	beq	:71		25642
endif				25643
	cmp	#Drv1581		25644
if ENABLE_DRVMODE = TRUE				25645
	bcc	:41		25646
endif				25647
	beq	:81		25648

```

25649      cmp    #DrvPCDOS
25650      bne     :err          ; => Ungültig...
25651
25652  ::pc      lda     #OFF_CBMP
25653          b $2c
25654  ::81      lda     #OFF_CBM81
25655          b $2c
25656  if ENABLE_DRVMODE = FALSE
25657  ::71      lda     #OFF_CBM71
25658          b $2c
25659  endif
25660  ::41      lda     #OFF_CBM41
25661
25662          jsr     z7170      ;Icon erzeugen.
25663
25664          sec
25665          rts              ;Laufwerk erkannt.
25666
25667  ::sd      jsr     z7163      ;Laufwerksmodus.
25668          bcs     :err          ; => Ungültig...
25669          cmp     #DrvNative +1
25670          bcs     :err          ; => Ungültig...
25671
25672  if ENABLE_DRVMODE = FALSE
25673          clc
25674          adc     #OFF_SD -1    ;Offset Tabelle.
25675  else
25676          lda     #OFF_SD      ;Offset für SD2IEC.
25677  endif
25678
25679          jsr     z7170      ;Icon erzeugen.
25680
25681          sec
25682          rts              ;Laufwerk erkannt.
25683
25684  ::err     clc
25685          rts              ;Nicht erkannt.
25686
25687  ; Auf CMD-FD testen.
25688  :z7164    ldx     curDrive
25689          lda     RealDrvType -8,x
25690          bmi     :err          ; => RAMDisk...
25691          and     #DrvCMD
25692          cmp     #DrvFD      ;CMD-FD?
25693          bne     :err          ; => Nein, Ende...
25694
25695          jsr     z7163      ;Laufwerksmodus.
25696          bcs     :err          ; => Ungültig...
25697
25698          jsr     z7440      ;ROM-Info einlesen.
25699          txa
25700          bne     :err          ; Fehler?
25701
25702          lda     v753C
25703          cmp     #"2"        ;FD2000?

```



```

25759 ::1          tya
25760          jsr    z7170          ;Icon erzeugen.
25761          sec
25762          rts          ;Laufwerk erkannt.
25763
25764 ::err          clc          ;Nicht erkannt.
25765          rts
25766
25767 ; Auf RAMDisk testen.
25768 :tmpram = SET_MODE_SRAM ! SET_MODE_CRAM ! SET_MODE_GRAM
25769 :z7168          ldx    curDrive
25770          lda    RealDrvType -8,x
25771          bpl    :err          ; => Keine RAMDisk...
25772
25773 ;          ldx    curDrive
25774          lda    RealDrvMode -8,x
25775          and    #tmpram
25776          bne    :extram          ; => ExtRAM-Lfwk...
25777
25778 ; RAM1541/71/81/NM:
25779          jsr    z7163          ;Laufwerksmodus.
25780          bcs    :err          ; => Ungültig...
25781          cmp    #DrvNative +1
25782          bcs    :err          ; => Ungültig...
25783
25784 if ENABLE_DRVMODE = FALSE
25785          clc
25786          adc    #OFF_RAM -1          ;Offset Tabelle.
25787 else
25788          lda    #OFF_RAM          ;Offset RAMDisk.
25789 endif
25790
25791          jsr    z7170          ;Icon erzeugen.
25792          sec
25793          rts
25794
25795 ; SuperCPU/C=REU/GeoRAM:
25796 ::extram          lsr
25797          lsr
25798          lsr
25799          bcs    :geo
25800          lsr
25801          bcs    :reu
25802 ::sup          lda    #OFF_EXTRAM +0          ;SuperCPU/Native.
25803          b $2c
25804 ::reu          lda    #OFF_EXTRAM +1          ;C=REU/Native.
25805          b $2c
25806 ::geo          lda    #OFF_EXTRAM +2          ;GeoRAM/Native.
25807
25808          jsr    z7170          ;Icon erzeugen.
25809          sec          ;Laufwerk erkannt.
25810          rts
25811
25812 ::err          clc          ;Nicht erkannt.
25813          rts

```

```

25814
; Auf CMD-RL/RD testen. 25815
:z716A    ldx    curDrive 25816
          lda    RealDrvType -8,x 25817
          bmi    :err      ; => RAMDisk... 25818
          and    #DrvCMD 25819
          cmp    #DrvRAMLink ;CMD-RL/RD? 25820
          bne    :err      ; => Nein, Ende... 25821
          25822
          bit    C0508      ;RL/RD bereits 25823
          bmi    :x         ;erkannt? 25824
          bvs    :x         25825
          lda    #%10000000 ; => Nein, RL setzen. 25826
          sta    C0508      ;RAMLink vorhanden. 25827
          25828
::x        jsr    z7163      ;Laufwerksmodus. 25829
          bcs    :err      ; => Ungültig... 25830
          cmp    #DrvNative +1 25831
          bcs    :err      ; => Ungültig... 25832
          25833
          bit    C0508      ;RL/RD? 25834
          bpl    :rd        ; => Nicht RL... 25835
          25836
::rl       ldy    #OFF_RL    ;RAMLink. 25837
          b $2c 25838
::rd       ldy    #OFF_RD    ;RAMDrive. 25839
          25840
if ENABLE_DRVMODE = FALSE 25841
          ldx    curDrive 25842
          lda    driveType -8,x 25843
          and    #ST_DMODES 25844
          cmp    #DrvNative 25845
          bne    :1 25846
          25847
          iny 25848
endif 25849
          25850
::1        tya 25851
          jsr    z7170      ;Icon erzeugen. 25852
          sec      ;Laufwerk erkannt. 25853
          rts 25854
          25855
::err      clc      ;Nicht erkannt. 25856
          rts 25857
          25858
; Laufwerksmodus einlesen. 25859
:z7163     ldx    curDrive 25860
          lda    driveType -8,x 25861
          and    #ST_DMODES 25862
          beq    :err 25863
          cmp    #$06      ;6+7=??? 25864
          bcs    :err      ;Modus unbekannt. 25865
          ; clc      ; => Lfwk. OK. 25866
          rts 25867
          25868

```

```

25869 ::err          sec          ; => Lfwk. unbekannt.
25870              rts
25871
25872 ; Laufwerk-Icon definieren.
25873 ; Übergabe: A = Eintrag 0-63.
25874 :z7170          tax
25875              lda    v7172,x
25876              sta    r3L          ;Geräte-Typ.
25877
25878              txa
25879              asl
25880              tax
25881
25882              lda    v7170 +0,x
25883              sta    r0L
25884              lda    v7170 +1,x
25885              sta    r0H          ;Standard-Icon.
25886
25887              txa
25888              asl
25889              tax
25890
25891              lda    v7171 +0,x
25892              sta    r1L
25893              lda    v7171 +1,x
25894              sta    r1H          ;Patch-Daten.
25895
25896              lda    v7171 +2,x
25897              sta    r2L          ;Offset Patch-Daten.
25898              lda    v7171 +3,x
25899              sta    r2H          ;Anzahl Patch-Bytes.
25900
25901              lda    r10L
25902              sec
25903              sbc    #8
25904              asl
25905              tax
25906              lda    V72DD +0,x    ;Zeiger auf Speicher
25907              sta    r4L          ;für Laufwerk-Icon.
25908              lda    V72DD +1,x
25909              sta    r4H
25910
25911              ldy    #1 + DRVICON_X*DRVICON_Y -1
25912 ::2              lda    (r0),y    ;Standard-Icon für
25913              sta    (r4),y    ;Lfwk.-Icon kopieren.
25914              dey
25915              bpl    :2
25916
25917              lda    r2H          ;Patch anwenden?
25918              beq    :done        ; => Nein, weiter...
25919
25920              ldy    #0          ;Patch auf Standard-
25921 ::3              tya          ;Icon anwenden.
25922              pha
25923              lda    (r1),y

```

	ldy	r2L	;Offset.	25924
	sta	(r4),y		25925
	inc	r2L		25926
	pla			25927
	tay			25928
	iny			25929
	cpy	r2H	;Patch komplett?	25930
	bcc	:3	; => Nein, weiter...	25931
				25932
::done	ldx	r10L		25933
	lda	r3L		25934
	sta	V04B8 -8,x	;Geräte-Typ.	25935
	rts			25936
				25937
; Laufwerksdaten aus ROM einlesen.				25938
:z7440	lda	#NULL		25939
	sta	v752C	;Lfwk.-Info löschen.	25940
				25941
	lda	#< v752C	;Laufwerk-Info.	25942
	sta	r0L		25943
	lda	#> v752C		25944
	sta	r0H		25945
				25946
	jsr	InitForIO		25947
				25948
	ldx	#< v7522	;"M-R \$FEA0"	25949
	ldy	#> v7522		25950
	jsr	z749A	;Befehl senden.	25951
	txa			25952
	bne	:err		25953
				25954
	ldx	#< v752A	;Lfwk.-Info einlesen.	25955
	ldy	#> v752A		25956
	jsr	z74DD	;Daten empfangen.	25957
				25958
	ldx	#< v7532	;"M-R \$FEF0"	25959
	ldy	#> v7532		25960
	jsr	z749A	;Befehl senden.	25961
	txa			25962
	bne	:err		25963
				25964
	ldx	#< v753A	;CMDFD-Info einlesen.	25965
	ldy	#> v753A		25966
	jsr	z74DD	;Daten empfangen.	25967
;	txa			25968
;	bne	:err		25969
				25970
::err	jmp	DoneWithIO		25971
				25972
; Zeiger auf Befehl setzen.				25973
:z7481	ldy	#\$00		25974
	lda	(r15),y		25975
	sta	r14L		25976
	iny			25977
	lda	(r15),y		25978

```

25979          sta    r14H
25980
25981          lda    r15L
25982          clc
25983          adc    #< 2
25984          sta    r15L
25985          lda    r15H
25986          adc    #> 2
25987          sta    r15H
25988          rts
25989
25990 ; Byte-Zähler korrigieren.
25991 :z7488          lda    r14L
25992          sec
25993          sbc    #< 1
25994          sta    r14L
25995          lda    r14H
25996          sbc    #> 1
25997          sta    r14H
25998          rts
25999
26000 ; Befehl an Laufwerk senden.
26001 ; Übergabe: X/Y = Zeiger auf Daten.
26002 ; Rückgabe: X = Fehler.
26003 :z749A          stx    r15L
26004          sty    r15H
26005
26006          lda    #$00
26007          sta    STATUS
26008
26009          jsr    UNLSN
26010
26011          lda    curDrive
26012          jsr    LISTEN
26013          lda    #$ff
26014          jsr    SECOND
26015
26016          lda    STATUS
26017          beq    :ok
26018
26019          jsr    UNLSN
26020
26021          ldx    #$ff
26022          rts
26023
26024 ::ok           jsr    z7481          ;Bytes/Befehl setzen.
26025
26026          ldy    #$00
26027 ::next        jsr    z7488
26028          bcc    :done
26029
26030          lda    (r15),y
26031          jsr    CIOUT
26032          iny
26033          bne    :next

```



```

        inc    r15H                                26034
        bne    :next                                26035
                                                26036
::done    jsr    UNLSN                                26037
                                                26038
        ldx    #$00                                26039
        rts                                         26040
                                                26041
; Daten empfangen.                                26042
; Übergabe: X/Y = Zeiger auf Daten.                26043
; Rückgabe: X = Fehler.                             26044
:z74DD    stx    r15L                                26045
        sty    r15H                                26046
                                                26047
        lda    #$00                                26048
        sta    STATUS                                26049
                                                26050
        jsr    UNTALK                                26051
                                                26052
        lda    curDrive                                26053
        jsr    TALK                                    26054
        lda    #$ff                                26055
        jsr    TKSA                                    26056
                                                26057
        lda    STATUS                                26058
        beq    :ok                                    26059
                                                26060
        jsr    UNTALK                                26061
                                                26062
        ldx    #$ff                                26063
        rts                                         26064
                                                26065
::ok      jsr    z7481                                ;Bytes/Befehl setzen. 26066
                                                26067
        ldy    #$00                                26068
::next    jsr    z7488                                26069
        bcc    :done                                26070
                                                26071
        jsr    ACPTR                                26072
        sta    (r15),y                                26073
        iny                                         26074
        bne    :next                                26075
        inc    r15H                                26076
        bne    :next                                26077
                                                26078
::done    jsr    UNTALK                                26079
                                                26080
        ldx    #$00                                26081
        rts                                         26082
                                                26083
; CMD-Gerätetyp abfragen.                          26084
:v7522    w    $0006                                26085
        b    "M-R", $a0, $fe, $06                26086
:v752A    w    $0006                                26087
:v752C    s    6    ;Laufwerk-Info.                26088

```

```

26089
26090 :v7540          b "CMD RL"
26091
26092 ; Hinweis:
26093 ; Beim erzeugen der Icons nutzt das
26094 ; Programm die Informationen von MP3.
26095 ; Test auf RD/FD/HD nicht mehr nötig.
26096 ;:v7546          b "CMD RD"
26097 ;:v754C          b "CMD FD"
26098 ;:v7552          b "CMD HD"
26099
26100 ; CMD-FD-Gerätetyp abfragen.
26101 :v7532          w $0006
26102                b "M-R", $f0, $fe, $04
26103 :v753A          w $0004
26104 :v753C          s 4      ;CMDFD-Info.
26105
26106 :v7558          b "4000"
26107
26108 ; Tabelle mit Gerätetypen.
26109 :v7172
26110 if ENABLE_DRVMODE = FALSE
26111 :_id01a
26112 :OFF_64NET41    = _id01a - v7172
26113                b $80      ;64Net/1541
26114 :_id01b
26115 :OFF_64NET71    = _id01b - v7172
26116                b $80      ;64Net/1571
26117 else
26118 :_id01a
26119 :OFF_64NET41    = _id01a - v7172
26120                b $80      ;64Net/1541/71
26121 endif
26122 :_id01c
26123 :OFF_64NET81    = _id01c - v7172
26124                b $80      ;64Net/1581
26125
26126 if ENABLE_DRVMODE = FALSE
26127 :_id02a
26128 :OFF_CBM41      = _id02a - v7172
26129                b $00      ;1541
26130 :_id02b
26131 :OFF_CBM71      = _id02b - v7172
26132                b $00      ;1571
26133 else
26134 :_id02a
26135 :OFF_CBM41      = _id02a - v7172
26136                b $00      ;1541/71
26137 endif
26138
26139 :_id02c
26140 :OFF_CBM81      = _id02c - v7172
26141                b $00      ;1581
26142 :_id02d
26143 :OFF_CBMP      = _id02d - v7172

```

```

                                b $00      ;PCDOS                                26144
                                                                26145
:_id03                                                                26146
:OFF_SD                        = _id03 - v7172                                26147
if ENABLE_DRVMODE = FALSE                                            26148
::07                          b $00      ;SD41                                26149
                                b $00      ;SD71                                26150
                                b $00      ;SD81                                26151
                                b $00      ;SDNative                            26152
else                                                                    26153
::07                          b $00      ;SD2IEC                            26154
endif                                                                    26155
                                                                26156
:_id04                                                                26157
:OFF_FD2                       = _id04 - v7172                                26158
::0b                          b $04      ;FD2                                26159
if ENABLE_DRVMODE = FALSE                                            26160
                                b $04      ;FD2/Native                            26161
                                b $04      ;FD2/DOS                            26162
endif                                                                    26163
                                                                26164
:_id05                                                                26165
:OFF_FD4                       = _id05 - v7172                                26166
::0e                          b $08      ;FD4                                26167
if ENABLE_DRVMODE = FALSE                                            26168
                                b $08      ;FD4/Native                            26169
                                b $08      ;FD4/DOS                            26170
endif                                                                    26171
                                                                26172
:_id06                                                                26173
:OFF_HD                        = _id06 - v7172                                26174
::11                          b $10      ;HD                                26175
if ENABLE_DRVMODE = FALSE                                            26176
                                b $10      ;HD/Native                            26177
endif                                                                    26178
                                                                26179
:_id07                                                                26180
:OFF_RAM                       = _id07 - v7172                                26181
if ENABLE_DRVMODE = FALSE                                            26182
::13                          b $00      ;RAM41                                26183
                                b $00      ;RAM71                                26184
                                b $00      ;RAM81                                26185
                                b $00      ;RAMNative                            26186
else                                                                    26187
::13                          b $00      ;RAMDisk                            26188
endif                                                                    26189
                                                                26190
:_id08                                                                26191
:OFF_EXTRAM                    = _id08 - v7172                                26192
::17                          b $00      ;SuperCPU/Native                    26193
                                b $00      ;C=REU/Native                        26194
                                b $00      ;GeoRAM/Native                        26195
                                                                26196
:_id09                                                                26197
:OFF_RL                        = _id09 - v7172                                26198

```

```

26199 ::1a          b $01      ;RAMLink
26200 if ENABLE_DRVMODE = FALSE
26201          b $01      ;RAMLink/Native
26202 endif
26203
26204 :_id10
26205 :OFF_RD          = _id10 - v7172
26206 ::1c          b $02      ;RAMDrive
26207 if ENABLE_DRVMODE = FALSE
26208          b $02      ;RAMDrive/Native
26209 endif
26210
26211 :_id11
26212 :OFF_UNKNOWN     = _id11 - v7172
26213 ::1e          b $00      ;Unbekannt
26214
26215 :_idxx
26216 :MAX_ICONS       = _idxx - v7172
26217
26218 ; Tabelle mit Daten für Laufwerk-Icon.
26219 :v7170
26220 if ENABLE_DRVMODE = FALSE
26221          w GR79D8    ;64Net/1541
26222          w GR79D8    ;64Net/1571
26223 else
26224          w GR76EB    ;64Net/1541/71
26225 endif
26226          w GR7752    ;64Net/1581
26227
26228 if ENABLE_DRVMODE = FALSE
26229          w GR76EB    ;1541
26230          w GR76EB    ;1571
26231 else
26232          w GR76EB    ;1541/71
26233 endif
26234
26235          w GR7752    ;1581
26236          w GR7752    ;PCDOS
26237
26238 if ENABLE_DRVMODE = FALSE
26239          w GR7A80    ;SD41
26240          w GR7A80    ;SD71
26241          w GR7A80    ;SD81
26242          w GR7A80    ;SDNative
26243 else
26244          w GR7A80    ;SD2IEC
26245 endif
26246
26247          w GR7752    ;FD2
26248 if ENABLE_DRVMODE = FALSE
26249          w GR7752    ;FD2/Native
26250          w GR7752    ;FD2/DOS
26251 endif
26252
26253          w GR7752    ;FD4

```

```

if ENABLE_DRVMODE = FALSE                                     26254
    w GR7752    ;FD4/Native                                   26255
    w GR7752    ;FD4/DOS                                     26256
endif                                                         26257
                                                             26258
    w GR7752    ;HD                                           26259
if ENABLE_DRVMODE = FALSE                                     26260
    w GR7752    ;HD/Native                                    26261
endif                                                         26262
                                                             26263
if ENABLE_DRVMODE = FALSE                                     26264
    w GR77D9    ;RAM41                                         26265
    w GR77D9    ;RAM71                                         26266
    w GR77D9    ;RAM81                                         26267
    w GR77D9    ;RAMNative                                    26268
else                                                         26269
    w GR77D9    ;RAMDisk                                       26270
endif                                                         26271
                                                             26272
    w GR77D9    ;SuperCPU/Native                              26273
    w GR77D9    ;C=REU/Native                                 26274
    w GR77D9    ;GeoRAM/Native                                26275
                                                             26276
    w GR78B1    ;RAMLink                                       26277
if ENABLE_DRVMODE = FALSE                                     26278
    w GR78B1    ;RAMLink/Native                               26279
endif                                                         26280
                                                             26281
    w GR792D    ;RAMDrive                                       26282
if ENABLE_DRVMODE = FALSE                                     26283
    w GR792D    ;RAMDrive/Native                              26284
endif                                                         26285
                                                             26286
    w GR76EB    ;Unbekannt                                     26287
                                                             26288
; Tabelle mit Patch-Daten:                                   26289
;   w $0000    ;Adr. Patch-Daten.                             26290
;   b $00      ;Offset Patch-Daten.                           26291
;   b $00      ;Anzahl Patch-Bytes.                           26292
:v7171                                                         26293
if ENABLE_DRVMODE = FALSE                                     26294
    w $0000     ;64Net/1541                                     26295
    b $00                                              26296
    b $00                                              26297
    w GR773E    ;64Net/1571                                     26298
    b $04                                              26299
    b $0d                                              26300
else                                                         26301
    w GR7A55    ;64Net/1541/71                                 26302
    b $28                                              26303
    b $0f                                              26304
endif                                                         26305
    w GR7A56    ;64Net/1581                                     26306
    b $07                                              26307
    b $12                                              26308

```

```

26309
26310 if ENABLE_DRVMODE = FALSE
26311     w $0000    ;1541
26312     b $00
26313     b $00
26314     w GR773E   ;1571
26315     b $04
26316     b $0d
26317 else
26318     w $0000    ;1541/71
26319     b $00
26320     b $00
26321 endif
26322     w $0000    ;1581
26323     b $00
26324     b $00
26325     w GR7A6B   ;PCDOS
26326     b $07
26327     b $12
26328
26329 if ENABLE_DRVMODE = FALSE
26330     w $0000    ;SD41
26331     b $00
26332     b $00
26333     w GR7ACD   ;SD71
26334     b $08
26335     b $11
26336     w GR7ADE   ;SD81
26337     b $08
26338     b $11
26339     w GR7AEF   ;SDNative
26340     b $08
26341     b $11
26342 else
26343     w $0000    ;SD2IEC
26344     b $00
26345     b $00
26346 endif
26347
26348     w GR796F   ;FD2
26349     b $07
26350     b $12
26351 if ENABLE_DRVMODE = FALSE
26352     w GR7999   ;FD2/Native
26353     b $07
26354     b $12
26355     w GR7A6B   ;FD2/DOS
26356     b $07
26357     b $12
26358 endif
26359
26360     w GR7984   ;FD4
26361     b $07
26362     b $12
26363 if ENABLE_DRVMODE = FALSE

```

```

        w GR7999    ;FD4/Native      26364
        b $07      26365
        b $12      26366
        w GR7A6B    ;FD4/DOS        26367
        b $07      26368
        b $12      26369
endif      26370
      26371
        w GR79AE    ;HD              26372
        b $07      26373
        b $12      26374
if ENABLE_DRVMODE = FALSE      26375
        w GR79C3    ;HD/Native      26376
        b $07      26377
        b $12      26378
endif      26379
      26380
if ENABLE_DRVMODE = FALSE      26381
        w $0000     ;RAM41          26382
        b $00      26383
        b $00      26384
        w GR782D    ;RAM71          26385
        b $16      26386
        b $15      26387
        w GR7843    ;RAM81          26388
        b $16      26389
        b $15      26390
        w GR7859    ;RAMNative      26391
        b $16      26392
        b $15      26393
else      26394
        w $0000     ;RAMDisk        26395
        b $00      26396
        b $00      26397
endif      26398
      26399
        w GR786F    ;SuperCPU/Native 26400
        b $16      26401
        b $15      26402
        w GR789B    ;C=REU/Native   26403
        b $16      26404
        b $15      26405
        w GR7885    ;GeoRAM/Native   26406
        b $16      26407
        b $15      26408
      26409
        w $0000     ;RAMLink        26410
        b $00      26411
        b $00      26412
if ENABLE_DRVMODE = FALSE      26413
        w GR78EF    ;RAMLink/Native 26414
        b $07      26415
        b $0f      26416
endif      26417
        w $0000     ;RAMDrive       26418

```

```

26419      b $00
26420      b $00
26421  if ENABLE_DRVMODE = FALSE
26422      w GR792E    ;RAMDrive/Native
26423      b $0d
26424      b $0f
26425  endif
26426
26427      w GR773F    ;Unbekannt
26428      b $04
26429      b $10
26430
26431  ; Hinweis:
26432  ; DoIcons definiert die Größe aller
26433  ; Laufwerk-Icons mit x=3 / y=20 Pixel.
26434  ;
26435  ; 1541-Icon.
26436  if ENABLE_DRVMODE = FALSE
26437  :GR76EB      b $80 +DRVICON_X *DRVICON_Y
26438              b %01111111,%11111111,%11111110
26439              b %01101110,%11111111,%11111110
26440              b %01101010,%11111111,%11111110
26441              b %01100010,%11111111,%11111100
26442              b %01111010,%11111111,%11111100
26443              b %01111010,%11111111,%11111110
26444              b %01111111,%11111111,%11111110
26445              b %01111111,%11000011,%11111110
26446              b %01111111,%10011001,%11111110
26447              b %01111111,%10100101,%11111110
26448              b %01111111,%10011001,%11111110
26449              b %01111111,%11000011,%11111110
26450              b %01111111,%11111111,%11111110
26451              b %01111111,%11111111,%11111110
26452              b %01111111,%11111111,%11111110
26453              b %01111111,%11100111,%11111110
26454              b %01111111,%11100111,%11111110
26455              b %01111111,%11100111,%11111110
26456              b %01111111,%11111111,%11111110
26457              b %01111111,%01111110,%11111110
26458  else
26459  :GR76EB      b $80 +DRVICON_X *DRVICON_Y
26460              b %01111111,%11111111,%11111110
26461              b %01000000,%01111111,%11111110
26462              b %01000000,%01111111,%11111110
26463              b %01000000,%01111111,%11111100
26464              b %01000000,%01111111,%11111100
26465              b %01111111,%11111111,%11111110
26466              b %01111111,%11111111,%11111110
26467              b %01111111,%11000011,%11111110
26468              b %01111111,%10011001,%11111110
26469              b %01111111,%10100101,%11111110
26470              b %01111111,%10011001,%11111110
26471              b %01111111,%11000011,%11111110
26472              b %01111111,%11111111,%11111110
26473              b %01111111,%11111111,%11111110

```



```

        b %01111111,%11111111,%11111110 26474
        b %01111111,%11100111,%11111110 26475
        b %01111111,%11100111,%11111110 26476
        b %01111111,%11100111,%11111110 26477
        b %01111111,%11111111,%11111110 26478
        b %01111111,%01111110,%11111110 26479
endif 26480
26481
if ENABLE_DRVMODE = FALSE 26482
; Kennung "41". 26483
;:GR772A        b %01101110,%11111111,%11111110 26484
;               b %01101010,%11111111,%11111110 26485
;               b %01100010,%11111111,%11111100 26486
;               b %01111010,%11111111,%11111100 26487
;               b %01111010 26488
26489
; Kennung "71". 26490
:GR773E        b %01100010,%11111111,%11111110 26491
               b %01111010,%11111111,%11111110 26492
               b %01111010,%11111111,%11111100 26493
               b %01110110,%11111111,%11111100 26494
               b %01110110 26495
endif 26496
26497
; Kennung "?". 26498
:GR773F        b %01110001,%11111111,%11111110 26499
               b %01101110,%11111111,%11111110 26500
               b %01111101,%11111111,%11111100 26501
               b %01111011,%11111111,%11111100 26502
               b %01111111,%11111111,%11111110 26503
               b %01111011 26504
26505
; 1581-Icon. 26506
if ENABLE_DRVMODE = FALSE 26507
:GR7752        b $80 +DRVICON_X *DRVICON_Y 26508
               b %00111111,%11111111,%11111100 26509
               b %00111000,%00000000,%00001110 26510
               b %00111010,%11100110,%01001110 26511
               b %00111010,%10001001,%01001110 26512
               b %00111010,%11000110,%01001110 26513
               b %00111010,%00101001,%01001110 26514
               b %00111010,%00101001,%01001110 26515
               b %00111010,%11000110,%01001110 26516
               b %00111000,%00000000,%00001110 26517
               b %00111111,%11111111,%11111110 26518
               b %00111111,%11111111,%11111110 26519
               b %00111111,%11111111,%11111110 26520
               b %00111111,%01010101,%00001110 26521
               b %00111110,%10101010,%10001110 26522
               b %00111111,%00000101,%00001110 26523
               b %00111110,%10001010,%10001110 26524
               b %00111111,%00000101,%00001110 26525
               b %00111110,%10101010,%10001110 26526
               b %00011111,%01010101,%01111100 26527
               b %00000000,%00000000,%00000000 26528

```

```

26529 else
26530 :GR7752          b $80 +DRVICON_X *DRVICON_Y
26531                b %00111111,%11111111,%11111100
26532                b %00111000,%00000000,%00001110
26533                b %00111000,%00110000,%00001110
26534                b %00111000,%01000111,%00001110
26535                b %00111000,%01000110,%00001110
26536                b %00111000,%01000110,%00001110
26537                b %00111000,%01000111,%00001110
26538                b %00111000,%00110000,%00001110
26539                b %00111000,%00000000,%00001110
26540                b %00111111,%11111111,%11111110
26541                b %00111111,%11111111,%11111110
26542                b %00111111,%11111111,%11111110
26543                b %00111111,%01010101,%00001110
26544                b %00111110,%10101010,%10001110
26545                b %00111111,%00000101,%00001110
26546                b %00111110,%10001010,%10001110
26547                b %00111111,%00000101,%00001110
26548                b %00111110,%10101010,%10001110
26549                b %00011111,%01010101,%01111100
26550                b %00000000,%00000000,%00000000
26551 endif
26552
26553 ; Kennung "1581".
26554 :GR778F          b %00111010,%11100110,%01001110
26555                b %00111010,%10001001,%01001110
26556                b %00111010,%11000110,%01001110
26557                b %00111010,%00101001,%01001110
26558                b %00111010,%00101001,%01001110
26559                b %00111010,%11000110,%01001110
26560
26561 ; Kennung "CMD-FD/FD2".
26562 :GR796F          b %00111011,%10110000,%00001110
26563                b %00111010,%00101000,%11001110
26564                b %00111011,%00101001,%00101110
26565                b %00111010,%00101000,%01001110
26566                b %00111010,%00101000,%10001110
26567                b %00111010,%00110001,%11101110
26568
26569 ; Kennung "CMD-FD/FD4".
26570 :GR7984          b %00111011,%10110000,%00001110
26571                b %00111010,%00101001,%00001110
26572                b %00111011,%00101001,%01001110
26573                b %00111010,%00101001,%11101110
26574                b %00111010,%00101000,%01001110
26575                b %00111010,%00110000,%01001110
26576
26577 ; Kennung "CMD-FD/FDNative".
26578 if ENABLE_DRVMODE = FALSE
26579 :GR7999          b %00111011,%10110000,%00001110
26580                b %00111010,%00101001,%00101110
26581                b %00111011,%00101001,%10101110
26582                b %00111010,%00101001,%01101110
26583                b %00111010,%00101001,%00101110

```

```

                                b %00111010,%00110001,%00101110      26584
endif                                                                    26585
                                                                    26586
; Kennung "CMD-HD".                                                    26587
:GR79AE                        b %00111000,%01010110,%00001110      26588
                                b %00111000,%01010101,%00001110      26589
                                b %00111000,%01110101,%00001110      26590
                                b %00111000,%01010101,%00001110      26591
                                b %00111000,%01010101,%00001110      26592
                                b %00111000,%01010110,%00001110      26593
                                                                    26594
; Kennung "CMD-HD/Native".                                              26595
if ENABLE_DRVMODE = FALSE                                              26596
:GR79C3                        b %00111010,%10110000,%00001110      26597
                                b %00111010,%10101001,%00101110      26598
                                b %00111011,%10101001,%10101110      26599
                                b %00111010,%10101001,%01101110      26600
                                b %00111010,%10101001,%00101110      26601
                                b %00111010,%10110001,%00101110      26602
endif                                                                    26603
                                                                    26604
; Kennung "DOS".                                                        26605
:GR7A6B                        b %00111001,%10001000,%11001110      26606
                                b %00111001,%01010101,%00001110      26607
                                b %00111001,%01010100,%10001110      26608
                                b %00111001,%01010100,%01001110      26609
                                b %00111001,%01010100,%01001110      26610
                                b %00111001,%10001001,%10001110      26611
                                                                    26612
; RAMDisk                                                                26613
if ENABLE_DRVMODE = FALSE                                              26614
:GR77D9                        b $80 +DRVICON_X *DRVICON_Y          26615
                                b %01111111,%11111111,%11111110      26616
                                b %01000000,%00000000,%00000010      26617
                                b %01011111,%11111111,%11111010      26618
                                b %01000000,%00000000,%00000010      26619
                                b %01011111,%11111111,%11111010      26620
                                b %01000000,%00000000,%00000010      26621
                                b %01011111,%11111110,%00000010      26622
                                b %01000000,%00000000,%10001010      26623
                                b %01011100,%10010010,%10101010      26624
                                b %01010101,%01010110,%10101010      26625
                                b %01010101,%01011010,%11101010      26626
                                b %01011101,%11010010,%00101010      26627
                                b %01010101,%01010010,%00101010      26628
                                b %01010101,%01010010,%00101010      26629
                                b %01000000,%00000000,%00000010      26630
                                b %01111111,%11111111,%11111110      26631
                                b %00010000,%00010000,%00000000      26632
                                b %00010111,%11010000,%00000000      26633
                                b %00010000,%00010000,%00000000      26634
                                b %00011111,%11110000,%00000000      26635
                                                                    26636
; Kennung "RAM1541".                                                    26637
;:GR7817                      b %01000000,%00000000,%10001010      26638

```

```

26639 ; b %01011100,%10010010,%10101010
26640 ; b %01010101,%01010110,%10101010
26641 ; b %01010101,%01011010,%11101010
26642 ; b %01011101,%11010010,%00101010
26643 ; b %01010101,%01010010,%00101010
26644 ; b %01010101,%01010010,%00101010
26645
26646 ; Kennung "RAM1571".
26647 :GR782D b %01000000,%00000000,%11101010
26648 b %01011100,%10010010,%00101010
26649 b %01010101,%01010110,%00101010
26650 b %01010101,%01011010,%01001010
26651 b %01011101,%11010010,%01001010
26652 b %01010101,%01010010,%01001010
26653 b %01010101,%01010010,%01001010
26654
26655 ; Kennung "RAM1581".
26656 :GR7843 b %01000000,%00000000,%01001010
26657 b %01011100,%10010010,%10101010
26658 b %01010101,%01010110,%10101010
26659 b %01010101,%01011010,%01001010
26660 b %01011101,%11010010,%10101010
26661 b %01010101,%01010010,%10101010
26662 b %01010101,%01010010,%01001010
26663
26664 ; Kennung "RAMNative".
26665 :GR7859 b %01000000,%00000000,%01001010
26666 b %01011100,%10010010,%01101010
26667 b %01010101,%01010110,%01101010
26668 b %01010101,%01011010,%01011010
26669 b %01011101,%11010010,%01011010
26670 b %01010101,%01010010,%01001010
26671 b %01010101,%01010010,%01001010
26672 else
26673 :GR77D9 b $80 +DRVICON_X *DRVICON_Y
26674 b %01111111,%11111111,%11111110
26675 b %01000000,%00000000,%00000010
26676 b %01011111,%11111111,%11111010
26677 b %01000000,%00000000,%00000010
26678 b %01011111,%11111111,%11111010
26679 b %01000000,%00000000,%00000010
26680 b %01011111,%11111110,%00000010
26681 b %01000000,%00000000,%00000010
26682 b %01011100,%10010001,%00000010
26683 b %01010101,%01011011,%00000010
26684 b %01010101,%01010101,%00000010
26685 b %01011101,%11010001,%00000010
26686 b %01010101,%01010001,%00000010
26687 b %01010101,%01010001,%00000010
26688 b %01000000,%00000000,%00000010
26689 b %01111111,%11111111,%11111110
26690 b %00010000,%00010000,%00000000
26691 b %00010111,%11010000,%00000000
26692 b %00010000,%00010000,%00000000
26693 b %00011111,%11111000,%00000000

```

```

endif 26694
26695
; Kennung "SuperRAM/Native". 26696
:GR786F b %01000000,%00000000,%01001010 26697
b %01001101,%01011000,%01101010 26698
b %01010001,%01010100,%01101010 26699
b %01001001,%01010100,%01011010 26700
b %01000101,%01011000,%01011010 26701
b %01000101,%01010000,%01001010 26702
b %01011001,%10010000,%01001010 26703
26704
; Kennung "GeoRAM/Native". 26705
:GR7885 b %01000000,%00000000,%01001010 26706
b %01011101,%11001000,%01101010 26707
b %01010001,%000010100,%01101010 26708
b %01011101,%10010100,%01011010 26709
b %01010101,%000010100,%01011010 26710
b %01010101,%000010100,%01001010 26711
b %01011101,%11001000,%01001010 26712
26713
; Kennung "C=REU/Native". 26714
:GR789B b %01000000,%00000000,%01001010 26715
b %01011001,%11010100,%01101010 26716
b %01010101,%000010100,%01101010 26717
b %01010101,%10010100,%01011010 26718
b %01011001,%000010100,%01011010 26719
b %01010101,%000010100,%01001010 26720
b %01010101,%11011000,%01001010 26721
26722
; RAMLink 26723
:GR78B1 b $80 +DRVICON_X *DRVICON_Y 26724
b %11111111,%11111111,%11111110 26725
b %11111111,%11111111,%11101010 26726
b %11000000,%00011111,%11101110 26727
b %11000000,%00011111,%11111010 26728
b %11111111,%11111111,%11111110 26729
b %11111111,%11111111,%11101010 26730
b %11000000,%00011111,%11101110 26731
b %11000000,%00011111,%11111010 26732
b %11111111,%11111111,%11101110 26733
b %11000011,%11111111,%11101010 26734
b %11100001,%11111111,%11111110 26735
b %11000011,%11111111,%11111110 26736
b %11111111,%11111111,%11111010 26737
b %10111111,%01111011,%11111010 26738
b %10000000,%01000001,%11111110 26739
b %10100000,%01000011,%11111010 26740
b %10000000,%00000001,%11111010 26741
b %11111111,%11111111,%11111110 26742
b %00111111,%11100000,%00000000 26743
b %00101010,%10100000,%00000000 26744
26745
; Kennung "RAMLink/Native" 26746
if ENABLE_DRVMODE = FALSE 26747
:GR78EF b %11000000,%00011011,%01101110 26748

```

```

26749      b %11000000,%00011001,%01111010
26750      b %11111111,%11111000,%01111110
26751      b %11111111,%11111010,%01101010
26752      b %11000000,%00011011,%01101110
26753  endif
26754
26755  ; RAMDrive
26756  :GR792D      b $80 +DRVICON_X *DRVICON_Y
26757      b %01111111,%11111111,%11111111
26758      b %01100111,%11101111,%11111001
26759      b %01100111,%11101111,%11111001
26760      b %01111111,%11111111,%11111111
26761      b %01111111,%11111111,%111110101
26762      b %01111111,%11111111,%11111111
26763      b %01111111,%11111111,%111111101
26764      b %01111111,%11111111,%11111111
26765      b %01111111,%11111111,%111111101
26766      b %01100011,%01111111,%11111111
26767      b %01101001,%00111111,%111111101
26768      b %01100011,%01111111,%11111111
26769      b %01111111,%11111111,%11111111
26770      b %01101101,%11010111,%11111111
26771      b %01110011,%00110010,%11001001
26772      b %01101001,%00110010,%11011011
26773      b %01110011,%00110110,%11011001
26774      b %01111111,%11111111,%11111111
26775      b %00001100,%11001100,%11001100
26776      b %00001011,%00110011,%00110100
26777
26778  ; Kennung "RAMDrive/Native"
26779  if ENABLE_DRVMODE = FALSE
26780  :GR792E      b %01111111,%11111011,%01110101
26781      b %01111111,%11111001,%01111111
26782      b %01111111,%11111000,%011111101
26783      b %01111111,%11111010,%01111111
26784      b %01111111,%11111011,%011111101
26785  endif
26786
26787  ; 64Net/41-Icon.
26788  if ENABLE_DRVMODE = FALSE
26789  :GR79D8      b $80 +DRVICON_X *DRVICON_Y
26790      b %01111111,%11111111,%11111110
26791      b %01101110,%11111111,%11111110
26792      b %01101010,%11111111,%11111110
26793      b %01100010,%11111111,%111111100
26794      b %01111010,%11111111,%111111100
26795      b %01111010,%11111111,%11111110
26796      b %01111111,%11111111,%11111110
26797      b %01111111,%11000011,%11111110
26798      b %01111111,%10011001,%11111110
26799      b %01111111,%10100101,%11111110
26800      b %01111111,%10011001,%11111110
26801      b %01111111,%11000011,%11111110
26802      b %01111111,%11111111,%11111110
26803      b %01110110,%11111110,%00100010

```

```

        b %01110010,%11111110,%11110110      26804
        b %01110100,%11100110,%01110110      26805
        b %01110110,%11100110,%11110110      26806
        b %01110110,%11100110,%00110110      26807
        b %01111111,%11111111,%11111110      26808
        b %01111111,%10111101,%11111110      26809
        26810
; Kennung "64Net/1571" (wie C=1571)             26811
;:GR7A17      b %01100010,%11111111,%11111110      26812
;      b %01111010,%11111111,%11111110      26813
;      b %01111010,%11111111,%11111100      26814
;      b %01110110,%11111111,%11111100      26815
;      b %01110110                                26816
else                                              26817
; Kennung "64Net/1541/71"                       26818
:GR7A55      b %01110110,%11111110,%00100010      26819
        b %01110010,%11111110,%11110110      26820
        b %01110100,%11100110,%01110110      26821
        b %01110110,%11100110,%11110110      26822
        b %01110110,%11100110,%00110110      26823
endif                                             26824
        26825
; Kennung "64Net/1581"                         26826
:GR7A56      b %00111001,%00101110,%11101110      26827
        b %00111001,%10101000,%01001010      26828
        b %00111001,%01101100,%01001110      26829
        b %00111001,%00101000,%01001110      26830
        b %00111001,%00101110,%01001110      26831
        b %00111000,%00000000,%00001110      26832
        26833
; SD2IEC-Icon                                  26834
if ENABLE_DRVMODE = FALSE                       26835
:GR7A80      b $80 +DRVICON_X *DRVICON_Y          26836
        b %00111111,%11111111,%11111000      26837
        b %00111000,%00000000,%00001100      26838
        b %00011011,%10001100,%10001110      26839
        b %00111010,%01010000,%10101110      26840
        b %00011010,%01011100,%10101110      26841
        b %00111010,%01010010,%11101100      26842
        b %00011010,%01010010,%00101100      26843
        b %00011011,%10011100,%00101110      26844
        b %00011000,%00000000,%00001110      26845
        b %00111111,%11111111,%11111110      26846
        b %00111111,%11111111,%11111110      26847
        b %00111111,%00011000,%11111110      26848
        b %00111110,%11111011,%01111110      26849
        b %00111111,%00111011,%01111110      26850
        b %00111111,%11011011,%01111110      26851
        b %00111110,%00111000,%11111110      26852
        b %00111111,%11111111,%11111110      26853
        b %00111111,%11111111,%11111110      26854
        b %00011111,%01010101,%01111110      26855
        b %00000000,%00000000,%00000000      26856
        26857
; Kennung "SD2IEC/D64".                        26858

```

```

26859 ;:GR7ABC          b          %10001100,%10001110
26860 ;                b %00111010,%01010000,%10101110
26861 ;                b %00011010,%01011100,%10101110
26862 ;                b %00111010,%01010010,%11101100
26863 ;                b %00011010,%01010010,%00101100
26864 ;                b %00011011,%10011100,%00101110
26865
26866 ; Kennung "SD2IEC/D71".
26867 :GR7ACD          b          %10011110,%01001110
26868                b %00111010,%01000010,%11001110
26869                b %00011010,%01000010,%01001110
26870                b %00111010,%01000100,%01001100
26871                b %00011010,%01000100,%01001100
26872                b %00011011,%10000100,%01001110
26873
26874 ; Kennung "SD2IEC/D81".
26875 :GR7ADE          b          %10001100,%01001110
26876                b %00111010,%01010010,%11001110
26877                b %00011010,%01001100,%01001110
26878                b %00111010,%01010010,%01001100
26879                b %00011010,%01010010,%01001100
26880                b %00011011,%10001100,%01001110
26881
26882 ; Kennung "SD2IEC/DNP".
26883 :GR7AEF          b          %10010010,%11001110
26884                b %00111010,%01011010,%10101110
26885                b %00011010,%01011110,%10101110
26886                b %00111010,%01010110,%11001100
26887                b %00011010,%01010010,%10001100
26888                b %00011011,%10010010,%10001110
26889 else
26890 :GR7A80          b $80 +DRVICON_X *DRVICON_Y
26891                b %00111111,%11111111,%11111000
26892                b %00111000,%00000000,%00001100
26893                b %00011000,%00110110,%00001110
26894                b %00111000,%01000101,%00001110
26895                b %00011000,%00100101,%00001110
26896                b %00111000,%00010101,%00001100
26897                b %00011000,%00010101,%00001100
26898                b %00011000,%01100110,%00001110
26899                b %00011000,%00000000,%00001110
26900                b %00111111,%11111111,%11111110
26901                b %00111111,%11111111,%11111110
26902                b %00111111,%11111111,%11111110
26903                b %00111111,%11111111,%11111110
26904                b %00111111,%11111111,%11111110
26905                b %00111111,%11111111,%11111110
26906                b %00111111,%11111111,%11111110
26907                b %00111111,%11111111,%11111110
26908                b %00111111,%11111111,%11111110
26909                b %00011111,%01010101,%01111110
26910                b %00000000,%00000000,%00000000
26911 endif
26912
26913 ; Hinweis:

```





```

26928
26929 ;
26930 ; SourceCode: src.mod#08.s
26931 ;
26932
26933 ;
26934 ; Modul-Info:
26935 ; 0 - DiskImage erstellen
26936 ; 1 - DiskCopy
26937 ;
26938 if .p
26939             t "TopSym"
26940             t "TopSym.MP3"
26941             t "TopSym.ROM"
26942             t "DTopDeskI18N.ext"
26943             t "src.DTopDesk.ext"
26944
26945 ; Anzahl Sektoren je Track:
26946 :TRACK41      = 35
26947 :TRACK71      = 70
26948 :TRACK81      = 80
26949
26950 ; Datenkanel für SD2IEC:
26951 :IEC_DATA_LFN    = 2
26952 :IEC_DATA_SEK    = 2
26953 :IEC_DEV_CHAN    = 15
26954
26955 ; Speicher für DiskCopy:
26956 :COPYMEM_DISK    = $7600
26957
26958 :DCOPYSTART      = COPYMEM_DISK ;$7600
26959 :DCOPYEND        = OS_VARS      ;$8000
26960 endif
26961
26962             o BASE_MOD_A
26963             n "obj.#08"
26964
26965 ::ModEntry_0      jmp    N7768 ;DiskImage erstellen.
26966 ::ModEntry_1      jmp    N6EB2 ;DiskCopy
26967
26968 ; 1: CopyDisk
26969 ; TD13: DiskCopy
26970 :N6EB2            jsr    S5C89 ;Hole Disk/Fenster.
26971                 jsr    S5245 ;Dateien invertieren.
26972
26973                 lda    #$00 ;Kein VLIR-Modul mehr
26974                 sta    V0422 ;im Speicher!
26975
26976                 jsr    S59F1 ;Diskname akt.Fens.
26977
26978                 lda    #< V0483 +2 ;Name Quelldisk.
26979                 sta    r6L
26980                 lda    #> V0483 +2
26981                 sta    r6H
26982                 jsr    z71D9 ;DiskCopy.

```

			26983
	lda	#\$00 ;Status "Datei/Disk"	26984
	sta	V5611 ;zurücksetzen.	26985
			26986
	txa		26987
	beq	:done	26988
	cmp	#CANCEL_ERR	26989
	bne	:err	26990
	rts		26991
			26992
::err	jmp	S5CF3 ;Fehler ausgeben.	26993
			26994
::done	jsr	S5255 ;Mark.Dat=0/Anzeige.	26995
			26996
		; Menü starten um Status zu löschen.	26997
		; jsr S54F0 ;Hauptmenü starten.	26998
			26999
		; Hinweis:	27000
		; Beim kopieren/löschen von Dateien	27001
		; wird der Status erst nach dem Update	27002
		; des Fensters gelöscht.	27003
		; Die Routine kann aber nicht nach dem	27004
		; Update über JSR aufgerufen werden, da	27005
		; im Textmodus der Modulspeicher zum	27006
		; Teil überschrieben wird.	27007
	lda	#> S54F0 -1 ;Hauptmenü starten.	27008
	pha		27009
	lda	#< S54F0 -1	27010
	pha		27011
			27012
		; Hinweis:	27013
		; Im Textmodus wird der Bereich für	27014
		; VLIR-Module überschrieben!	27015
	jmp	S4321 ;Alle Fenster neu.	27016
			27017
:v6EE8	b	\$00 ;driveMode Source (41/71/81/NM)	27018
:v6EE9	b	\$00 ;MyInterleave	27019
:v6EEA	b	\$00 ;driveType für Source-Disk	27020
:v6EEB	b	\$00 ;Make2SFlag, \$FF=BAM#2 löschen.	27021
			27022
		; Hinweis:	27023
		; Register müssen zusammenbleiben, da	27024
		; über das Y-Register indexiert.	27025
		;---	27026
:v6EEC	b	\$00 ;Lfwk. Source-Disk.	27027
:v6EED	b	\$00 ;Lfwk. Target-Disk.	27028
		;---	27029
:v6EEE	w	\$0000 ;SourceName	27030
			27031
		; Hinweis:	27032
		; Damit der Programmcode nicht geändert	27033
		; werden muss, Flag für 1541/71/81 oder	27034
		; NativeMode setzen.	27035
:v6EEF	b	\$00 ;\$FF = Native.	27036
			27037

```

27038 ; Max. Tracks für 1541/71/81.
27039 :v7745          b TRACK41 +1
27040              b TRACK71 +1
27041              b TRACK81 +1
27042
27043 :v7748          b $00  ;Typ Source-Disk.
27044 :v7749          b $00  ;Anz.Tracks übrig.
27045 :v774A          b $00  ;Anzahl Tracks Source.
27046 :v774B          s 5    ;Native: Disk-ID.
27047 :v7750          s 18   ;Native: Diskname.
27048
27049 ; DiskCopy initialisieren.
27050 ; Übergabe: r6 = Zeiger auf Diskname.
27051 :z71D9          lda    r6L                ;Zeiger auf Diskname
27052              sta    v6EEE +0              ;für Source-Disk.
27053              lda    r6H
27054              sta    v6EEE +1
27055
27056              lda    curType                ;Source-Disk.
27057              cmp    #Drv1571              ;Echte 1571?
27058              bne    :1                    ; => Nein, weiter...
27059
27060              lda    #$08
27061              sta    interleave
27062
27063 ::1              jsr    OpenDisk
27064              txa
27065              bne    :err
27066
27067              lda    curDirHead
27068              sta    v6FB8                  ;BAM-Track setzen.
27069              lda    interleave
27070              sta    v6EE9                  ;MyInterleave.
27071
27072              lda    curType
27073              sta    v6EEA                  ;driveType Source.
27074              and    #ST_DM0DES
27075              sta    v6EE8                  ;driveMode Source.
27076              cmp    #Drv1571              ;1571?
27077              bne    :2                    ; => Nein, weiter...
27078
27079              ldx    curDirHead +3          ;Doppelseitig?
27080              bmi    :2                      ; => Ja, weiter...
27081
27082              lda    #Drv1541                ;Einseitig = 1541.
27083 ::2              sta    v7748              ;Typ Source-Disk.
27084
27085              lda    curDrive
27086              sta    v6EEC                  ;Lfwk. Source-Disk.
27087 ;              sta    v6EED                  ;Lfwk. Target-Disk.
27088
27089 ; Laufwerksliste für DBox-Icons erstellen.
27090 ; Rückgabe: Y = Anzahl Laufwerke.
27091              jsr    z7350                  ;Laufwerksliste.
27092

```



```

27148 ::1          sty    v72D6
27149
27150 ; Verify-Option 1541/71 definieren.
27151          tay          ;RAMDisk?
27152          bmi    :2          ; => Ja, weiter...
27153          ldx    #1          ;41/71: Verify.
27154          lda    v7748          ;Typ Source-Disk.
27155          cmp    #Drv1581
27156          bcc    :3
27157 ::2          ldx    #11          ;RAM/81/NM: Kein Ver.
27158 ::3          stx    v7065          ;Verify-Flag.
27159
27160 ; Einsprung bei unterschiedlichen
27161 ; Native-Größen bei DiskCopy.
27162 :z7260      ldx    #0          ;Laufwerke für
27163 ::1          lda    driveType,x          ;MP3/Lfwk.-Icons
27164          pha          ;vorbereiten.
27165          lda    v7351,x
27166          sta    driveType,x
27167          inx
27168          cpx    #4
27169          bcc    :1
27170
27171          lda    #< d72B3          ;Lfwk. wählen.
27172          ldx    #> d72B3
27173          jsr    S2090          ;DBox ohne Recover.
27174
27175          ldx    #4 -1
27176 ::2          pla          ;Laufwerke wieder
27177          sta    driveType,x          ;zurücksetzen.
27178          dex
27179          bpl    :2
27180
27181          lda    r0L          ;Laufwerk?
27182          bmi    :drive          ; => Ja, weiter...
27183          cmp    #OK          ;"OK"?
27184          beq    :ok          ; => Ja, weiter...
27185          cmp    #DISK          ;"DISK"?
27186          beq    :disk          ; => Ja, weiter...
27187          bne    :cancel          ; => Abbruch...
27188
27189 ; Dialogbox: Laufwerk.
27190 ::drive      and    #%00001111
27191          sta    v6EED          ;Lfwk. Target-Disk.
27192          jsr    SetDevice
27193
27194          lda    #$00
27195          sta    v6ED8          ;Name Target-Disk.
27196
27197 ; Dialogbox: Disk.
27198 ::disk      jsr    z73A3          ;Neue Diskette.
27199          txa          ;Fehler?
27200          bne    :err          ; => Ja, Abbruch...
27201          beq    x7249          ; => Neues Laufwerk.
27202

```

```

; Dialogbox: OK.                                     27203
::ok          lda    curDirHead +$bd                 27204
              cmp    #"P"                           ;Hauptdiskette? 27205
              beq    :sysdisk                         27206
              cmp    #"B"                           ;Systemdiskette? 27207
              beq    :sysdisk                         27208
              jmp    z7586                            ;Disk kopieren. 27209
              27210
; Fehler: Target-Disk ist System-/Hauptdiskette.      27211
::sysdisk     lda    #< d730F                       27213
              ldx    #> d730F                       27214
              jsr    S2090                            ;DBox ohne Recover. 27215
              27216
::cancel      ldx    #CANCEL_ERR                     27217
::err         rts                                     27218
              27219
; Laufwerksliste initialisieren.                      27220
; Rückgabe: Y = Anzahl Ziel-Laufwerke.                27221
:z7350        ldy    #0                              ;Anzahl Laufwerke. 27222
              ldx    #8                              27223
::loop        lda    driveType -8,x                  27224
              sta    v7351 -8,x                      ;driveType speichern. 27225
              beq    :next                            ; => 0, weiter... 27226
              27227
              bit    v6EEA                            ;driveType Source. 27228
              bpl    :1                              ; => Keine RAMDisk... 27229
              27230
              cpx    v6EEC                            ;Laufwerk = Quelle? 27231
              beq    :4                              ; => Ja, ungültig. 27232
              bne    :2                              ; => Lfwk. testen... 27233
              27234
::1           cpx    v6EEC                            ;Laufwerk = Quelle? 27235
              beq    :3                              ; => Ja, weiter... 27236
              27237
::2           and    #ST_DM0DES                      ;driveMode Laufwerk 27238
              cmp    v6EE8                            ; = driveMode Source? 27239
              beq    :3                              ; => Ja, gültig. 27240
              27241
              cmp    #Drv1571                        ;Laufwerk = 1571? 27242
              bne    :4                              ; => Ja, ungültig. 27243
              lda    v6EE8                            ;driveMode Source. 27244
              cmp    #Drv1541                        ;Source = 1541? 27245
              bne    :4                              ; => Nein, ungültig. 27246
              27247
::3           iny                                     ;Laufwerke +1. 27248
              bne    :next                            27249
              27250
::4           lda    #$00                            ;Laufwerk löschen. 27251
              sta    v7351 -8,x                      27252
              27253
::next        inx                                     27254
              cpx    #12                              ;Alle Laufwerke? 27255
              bcc    :loop                            ; => Nein, weiter... 27256
              27257

```

```

27258             rts
27259
27260 :v7351          s 4 ;Laufwerksliste.
27261
27262 ; Nächstes Ziel-Laufwerk suchen.
27263 ; Darf Y nicht verändern!
27264 ; Rückgabe: X = Target-Disk.
27265 :z734E          ldx  v6EEC          ;Nächstes Lfwk. für
27266 ::1             inx                ;Target-Disk suchen.
27267             cpx  #12
27268             bcc  :2
27269             ldx  #8                ;Erstes Laufwerk.
27270 ::2             cpx  v6EEC
27271             beq  :3
27272             lda  v7351 -8,x        ;Lfwk. verfügbar?
27273             beq  :1                ; => Nein, weiter...
27274 ::3             rts
27275
27276 ; Mausabfrage Dialogbox für Verify.
27277 :r7320          lda  v7065          ;Verify möglich?
27278             cmp  #2                ;(Nur Werte 0/1)
27279             bcs  :exit             ; => Nein, Ende...
27280
27281             ldx  #(r4H - r2L)
27282 ::1             lda  opt_verify,x
27283             sta  r2,x
27284             dex
27285             bpl  :1
27286
27287             jsr  IsMseInRegion      ;Mausklick Option?
27288             bne  :2                ; => Ja, weiter...
27289 ::exit          rts
27290
27291 ::2             jsr  S52AD           ;Maustaste loslassen.
27292
27293             lda  v7065              ;Verify-Flag.
27294             eor  #%000000001
27295             sta  v7065
27296
27297 ; Dialogbox: Verify-Status anzeigen.
27298 :r7321          ldx  #(r4H - r2L)
27299 ::1             lda  opt_verify,x
27300             sta  r2,x
27301             dex
27302             bpl  :1
27303
27304             lda  v7065              ;Verify-Flag.
27305             jsr  S140C              ;Rechteck zeichnen.
27306
27307             lda  #%11111111
27308             jmp  FrameRectangle
27309
27310 :opt_verify     b $20 +4*8
27311             b $20 +4*8 +7
27312             w $0040 +17*8

```



```

                                w $0040 +17*8 +7                                27313
                                                                27314
; Neue Diskette einlesen.                                27315
; TD13: InsertNewDisk                                27316
:z73A3        lda    curDrive                ;Aktuelles Laufwerk =    27317
                                cmp    v6EEC                ;Lfwk. Source-Disk?    27318
                                bne    :open                ; => Nein, weiter...    27319
                                                                27320
::retry       lda    curDrive                ;Laufwerk für            27321
                                clc                        ;"Diskette einlegen".    27322
                                adc    #"A" -8              27323
                                sta    t744F                27324
                                sta    t748C                27325
                                                                27326
                                lda    #< d7420              ;Ziel-Disk einlegen.    27327
                                ldx    #> d7420              27328
                                jsr    S2090                ;DBox ohne Recover.    27329
                                                                27330
                                lda    r0L                  27331
                                cmp    #OK                  ;"OK"?                27332
                                beq    :open                ; => Ja, weiter...    27333
                                                                27334
                                ldx    #CANCEL_ERR           ; => Abbruch...    27335
::err         rts                                          27336
                                                                27337
::open        jsr    OpenDisk                    27338
                                txa                        27339
                                bne    :err              27340
                                                                27341
::disk        ldy    #16 -1                      ;Diskname einlesen.    27342
::1           lda    curDirHead +$90,y            27343
                                cmp    #$a0              27344
                                bne    :2                27345
                                lda    #$00              27346
::2           sta    v6ED8,y                      ;Name Target-Disk.    27347
                                dey                      27348
                                bpl    :1                27349
                                                                27350
                                lda    interleave           27351
                                cmp    v6EE9              ;MyInterleave.    27352
                                bcc    :3                27353
                                sta    v6EE9              ;MyInterleave.    27354
                                                                27355
::3           ldx    #$00                      ;Einseitig.            27356
                                lda    v7748              ;Typ Source-Disk.    27357
                                cmp    #Drv1541           27358
                                bne    :4                27359
                                                                27360
                                lda    curType              ;Ziel-Diskette.    27361
                                and    #ST_DM0DES          27362
                                cmp    #Drv1571           ;1571-Laufwerk?    27363
                                bne    :4                ; => Nein, weiter...    27364
                                                                27365
; Hinweis:                                27366
; Bei 1541->1571 muss am Ende die BAM#2            27367

```

```

27368 ; auf der Rückseite gelöscht werden.
27369         lda    curDirHead +3          ;Ziel-Diskette.
27370         bpl    :4
27371         dex                      ;1541 -> 1571:
27372 ::4         stx    v6EEB              ;$FF = BAM#2 löschen.
27373
27374         lda    v7748                  ;Typ Source-Disk.
27375         cmp    #Drv1571              ;1571-Laufwerk?
27376         bne    :5                    ; => Nein, weiter...
27377
27378         lda    curType                ;Ziel-Diskette.
27379         and    #ST_DM0DES
27380         cmp    #Drv1571              ;1571-Laufwerk?
27381         bne    :5                    ; => Nein, Ende...
27382
27383         lda    curDirHead +3          ;Doppelseitig?
27384         bpl    :retry                ; => Nein, weiter...
27385
27386 ::5         ldx    #NO_ERROR
27387         rts
27388
27389 ; "Diskette kopieren" ausführen.
27390 :z7586         ldx    #$01              ;Status auf "Disk"
27391         stx    v5611                  ;umstellen.
27392         dex                      ;Status "Kop.Disk".
27393         stx    v5623                  ; => Kein "Ersetze".
27394
27395 ;         ldx    #STAT_COPY              ;Status "Kopiere".
27396         jsr    S55B4                  ;Status anzeigen.
27397
27398         lda    v7748                  ;Typ Source-Disk.
27399         and    #ST_DM0DES
27400         tay
27401         cmp    #DrvNative
27402         beq    z7617
27403
27404 ; DiskCopy für 1541/71/81.
27405 :z75A4         lda    v7745 -1,y
27406         sta    v7749                  ;Anz.Tracks übrig.
27407
27408         jmp    z6EF0                  ;DiskCopy ausführen.
27409
27410 ; DiskCopy für NativeMode.
27411 :z7617         lda    v6EED              ;Lfwk. Target-Disk.
27412         jsr    z760A                  ;Lfwk./BAM einlesen.
27413
27414         ldy    #0
27415 ::1         lda    dir2Head +4,y
27416         sta    v774B,y
27417         iny
27418         cpy    #5                    ;Disk-ID kopieren.
27419         bcc    :1
27420
27421         lda    v6EEC                  ;Lfwk. Source-Disk.
27422         jsr    z760A                  ;Lfwk./BAM einlesen.

```

		27423
; Hinweis:		27424
; Hier wurde ":GetDirHead" aufgerufen,		27425
; daher ist Sektor \$01/\$02 bereits in		27426
; ":dir2Head" im Speicher.		27427
lda dir2Head +8		27428
sta v774A	;Anz. Tracks Source.	27429
sta v7122	;Anz. Tracks Copy.	27430
clc		27431
adc #\$01		27432
sta v7749	;Anz.Tracks übrig.	27433
		27434
lda v774B +4	;Anz. Tracks Target.	27435
cmp v774A	;Anz. Tracks Source.	27436
bcs :2		27437
		27438
; Fehler, nur DiskCopy NativeMode,		27439
; wenn Disk-Größe unterschiedlich.		27440
jsr S54F0	;Hauptmenü starten.	27441
jmp z7260	;Zieldisk wählen.	27442
		27443
::2	ldy #0	27444
::3	lda curDirHead +\$ab,y	27445
	sta v7750,y	27446
	iny	27447
	cpy #18	;Diskname kopieren.
	bcc :3	27449
		27450
	jsr z7704	;BAM-Block \$01/\$02
	jsr GetBlock	;einlesen.
	txa	27453
	bne :err	27454
		27455
	lda v6EED	;Lfwk. Target-Disk.
	jsr SetDevice	27457
	jsr OpenDisk	27458
	txa	27459
	bne :err	27460
		27461
	ldy #0	;Disk-ID übernehmen.
::4	lda v774B,y	27463
	sta DCOPYSTART +4,y	27464
	iny	27465
	cpy #5	27466
	bcc :4	27467
		27468
	jsr z7704	;BAM-Block \$01/\$02
	jsr PutBlock	;auf Disk schreiben.
	txa	27471
	bne :err	27472
		27473
	jsr z7709	;BAM-Block \$01/\$01
	jsr GetBlock	;einlesen.
	txa	27476
	bne :err	27477

```

27478
27479          ldy    #0                      ;Diskname übernehmen.
27480 ::5          lda    v7750,y
27481          sta    DCOPYSTART +$ab,y
27482          iny
27483          cpy    #18
27484          bcc    :5
27485
27486          jsr    z7709                    ;BAM-Block $01/$01
27487          jsr    PutBlock                ;auf Disk schreiben.
27488          txa
27489          bne    :err
27490
27491          jmp    z6EF0                    ;DiskCopy ausführen.
27492
27493 ::err        ldx    #INV_TRACK
27494          rts
27495
27496 ; NativeMode: Laufwerk öffnen/BAM einlesen.
27497 ; Übergabe: A = Laufwerk 8-11,
27498 :z760A      jsr    SetDevice
27499
27500 ; Hinweis:
27501 ; Sektor $01/$02 mit der Anzahl der
27502 ; Tracks im Image einlesen.
27503          jsr    GetDirHead
27504          txa
27505          beq    :1
27506
27507          ldx    #INV_TRACK
27508 ::1          rts
27509
27510 ; DiskCopy ausführen.
27511 :z6EF0      ldy    #$00                    ;Source-Disk.
27512          lda    #$00                    ;Neue Disk...
27513          jsr    z6FEA                    ;Disk wechseln.
27514          txa
27515          beq    :1
27516          rts
27517
27518 ::1          ldx    v7749                    ;Max. Track +1.
27519          dex                      ;Akt. Track -1.
27520          stx    v04B7                    ;Anzahl mark.Dateien.
27521          jsr    S5802                    ;Anzahl anzeigen.
27522
27523          lda    curType
27524          sta    v7748                    ;Typ Source-Disk.
27525
27526          ldx    #FALSE
27527          and    #ST_DM0DES
27528          cmp    #DrvNative
27529          bne    :2
27530          ldx    #TRUE
27531 ::2          stx    v6EEF
27532

```

```

; Hinweis: 27533
; Ersten Sektor setzen. 27534
    ldx #1 ;1541/71/81. 27535
    ldy #0 27536
    bit v6EEF ;NativeMode? 27537
    bpl :3 ; => Nein, weiter... 27538
; ldx #1 ;NativeMode. 27539
    ldy #3 27540
::3 stx r1L ;Aktueller Sektor 27541
    sty r1H ;Quelldisk. 27542
    stx r11L ;Aktueller Sektor 27544
    sty r11H ;Zieldisk. 27545
; Sektoren kopieren. 27547
::loop lda #< ReadBlock 27548
    sta r0L 27549
    lda #> ReadBlock 27550
    sta r0H 27551
    jsr z6F75 27552
    cpx #INV_TRACK 27553
    beq :11 27554
    txa 27555
    bne :err 27556
::11 lda r1L ;Tr/Se Source-Disk. 27558
    sta r10L 27559
    lda r1H 27560
    sta r10H 27561
    ldy #$01 ;Target-Disk. 27563
    lda #$01 ;Keine neue Disk... 27564
    jsr z6FEA ;Disk wechseln. 27565
    txa 27566
    bne :err 27567
    lda r11L 27569
    sta r1L 27570
    lda r11H 27571
    sta r1H 27572
    lda #< z6FB9 27574
    sta r0L 27575
    lda #> z6FB9 27576
    sta r0H 27577
    jsr z6F75 27578
    cpx #INV_TRACK 27579
    beq :done 27580
    txa 27581
    bne :err 27582
    ldy #$00 ;Source-Disk. 27584
    lda #$01 ;Keine neue Disk... 27585
    jsr z6FEA ;Disk wechseln. 27586
    txa 27587

```

```

27588             bne    :err
27589
27590             lda     r10L
27591             sta     r1L
27592             sta     r11L
27593             lda     r10H
27594             sta     r1H
27595             sta     r11H
27596
27597             jmp     :loop
27598
27599 ; DiskCopy beenden.
27600 ::done          ldx     v6EEB                ;BAM#2 löschen?
27601             beq     :err                ; => Nein, weiter...
27602
27603 ; TODO:
27604 ; Wozu Diskwechsel über Dialogbox?
27605 ; => Target-Disk ist bereits aktiv.
27606 ;---
27607 ;             ldy     #$01                ;Target-Disk.
27608 ;             lda     #$00                ;Neue Disk...
27609 ;             jsr     z6FEA                ;Disk wechseln.
27610 ;---
27611             jsr     OpenDisk                ;Diskette öffnen.
27612 ;---
27613             txa
27614             bne     :err
27615
27616             jsr     z712F                ;BAM DoubleSide-Disk.
27617
27618             ldx     #NO_ERROR
27619 ::err          rts
27620
27621 ; BAM#2 erzeugen.
27622 :z712F          jsr     i_FillRam
27623             w 256
27624             w dir2Head
27625             b $00
27626
27627             lda     #< v7145
27628             sta     r0L
27629             lda     #> v7145
27630             sta     r0H
27631             jsr     InitRam
27632
27633             jmp     PutDirHead
27634
27635 :v7145          w curDirHead +$dd
27636             b 35
27637             b $15,$15,$15,$15,$15,$15,$15,$15
27638             b $15,$15,$15,$15,$15,$15,$15,$15
27639             b $15,$00,$13,$13,$13,$13,$13,$13
27640             b $12,$12,$12,$12,$12,$12,$11,$11
27641             b $11,$11,$11
27642

```

	w dir2Head	27643
	b 105	27644
	b \$ff,\$ff,\$1f, \$ff,\$ff,\$1f	27645
	b \$ff,\$ff,\$1f, \$ff,\$ff,\$1f	27646
	b \$ff,\$ff,\$1f, \$ff,\$ff,\$1f	27647
	b \$ff,\$ff,\$1f, \$ff,\$ff,\$1f	27648
	b \$ff,\$ff,\$1f, \$ff,\$ff,\$1f	27649
	b \$ff,\$ff,\$1f, \$ff,\$ff,\$1f	27650
	b \$ff,\$ff,\$1f, \$ff,\$ff,\$1f	27651
	b \$ff,\$ff,\$1f, \$ff,\$ff,\$1f	27652
	b \$ff,\$ff,\$1f	27653
		27654
::dir_tr_18	b \$00,\$00,\$00	27655
		27656
	b \$ff,\$ff,\$07, \$ff,\$ff,\$07	27657
	b \$ff,\$ff,\$07, \$ff,\$ff,\$07	27658
	b \$ff,\$ff,\$07, \$ff,\$ff,\$07	27659
	b \$ff,\$ff,\$03, \$ff,\$ff,\$03	27660
	b \$ff,\$ff,\$03, \$ff,\$ff,\$03	27661
	b \$ff,\$ff,\$03, \$ff,\$ff,\$03	27662
	b \$ff,\$ff,\$01, \$ff,\$ff,\$01	27663
	b \$ff,\$ff,\$01, \$ff,\$ff,\$01	27664
	b \$ff,\$ff,\$01	27665
		27666
	w NULL	27667
		27668
	; Buffer von Disk einlesen/auf Disk schreiben.	27669
	; Übergabe: r0 auf Disk-Routine ReadBlock/WriteBlock.	27670
:z6F75	jsr z709E ;InitCount.	27671
	txa	27672
	beq :start	27673
::exit	rts	27674
		27675
::start	lda #< DCOPYSTART	27676
	sta r4L	27677
	lda #> DCOPYSTART	27678
	sta r4H	27679
		27680
	jsr EnterTurbo	27681
	txa	27682
	bne :exit	27683
		27684
	jsr InitForIO	27685
		27686
	; 1541/71/81:	27687
	; Nur belegte Sektoren kopieren.	27688
	;	27689
	; Native:	27690
	; Alle Sektoren kopieren.	27691
::loop	bit v6EEF ;NativeMode?	27692
	bmi :1 ; => Ja, schreiben...	27693
		27694
	lda r1L	27695
	sta r6L	27696
	lda r1H	27697

```

27698          sta    r6H
27699
27700          jsr    FindBAMBit          ;Block belegt?
27701          bne    :2                  ; => Nein, weiter...
27702
27703  ::1          lda    r0L
27704          ldx    r0H
27705          jsr    CallRoutine          ;Block read/write.
27706          txa
27707          bne    :err
27708
27709          ldy    r4H
27710          iny
27711          cpy    #> DCOPYEND
27712          beq    :ok
27713          sty    r4H
27714
27715  ::2          jsr    z7066
27716          txa
27717          beq    :loop                ;Nächster Sektor.
27718          bne    :err                ;Fehler, Abruch...
27719
27720  ::ok          ldx    #NO_ERROR
27721  ::err          jmp    DoneWithIO
27722
27723  ; Block schreiben.
27724  ; Bei BAM-Block Name anpassen.
27725  ; TD13: DirTrack
27726  :v6FB8        b $00    ;Track Directory.
27727  :z6FB9        bit    v6EEF          ;NativeMode?
27728          bmi    :write              ; => Ja, schreiben...
27729
27730          lda    v6FB8                ;1541/71=$12, 81=$40.
27731          cmp    r1L
27732          bne    :write
27733          lda    r1H
27734          bne    :write
27735
27736          ldy    #$90                ;Diskname übernehmen.
27737  ::1          lda    v6ED8 -$90,y
27738          beq    :2
27739          sta    (r4),y
27740          iny
27741          cpy    #$a0
27742          bcc    :1
27743          bcs    :3
27744
27745  ::2          lda    #$a0
27746          sta    (r4),y
27747          iny
27748          cpy    #$a0
27749          bcc    :2
27750
27751  ::3          ldx    v6EEB          ;BAM#2 löschen?
27752          beq    :write              ; => Nein, weiter...

```



```

27753
27754         ldy    #$03
27755         lda    #$80                ;1571: Diskmodus:
27756         sta    (r4),y             ;Doppelseitig.
27757
27758 ::write      jsr    WriteBlock      ;Block schreiben.
27759         txa
27760         bne    :exit
27761
27762         ldy    v7065              ;Verify-Flag.
27763         dey
27764         bne    :exit              ;$01 = Verify?
27765                                     ; => Nein, Ende...
27766         jsr    VerWriteBlock      ;1541/71: Verify.
27767         txa
27768         bne    :exit
27769
27770 ::exit       rts
27771
27772 :v7065       b $01    ;$01 = Verify.
27773
27774 ; Blocks zählen.
27775 ; Zählt in Abhängigkeit von MyInterleave 1 Block weiter.
27776 ; Übergabe: -
27777 ; Rückgabe: r1L/r1H = Nächster Tr/Se.
27778 ;           X = Fehler.
27779 ; TD13: CountBlock
27780 :z7066       bit    v6EEF          ;NativeMode?
27781         bmi    :native            ; => Ja, weiter...
27782
27783 ; 1541/71/81:
27784 ;---
27785         lda    v70D7              ;Aktueller Sektor.
27786         clc                        ;1541/71/81.
27787         adc    v6EE9              ;MyInterleave.
27788         sta    v70D7              ;Nächster Sektor.
27789         sta    r1H
27790
27791         cmp    v70D8              ;Sektor per Track.
27792         bcc    :cont              ;1541/71/81.
27793
27794         inc    v70D9
27795         lda    v70D9
27796         cmp    v6EE9              ;MyInterleave.
27797         bcs    :done
27798
27799         sta    r1H
27800         sta    v70D7              ;Nächster Sektor.
27801         bcc    :cont
27802 ;---
27803
27804 ; NativeMode:
27805 ;---
27806 ::native     lda    v70D7          ;Aktueller Sektor.
27807         cmp    v70D8              ;Sektor per Track.

```

```

27808          bcc    :1          ;NativeMode.
27809          beq    :done
27810
27811 ::1          inc    v70D7          ;Nächster Sektor.
27812          lda    v70D7          ;Sektor setzen.
27813          sta    r1H
27814 ;---
27815
27816 ::cont      lda    v70D6          ;Akt. Track/Status.
27817          sta    r1L
27818
27819          ldx    #NO_ERROR
27820          rts
27821
27822 ::done      jsr    DoneWithIO
27823          jsr    z752A          ;Track-Anzeige.
27824          jsr    InitForIO
27825
27826          lda    v70D6          ;Akt. Track/Status.
27827          sta    r1L
27828          lda    #$00
27829          sta    r1H
27830
27831 ; Initialisiert CountBlock.
27832 ; TD13: InitCount
27833 :z709E      lda    r1L
27834          sta    v70D6          ;Akt. Track/Status.
27835
27836          lda    r1H
27837          pha
27838
27839          sta    v70D7          ;Nächster Sektor.
27840          jsr    z70DA          ;Letzten Sektor.
27841          txa          ;Ende erreicht?
27842          bne    :exit          ; => Ja, Ende...
27843
27844          lda    r1H
27845          sta    v70D8          ;Sektor per Track.
27846
27847          bit    v6EEF          ;NativeMode?
27848          bmi    :ok          ; => Ja, weiter...
27849
27850          lda    #$00          ;1541/71/81.
27851          sta    r6H
27852          sta    r7H
27853
27854          lda    v70D7          ;Akt. Sektor.
27855          sta    r6L
27856          lda    v6EE9          ;MyInterleave.
27857          sta    r7L
27858
27859          ldx    #r6
27860          ldy    #r7
27861          jsr    Ddiv
27862

```

	lda	r8L		27863
	sta	v70D9	;Erster Sektor.	27864
				27865
::ok	ldx	#NO_ERROR		27866
				27867
::exit	pla			27868
	sta	r1H		27869
	rts			27870
				27871
:v70D6	b	\$00	;Akt. Track/Status.	27872
:v70D7	b	\$00	;Aktueller Sektor.	27873
:v70D8	b	\$00	;Sektor per Track.	27874
:v70D9	b	\$00	;Erster Sektor.	27875
				27876
			; Auf letzten Track testen, letzten Sektor ermitteln.	27877
			; Rückgabe: r1H = Letzter Block/Track.	27878
			; X = \$00, OK.	27879
			; \$02, Ende erreicht.	27880
			; TD13: GetSectors	27881
:z70DA	lda	v7748	;Typ Source-Disk.	27882
	and	#ST_DMODES		27883
	beq	:err		27884
	cmp	#DrvPCDOS		27885
	bcs	:err		27886
				27887
	tay			27888
				27889
	lda	r1L	;Track = \$00?	27890
	beq	:err	; => Invalid Track.	27891
				27892
	dey		;1541?	27893
	bne	:1571	; => Nein, weiter...	27894
				27895
::1541	cmp	#TRACK41 +1		27896
	bcc	:getsek		27897
	bcs	:err		27898
				27899
::1571	dey		;1571?	27900
	bne	:1581	; => Nein, weiter...	27901
				27902
	cmp	#70 +1		27903
	bcs	:err	; => Invalid Track.	27904
				27905
::getsek	ldy	#\$07		27906
::1	cmp	v7123,y		27907
	bcs	:2		27908
	dey			27909
	bpl	:1		27910
	bmi	:err	; => Invalid Track.	27911
				27912
::2	tya			27913
	and	#\$03		27914
	tay			27915
	lda	v712B,y		27916
	bne	:setsek		27917

```

27918
27919 ::1581          dey          ;1581?
27920              bne      :native      ; => Nein, weiter...
27921
27922              cmp      #TRACK81 +1
27923              bcs      :err          ; => Invalid Track.
27924
27925              lda      #40
27926              bne      :setsek
27927
27928 ; NativeMode:
27929 ::native          cmp      v7122          ;Max. Track?
27930              beq      :3              ; => Ja, weiter...
27931              bcs      :err          ; => Invalid Track.
27932
27933 ::3              lda      #$ff          ;Letzter Sektor.
27934
27935 ::setsek          sta      r1H
27936              ldx      #NO_ERROR
27937              rts
27938
27939 ; Fehler: Invalid Track.
27940 ::err            ldx      #INV_TRACK
27941              rts
27942
27943 :v7122            b $00      ;Max. Track NativeMode.
27944 :v7123            b $01,$12,$19,$1f,$24,$35,$3c,$42
27945 :v712B            b $15,$13,$12,$11
27946
27947 ; Wechselt zwischen Source- und Target-Disk.
27948 ; Übergabe: Y=0 = Source-Disk einlegen.
27949 ;           1 = Target-Disk einlegen.
27950 ;           A=0 = Vollständiger Diskwechsel.
27951 ;           1 = Keine neue Disk.
27952 ; TD13: ChangeDisk
27953 :z6FEA            pha
27954              lda      v6EEC          ;Lfwk. Source-Disk =
27955              cmp      v6EED          ;Lfwk. Target-Disk?
27956              bne      :1
27957              tya
27958              pha
27959              beq      :source
27960
27961 ::target          lda      #< d7420          ;Ziel-Disk einlegen.
27962              ldx      #> d7420
27963              bne      :askuser
27964
27965 ::source          lda      #< d745C          ;Start-Disk einlegen.
27966              ldx      #> d745C
27967
27968 ::askuser          sta      r0L
27969              stx      r0H
27970
27971              ldx      #(r11H - r10L)
27972 ::sv              lda      r10,x

```

	pha		27973
	dex		27974
	bpl	:sv	27975
			27976
	lda	r0L	27977
	ldx	r0H	27978
	jsr	S2080 ;DBox, k. Maustaste.	27979
			27980
	ldx	#0	27981
::ld	pla		27982
	sta	r10,x	27983
	inx		27984
	cpx	#(r11H - r10L) +1	27985
	bcc	:ld	27986
			27987
	pla		27988
	tay		27989
::1	pla		27990
	tax		27991
	bne	:swap ;Keine neue Disk.	27992
			27993
; Neue Diskette öffnen.			27994
	lda	v6EEC,y ;Lfwk. Source/Target.	27995
	jsr	SetDevice	27996
			27997
	lda	curType	27998
	cmp	#Drv1571 ;Echte 1571?	27999
	bne	:open ; => Nein, weiter...	28000
			28001
	lda	#\$08	28002
	sta	interleave	28003
			28004
::open	jmp	OpenDisk	28005
			28006
; Diskette wechseln.			28007
; Bei 1581: BAM-Block #3 retten.			28008
::swap	lda	v7748 ;Typ Source-Disk.	28009
	and	#ST_DM0DES	28010
	cmp	#Drv1581	28011
	bne	:11	28012
			28013
	tya	;Y=Quelle/Ziel.	28014
	pha		28015
			28016
	jsr	i_MoveData	28017
	w	dir3Head	28018
	w	fileHeader	28019
	w	256	28020
			28021
	pla		28022
	tay	;Y=Quelle/Ziel.	28023
			28024
::11	lda	v6EEC,y ;Lfwk. Source/Target.	28025
	jsr	SetDevice	28026
			28027

```

28028          lda    curType
28029          cmp    #Drv1571          ;Echte 1571?
28030          bne    :12              ; => Nein, weiter...
28031
28032          lda    #$08
28033          sta    interleave
28034
28035 ::12          txa                  ;Fehler?
28036          bne    :exit            ; => Ja, Ende...
28037
28038 ; Bei 1581: BAM-Block #3 zurücksetzen.
28039          lda    v7748            ;Typ Source-Disk.
28040          and    #ST_DM0DES
28041          cmp    #Drv1581
28042          bne    :exit
28043
28044 ; Hinweis:
28045 ; MoveData ändert nicht das X-Register!
28046          jsr    i_MoveData
28047          w fileHeader
28048          w dir3Head
28049          w 256
28050
28051 ;          ldx    #NO_ERROR
28052 ::exit      rts
28053
28054 ; Dialogbox: Laufwerk wählen.
28055 :OFF_DBTXT_Y    = $20 +$2b
28056 :OFF_DBTXT_X    = $0040 +$0a
28057 :d72B3          b %10000001
28058
28059          b DBTXTSTR , $08, $0e    ;Titel.
28060          w t72DA
28061
28062          b DBTXTSTR , $08, $1c    ;Startdiskette.
28063          w t72F3
28064          b DBTXTSTR , $08, $27    ;Laufwerk.
28065          w t7360
28066          b DBTXTSTR , $1a, $27    ;Diskname.
28067          w V0483 +2
28068
28069          b DBTXTSTR , $08, $34    ;Zieldiskette.
28070          w t7301
28071          b DBTXTSTR , $08, $3f    ;Laufwerk.
28072          w t7362
28073          b DBTXTSTR , $1a, $3f    ;Diskname.
28074          w v6ED8
28075
28076          b DBOPVEC                ;Mausabfrage.
28077          w r7320
28078          b DB_USR_ROUT            ;Verify-Flag.
28079          w r7321
28080          b DBTXTSTR , $93, $26    ;"Verify".
28081          w t72DB
28082

```

```

                b OK          , $11, $06                28083
                b CANCEL      , $11, $48                28084
                b DRIVE       , $07, $48                28085
                b DBTXTSTR    , $08, $54                28086
                w t7302                                28087
                b DBTXTSTR    , $08, $54                28088
                w t7302                                28089
                ; Bei RAMDisk "Disk"-Icon abschalten.    28090
;v72D6          b DISK       , $11, $30                28091
                b NULL                                28092
                b NULL                                28093
                b NULL                                28094
if LANG = LANG_DE                                28095
:t72DA          b BOLDON                                28096
                b "Diskette kopieren", 0              28097
                b "Verify", 0                          28098
:t72DB          b "Verify", 0                          28099
                b BOLDON                                28100
:t72F3          b BOLDON                                28101
                b "Startdiskette:", 0                 28102
:t7301          b BOLDON                                28103
                b "Zieldiskette wählen:", 0           28104
                b PLAINTEXT                             28105
:t7302          b "Laufwerke:"                         28106
                b NULL                                  28107
;v72D9          b NULL                                  28108
endif                                                  28109
if LANG = LANG_EN                                28110
:t72DA          b BOLDON                                28111
                b "Copy disk", 0                      28112
                b "Verify", 0                          28113
:t72DB          b "Verify", 0                          28114
                b BOLDON                                28115
:t72F3          b BOLDON                                28116
                b "Source disk:", 0                   28117
:t7301          b BOLDON                                28118
                b "Select target disk:", 0             28119
                b PLAINTEXT                             28120
:t7302          b "Drives:"                            28121
                b NULL                                  28122
;v72D9          b NULL                                  28123
endif                                                  28124
:t7360          b "A:" ;IDD_SDr.                      28125
                b PLAINTEXT, 0                        28126
:t7362          b "B:" ;IDD_DDr.                      28127
                b PLAINTEXT, 0                        28128
                b PLAINTEXT, 0                        28129
                b PLAINTEXT, 0                        28130
;v6ED8          s 17 ;DestinationName                  28131
                b DBTXTSTR , $08, $1c                 28132
; Fehler: System- oder Hauptdiskette.                28133
:d730F          b %10000001                            28134
                b DBTXTSTR , $08, $1c                 28135
                w :t1                                28136
                w :t1                                28137

```

```

28138          b DBTXTSTR,$08,$27
28139          w :t2
28140
28141          b OK          , $11,$48
28142          b NULL
28143
28144 if LANG = LANG_DE
28145 ::t1          b "Die Zieldiskette darf keine",0
28146 ::t2          b "System- oder Hauptdiskette sein!",0
28147 endif
28148 if LANG = LANG_EN
28149 ::t1          b "The target disk must not be a",0
28150 ::t2          b "system or boot disk!",0
28151 endif
28152
28153 ; Dialogbox: Zieldiskette einlegen.
28154 :d7420          b %10000001
28155
28156          b DBTXTSTR,$08,$0e          ;Titel.
28157          w t72DA
28158
28159          b DBTXTSTR,$08,$22          ;"Target disk".
28160          w t7432
28161
28162          b DBTXTSTR,$08,$2e          ;"?".
28163          w T29D3
28164          b DBTXTSTR,$12,$2e          ;Diskname.
28165          w v6ED8
28166
28167          b OK          , $01,$48
28168          b CANCEL     , $11,$48
28169
28170          b NULL
28171
28172 if LANG = LANG_DE
28173 :t7432          b PLAINTEXT
28174          b "Zieldiskette in Laufwerk "
28175          b BOLDON
28176 :t744F          b "A:"
28177          b PLAINTEXT
28178          b " einlegen:"
28179          b BOLDON,0
28180 endif
28181 if LANG = LANG_EN
28182 :t7432          b PLAINTEXT
28183          b "Insert target disk in drive "
28184          b BOLDON
28185 :t744F          b "A:",0
28186 endif
28187
28188 ; Dialogbox: Startdiskette einlegen.
28189 :d745C          b %10000001
28190
28191          b DBTXTSTR,$08,$0e          ;Titel.
28192          w t72DA

```



		28193
	b DBTXTSTR,\$08,\$22 ;"Source disk".	28194
	w t746E	28195
		28196
	b DBTXTSTR,\$08,\$2e ;"?".	28197
	w T29D3	28198
	b DBTXTSTR,\$12,\$2e ;Diskname.	28199
	w V0483 +2	28200
		28201
	b OK , \$01,\$48	28202
	b CANCEL , \$11,\$48	28203
		28204
	b NULL	28205
		28206
if LANG = LANG_DE		28207
:t746E	b PLAINTEXT	28208
	b "Startdiskette in Laufwerk "	28209
	b BOLDON	28210
:t748C	b "A:"	28211
	b PLAINTEXT	28212
	b " einlegen:"	28213
	b BOLDON,0	28214
endif		28215
if LANG = LANG_EN		28216
:t746E	b PLAINTEXT	28217
	b "Insert source disk in drive "	28218
	b BOLDON	28219
:t748C	b "A:",0	28220
endif		28221
		28222
; Track-Anzeige.		28223
:z752A	lda r0H	28224
	pha	28225
	lda r0L	28226
	pha	28227
		28228
	lda r4H	28229
	pha	28230
	lda r4L	28231
	pha	28232
		28233
	lda r10H	28234
	pha	28235
	lda r10L	28236
	pha	28237
		28238
	lda r11H	28239
	pha	28240
	lda r11L	28241
	pha	28242
		28243
	lda v7749 ;Anz.Tracks übrig.	28244
	sec	28245
	sbc v70D6 ;Akt. Track/Status.	28246
	sta V04B7 ;Anzahl mark.Dateien.	28247

```

28248          jsr    S5802          ;Anzahl anzeigen.
28249
28250          inc     v70D6          ;Akt. Track/Status.
28251
28252          pla
28253          sta     r11L
28254          pla
28255          sta     r11H
28256
28257          pla
28258          sta     r10L
28259          pla
28260          sta     r10H
28261
28262          pla
28263          sta     r4L
28264          pla
28265          sta     r4H
28266
28267          pla
28268          sta     r0L
28269          pla
28270          sta     r0H
28271
28272          rts
28273
28274 ; Zeiger NativeMode = Erster BAM-Block $01/$02.
28275 :z7704      ldy     #2
28276          b $2c
28277
28278 ; Zeiger NativeMode = ROOT-Header $01/$01.
28279 :z7709      ldy     #1
28280          sty     r1H
28281
28282          ldy     #1
28283          sty     r1L
28284
28285          ldy     #< DCOPYSTART
28286          sty     r4L
28287          ldy     #> DCOPYSTART
28288          sty     r4H
28289
28290          rts
28291
28292 ;*** Endadresse testen:
28293          g DCOPYSTART
28294
28295 ; 0: DiskImage erstellen.
28296 :N7768      jsr     KB4E72          ;Shortcut "<-" =
28297          ;Ordner verlassen.
28298
28299          jsr     S5245          ;Dateien invertieren.
28300
28301          lda     curType
28302          and     #ST_DM0DES

```

```

        cmp    #DrvNative          ;NativeMode?          28303
        bne    :skip                ; => Nein, weiter...    28304
                                           28305
        jsr    z77A0                ;Größe DiskImage.      28306
        txa                                ;Abbruch?        28307
        bne    :cancel              ; => Ja, Ende...        28308
                                           28309
::skip      jsr    z77E7                ;Name/Img erstellen.  28310
                                           28311
        lda    #$00                ;Status "Datei/Disk"    28312
        sta    V5611                ;zurücksetzen.        28313
                                           28314
        txa                                28315
        beq    x77DB                28316
        cpx    #30                  ;SYNTAX ERROR?        28317
        beq    x77DB                ; => Ja, Ende...        28318
        cpx    #CANCEL_ERR          ;Abbruch?            28319
        bne    :err                 ; => Nein, Fehler...     28320
::cancel    rts                                28321
                                           28322
::err       jmp    S5CF3                ;Fehler ausgeben.   28323
                                           28324
; Reset Statuszeile, Ende.                28325
:x77DB      jsr    S5255                ;Mark.Dat=0/Anzeige.  28326
                                           28327
; Menü starten um Status zu löschen.        28328
;                               jsr    S54F0                ;Hauptmenü starten.  28329
                                           28330
; Hinweis:                                28331
; Beim kopieren/löschen von Dateien        28332
; wird der Status erst nach dem Update      28333
; des Fensters gelöscht.                  28334
; Die Routine kann aber nicht nach dem      28335
; Update über JSR aufgerufen werden, da     28336
; im Textmodus der Modulspeicher zum        28337
; Teil überschrieben wird.                28338
        lda    #> S54F0 -1          ;Hauptmenü starten.   28339
        pha                                28340
        lda    #< S54F0 -1          28341
        pha                                28342
                                           28343
; Hinweis:                                28344
; Im Textmodus wird der Bereich für         28345
; VLIR-Module überschrieben!              28346
        jmp    S4321                ;Alle Fenster neu.     28347
                                           28348
; NativeMode: Größe DiskImage eingeben.    28349
:z77A0      lda    #< v77A1          ;Größe DiskImage.     28350
        sta    a1L                  28351
        lda    #> v77A1            28352
        sta    a1H                  28353
                                           28354
        lda    #"0"                ;0=255=16Mb.           28355
        sta    v77A1 +0            28356
        lda    #NULL                28357

```

```

28358          sta    v77A1 +1
28359
28360          lda    #< d7B70          ;Größe eingeben.
28361          ldx    #> d7B70
28362          jsr    S2090              ;DBox ohne Recover.
28363
28364          lda    sysDBData
28365          cmp    #CANCEL
28366          beq    :cancel
28367
28368          lda    v77A1              ;Eingabe gültig?
28369          beq    z77A0              ; => Nein, Neustart.
28370
28371          jsr    z77A1              ;Größe berechnen.
28372          cmp    #2                ;Zahl gültig?
28373          bcs    :1                ; => Ja, weiter...
28374
28375 ; Hinweis:
28376 ; Sektor $01/$02 mit der Anzahl der
28377 ; Tracks im Image einlesen.
28378          jsr    GetDirHead        ;BAM einlesen.
28379          txa
28380          bne    :2                ; Fehler?
28381                                     ; => Ja, Vorgabewert.
28382          lda    dir2Head +8        ;Aktuelle Größe.
28383          cmp    #2                ;Anzahl gültig?
28384          bcs    :1                ; => Ja, weiter...
28385
28386 ::2          lda    #$40            ;Vorgabe DNP = 4Mb.
28387 ::1          sta    v7B25          ;Anzahl Tracks.
28388          jsr    z77A2              ;Größe nach ASCII.
28389
28390          ldx    #NO_ERROR
28391          rts
28392
28393 ::cancel     ldx    #CANCEL_ERR
28394          rts
28395
28396 ; Größe berechnen.
28397 :z77A1       ldx    #0
28398          stx    r0L
28399          stx    r0H
28400
28401          lda    #10
28402          sta    r1L
28403          stx    r1H
28404
28405 ;           ldx    #0
28406 ::1          lda    v77A1,x
28407          beq    :done
28408
28409          pha
28410          txa
28411          pha
28412

```

	ldx	#r0		28413
	ldy	#r1		28414
	jsr	DMult		28415
				28416
	pla			28417
	tax			28418
	pla			28419
	sec			28420
	sbc	#"0"		28421
	bcs	:2		28422
::null	lda	#"0"		28423
::2	cmp	#10		28424
	bcs	:null		28425
				28426
	clc			28427
	adc	r0L		28428
	sta	r0L		28429
	bcc	:3		28430
	inc	r0H		28431
				28432
::3	inx			28433
	bne	:1		28434
				28435
::done	lda	r0H	;Größer 255?	28436
	bne	:err	; => Ja, 255 Tracks.	28437
				28438
	lda	r0L	;Wert = 0?	28439
	beq	:ok	; => Ja, übernehmen.	28440
	cmp	#2	;Größer 2?	28441
	bcs	:ok	; => Ja, weiter...	28442
	lda	#2	;Mind. 2 Tracks.	28443
	b	\$2c		28444
::err	lda	#255	;Max. 255 Tracks.	28445
::ok	rts			28446
				28447
				28448
; Zahl nach ASCII wandeln				28448
:z77A2	ldx	#"0"		28449
	ldy	#"0"		28450
::1	cmp	#100		28451
	bcc	:2		28452
	iny			28453
;				28454
	sec			28454
	sbc	#100		28455
	bne	:1		28456
::2	cmp	#10		28457
	bcc	:3		28458
	inx			28459
;				28460
	sec			28460
	sbc	#10		28461
	bne	:2		28462
::3	clc			28463
	adc	#"0"		28464
				28465
	cpy	#"0"	;Führende "=" durch	28466
	bne	:4	;Space ersetzen.	28467

```

28468      ldy  #" "
28469      cpx  #"0"
28470      bne  :4
28471      ldx  #" "
28472  ::4      sty  t7BAA +0
28473      stx  t7BAA +1
28474      sta  t7BAA +2
28475      rts
28476
28477 ; Name eingeben.
28478 :z77E7    jsr  z77DD          ;Name definieren.
28479
28480      lda  #< v7B0F          ;Name DiskImage.
28481      sta  a1L
28482      lda  #> v7B0F
28483      sta  a1H
28484
28485      ldx  #< t7BA9          ;Native:
28486      ldy  #> t7BA9          ;Größe anzeigen.
28487
28488      lda  curType
28489      and  #ST_DM0DES
28490      cmp  #DrvNative        ;NativeMode?
28491      beq  :settx            ; => Nein, weiter...
28492
28493      ldx  #< t7BBF          ;1541/71/81:
28494      ldy  #> t7BBF          ;Nichts anzeigen.
28495  ::settx   stx  a2L
28496      sty  a2H
28497
28498      lda  #< d7B79          ;Name eingeben.
28499      ldx  #> d7B79
28500      jsr  S2090            ;DBox ohne Recover.
28501
28502      lda  sysDBData
28503      cmp  #CANCEL
28504      beq  :cancel
28505
28506      lda  v7B0F            ;Eingabe testen.
28507      beq  z77E7            ; => Neu eingeben.
28508
28509      jsr  z77DE            ;Name kopieren.
28510
28511      ldx  C042A            ;Oberstes Fenster.
28512      lda  #$00
28513      sta  C050B,x          ;Fensterposition.
28514
28515      lda  #$01            ;Status "Disk".
28516      sta  V5611
28517
28518      ldx  #STAT_CREATE     ;Status "Erstelle".
28519      jsr  S55B4            ;Status anzeigen.
28520
28521      lda  #< v7B73
28522      ldx  #> v7B73

```

	jsr	z7A8B		28523
				28524
	ldy	v7AF1		28525
	beq	:unknown	;Type = \$00.	28526
	dey			28527
	beq	:41	;Type = \$x1.	28528
	dey			28529
	beq	:71	;Type = \$x2.	28530
	dey			28531
	beq	:81	;Type = \$x3.	28532
	dey			28533
	beq	:native	;Type = \$x4.	28534
				28535
::unknown	ldx	#DEV_NOT_FOUND		28536
	rts			28537
				28538
::cancel	ldx	#CANCEL_ERR		28539
	rts			28540
				28541
::41	lda	#TRACK41	;1541.	28542
	bne	:size		28543
				28544
::71	lda	#TRACK71	;1571.	28545
	bne	:size		28546
				28547
::81	lda	#TRACK81	;1581.	28548
	bne	:size		28549
				28550
::native	lda	v7B25	;Native-Mode.	28551
				28552
::size	sta	v7AEF	;Dxx: Anzahl Tracks.	28553
	sta	v7749	;Anz.Tracks übrig.	28554
				28555
; Leeres DiskImage erstellen.				28556
	jsr	z7973	;OPEN ",P,W".	28557
				28558
	jsr	z7929	;DiskImage öffnen.	28559
	txa			28560
	bne	:err	; => Fehler, Ende...	28561
				28562
	jsr	z7968	;Datenk. schließen.	28563
				28564
; DiskImage auf Größe erweitern.				28565
	jsr	z7976	;OPEN ",P,A".	28566
				28567
	jsr	z79AB	;Buffer-Pointer.	28568
	txa		;Erstellt DiskImage.	28569
	bne	:err		28570
				28571
; DiskImage formatieren.				28572
	ldx	#0	;Name DiskImage	28573
::11	lda	v7AF2,x	;übernehmen.	28574
	beq	:12		28575
	sta	v7B44,x		28576
	inx			28577

```

28578      cpx    #16
28579      bcc    :11
28580
28581  ::12      txa                                ;DiskImage öffnen.
28582      clc
28583      adc    #< 3
28584      sta    v7B3F +0
28585      lda    #> 3
28586      sta    v7B3F +1
28587
28588      lda    #< v7B3F
28589      ldx    #> v7B3F
28590      jsr    z7A8B                                ;CD-Befehl senden.
28591
28592      ldx    #0                                    ;Name DiskImage in
28593  ::13      lda    v7AF2,x                        ;FORMAT übernehmen.
28594      beq    :14
28595      sta    v7B59,x
28596      inx
28597      cpx    #16
28598      bcc    :13
28599
28600  ::14      lda    #", "                        ;Disk-ID.
28601      sta    v7B59,x
28602      inx
28603      lda    #"0"
28604      sta    v7B59,x
28605      inx
28606      lda    #"1"
28607      sta    v7B59,x
28608      inx
28609
28610      txa                                ;Länge des
28611      clc                                ;Format-Befehls.
28612      adc    #< 2
28613      sta    v7B55 +0
28614      lda    #> 2
28615      sta    v7B55 +1
28616
28617      lda    #< v7B55
28618      ldx    #> v7B55
28619      jsr    z7A8B                                ;Format-Befehl.
28620
28621 ; Hinweis:
28622 ; NewDisk erforderlich um den Format-
28623 ; Befehl abzuwarten.
28624      jsr    NewDisk                                ;Disk öffnen.
28625
28626 ; Hinweis:
28627 ; Wird die Disk nicht neu initialisiert
28628 ; dann erhält GEOS beim lesen der BAM
28629 ; vom SD2IEC evtl. noch Reste des
28630 ; vorherigen DiskImage mit der falschen
28631 ; Anzahl an Tracks im DiskImage.
28632      lda    #< v7B6D

```



```

        ldx    #> v7B6D                                28633
        jsr    z7A8B                                ;Initialize Disk. 28634
                                                28635
        jsr    OpenDisk                                28636
;        txa                                          28637
;        bne    :err                                28638
::err    rts                                          28639
                                                28640
; Name für DiskImage initialisieren. 28641
:z77DD    ldx    #0                                ;Name DiskImage 28642
::1      lda    v7B0B,x                            ;initialisieren. 28643
        sta    v7B0F,x                                28644
        inx                                          28645
        cpx    #3                                28646
        bcc    :1                                28647
                                                28648
        ldx    V6168 +OFF_DAY +0                    28649
        lda    V6168 +OFF_DAY +2                    28650
        stx    v7B0F +3                            28651
        sta    v7B0F +4                            28652
                                                28653
        ldx    V6168 +OFF_MONTH +0                  28654
        lda    V6168 +OFF_MONTH +2                  28655
        stx    v7B0F +5                            28656
        sta    v7B0F +6                            28657
                                                28658
        lda    #" - "                                28659
        sta    v7B0F +7                            28660
                                                28661
        ldx    V6168 +OFF_HOUR +0                   28662
        lda    V6168 +OFF_HOUR +2                   28663
        stx    v7B0F +8                            28664
        sta    v7B0F +9                            28665
                                                28666
        ldx    V6168 +OFF_MINUTES +0                28667
        lda    V6168 +OFF_MINUTES +2                28668
        stx    v7B0F +10                           28669
        sta    v7B0F +11                           28670
                                                28671
        jsr    z77DC                                ;Typ DiskImage. 28672
                                                28673
        asl                                          28674
        tay                                          28675
        dey                                          28676
        dey                                          28677
                                                28678
        lda    v7B03 +0,y                            28679
        sta    v7B02 +2                            28680
        lda    v7B03 +1,y                            28681
        sta    v7B02 +3                            28682
                                                28683
        rts                                          28684
                                                28685
; Name für DiskImage kopieren. 28686
:z77DE    ldx    #0                                28687

```

```

28688 ::1      lda    v7B0F,x
28689          beq     :2
28690          jsr     z77DF          ;Name speichern.
28691          cpx     #12
28692          bcc     :1
28693
28694 ::2      lda     #". "          ;Erweiterung DiskImg.
28695          jsr     z77DF          ;Zeichen speichern.
28696          lda     #"D"
28697          jsr     z77DF          ;Zeichen speichern.
28698
28699          jsr     z77DC          ;Typ DiskImage.
28700          sta     v7AF1        ;Typ 1-4 für Dxx.
28701
28702          asl
28703          tay
28704          dey
28705          dey
28706
28707          lda     v7B03 +0,y
28708          jsr     z77DF          ;Zeichen speichern.
28709          lda     v7B03 +1,y
28710          jsr     z77DF          ;Zeichen speichern.
28711
28712          lda     #NULL          ;Ende markieren.
28713 ;          sta     v7AF2,x
28714 ;          sta     V0483 +2,x
28715 ;
28716 ;          rts
28717
28718 ; Zeichen in Diskname und Name DiskImage kopieren.
28719 :z77DF      sta     v7AF2,x
28720          sta     V0483 +2,x
28721          inx
28722          rts
28723
28724 ; DiskImage-Typ ermitteln.
28725 :z77DC      lda     curType
28726          and     #ST_DM0DES
28727          beq     :1
28728          cmp     #DrvNative +1
28729          bcc     :2
28730 ::1      lda     #Drv1581
28731 ::2      rts
28732
28733 ; Neues DiskImage erstellen.
28734 :z7929      lda     curDrive
28735          jsr     SetDevice
28736
28737          jsr     PurgeTurbo
28738          jsr     InitForIO
28739
28740          lda     #$00
28741          sta     STATUS
28742

```

	lda	v7B26	28743
	ldx	#< v7B27	28744
	ldy	#> v7B27	28745
	jsr	SETNAM ;Datei öffnen.	28746
			28747
	lda	#IEC_DATA_LFN	28748
	ldx	curDrive	28749
	ldy	#IEC_DATA_SEK	28750
	jsr	SETLFS	28751
			28752
	jsr	OPENCHN	28753
			28754
	lda	#IEC_DATA_LFN	28755
	jsr	CLOSE	28756
			28757
	bit	STATUS ;Fehler?	28758
	bpl	:ok ; => Nein, weiter...	28759
			28760
	jsr	z7968 ;Datenk. schließen.	28761
			28762
	ldx	#DEV_NOT_FOUND	28763
	rts		28764
			28765
	; Ausgabekanal setzen.		28766
::ok	jsr	OPENCHN	28767
			28768
	ldx	#IEC_DATA_LFN	28769
	jsr	CKOUT	28770
			28771
	ldx	#NO_ERROR	28772
	rts		28773
			28774
	; Datenkanal schließen.		28775
:z7968	lda	#IEC_DATA_LFN	28776
	jsr	CLOSE	28777
	jsr	CLRCHN	28778
			28779
	jmp	DoneWithIO	28780
			28781
	; OPEN-Befehl für SD2IEC setzen.		28782
:z7973	lda	#"W" ;Modus: Write.	28783
	b	\$2c	28784
:z7976	lda	#"A" ;Modus: Append.	28785
	pha		28786
			28787
	ldx	#0	28788
::1	lda	v7AF2,x	28789
	beq	:2	28790
	sta	v7B2A,x	28791
	inx		28792
	cpx	#16	28793
	bcc	:1	28794
			28795
::2	lda	#" , "	28796
	sta	v7B2A,x	28797

```

28798      inx
28799      lda    #"P"
28800      sta    v7B2A,x
28801      inx
28802      lda    #", "
28803      sta    v7B2A,x
28804      inx
28805      pla
28806      sta    v7B2A,x
28807      inx
28808      lda    #NULL
28809      sta    v7B2A,x
28810      inx
28811      inx
28812      inx
28813      stx    v7B26
28814      rts
28815
28816 ; Buffer-Pointer senden.
28817 :z79AB      ldx    #$00
28818      stx    v7A28 +0
28819      stx    v7A28 +1
28820      stx    v7A28 +2
28821      stx    v7A28 +3
28822      inx
28823      stx    v7AF0          ;Erster Track.
28824      stx    v70D6          ;Akt. Track/Status.
28825
28826 ::loop      jsr    z7A2D          ;Anz. Sek./Track.
28827      txa
28828      bne    :exit
28829
28830      jsr    z752A          ;Track-Anzeige.
28831
28832      jsr    z7929          ;DiskImage öffnen.
28833      txa
28834      bne    :exit
28835
28836 ::1          inc    v7A28 +1
28837      bne    :2
28838      inc    v7A28 +2
28839      beq    :err          ;>16Mb, Fehler...
28840
28841 ::2          dec    v7B78
28842      bne    :1
28843
28844      lda    #IEC_DEV_CHAN
28845      ldx    curDrive
28846      tay
28847      jsr    SETLFS
28848
28849      lda    #$00
28850      jsr    SETNAM
28851      jsr    OPENCHN          ;Befehlskanal.
28852

```

	jsr	UNTALK	28853
			28854
	lda	curDrive	28855
	jsr	LISTEN	28856
	lda	#\$ff	28857
	jsr	SECOND	28858
			28859
	ldy	#0	28860
		;"B-P"-Befehl	
::3	tya	;an Laufwerk senden.	28861
	pha		28862
	lda	v7A26,y	28863
	jsr	BSOUT	28864
	pla		28865
	iny		28866
	cpy	#v7A26len	28867
	bcc	:3	28868
			28869
	jsr	UNLSN	28870
		;Befehl ausführen.	
			28871
	lda	#IEC_DEV_CHAN	28872
	jsr	CLOSE	28873
			28874
	jsr	z7968	28875
		;Datenk. schließen.	
			28876
	lda	v7AF0	28877
	inc	v7AF0	28878
	cmp	v7AEF	28879
	bcc	:loop	28880
		; => Nein, weiter...	
			28881
	ldx	#NO_ERROR	28882
	rts		28883
			28884
			28885
; DiskImage >16Mb, Fehler...			
::err	jsr	z7968	28886
		;Datenk. schließen.	
			28887
	ldx	#INV_TRACK	28888
::exit	rts		28889
			28890
; P-Befehl für SD2IEC um Pointer auf			28891
; Byte-Position zu setzen.			28892
:v7A26	b	"P"	28893
	b	IEC_DATA_LFN	28894
:v7A28	s	4	28895
		;Pointer für Position in DiskImage.	
	b	CR	28896
:v7A2X			28897
:v7A26len	=	(v7A2X - v7A26)	28898
			28899
; Anzahl Sektoren im Track ermitteln.			28900
; Übergabe: v7AF1 = Laufwerkstyp 1-4.			28901
; Rückgabe: X = Fehler.			28902
; v7B78 = Anz. Sektoren/Track.			28903
:z7A2D	ldx	v7AF1	28904
	beq	:unknown	28905
		;Type = \$00.	
	dex		28906
	beq	:2	28907
		;Type = \$x1.	

```

28908          dex
28909          beq    :2                ;Type = $x2.
28910          dex
28911          bne    :1                ;Type = $x3.
28912
28913 ; 1581:
28914          lda    #40                ;D81: 40 Sektoren.
28915          sta    v7B78
28916
28917 ;          ldx    #NO_ERROR
28918          rts
28919
28920 ::1          dex
28921          bne    :unknown          ;Type = $x4.
28922
28923 ; Native:
28924 ;          ldx    #$00
28925          stx    v7B78                ;DNP: 256 Sektoren.
28926
28927 ;          ldx    #NO_ERROR
28928          rts
28929
28930 ; 1541/71:
28931 ::2          ldy    v7AF1
28932          dey
28933          bne    :3                ;1541?
28934                                     ; => Nein, weiter...
28935          lda    v7AF0                ; D64
28936          cmp    #TRACK41 +1
28937          bcc    :4
28938
28939 ::unknown    ldx    #INV_TRACK
28940          rts
28941
28942 ::3          dey
28943          bne    :unknown          ;1571?
28944                                     ; => Nein, Abbruch...
28945          lda    v7AF0                ; D71
28946          cmp    #TRACK71 +1
28947          bcs    :unknown
28948
28949 ::4          ldy    #$07                ;Max. Anzahl
28950 ::5          cmp    v7123,y            ;Sektoren/Track
28951          bcs    :6                ;ermitteln.
28952          dey
28953          bpl    :5
28954          bmi    :unknown
28955
28956 ::6          tya
28957          and    #%00000011          ;4 Sektor-Zonen.
28958          tay
28959          lda    v712B,y
28960          sta    v7B78                ;Anzahl Sektoren.
28961
28962 ;          ldx    #NO_ERROR

```

	rts		28963
			28964
	; I/O: Floppy-Befehl senden.		28965
:z7A8B	sta	r0L	28966
	stx	r0H	28967
			28968
	jsr	PurgeTurbo	28969
	jsr	InitForIO	28970
			28971
	jsr	z7A9B	28972
			28973
	jmp	DoneWithIO	28974
			28975
	; Befehl auf IEC-Bus senden.		28976
:z7A9B	jsr	UNLSN	28977
			28978
	lda	#\$00	28979
	sta	STATUS	28980
			28981
	lda	curDrive	28982
	jsr	LISTEN	28983
	lda	#\$ff	28984
	jsr	SECOND	28985
			28986
	lda	STATUS	28987
	bne	:err	28988
			28989
	ldy	#\$01	28990
	lda	(r0),y	28991
	sta	r1H	28992
	dey		28993
	lda	(r0),y	28994
	sta	r1L	28995
			28996
	lda	r0L	28997
	clc		28998
	adc	#2	28999
	sta	r0L	29000
	bcc	:next	29001
	inc	r0H	29002
	bcs	:next	29003
			29004
::loop	lda	(r0),y	29005
	jsr	CIOUT	29006
			29007
	iny		29008
	bne	:next	29009
	inc	r0H	29010
			29011
::next	lda	r1L	29012
	bne	:1	29013
	dec	r1H	29014
::1	dec	r1L	29015
			29016
	lda	r1H	29017

```

29018          bpl      :loop                ; => Nein, weiter...
29019
29020          jsr      UNLSN
29021
29022          ldx      #NO_ERROR
29023          rts
29024
29025 ::err          jsr      UNLSN
29026
29027          ldx      #$ff
29028          rts
29029
29030 :v7AEF          b $00      ;Dxx: Anzahl Tracks.
29031 :v7AF0          b $00                ;Aktueller Track.
29032 :v7AF1          b $00                ;Typ 1-4 für
29033 D64/D71/D81/DNP.
29034 :v7AF2          s 17                ;Name DiskImage.
29035
29036 ; Typ für DiskImage.
29037 :v7B03          b "647181NP"
29038
29039 ; Prefix DiskImage.
29040 :v7B0B          b "IMG",0
29041
29042 ; Vorgabe für Name DiskImage.
29043 :v7B0F          b "IMG"
29044 :v7B12          b "xxxx"
29045              b "-"
29046 :v7B17          b "xxxx"
29047              b NULL
29048
29049 :v7B25          b $40                ;Max.Tracks DiskImage.
29050 :v7B26          b $00                ;Länge Dateiname.
29051
29052 :v7B27          b $40,"0:"
29053 :v7B2A          b "1234567890123456,P,W",0
29054
29055 ; DiskImage öffnen.
29056 :v7B3F          w $0000
29057              b "CD:"
29058 :v7B44          s 17
29059
29060 ; DiskImage formatieren.
29061 :v7B55          w $0000
29062              b "N:"
29063 :v7B59          b "1234567890123456,01",0
29064
29065 ; DiskImage initialisieren.
29066 :v7B6D          w $0004
29067              b "I0:"
29068              b CR
29069
29070 ; DiskImage verlassen.
29071 :v7B73          w $0003
29072              b "CD_"

```



		29073
:v7B78	b \$00 ;Anzahl Sektoren/Track.	29074
		29075
; Dialogbox: Größe DiskImage/Native eingeben.		29076
:d7B70	b %10000001	29077
		29078
	b DBTXTSTR,\$09,\$10	29079
	w :t1	29080
	b DBTXTSTR,\$09,\$1a	29081
	w :t2	29082
		29083
	b DBGETSTRING	29084
	b \$10,\$24	29085
	b a1L,3	29086
		29087
	b DBTXTSTR,\$09,\$38	29088
	w :t3	29089
	b DBTXTSTR,\$09,\$42	29090
	w :t4	29091
		29092
	b CANCEL , \$11,\$48	29093
		29094
	b NULL	29095
		29096
if LANG = LANG_DE		29097
::t1	b BOLDON	29098
	b "Größe für DiskImage eingeben:"	29099
	b PLAINTEXT,0	29100
::t2	b "(Werte von 2 bis 255 oder 0 möglich)",0	29101
::t3	b BOLDON	29102
	b "Info: "	29103
	b PLAINTEXT	29104
	b "16 = 1Mb, 64 = 4Mb, 255 = 16Mb",0	29105
::t4	b "0 = Größe wie aktuelles DiskImage",0	29106
endif		29107
if LANG = LANG_EN		29108
::t1	b BOLDON	29109
	b "Set new disk image size:"	29110
	b PLAINTEXT,0	29111
::t2	b "(Allowed values 2 to 255 or 0)",0	29112
::t3	b BOLDON	29113
	b "Info: "	29114
	b PLAINTEXT	29115
	b "16 = 1Mb, 64 = 4Mb, 255 = 16Mb",0	29116
::t4	b "0 = Use current image size",0	29117
endif		29118
		29119
:v77A1	b "0"	29120
	e v77A1 +3 +1	29121
		29122
; Dialogbox: DiskImage erstellen/öffnen?		29123
:d7B79	b %10000001	29124
		29125
	b DBTXTSTR,\$08,\$10	29126
	w :t1	29127

```

29128      b DBTXTSTR,$08,$1a
29129      w :t2
29130
29131      b DBTXTSTR,$78,$2a
29132      w v7B02
29133
29134      b DBTXTSTR,$08,$2a
29135      w T29D5
29136      b DBGETSTRING
29137      b $10,$24
29138      b a1L,12
29139
29140      b DBVARSTR,$08,$42
29141      b a2L
29142
29143 ;      b OK      , $01,$48
29144      b CANCEL   , $11,$48
29145      b NULL
29146
29147 if LANG = LANG_DE
29148 ::t1      b BOLDON
29149           b "Name für DiskImage eingeben:"
29150           b PLAINTEXT,0
29151 ::t2      b "(Ohne Erweiterung .Dxx)"
29152           b BOLDON,0
29153
29154 :v7B02     b ".Dxx"
29155           b PLAINTEXT,0
29156
29157 :t7BA9     b BOLDON
29158           b "Größe: "
29159           b PLAINTEXT
29160 :t7BAA     b "xxx Tracks zu je 64Kb"
29161 :t7BBF     b 0
29162 endif
29163 if LANG = LANG_EN
29164 ::t1      b BOLDON
29165           b "Enter name for new disk image:"
29166           b PLAINTEXT,0
29167 ::t2      b "(Without extension .Dxx)"
29168           b BOLDON,0
29169
29170 :v7B02     b ".Dxx"
29171           b PLAINTEXT,0
29172
29173 :t7BA9     b BOLDON
29174           b "Size: "
29175           b PLAINTEXT
29176 :t7BAA     b "xxx Tracks of 64Kb each"
29177 :t7BBF     b 0
29178 endif
29179
29180 ;*** Endadresse testen:
29181      g OS_VARS
29182

```

```

;
; SourceCode: src.mod#09.s
;
;
;
; Modul-Info:
; 0 - Neues Verzeichnis
; 1 - Verzeichnis löschen
; 2 - Quicklink/Programm starten
; 3 - Hilfsmittel/Anwendungen/Dokumente/Drucker/Eingabe
; 4 - Verzeichnis kopieren.
; 5 - Verzeichnis duplizieren
;   (Aufruf nur durch mod.#02)
; 6 - Ordner kopieren
; 7 - Neuen Ordner erstellen
;
if .p
    t "TopSym"
    t "TopSym.MP3"
    t "TopSym.ROM"
    t "DTopDeskI18N.ext"
    t "src.DTopDesk.ext"

:SIZE_DBUF      = 128
endif

    o BASE_MOD_A
    n "obj.#09"

::ModEntry_0    jmp    N747E    ;Neues Verzeichnis.
::ModEntry_1    jmp    N76B7    ;Verzeichnis löschen.
::ModEntry_2    jmp    N7237    ;Quicklink öffnen.
::ModEntry_3    jmp    N72D7    ;Appl./DA/Dok./Print/Input.
::ModEntry_4    jmp    N73A9    ;Verzeichnis kopieren.
::ModEntry_5    jmp    N745D    ;Verzeichnis duplizieren.
::ModEntry_6    jmp    N7062    ;Ordner kopieren.
::ModEntry_7    jmp    N6EC4    ;Neuen Ordner erstellen.

; 2: Quicklink-Programm starten.
; Übergabe: V4191:
; $00 = geoHDscsi
; $11 = F2/Quicklink #1
; $22 = F4/Quicklink #2
; $33 = F6/Quicklink #3
; $44 = F8/Quicklink #4
; $FF = GEOS.Editor/GD.CONFIG
:N7237          ldx     #8                ;Erste RAMDisk
::find          lda     driveType -8,x    ;suchen.
                bmi     :1
                inx
                cpx     #12
                bcc     :find

                ldx     curDrive          ; => Keine RAMDisk.

```

```

29238
29239 ; Startlaufwerk setzen.
29240 ::1          stx    V087C          ;Startlaufwerk für
29241             txa
29242             jsr    SetDevice       ;Suche nach Datei.
29243
29244 ; Zeiger auf Dateiname.
29245             lda    V4191           ;GEOS/Setup?
29246             bpl    :appl          ; => Nein, weiter...
29247
29248 ; GEOS.Editor/GD.CONFIG.
29249 ::cfg         bit    sysVersion
29250             bmi    :gd64          ; => GD64.
29251
29252 ::ge64        ldx    #$0a          ;GEOS64.Editor.
29253             b $2c
29254 ::gd64        ldx    #$0c          ;GD.CONFIG.
29255             bne    :3
29256
29257 ; Programme LONG/F2/F4/F6/F8.
29258 ::appl        and    #%00000111   ;$11...$44 -> 1-4.
29259             cmp    #5
29260             bcs    :exit
29261             asl
29262             tax
29263
29264 ; Datei suchen.
29265 ::3           lda    v72B3 +0,x     ;Zeiger auf
29266             sta    r6L             ;Dateiname.
29267             lda    v72B3 +1,x
29268             sta    r6H
29269
29270             ldy    #$00
29271             lda    (r6),y          ;Dateiname definiert?
29272             beq    :exit           ; => Nein, Ende...
29273
29274 ; Suche auf aktuellem Laufwerk.
29275 ::loop        jsr    OpenDisk
29276             txa
29277             bne    :next           ;Diskette geöffnet?
29278                                     ; => Nein, Fehler...
29279
29280             lda    r6L             ;Dateiname.
29281             pha
29282             lda    r6H
29283             pha
29284             jsr    FindFile        ;Datei suchen.
29285             pla                   ;Dateiname.
29286             sta    r6H
29287             pla
29288             sta    r6L
29289
29290             txa
29291             bne    :next           ;Datei gefunden?
29292                                     ; => Nein, weiter...
29293
29294             lda    #$01           ;PrintFlag setzen:

```

```

        sta    r1L                ;ggf. Drucker laden.      29293
                                                                    29294
        jmp    S5040              ;Datei öffnen.           29295
                                                                    29296
::next    ldx    numDrives          29297
        dex                      ;Nur ein Laufwerk?        29298
        beq    :err                ; => Ja, Ende...         29299
                                                                    29300
        lda    curDrive            ;Nächstes Laufwerk.      29301
        tax                      29302
        sec                      29303
        sbc    #8 -1              29304
                                                                    29305
        cmp    numDrives          ;Max. Laufwerk?          29306
        bne    :4                  ; => Nein, weiter...      29307
                                                                    29308
        ldx    #8 -1              ;Erstes Laufwerk.        29309
                                                                    29310
::4        inx                      29311
        txa                      29312
        pha                      29313
        jsr    SetDevice          29314
        pla                      29315
        cmp    V087C              ;Startlaufwerk?          29316
        bne    :loop              ; => Nein, weiter...      29317
                                                                    29318
::err      ldx    #ERR_NO_APPL      29319
        jsr    S5CD0              ;Fehler A-D ausgeben.    29320
                                                                    29321
::exit     lda    V087C            ;Startlaufwerk          29322
        jmp    SetDevice          ;zurücksetzen.           29323
                                                                    29324
; Tabelle mit Zeiger auf Dateinamen.      29325
:v72B3     29326
::00      w    V4122 ;geoHDscsi      29327
::02      w    V4132 ;Quicklink#1    29328
::04      w    V4142 ;Quicklink#2    29329
::06      w    V4152 ;Quicklink#3    29330
::08      w    V4162 ;Quicklink#4    29331
::0A      w    v72BF ;GEOS64.Editor  29332
::0C      w    v72C1 ;GD.CONFIG      29333
                                                                    29334
; Konfigurationsprogramme.              29335
:v72BF     b    "GEOS64.Editor",0     29336
:v72C1     b    "GD.CONFIG",0        29337
                                                                    29338
; 3: Hilfsmittel/Anwendungen/Dokumente/Drucker/Eingabe  29339
; Übergabe: V40C2 = Menü-Nummer.        29340
;                                          29341
; Datei öffnen:                          29342
; - OFF_OPEN_APPL :    1=Anwendungen    29343
; - OFF_OPEN_DA   :    2=Hilfsmittel    29344
; - OFF_OPEN_DOC  :    3- 9=Dokumente   29345
; - OFF_OPEN_PRNT :    10=Druckertreiber 29346
; - OFF_OPEN_INPT :    11=Eingabetreiber 29347

```

```

29348 ; - OFF_OPEN_OTHER: 12/13=AutoExec/NichtGEOS
29349 :N72D7          jsr    S5245          ;Dateien invertieren.
29350              jsr    S5C89          ;Hole Disk/Fenster.
29351
29352              lda    #$00          ;Kein VLIR-Modul mehr
29353              sta    V0422          ;im Speicher!
29354
29355 ;              ldx    C042A          ;Bereits gesetzt.
29356              lda    C04F8,x        ;Lfwk. für Fenster.
29357
29358 ; Dialogbox initialisieren.
29359 ; Übergabe: A = Laufwerk.
29360 ::setdev          jsr    SetDevice
29361
29362              lda    #$00          ;GEOS-Klasse löschen.
29363              sta    r10L
29364              sta    r10H
29365
29366 ; Dokument öffnen?
29367              lda    V40C2          ;Menü-Nummer holen.
29368              cmp    #OFF_OPEN_DOC
29369              bcc    :1
29370              cmp    #(OFF_OPEN_DOC+MAX_OPEN_DOC)
29371              bcs    :1
29372
29373 ; Dokument: GEOS-Klasse setzen.
29374              sec
29375              sbc    #OFF_OPEN_DOC
29376              asl
29377              tay
29378              lda    :class +0,y
29379              sta    r10L
29380              lda    :class +1,y
29381              sta    r10H
29382
29383 ; Anwendung/DeskAccesory.
29384 ::1              ldy    V40C2          ;Menü-Nummer holen.
29385              dey
29386              lda    :types,y        ;Dateityp für
29387              sta    r7L            ;DBGETFILES.
29388
29389              lda    #< V0470        ;Speicher für
29390              sta    r5L            ;Dateiname.
29391              lda    #> V0470
29392              sta    r5H
29393
29394              lda    #NULL          ;Name löschen.
29395              sta    V0470
29396
29397              lda    #< :dbselect
29398              ldx    #> :dbselect
29399              jsr    S2090          ;DBox ohne Recover.
29400
29401              lda    r0L            ;Laufwerk wechseln?
29402              bpl    :2            ; => Nein, weiter...

```

```

29403
29404      and    #%00001111      ;Laufwerk ermitteln.
29405      bne    :setdev          ;DBox neu starten.
29406
29407  ::2      cmp    #CANCEL      ;Abbruch?
29408          bne    :3            ; => Nein, weiter...
29409  ::exit    rts
29410
29411  ::3      lda    #1            ;Drucker laden:
29412          sta    V4C54          ;PrintFlag setzen.
29413          jmp    S5033          ;Datei öffnen.
29414
29415  ::types   b DESK_ACC        ;OFF_OPEN_APPL
29416          b APPLICATION      ;OFF_OPEN_DA
29417          b APPL_DATA        ;OFF_OPEN_DOC / Alle
29418          b APPL_DATA        ;OFF_OPEN_DOC / GeoWrite
29419          b APPL_DATA        ;OFF_OPEN_DOC / GeoPaint
29420          b APPL_DATA        ;OFF_OPEN_DOC / GeoFile
29421          b APPL_DATA        ;OFF_OPEN_DOC / GeoCalc
29422          b APPL_DATA        ;OFF_OPEN_DOC / Publish
29423          b APPL_DATA        ;OFF_OPEN_DOC / GeoChart
29424          b PRINTER          ;OFF_OPEN_PRNT
29425          b INPUT_DEVICE     ;OFF_OPEN_INPT
29426          b AUTO_EXEC        ;OFF_OPEN_OTHER / AutoExec
29427          b NULL             ;OFF_OPEN_OTHER / Nicht GEOS
29428
29429  ::class   w :c03            ;Alle Dokumente
29430          w :c04            ;GeoWrite
29431          w :c05            ;GeoPaint
29432          w :c06            ;GeoFile
29433          w :c07            ;GeoCalc
29434          w :c08            ;Publish
29435          w :c09            ;GeoChart
29436
29437  ::c03     b 0
29438  ::c04     b "Write Image",0
29439  ::c05     b "Paint Image",0
29440  ::c06     b "GEOFILE IM",0
29441  ::c07     b "geoCalc Img",0
29442  ::c08     b "Publish Doc",0
29443  ::c09     b "GEOCHART",0
29444
29445  ; Dateiauswahlbox.
29446  ::dbselect b %10000001
29447
29448          b DBGETFILES!DBSETDRVICON,0,0
29449
29450          b OPEN      ,0,0
29451          b CANCEL    ,0,0
29452
29453          b NULL
29454
29455  ; 7: Neuen Ordner anlegen.
29456  ; TD13: NeuerOrdner
29457  :N6EC4     jsr    S5245          ;Dateien invertieren.

```

```

29458
29459          jsr    S5C89          ;Hole Disk/Fenster.
29460          tax
29461          bne     :err
29462
29463 ;          lda     #NULL          ;Name löschen.
29464          sta     V0470
29465
29466 ;          lda     #NULL
29467          b $2c
29468 ::getname          lda     #DBTXTSTR
29469          sta     v6F15          ;Dialogbox anpassen.
29470
29471          lda     #< V0470        ;Zeiger auf Name
29472          sta     a1L            ;für neuen ordner.
29473          lda     #> V0470
29474          sta     a1H
29475
29476          lda     #< t7461        ;"Neuer Ordner".
29477          sta     a2L
29478          lda     #> t7461
29479          sta     a2H
29480
29481          lda     #< d6F07        ;Neuen Namen für
29482          ldx     #> d6F07        ;Ordner eingeben.
29483          jsr     S2090          ;DBox ohne Recover.
29484
29485          lda     V0470          ;Name eingegeben?
29486          beq     :exit          ; => Nein, Ende...
29487
29488          lda     r0L
29489          cmp     #CANCEL        ;Abbruch?
29490          bne     :ok           ; => Ja, Ende...
29491
29492 ::exit          rts            ;Ende.
29493
29494 ::ok          ldx     C042A      ;Oberstes Fenster.
29495          jsr     z6F2E          ;Neuer Ordner.
29496          cpx     #FILE_EXISTS  ;Bereits vorhanden?
29497          beq     :getname      ; => Neuer Name...
29498          txa
29499          bne     :err          ; => Fehler...
29500
29501          jmp     S4296          ;Update akt.Fenster.
29502
29503 ::err          cpx     #CANCEL_ERR ;Abbruch?
29504          beq     :exit          ; => Ja, Ende...
29505
29506          jmp     S5CF3          ;Fehler ausgeben.
29507
29508 ; Dialogbox: Name eingeben:
29509 :d6F07          b %10000001
29510
29511          b CANCEL , $11, $48
29512

```



	b DBVARSTR,\$08,\$10	29513
	b a2L	29514
		29515
	b DBTXTSTR,\$08,\$1c	29516
	w T29D5	29517
		29518
	b DBGETSTRING	29519
	b \$10,\$16	29520
	b a1L,16	29521
		29522
	; Wird verändert.	29523
	; Ist die Datei bereits vorhanden, dann	29524
	; wird hier DBTXTSTR eingesetzt und der	29525
	; Text "bereits vorhanden" angezeigt.	29526
:v6F15	b NULL , \$08,\$4a ;"Fehler".	29527
	w T7464	29528
	b DBTXTSTR,\$08,\$54 ;"Bereits vergeben".	29529
	w t7470	29530
		29531
	b NULL	29532
		29533
if LANG = LANG_DE		29534
:t7461	b BOLDON,"Name für neuen Ordner:",0	29535
:t7462	b BOLDON,"Name für neues Verzeichnis:",0	29536
		29537
:t7470	b "Name bereits vergeben!",0	29538
endif		29539
if LANG = LANG_EN		29540
:t7461	b BOLDON,"Enter new folder name:",0	29541
:t7462	b BOLDON,"Enter new directory name:",0	29542
		29543
:t7470	b "The name is already in use!",0	29544
endif		29545
		29546
	; Neuen Ordner erstellen.	29547
	; Übergabe: X = Fenster-Nr.	29548
	; TD13: MyNewSubDir	29549
:z6F2E	txa ;Fenster-Nr.	29550
	pha	29551
		29552
	lda #< C051F ;8-Byte-Ordnerliste.	29553
	sta r0L	29554
	lda #> C051F	29555
	sta r0H	29556
		29557
	lda #< v777B ;128 Byte für	29558
	sta r3L ;Tr/Se Infoblock der	29559
	lda #> v777B ;Ordner in Liste.	29560
	sta r3H	29561
		29562
	jsr z6F65 ;Liste mark. Dateien.	29563
		29564
	lda #< V0470 ;Ordner-Name.	29565
	sta r1L	29566
	lda #> V0470	29567

```

29568          sta    r1H
29569          jsr    z6FE3          ;Ordner erstellen.
29570
29571          pla          ;Fenster-Nr.
29572
29573          cpx    #NO_ERROR
29574          bne    :err          ; => Fehler...
29575
29576          tax          ;Fenster-Nr.
29577          lda    C051B,x        ;Aktive Ordner-Nr.
29578          sta    r10L
29579          jsr    S2832          ;Datei in Ordner.
29580
29581 ::err          rts
29582
29583 ; Liste mit Ordnern erstellen.
29584 ; Übergabe: r0 = Zeiger 8-Byte-Puffer für Ordner-Liste.
29585 ;           r3 = Zeiger 128-Byte-Puffer Tr/Se Infoblock.
29586 ; Rückgabe: r0 = Zeiger 8-Byte-Puffer, jedes der 64Bit
29587 ;             definiert ob die Ordner-Nr. frei (0) oder
29588 ;             bereits belegt (1) ist.
29589 ; TD13: MakeDirList
29590 :z6F65          ldy    #8 -1
29591                  lda    #$00
29592 ::1              sta    (r0),y
29593                  dey
29594                  bne    :1
29595
29596                  lda    #1          ;Kein Ordner Nr=0.
29597                  sta    (r0),y
29598
29599                  lda    r0H
29600                  pha
29601                  lda    r0L
29602                  pha
29603
29604                  ldy    #SIZE_DBUF -1 ;Speicher für Tr/Se
29605                  lda    #$00          ;für Infoblock/Ordner
29606 ::2              sta    (r3),y        ;löschen.
29607                  dey
29608                  bpl    :2
29609                  sta    r11H
29610
29611 ; r3   = Zeiger auf 128Byte Puffer für Ordner-Einträge.
29612 ; r10L = %10000000 / Alle Ordner suchen.
29613 ;      Bit%0-5=0 / Keine Ordner-Nr.
29614 ; r11L = max. 64 Dateien finden.
29615 ; r12L = %11000000 / Gelöschte/Leere Einträge ignorieren.
29616 ; r12H = Dateityp.
29617 ; r13L = 2 Bytes aus Eintrag einlesen.
29618 ; r13H = 20 Bytes aus Eintrag überlesen.
29619          lda    #%11000000
29620          sta    r10L
29621          sta    r12L
29622          lda    #TDFOLDER

```

	sta	r12H	29623
	lda	#64	29624
	sta	r11L	29625
	lda	#2	29626
	sta	r13L	29627
	lda	#20	29628
	sta	r13H	29629
			29630
	lda	r3H	29631
	pha		29632
	lda	r3L	29633
	pha		29634
			29635
	jsr	S2603 ;Ordner suchen.	29636
			29637
	pla		29638
	sta	r3L	29639
	pla		29640
	sta	r3H	29641
			29642
	pla		29643
	sta	r0L	29644
	pla		29645
	sta	r0H	29646
			29647
	txa		29648
	bne	:err ; => Fehler...	29649
			29650
	lda	#< diskBlkBuf	29651
	sta	r4L	29652
	lda	#> diskBlkBuf	29653
	sta	r4H	29654
			29655
	ldy	#\$00 ;Infoblock des	29656
::loop	lda	(r3),y ;nächsten Ordners	29657
	sta	r1L ;einlesen.	29658
	beq	:done	29659
	iny		29660
	lda	(r3),y	29661
	sta	r1H	29662
	iny		29663
	tya		29664
	pha		29665
	jsr	GetBlock	29666
	pla		29667
	tay		29668
	txa		29669
	bne	:err ; => Fehler...	29670
			29671
	tya		29672
	pha		29673
			29674
	ldy	#OFF_DIR_NUM	29675
	lda	diskBlkBuf,y	29676
	jsr	z7041 ;Ordner in Liste.	29677

```

29678
29679          pla
29680          tay
29681
29682          clv
29683          bvc    :loop          ;Nächster Ordner.
29684
29685 ::done          ldx    #NO_ERROR
29686 ::err           rts
29687
29688 ; Ordner erstellen.
29689 ; Übergabe: r0  = Zeiger 8-Byte-Puffer für Ordner-Liste.
29690 ;             r1  = Zeiger auf Dateiname.
29691 ;             r10L = Eltern-Ordner.
29692 ; TD13: MakeDir
29693 :z6FE3          jsr    z701F          ;Suche fr.Ordner-Nr.
29694                sta    v7236          ;Freie Ordner-Nr.
29695                txa
29696                bne    :err          ; => Fehler...
29697
29698                lda    r1L          ;Zeiger auf
29699                sta    v71C1 +0        ;Dateiname speichern.
29700 ;                sta    r6L
29701                ldx    r1H
29702                stx    v71C1 +1
29703 ;                sta    r6H
29704
29705                jsr    S1414
29706                txa          ;Datei gefunden?
29707                beq    :exist          ; => Ja, Ende...
29708                cpx    #FILE_NOT_FOUND
29709                beq    :1          ; => Nein, weiter...
29710                bne    :err          ; => Fehler, Ende...
29711
29712 ::exist          ldx    #FILE_EXISTS
29713                rts
29714
29715 ::1              lda    r0L          ;Zeiger/Tabelle für
29716                sta    r14L          ;Ordner-Liste.
29717                lda    r0H
29718                sta    r14H
29719
29720                lda    #< v71C1        ;Infoblock.
29721                sta    r9L
29722                lda    #> v71C1
29723                sta    r9H
29724
29725 ; Hinweis:
29726 ; Durch den Wert #1 werden Ordner immer
29727 ; erst ab dem zweiten Verzeichnisblock
29728 ; erstellt/gespeichert.
29729 ;                lda    #1          ;Vorgabe: TopDesk.
29730                lda    #0          ;Suche nach freiem
29731                sta    r10L          ;Eintrag ab Seite 0.
29732

```

	jsr	SaveFile	29733
			29734
::err	rts		29735
			29736
		; Infoblock für neuen Ordner.	29737
:v71C1	w	\$0000 ;Zeiger auf Dateiname.	29738
::HdrB002	b	\$03,\$15,\$bf	29739
::HdrB005	b	%00000000,%00000001,%11111110	29740
	b	%00000000,%00000010,%00000001	29741
	b	%00111111,%11111110,%00000001	29742
	b	%01000000,%00000000,%00000001	29743
	b	%01000000,%00000000,%00000001	29744
	b	%01111111,%11111111,%11111101	29745
	b	%10000000,%00000000,%00000011	29746
	b	%10000000,%00000000,%00000011	29747
	b	%10000000,%00000000,%00000011	29748
	b	%10000000,%00000000,%00000011	29749
	b	%10000000,%00000000,%00000011	29750
	b	%10000000,%00000000,%00000011	29751
	b	%10000000,%00000000,%00000011	29752
	b	%10000000,%00000000,%00000011	29753
	b	%10000000,%00000000,%00000011	29754
	b	%10000000,%00000000,%00000011	29755
	b	%10000000,%00000000,%00000011	29756
	b	%10000000,%00000000,%00000011	29757
	b	%10000000,%00000000,%00000011	29758
	b	%10000000,%00000000,%00000010	29759
	b	%01111111,%11111111,%11111100	29760
			29761
::HdrB068	b	%10000000 ! USR	29762
::HdrB069	b	TDFOLDER	29763
::HdrB070	b	SEQUENTIAL	29764
::HdrB071	w	\$0000,\$0000,\$0000	29765
::HdrB077	b	"Directory V1.0"	29766
	b	NULL	29767
::HdrB094	b	\$00	29768
::HdrB095	b	\$00	29769
::HdrB096	b	\$00	29770
::HdrB097	b	"DPT-Team"	29771
	e	:HdrB097 +20	29772
			29773
::HdrB117		; = OFF_DIR_NUM	29774
:v7236	b	\$00 ;Freie Ordner-Nr.	29775
			29776
		; Freie Ordner-Nr. suchen.	29777
		; Hinweis: Die Ordner-Nr wird nicht als	29778
		; "belegt" markiert!	29779
		; Übergabe: r0 = Zeiger 8-Byte-Puffer für Ordner-Liste.	29780
		; Rückgabe: A = Ordner-Nr.	29781
		; X = Fehler.	29782
		; Ändert : A,X,Y,r6L	29783
		; TD13: NextFreeDirNum	29784
:z701F	ldy	#\$00 ;Ordner-Nr. löschen.	29785
	sty	r6L	29786
			29787

```

29788 ;          ldy  #0
29789 ::1      ldx  #8 -1          ;Max. 8 Bit testen.
29790          lda  (r0),y
29791 ::2      lsr          ;Ordner frei?
29792          bcc  :ok          ; => Ja, Ende...
29793          inc  r6L
29794          dex          ;Alle Bit getestet?
29795          bpl  :2          ; => Nein, weiter...
29796          iny
29797          cpy  #8          ;Alle Byte getestet?
29798          bne  :2          ; => Nein, weiter...
29799
29800          ldx  #TDLIST_FULL  ;Fehler: Kein Ordner.
29801          rts
29802
29803 ::ok      ldx  #NO_ERROR    ;Kein Fehler.
29804          lda  r6L          ;Freie Ordner-Nr.
29805          rts
29806
29807 ; Ordner-Nr. in Liste eintragen.
29808 ; Übergabe: r0 = Zeiger 8-Byte-Puffer für Ordner-Liste.
29809 ;          A = Ordner-Nr.
29810 ; Rückgabe: r0 = Zeiger 8-Byte-Puffer für Ordner-Liste.
29811 ; Ändert : A,Y,r1L
29812 ; TD13: PutinList
29813 :z7041    pha
29814          and  #%00000111
29815          tay
29816
29817          lda  #%00000001
29818          cpy  #$00          ;Ordner #0?
29819          beq  :2          ; => Ja, weiter...
29820 ::1      asl
29821          dey
29822          bne  :1
29823 ::2      sta  r1L          ;Ordner-Bits.
29824
29825          pla
29826          and  #%001111000
29827          lsr
29828          lsr
29829          lsr
29830          tay          ;Zeiger Ordner-Byte.
29831
29832          lda  (r0),y
29833          ora  r1L          ;Ordner-Bits setzen.
29834          sta  (r0),y
29835          rts
29836
29837 ; 6: Ordner kopieren.
29838 ; TD13: _CopyDir
29839 :N7062    lda  #$00          ;Modus setzen:
29840          sta  V40C3          ;Alle Dat. kopiert.
29841
29842          lda  V04AA          ;Akt. Datei-Nr.

```

	sta	v71BC	; zwischenspeichern.	29843
				29844
	inc	V04B0	; Zähler Instanz +1.	29845
				29846
	lda	V0A72	; Lfwk. Source-Disk.	29847
	jsr	S06AB	; Source-Disk öffnen.	29848
				29849
	ldy	#16 -1		29850
::11	lda	V0470,y		29851
	pha			29852
	dey			29853
	bpl	:11		29854
				29855
	jsr	z7070	; Ordner kopieren.	29856
				29857
	ldy	#\$00	; => Datei/Ordner.	29858
	sty	v745C	; Dateityp setzen.	29859
				29860
; ::12	ldy	#0		29861
	pla			29862
	sta	V0470,y		29863
	iny			29864
	cpy	#16		29865
	bcc	:12		29866
				29867
	txa			29868
	beq	z715C	; => OK...	29869
	cpx	#FILE_EXISTS		29870
	bne	z7151	; => Fehler...	29871
	beq	x71B6	; => Ende...	29872
				29873
; Ordner/Verzeichnis kopieren fortsetzen.				29874
:z715C	lda	V0A72	; Lfwk. Source-Disk.	29875
	jsr	S06AB	; Source-Disk öffnen.	29876
	txa			29877
	bne	z714E		29878
				29879
	ldx	V04A9	; Kop.od.verschieben?	29880
	beq	x71B6	; => Kopieren.	29881
				29882
; Modus: Verschieben.				29883
	ldx	V04AB	; Abbruch?	29884
	bne	x71B6	; => Ja, Ende...	29885
				29886
; Quelle löschen.				29887
	lda	v745C	; Verz. kopieren?	29888
	beq	:21	; => Nein, weiter...	29889
				29890
; -> Verzeichnis löschen.				29891
;				29892
; Hinweis:				29893
; Systemroutine zum löschen verwenden,				29894
; damit Verz. mit Schreibschutz nicht				29895
; gelöscht wird.				29896
;	jsr	N76B7	; Verzeichnis löschen.	29897

```

29898          jsr    S5B38          ;Verzeichnis löschen.
29899          bcs    x71B6          ; => Nicht leer...
29900          bcc    :22            ; => OK, weiter...
29901
29902 ; -> Ordner löschen.
29903 ::21          lda    V40C3          ;Alle Dat. gelöscht?
29904          bne    x71B6          ; => Nein, weiter...
29905
29906 ; Hinweis:
29907 ; Ein Ordner-Flag setzen, damit beim
29908 ; löschen eines Verzeichnisses in einem
29909 ; Ordner der Fehler "DIR_NOT_EMPTY"
29910 ; nicht 2x angezeigt wird:
29911 ; 1x beim Verzeichnis verschieben und
29912 ; 1x beim Ordner löschen.
29913          lda    v71C0          ;Ordner-Flag setzen.
29914          pha
29915          lda    #$ff          ;Ordner-Inhalt
29916          sta    v71C0          ;löschen.
29917          jsr    S5B38          ;Eintrag löschen.
29918          pla          ;Ordner-Flag wieder
29919          sta    v71C0          ;zurücksetzen.
29920
29921          txa
29922          bne    x71B6          ; => Fehler...
29923
29924 ::22          dec    V04B0          ;Letzte Instanz?
29925          bne    x71BB          ; => Nein, weiter...
29926
29927          ldx    V04A9          ;Kop.od.verschieben?
29928          beq    x71BB          ; => Kopieren.
29929
29930 ; Index mark.Dateien korrigieren.
29931 ; Siehe auch Routine in CopyFile.
29932          ldx    #$ff
29933 ::31          inx
29934          lda    V634F,x          ;Tab. mark.Dateien.
29935          cmp    #$ff
29936          beq    x71B9
29937          cmp    v71BC          ;Nr. mark.Datei.
29938          bcc    :31
29939          dec    V634F,x          ;Tab. mark.Dateien.
29940          bcs    :31
29941
29942 ; Ende...
29943 :x71B6        dec    V04B0          ;Zähler Instanz -1.
29944
29945 :x71B9        ldx    #NO_ERROR
29946 :x71BB        clc
29947          rts
29948
29949 ; Fehler ausgeben.
29950 :z714E        jsr    S5CF3          ;Fehler ausgeben.
29951
29952 ; Datei Nicht kopiert.

```



:z7151	lda	#\$01		29953
	sta	V04AB	;Abbruch.	29954
				29955
	dec	V04B0	;Zähler Instanz -1.	29956
				29957
	sec		;C=1: Ende.	29958
	rts			29959
				29960
:v71BC	b	\$00	;Nr. markierte Datei.	29961
:v71BD	b	\$00	;Nr. Ordner/Source.	29962
:v71BE	b	\$00	;Nr. Ordner/Target.	29963
:v71BF	b	\$00	;Dateizähler.	29964
:v71C0	b	\$00	;\$FF=Ordnerinhalt löschen.	29965
:v745C	b	\$00	;Lfwk./Source für Verz. kopieren.	29966
				29967
			; Ordner kopieren.	29968
:z7070	jsr	S1412	;Datei suchen.	29969
	txa			29970
	bne	:err		29971
				29972
	jsr	S09C7	;Infoblock einlesen.	29973
	txa			29974
	bne	:err		29975
				29976
	lda	diskBlkBuf +OFF_DIR_NUM		29977
	sta	v71BD	;Nr. Ordner/Source.	29978
				29979
	lda	C042A	;Oberstes Fenster.	29980
	pha			29981
				29982
	lda	WIN042F	;Aktives Fenster.	29983
	sta	C042A	;Oberstes Fenster.	29984
				29985
	lda	V0A73	;Lfwk. Target-Disk.	29986
	jsr	S06AB	;Target-Disk öffnen.	29987
				29988
	pla			29989
	sta	C042A	;Oberstes Fenster.	29990
	txa			29991
	bne	:err	; => Fehler, Abbruch.	29992
				29993
			; Neuen Ordner erstellen.	29994
::retry	ldx	WIN042F	;Aktives Fenster.	29995
	jsr	z6F2E	;Neuer Ordner.	29996
	cpx	#FILE_EXISTS		29997
	beq	:found	; => File exists...	29998
	txa			29999
	beq	:copy	; => Ordner erstellt.	30000
				30001
			; Fehler.	30002
::err	jmp	z7151	;Abbruch, Ende.	30003
				30004
			; Datei existiert bereits.	30005
::found	lda	dirEntryBuf +1	;Ordner?	30006
	beq	:copy	; => Ja, öffnen.	30007

```

30008
30009 if TRUE
30010 ; Dateien in Hauptverzeichnis kopieren.
30011         lda    #$00                ;Hauptverzeichnis.
30012         beq    :init              ; => Nur Dateien.
30013 else
30014 ; Alternativ:
30015 ; Wenn Datei nicht ersetzt werden soll,
30016 ; Ordner+Dateien nicht kopieren.
30017         ldx    #FILE_EXISTS
30018         rts
30019 endif
30020
30021 ; Ziel-Ordner/Target öffnen.
30022 ::copy      jsr    S1412            ;Ordner suchen.
30023             txa
30024             bne    :err            ; => Fehler, Ende...
30025
30026             jsr    S09C7           ;Infoblock einlesen.
30027             txa
30028             bne    :err            ; => Fehler, Ende...
30029
30030 ; Hinweis:
30031 ; Ordner-Byte für vorhandenen Ordner.
30032         lda    diskBlkBuf +OFF_DIR_NUM
30033
30034 ; Ordner kopieren.
30035 ; Übergabe: A = Ordner-Nr. Target.
30036 ;           0=Hauptverzeichnis.
30037 ::init      sta    v71BE            ;Nr. Ordner/Target.
30038
30039             lda    V0A72            ;Lfwk. Source-Disk.
30040             jsr    S06AB            ;Source-Disk öffnen.
30041             txa
30042             bne    :err            ; => Fehler, Ende...
30043
30044 ;           lda    #$00            ;Dateizähler.
30045             sta    v71BF
30046
30047 ::loop      ldy    C042A            ;Fenster/Source.
30048             lda    C051B,y          ;Aktive Ordner-Nr.
30049             pha                    ;zwischenspeichern.
30050             lda    v71BD            ;Nr. Ordner/Source.
30051             sta    C051B,y          ;Aktive Ordner-Nr.
30052
30053             ldy    WIN042F          ;Fenster/Target.
30054             lda    C051B,y          ;Aktive Ordner-Nr.
30055             pha                    ;zwischenspeichern.
30056             lda    v71BE            ;Nr. Ordner/Target.
30057             sta    C051B,y          ;Aktive Ordner-Nr.
30058
30059             lda    V40C3            ;Zähler für fehlende
30060             pha                    ;Dateien speichern.
30061             lda    v71BF            ;Dateizähler.
30062             pha

```

```

;                                     30063
;         lda    v71BF                ;Dateizähler.                30064
;         jsr    S5537                ;CopyService.                30065
;         sta    r0L                  ;$00=Datei im Ordner.          30066
;                                     30067
;         pla                                     30068
;         sta    v71BF                ;Dateizähler.                30069
;         pla                                     ;Zähler für fehlende      30070
;         sta    V40C3                ;Dat. zurücksetzen.          30071
;                                     30072
;         ldy    WIN042F              ;Fenster/Target.            30073
;         lda    C051B,y              ;Aktive Ordner-Nr.          30074
;         sta    v71BE                ;Nr. Ordner/Target.          30075
;         pla                                     30076
;         sta    C051B,y              ;Aktive Ordner-Nr.          30077
;                                     30078
;         ldy    C042A                ;Fenster/Source.            30079
;         lda    C051B,y              ;Aktive Ordner-Nr.          30080
;         sta    v71BD                ;Nr. Ordner/Source.          30081
;         pla                                     30082
;         sta    C051B,y              ;Aktive Ordner-Nr.          30083
;                                     30084
;         txa                                     ;Fehler?                30085
;         beq    :ok                  ; => Nein, weiter...          30086
;                                     30087
;         inc    V40C3                ;Datei nicht kopiert.          30088
;                                     30089
; Datei gefunden?                      30090
::ok          lda    r0L                ;Datei im Ordner?          30091
              bne    :done              ; => Nein, weiter...          30092
;                                     30093
; Datei verschoben oder Fehler.          30094
              bcs    :err                ; => C=1, Fehler...          30095
;                                     30096
; Dateien verschieben?                  30097
              lda    V04A9              ;Kop.od.verschieben?      30098
              beq    :next              ; => verschieben.          30099
              txa                                     ;Datei verschoben?      30100
              beq    :loop              ; => Ja, weiter...          30101
;                                     30102
; Hinweis:                              30103
; Datei nicht verschoben, Zähler auf     30104
; nächste Datei setzen.                  30105
;                                         30106
; Hinweis:                              30107
; Aus jedem Unterordner werden hier      30108
; max. 256 Dateien kopiert.              30109
::next        inc    v71BF              ;Dateizähler +1.          30110
              bne    :loop              ; => Nächste Datei...      30111
;                                     30112
::done        ldx    #NO_ERROR          30113
              rts                      30114
;                                     30115
; 5: Verzeichnis duplizieren.            30116
; Aufruf aus mod.#02.                   30117

```

```

30118 :N745D          jsr    z7400          ;SubDir-Hinweis.
30119
30120          jsr    z7401          ;Dateiname kopieren.
30121          jsr    z7497          ;SubDir erstellen.
30122          txa
30123          beq     :1
30124
30125          sta     V04AB          ;>$00 = Abbruch.
30126
30127 ::1            jmp     S39A7          ;Zurück zu mod.#02:
30128                                     ;Dateien Duplizieren.
30129
30130 ; 4: Verzeichnis kopieren.
30131 ;
30132 ; Hinweis:
30133 ; Modul wird nur noch aufgerufen, wenn
30134 ; keine Ziel-Datei existiert bzw. ein
30135 ; leeres Verzeichnis gelöscht wurde.
30136 ; Ist Ziel-Laufwerk kein Native, dann
30137 ; erfolgt ebenfalls kein Aufruf mehr.
30138 :N73A9          lda     V04AA          ;Akt. Datei-Nr.
30139          sta     v71BC          ;zwischenspeichern.
30140
30141          inc     V04B0          ;Zähler Instanz +1.
30142
30143          lda     V0A72          ;Lfwk. Source-Disk.
30144          sta     v745C          ;Modus: Kopieren.
30145          jsr     S06AB          ;Source-Disk öffnen.
30146
30147          jsr     S1412          ;Datei suchen.
30148          txa
30149          bne     :err          ; => Fehler, Ende...
30150
30151          lda     C042A          ;Oberstes Fenster.
30152          pha
30153
30154          lda     WIN042F          ;Aktives Fenster.
30155          sta     C042A          ;Oberstes Fenster.
30156          lda     V0A73          ;Lfwk. Target-Disk.
30157          jsr     S06AB          ;Target-Disk öffnen.
30158
30159          pla
30160          sta     C042A          ;Oberstes Fenster.
30161          txa
30162          bne     :err          ; => Fehler, Ende...
30163
30164          jsr     z7400          ;SubDir-Hinweis.
30165
30166          jsr     z7401          ;Dateiname kopieren.
30167          jsr     z74C1          ;SubDir erstellen.
30168          txa
30169          bne     :err          ; => Fehler...
30170
30171 ::done          jmp     z715C          ;Kopieren fortsetzen.
30172

```

```

; Fehler ausgeben, Ende... 30173
::err          jmp      z714E          ; => Abbruch, Ende... 30174
                                                    30175
; Dateiname kopieren. 30176
:z7401          ldx      #$00          30177
                  ldy      #$00          30178
::1            lda      V0470,x        30179
                  beq      :2          30180
                  sta      v76A6,x     30181
                  inx              30182
                  tya              30183
                  sta      v76A6,x     30184
                  cpx      #16 +1     30185
                  bcc      :1          30186
::2            rts                    30187
                                                    30188
; SubDir-Hinweis anzeigen. 30189
:z7400          ldx      V5596          ;Info anzeigen? 30190
                  bne      :1          ; => Nein, weiter... 30191
                  inx              30192
                  stx      V5596       ;SubDir-Info setzen. 30193
                                                    30194
                  lda      #< d742B    ;Hinweis: SubDir 30195
                  ldx      #> d742B    ;wird leer kopiert. 30196
                  jsr      S2090       ;DBox ohne Recover. 30197
                                                    30198
::1            rts                    30199
                                                    30200
; Dialogbox: UV wird ohne Inhalt kopiert. 30201
:d742B          b %10000001          30202
                                                    30203
                  b DBTXTSTR,$08,$10  30204
                  w T29CF             30205
                                                    30206
                  b DBTXTSTR,$08,$20  30207
                  w :t1               30208
                  b DBTXTSTR,$08,$2a  30209
                  w :t2               30210
                                                    30211
                  b DBTXTSTR,$08,$3a  30212
                  w :t3               30213
                                                    30214
                  b OK                , $11,$48 30215
                                                    30216
                  b NULL              30217
                                                    30218
if LANG = LANG_DE 30219
::t1            b "Verzeichnisse werden ohne Inhalte wie",0 30220
::t2            b "Dateien oder Ordner kopiert/dupliziert!",0 30221
                                                    30222
::t3            b "(Hinweis wird nur einmal angezeigt)",0 30223
endif          30224
if LANG = LANG_EN 30225
::t1            b "Directories are copied/duplicated",0 30226
::t2            b "without content such as files/folders!",0 30227

```

```

30228
30229 ::t3                b "(Message is only displayed once)",0
30230 endif
30231
30232 ; 0: Neues Verzeichnis erstellen.
30233 :N747E                ldx    curDrive
30234                    lda    RealDrvMode -8,x
30235                    and    #SET_MODE_SUBDIR
30236                    beq    :folder            ; => Kein Native...
30237
30238                    lda    #NULL                ;Name Verzeichnis
30239                    sta    v76A6                ;löschen.
30240
30241                    jsr    z7497                ;Neues Verzeichnis.
30242                    txa
30243                    beq    :refresh
30244
30245                    rts
30246
30247 ; Kein NativeMode, Ordner erstellen.
30248 ::folder            jmp    N6EC4                ;Ordner erstellen.
30249
30250 ; Hinweis:
30251 ; Im Textmodus wird der Bereich für
30252 ; VLIR-Module überschrieben!
30253 ::refresh            jmp    S4321                ;Alle Fenster neu.
30254
30255 ; Aktuellen Namen bearbeiten.
30256 :z7497                lda    #NULL
30257                    b    $2c
30258
30259 ; Neuen Namen für Verzeichnis eingeben.
30260 :z7498                lda    #DBTXTSTR
30261                    sta    v6F15                ;Dialogbox anpassen.
30262
30263                    lda    #< v76A6                ;Vorgabe Verzeichnis-
30264                    sta    a1L                ;name übernehmen.
30265                    lda    #> v76A6
30266                    sta    a1H
30267
30268                    lda    #< t7462                ;"Neues Verzeichnis".
30269                    sta    a2L
30270                    lda    #> t7462
30271                    sta    a2H
30272
30273                    lda    #< d6F07                ;Name eingeben.
30274                    ldx    #> d6F07
30275                    jsr    S2090                ;DBox ohne Recover.
30276
30277                    lda    v76A6                ;Name gültig?
30278                    beq    :exit                ; => Nein, Ende...
30279
30280                    lda    r0L
30281                    cmp    #CANCEL                ;Abbruch?
30282                    bne    z74C1                ; => Nein, weiter...

```

		30283
	ldx #CANCEL_ERR	30284
::exit	rts	30285
		30286
	; Verzeichnis erstellen.	30287
	; Übergabe: v76A6 = Name.	30288
:z74C1	lda #< v76A6	30289
	ldx #> v76A6	30290
	jsr S1414	30291
	cpx #FILE_NOT_FOUND	30292
	beq :1 ; => OK, weiter...	30293
	txa	30294
	beq z7498 ; => Neuer Name...	30295
	bne :err ; => Fehler, Fens.neu.	30296
		30297
::1	jsr OpenDisk ;Diskette/BAM öffnen.	30298
	txa	30299
	bne :err	30300
		30301
	lda #\$01 ;Erster Datenblock	30302
	sta r3L ;auf NativeMode.	30303
	lda #\$40	30304
	sta r3H	30305
	jsr SetNextFree	30306
	txa	30307
	bne :err	30308
		30309
	; Ein Unterverzeichnis erfordert hier	30310
	; mind. zwei Blöcke in Folge.	30311
	lda r3L	30312
	ldx r3H	30313
::loop	sta v769F	30314
	stx v76A0	30315
	jsr SetNextFree	30316
	txa	30317
	bne :err	30318
		30319
	lda r3L	30320
	ldx r3H	30321
	cmp v769F	30322
	bne :fail	30323
	cpx #\$00	30324
	beq :fail	30325
	dex	30326
	cpx v76A0	30327
	beq :found	30328
		30329
::fail	lda v769F	30330
	sta r6L	30331
	lda v76A0	30332
	sta r6H	30333
	jsr FreeBlock	30334
	txa	30335
	bne :err	30336
		30337

```

30338          ldx    r3H
30339          lda    r3L          ;Track gültig?
30340          bne    :loop        ; => Ja, weiter...
30341
30342          ldx    #INV_TRACK
30343 ::err      jsr    S5CF3        ;Fehler ausgeben.
30344          jmp    S157E        ;Alle Fens. anzeigen.
30345
30346 ; Zwei freie Blocks gefunden.
30347 ::found    inx
30348          sta    v76A1        ;Zweiten Verzeichnis-
30349          stx    v76A2        ;Sektor speichern.
30350
30351          jsr    z7550        ;Freien Eintrag im
30352          txa                    ;Verzeichnis suchen/
30353          bne    :err        ;Eintrag speichern.
30354
30355          jsr    z75DC        ;Neuer Verz.header.
30356          txa
30357          bne    :err
30358
30359 ; Leeren Sektor erzeugen.
30360 ;          ldx    #$00        ;Erster Verz.block.
30361 ;          txa
30362 ::2        sta    diskBlkBuf,x
30363          inx
30364          bne    :2
30365          dex
30366          stx    diskBlkBuf +1
30367
30368          lda    v76A1
30369          sta    r1L
30370          lda    v76A2
30371          sta    r1H
30372          jsr    PutBlock_dskBuf
30373          txa
30374          bne    :err
30375
30376          jsr    PutDirHead    ;BAM speichern.
30377 ;          txa
30378 ;          bne    :err
30379 ;
30380 ; Hinweis:
30381 ; Dieser Befehl sorgt dafür, das beim
30382 ; duplizieren mehrerer Verzeichnisse
30383 ; nach jedem Duplizieren alle Fenster
30384 ; neu aufgebaut werden.
30385 ;          lda    V5596        ;SubDir-Info aktiv?
30386 ;          bne    :done        ; => Ja, Ende...
30387 ;
30388 ;          jmp    S4321        ;Alle Fenster neu.
30389
30390 ::done      rts
30391
30392 ; Freien Eintrag im akt.Verzeichnis suchen.

```



:z7550	lda	#\$00		30393
	sta	r10L		30394
	jsr	GetFreeDirBlk		30395
	txa			30396
	beq	:1		30397
	rts			30398
				30399
::1	lda	r1L	;Tr/Se für Block mit	30400
	sta	v76A3	;neuem Eintrag.	30401
	lda	r1H		30402
	sta	v76A4		30403
	sty	v76A5	;Offset im Block.	30404
				30405
	ldx	WIN042F	;Aktives Fenster.	30406
	lda	C051B,x	;Aktive Ordner-Nr.	30407
	pha			30408
				30409
	tya		;Offset auf Eintrag.	30410
	cmp	#2	;Eintrag #0?	30411
	bne	:2	; => Nein, weiter...	30412
				30413
	clc		;Sonderbehandlung.	30414
	adc	#TDFOLDER_POS0 -1		30415
				30416
::2	tax			30417
	dex		;Offset Ordner-Nr.	30418
				30419
	pla			30420
	sta	diskBlkBuf,x	;Ordner-Nr. setzen.	30421
				30422
				30423
				30424
				30425
	lda	##10000000!FTYPE_DIR		30426
	sta	diskBlkBuf,y		30427
	iny			30428
	lda	v769F	;Tr Verz.-Header.	30429
	sta	diskBlkBuf,y		30430
	iny			30431
	lda	v76A0	;Se Verz.-Header.	30432
	sta	diskBlkBuf,y		30433
	iny			30434
	jsr	z7681	;Name kopieren.	30435
				30436
	lda	#\$00		30437
	sta	diskBlkBuf,y		30438
	iny			30439
	sta	diskBlkBuf,y		30440
	iny			30441
	sta	diskBlkBuf,y		30442
	iny			30443
	sta	diskBlkBuf,y		30444
	iny			30445
				30446
	ldx	#0	;Datum und Uhrzeit	30447
::3	lda	year,x	;übernehmen.	

```

30448          sta  diskBlkBuf,y
30449          iny
30450          inx
30451          cpx  #3 +2
30452          bcc  :3
30453
30454          lda  #< 2          ;Verzeichnisgröße
30455          sta  diskBlkBuf,y  ;zu Beginn 2 Blocks.
30456          iny
30457          lda  #> 2
30458          sta  diskBlkBuf,y
30459          jmp  PutBlock_dskBuf
30460
30461 ; Verzeichnisheader erstellen/speichern.
30462 :z75DC      ldx  #0
30463 ::1        lda  curDirHead,x
30464          sta  diskBlkBuf,x
30465          inx
30466          cpx  #39
30467          bcc  :1
30468          lda  #$00
30469 ::2        sta  diskBlkBuf,x
30470          inx
30471          bne  :2
30472
30473          lda  v76A1
30474          sta  diskBlkBuf +0
30475          lda  v76A2
30476          sta  diskBlkBuf +1
30477
30478          ldy  #$04
30479          jsr  z7681          ;Name kopieren.
30480
30481          lda  #$a0
30482          sta  diskBlkBuf +$14
30483          sta  diskBlkBuf +$15
30484
30485          ldx  curDirHead +$a2
30486          stx  diskBlkBuf +$16
30487          ldx  curDirHead +$a3
30488          stx  diskBlkBuf +$17
30489          sta  diskBlkBuf +$18
30490          ldx  curDirHead +$a5
30491          stx  diskBlkBuf +$19
30492          ldx  curDirHead +$a6
30493          stx  diskBlkBuf +$1a
30494          sta  diskBlkBuf +$1b
30495          sta  diskBlkBuf +$1c
30496
30497          lda  #$00
30498          sta  diskBlkBuf +$1d
30499          sta  diskBlkBuf +$1e
30500          sta  diskBlkBuf +$1f
30501
30502          lda  v769F

```

```

        sta    diskBlkBuf +$20      30503
        sta    r1L                  30504
        lda    v76A0                30505
        sta    diskBlkBuf +$21      30506
        sta    r1H                  30507
                                   30508
        lda    curDirHead +$20      30509
        sta    diskBlkBuf +$22      30510
        lda    curDirHead +$21      30511
        sta    diskBlkBuf +$23      30512
                                   30513
        lda    v76A3                ;Tr für Verzeichnis. 30514
        sta    diskBlkBuf +$24      30515
        lda    v76A4                ;Se für Verzeichnis. 30516
        sta    diskBlkBuf +$25      30517
        lda    v76A5                ;Offset im Block.    30518
        sta    diskBlkBuf +$26      30519
        jmp    PutBlock_dskBuf      30520
                                   30521
; Name kopieren und mit $a0 auffüllen. 30522
:z7681    ldx    #0                  30523
::1      lda    v76A6,x             30524
        beq    :2                  30525
        sta    diskBlkBuf,y         30526
        iny                     30527
        inx                     30528
        cpx    #16                 30529
        bcc    :1                  30530
        bcs    :4                  30531
::2      lda    #$a0               30532
::3      sta    diskBlkBuf,y        30533
        iny                     30534
        inx                     30535
        cpx    #16                 30536
        bcc    :3                  30537
::4      rts                      30538
                                   30539
:v769F    b $00    ;Tr Verzeichnisheader. 30540
:v76A0    b $00    ;Se Verzeichnisheader. 30541
:v76A1    b $00    ;Tr für ersten Verzeichnisblock. 30542
:v76A2    b $00    ;Se für ersten Verzeichnisblock. 30543
:v76A3    b $00    ;Tr Block / Verzeichnis-Eintrag. 30544
:v76A4    b $00    ;Se Block / Verzeichnis-Eintrag. 30545
:v76A5    b $00    ;Offset im Block.    30546
:v76A6    s 17     ;Zwischensp. Dateiname. 30547
                                   30548
; 1: Verzeichnis löschen. 30549
:N76B7    jsr    S1412              ;Verzeichnis suchen. 30550
        txa                      30551
        bne    :err                30552
                                   30553
; Hinweis: 30554
; Verzeichnis löschen wird auch von 30555
; ":CopyFile" aus aufgerufen. Daher 30556
; dürfen die Register ":r9" bis ":r15" 30557

```

```

30558 ; nicht verändert werden.
30559         lda    r1L                ;Track/Sektor für
30560         sta    r8L                ;Verzeichniseintrag.
30561         lda    r1H
30562         sta    r8H
30563
30564         lda    r5L                ;Adresse in
30565         sta    r7L                ;":diskBlkBuf" für
30566 ;         lda    r5H                ;Verzeichniseintrag.
30567 ;         sta    r7H
30568
30569         jsr    S09BC                ;1.Block > r1L/r1H.
30570         jsr    OpenSubDir          ;Verzeichnis öffnen.
30571
30572 ; Testen ob Verzeichnis leer.
30573         jsr    Get1stDirEntry      ;Erster Eintrag.
30574         txa
30575         bne    :err                ; => Fehler...
30576
30577 ;         ldy    #$00                ;Durch Get1stDirEntry
30578 ::loop    lda    (r5),y            ;Datei vorhanden?
30579         bmi    :found              ; => Ja, Fehler...
30580
30581         jsr    GetNxtDirEntry      ;Nächster Eintrag.
30582         txa
30583         bne    :err                ; => Fehler...
30584         tya
30585         beq    :loop               ; => Weiter suchen...
30586         bne    :ok                 ; => Verzeichniseinde.
30587
30588 ; Fehler: Verzeichnis nicht leer.
30589 ::found    jsr    z776B            ;Verzeichnis zurück.
30590
30591         ldx    #DIR_NOT_EMPTY
30592
30593 ; Fehler ausgeben.
30594 ::err      lda    v71C0            ;Ordn. löschen aktiv?
30595         bne    :1                  ; => Ja, weiter...
30596
30597         jsr    S5CE1                ;Fehler ohne Refresh.
30598
30599 ::1        sec                      ;Fehler.
30600         rts
30601
30602 ; Verzeichnis ist leer.
30603 ::ok      jsr    z776B            ;Verzeichnis zurück.
30604
30605         jsr    S0B30                ;Datei löschen.
30606 ;         txa                      ;Z-Flag ist gesetzt.
30607         bne    :err                ; => Fehler...
30608
30609         lda    r8L                ;Track/Sektor für
30610         sta    r1L                ;Verzeichnisblock.
30611         lda    r8H
30612         sta    r1H

```

```

;                                     30613
;         lda    #< diskBlkBuf        ;Zeiger auf          30614
;         sta    r4L                  ;Zwischenspeicher.    30615
;         lda    #> diskBlkBuf        30616
;         sta    r4H                  30617
;         jsr    GetBlock_dskBuf      30618
;         txa                          30619
;         bne    :err                 ; => Fehler...        30620
;                                     30621
; Verzeichnisblock einlesen.          30622
;         lda    r7L                  30623
;         tay                      ;Offset Eintrag.        30624
;         cmp    #2                   ;Eintrag #0?         30625
;         bne    :2                   ; => Nein, weiter...    30626
;                                     30627
;         clc                          ;Sonderbehandlung.  30628
;         adc    #TDFOLDER_POS0 -1    30629
;                                     30630
::2          tax                      30631
;         dex                          ;Offset auf Eintrag.  30632
;                                     30633
; Eintrag löschen. Damit kann ein    30634
; Unterverzeichnis nicht wieder      30635
; zurückgeholt werden (UNDELETE).    30636
;         lda    #NULL                30637
;         sta    diskBlkBuf,x         ;Ordner-Nr. löschen.  30638
;                                     30639
;         lda    #0                   30640
;         tax                          30641
::3          sta    diskBlkBuf,y       ;Eintrag löschen.    30642
;         iny                          30643
;         inx                          30644
;         cpx    #30                  30645
;         bne    :3                   30646
;                                     30647
;         lda    #< diskBlkBuf        ;Zeiger auf          30648
;         sta    r4L                  ;Zwischenspeicher.    30649
;         lda    #> diskBlkBuf        30650
;         sta    r4H                  30651
;         jsr    PutBlock_dskBuf      30652
;         jsr    PutBlock             30653
;         txa                          30654
;         bne    :err                 ; => Fehler...        30655
;                                     30656
::done       clc                      ;Kein Fehler.          30657
;         rts                          30658
;                                     30659
; Zum Elternverzeichnis wechseln.    30660
:z776B      jsr    GetDirHead          30661
;                                     30662
;         lda    curDirHead +$22      30663
;         sta    r1L                  30664
;         lda    curDirHead +$23      30665
;         sta    r1H                  30666
;         jmp    OpenSubDir           30667

```

```
30668
30669 ; Zwischenspeicher:
30670 ; SIZE_DBUF = 128Byte für Ordner-Liste.
30671 :v777B
30672
30673 ;*** Endadresse testen:
30674         g OS_VARS -SIZE_DBUF
30675
```

```

;
; SourceCode: src.mod#10.s
;
;
;
; Modul-Info:
; 0 - RAM-TopDesk
; 1 - TopDesk.win einlesen
; 2 - TopDesk.win speichern
;
if .p
    t "TopSym"
    t "TopSym.MP3"
    t "TopSym.ROM"
    t "DTopDeskI18N.ext"
    t "src.DTopDesk.ext"

:VMOD_BASE      = $1000 ;Temp. Ladeadresse VLIR-Modul.
:VMOD_SIZE      = $2000 ;Max. Größe VLIR-Modul.
:VMOD_MAX       = $f000 ;Max. Speicher für VLIR-Module.
:BASE_ENTERDT   = $7e00 ;Startadr. EnterDesktop-Routine.
endif

o BASE_MOD_A
n "obj.#10"

::ModEntry_0    jmp    N6EB5 ;RAM-TopDesk ein/aus.
::ModEntry_1    jmp    N70C3 ;TopDesk.win einlesen.
;:ModEntry_2    jmp    N6E00 ;TopDesk.win speichern.

; 2: TopDesk.win speichern.
; TD13: SaveWindowStat
:N6E00          jsr    S38DB ;Fensterdat. sichern.

if PANEL_RIGHT = FALSE
    jsr    z7120 ;Fens.pos. anpassen.
endif

    jsr    S07D3 ;Systemlaufwerk.
    txa
    bne    :err

    lda    #< F3956 ;"TopDesk.win".
    ldx    #> F3956
    jsr    S0B34 ;Datei löschen.
;    txa ;Z-Flag ist gesetzt.
    beq    :1
    cpx    #FILE_NOT_FOUND
    bne    :err

::1            lda    #< V385E
                sta    r9L
                lda    #> V385E
                sta    r9H

```

```

30731      lda    #$00
30732      sta    r10L
30733      jsr    SaveFile          ;Datei speichern.
30734      txa
30735      beq    :ok
30736
30737 ::err      jsr    S5CF3          ;Fehler ausgeben.
30738
30739 ::ok       jsr    S07C0          ;Lfwk. zurücksetzen.
30740      jmp    S4321          ;Alle Fenster neu.
30741
30742 ; Infoblock für TopDesk.win.
30743 :V385E      w    F3956          ;"TopDesk.win".
30744 ::HdrB002  b    3,21          ;Icon-Größe.
30745      b    $bf
30746 ::HdrB005  b    %11111111,%11111111,%11111111
30747      b    %10000000,%00000000,%00000001
30748      b    %10000000,%00000000,%00000001
30749      b    %10111111,%11111101,%11111101
30750      b    %10100000,%00000101,%00000101
30751      b    %10101010,%10100101,%01010101
30752      b    %10100000,%00000101,%00000101
30753      b    %10111111,%11111101,%01010101
30754      b    %10000000,%00000001,%00000101
30755      b    %10111111,%11111101,%01010101
30756      b    %10100000,%00000101,%00000101
30757      b    %10101010,%10100101,%01010101
30758      b    %10100000,%00000101,%00000101
30759      b    %10111111,%11111101,%01010101
30760      b    %10000000,%00000001,%00000101
30761      b    %10111111,%11111101,%01010101
30762      b    %10100000,%00000101,%00000101
30763      b    %10111111,%11111101,%11111101
30764      b    %10000000,%00000000,%00000001
30765      b    %10000000,%00000000,%00000001
30766      b    %11111111,%11111111,%11111111
30767 ::HdrB068  b    %10000000 ! PRG
30768 ::HdrB069  b    DATA
30769 ::HdrB070  b    SEQUENTIAL
30770
30771 ::HdrB071  w    BASE_TDWIN
30772      w    BASE_TDWIN +SIZE_TDWIN
30773      w    $0000
30774
30775 ::HdrB077  b    "TopDesk.winV5.0"
30776      e    :HdrB077 +16
30777      b    NULL
30778
30779 ::HdrB095  b    $00
30780 ::HdrB096  b    $40 ;Bildschirmflag.
30781
30782 ::HdrB097  b    "TD64 - Friends"
30783      e    :HdrB097 +19
30784      b    NULL
30785

```



: :HdrB117	b NULL		30786
	e :HdrB117 +16		30787
	b NULL		30788
			30789
:F3956	b "TopDesk.win",0		30790
			30791
; Speicherbelegung RAM-TopDesk:			30792
; \$0000-\$00FF = VLIR-Daten.			30793
; \$0100-\$02FF = Backup EnterDeskTop			30794
; \$0300-\$03FF = Systemkonfiguration			30795
; \$0400-\$07FF = TD.WIN			30796
; \$0800-\$0FFF = Frei			30797
; \$1000-\$F000 = VLIR-Module			30798
; \$F000-\$FA3F = Fensterdaten #1/#2 oder #3/#4			30799
; \$FA40-\$FFFF = Frei			30800
			30801
; 0: RAM-TopDesk ein/aus.			30802
:N6EB5	jsr SetADDR_EnterDT		30803
	jsr FetchRAM		30804
			30805
	ldy #RAMDTCHKLOW	;RAM-TopDesk aktiv?	30806
: :1	lda (r0),y		30807
	cmp newEnterDT,y		30808
	bne x6F10	; => Nein, weiter...	30809
	iny		30810
	cpy #RAMDTCHKHIGH		30811
	bne :1		30812
			30813
; Hinweis:			30814
; Aktive Speicherbank in der Routine			30815
; EnterDeskTop für die De-Installation			30816
; von RAM-TopDesk verwenden.			30817
	ldy #RAMDTPOFF +6		30818
	lda (r0),y		30819
	sta C0427	;Akt.Bank RAM-TD.	30820
			30821
; TODO:			30822
; Befehl ist doppelt, kann entfallen.			30823
	jsr SetADDR_EnterDT		30824
			30825
	lda #< DACC_DTBAK	;EnterDT wieder	30826
	sta r1L	;herstellen.	30827
	lda #> DACC_DTBAK		30828
	sta r1H		30829
			30830
	lda C0427	;64K-Bank RAM-TD.	30831
	sta r3L		30832
			30833
	jsr FetchRAM		30834
			30835
	jsr SetADDR_EnterDT		30836
	jsr StashRAM		30837
			30838
	jsr z6EE2	;RAM-TD abschalten.	30839
			30840

```

30841          jsr   ResetScreen          ;Grafik löschen.
30842
30843          jmp   EnterDeskTop          ;Zum DeskTop.
30844
30845 ; RAM-TopDesk abschalten.
30846 :z6EE2      lda   C0427              ;64K-Bank RAM-TD.
30847          jsr   z7225              ;64K-Bank freigeben.
30848
30849 :z6EE8      lda   #$00
30850          sta   C0428              ;Kein RAM-TopDesk.
30851          sta   C0427              ;64K-Bank RAM-TD.
30852          sta   v708F              ;64K-Bank EnterDT.
30853          sta   C0429              ;Kein System in REU.
30854
30855          jsr   S09DD              ;Konfig in REU/Datei.
30856
30857          lda   C0424              ;TopDesk/Laufwerk.
30858          jsr   SetDevice
30859
30860          jsr   OpenDisk            ;Disk öffnen.
30861          txa
30862          bne   :exit              ; => Fehler, Ende...
30863
30864          ldx   C050A              ;TopDesk/Partition.
30865          beq   :exit              ; => Nicht definiert.
30866
30867          jsr   S0723              ;Partition öffnen.
30868
30869 ::exit      rts
30870
30871 ; RAM-TopDesk installieren.
30872 :x6F10      lda   C0428              ;RAM-TopDesk?
30873          beq   :11                ; => Nein, weiter...
30874          rts
30875
30876 ::11       jsr   z7237              ;64K-Bank suchen.
30877          tax
30878          bne   :12
30879
30880          jmp   z6EE8              ;RAM-TD abschalten.
30881
30882 ::12       sta   v708F              ;64K-Bank EnterDT.
30883          sta   C0427              ;64K-Bank RAM-TD.
30884
30885          lda   #$01
30886          sta   C0428              ;RAM-TopDesk aktiv.
30887          sta   C0429              ;System in REU.
30888
30889          jsr   S09DD              ;Konfig in REU/Datei.
30890
30891          lda   curDrive            ;Aktuelles Laufwerk
30892          sta   V07D4              ;zwischenspeichern.
30893
30894          jsr   S07D6              ;Systemlaufwerk.
30895

```

; Speicherdaten initialisieren.			30896
	lda	#< APP_RAM	30897
	sta	r0L	30898
	lda	#> APP_RAM	30899
	sta	r0H	30900
			30901
	lda	#< DACC_CORE	30902
	sta	r1L	30903
	sta	a0L	30904
	lda	#> DACC_CORE	30905
	sta	r1H	30906
	sta	a0H	30907
			30908
	lda	#< BASE_MOD_A - APP_RAM	30909
	sta	r2L	30910
	sta	v708D +0	30911
	lda	#> BASE_MOD_A - APP_RAM	30912
	sta	r2H	30913
	sta	v708D +1	30914
			30915
	lda	C0427 ;64K-Bank RAM-TD.	30916
	sta	r3L	30917
			30918
	jsr	StashRAM ;Hauptmod. speichern.	30919
			30920
	lda	r2L ;Zeiger auf Bereich	30921
	clc	;für VLIR-Module.	30922
	adc	a0L	30923
	sta	a0L	30924
	lda	r2H	30925
	adc	a0H	30926
	sta	a0H	30927
			30928
; Speicher freigeben.			30929
	jsr	z7090 ;REU-MoveData aus.	30930
	jsr	z70A4 ;APP_RAM zwischensp.	30931
			30932
; VLIR-Datei öffnen.			30933
	lda	#< C0400	30934
	sta	r0L	30935
	lda	#> C0400	30936
	sta	r0H	30937
	jsr	OpenRecordFile	30938
	txa		30939
	bne	:err	30940
			30941
	lda	#1 ;Hauptmodul #0 ist	30942
	jsr	PointRecord ;bereits im Speicher.	30943
			30944
	lda	#\$00 ;Zeiger auf	30945
	sta	a1L ;VLIR-Tabelle.	30946
			30947
; VLIR-Module laden und in RAM speichern.			30948
::loop	lda	#< VMOD_BASE	30949
	sta	r7L	30950

```

30951      lda    #> VMOD_BASE
30952      sta    r7H
30953
30954      lda    #< VMOD_SIZE
30955      sta    r2L
30956      lda    #> VMOD_SIZE
30957      sta    r2H
30958      jsr    ReadRecord
30959      txa
30960      bne    :err
30961
30962      lda    r7L                ;Größe VLIR-Modul
30963      sec                ;berechnen.
30964      sbc    #< VMOD_BASE
30965      sta    r7L
30966      lda    r7H
30967      sbc    #> VMOD_BASE
30968      sta    r7H
30969
30970      lda    #< VMOD_BASE
30971      sta    r0L
30972      lda    #> VMOD_BASE
30973      sta    r0H
30974
30975      lda    a0L                ;Startadresse in RAM.
30976      sta    r1L
30977      lda    a0H
30978      sta    r1H
30979
30980      lda    r7L                ;Größe VLIR-Modul.
30981      sta    r2L
30982      lda    r7H
30983      sta    r2H
30984
30985      lda    C0427                ;64K-Bank RAM-TD.
30986      sta    r3L
30987
30988      jsr    StashRAM                ;VLIR-Mod. speichern.
30989
30990      jsr    z7011                ;VLIR-Info speichern.
30991
30992      lda    a0L                ;Statadresse in RAM
30993      clc                ;anpassen.
30994      adc    r7L
30995      sta    a0L
30996      lda    a0H
30997      adc    r7H
30998      sta    a0H
30999
31000      cmp    #> VMOD_MAX                ;Speicher voll?
31001      bcs    :buf_full                ; => Ja, Abbruch...
31002
31003      jsr    NextRecord                ;Nächstes VLIR-Modul.
31004      txa
31005      beq    :21

```

			31006
	cpx	#INV_RECORD	31007
	beq	:ok	31008
	bne	:err ; => Fehler...	31009
			31010
::21	lda	curRecord ;Max.Mod. eingelesen?	31011
	cmp	#TD_MAX_VLIR	31012
	bcc	:loop ; => Weiter...	31013
			31014
	lda	#OUT_OF_RECORDS	31015
	b	\$2c	31016
::ok	lda	#NO_ERROR	31017
	b	\$2c	31018
::buf_full	lda	#BFR_OVERFLOW	31019
::err	pha		31020
	tax	;Fehler?	31021
	bne	:31 ; => Ja, weiter...	31022
			31023
	jsr	z7011 ;Endadr. speichern.	31024
			31025
::31	jsr	z70A4 ;APP_RAM zurückholen.	31026
	jsr	z709C ;Reset REU-MoveData.	31027
			31028
	jsr	S07C0 ;Lfwk. zurücksetzen.	31029
			31030
	pla	;Fehler?	31031
	tax		31032
	beq	:32 ; => Nein, weiter...	31033
			31034
	jsr	S5CE1 ;Fehler ohne Refresh.	31035
	jmp	z6EE2 ;RAM-TD abschalten.	31036
			31037
::32	ldx	#(r2H - r0L)	31038
::33	lda	:data,x	31039
	sta	r0,x	31040
	dex		31041
	bpl	:33	31042
			31043
	lda	C0427 ;64K-Bank RAM-TD.	31044
	sta	r3L	31045
			31046
	jsr	StashRAM ;VLIR-Info speichern.	31047
			31048
; Original EnterDeskTop-Routine speichern.			31049
	jsr	SetADDR_EnterDT	31050
	jsr	FetchRAM	31051
			31052
	lda	#< DACC_DTBAK	31053
	sta	r1L	31054
	lda	#> DACC_DTBAK	31055
	sta	r1H	31056
			31057
	lda	C0427 ;64K-Bank RAM-TD.	31058
	sta	r3L	31059
			31060

```

31061          jsr    StashRAM
31062
31063 ; EnterDeskTop-Routine anpassen.
31064          jsr    SetADDR_EnterDT
31065
31066          lda    #< newEnterDT
31067          sta    r0L
31068          lda    #> newEnterDT
31069          sta    r0H
31070
31071          jmp    StashRAM
31072
31073 ::data          w v7022
31074                w DACC_VLIRDAT
31075                w SIZE_VLIRDAT
31076
31077 ; EnterDeskTop-Routine für TopDesk.
31078 ; Liegt ab $7E00 im Speicher.
31079 :newEnterDT     sei
31080                cld
31081                ldx    #$ff
31082                stx    firstBoot
31083                txs
31084                jsr    GEOS_InitSystem
31085                jsr    ResetScreen
31086
31087 ;--- Bereich für EnterDT-Überprüfung.
31088 :chkDTLow        ldx    #(r3L - r0L)
31089 ::1             lda    BASE_ENTERDT + RAMDTPOFF,x
31090                sta    r0,x
31091                dex
31092                bpl    :1
31093 :chkDTHigh
31094 ;---
31095
31096          jsr    FetchRAM
31097
31098          lda    #< MAININIT          ; => L3136.
31099          sta    r7L
31100          lda    #> MAININIT
31101          sta    r7H
31102          jmp    StartAppl
31103
31104 :RAMDTPDAT       w APP_RAM
31105                w DACC_CORE
31106 :v708D          w $0000 ;Größe Hauptmodul.
31107 :v708F          b $01   ;64K-Bank RAM-TD.
31108
31109 :RAMDTPOFF       = RAMDTPDAT - newEnterDT
31110
31111 :RAMDTCHKLOW     = (chkDTLow - newEnterDT)
31112 :RAMDTCHKHIGH    = (chkDTHigh - newEnterDT)
31113 ;---
31114
31115 ; VLIR-Info in Tabelle speichern.

```

```

; Word #0-#11: Startadr. Modul in DACC. 31116
; Word #12: Endadr. letztes Modul. 31117
; 31118
; Hinweis: 31119
; Mit diesen Angaben wird die Start- 31120
; Adresse und die Größe des VLIR-Moduls 31121
; ermittelt. 31122
:z7011 ldy a1L 31123
lda a0L 31124
sta v7022,y 31125
iny 31126
lda a0H 31127
sta v7022,y 31128
iny 31129
sty a1L 31130
rts 31131
31132
:v7022 s SIZE_VLIRDAT 31133
31134
; REU-MoveData temp. abschalten. 31135
:z7090 lda sysRAMFlg ;REU-MoveData 31136
sta v70A3 ;abschalten. 31137
and #%01111111 31138
sta sysRAMFlg 31139
rts 31140
31141
; REU-MoveData zurücksetzen. 31142
:z709C lda v70A3 ;REU-MoveData 31143
sta sysRAMFlg ;zurücksetzen. 31144
rts 31145
31146
:v70A3 b $00 ;Zwischenspeicher sysRAMFlg. 31147
31148
; Speicher temp. in Bank#0 sichern. 31149
:z70A4 ldx #(r3L - r0L) 31150
::1 lda :data,x 31151
sta r0,x 31152
dex 31153
bpl :1 31154
31155
jmp SwapRAM 31156
31157
::data w VMOD_BASE ;RAM-Speicher. 31158
w $0000 ;Bereich für REU-MoveData. 31159
w VMOD_SIZE ;Max. Größe VLIR-Modul. 31160
b $00 ;GEOS-Bank #0. 31161
31162
; 64K-Bank freigeben. 31163
:z7225 tax 31164
lsr 31165
lsr 31166
tay 31167
txa 31168
and #$03 31169
tax 31170

```

```

31171          lda    RamBankInUse,y
31172          and    v734B,x
31173          sta    RamBankInUse,y
31174          rts
31175
31176 ; Freie Speicherbank suchen.
31177 :z7237          lda    #$00
31178          sta    v7346
31179
31180          tay
31181 :x723D          lda    #$03
31182          sta    r0L
31183          lda    RamBankInUse,y
31184 :x7244          asl
31185          bcc    x726B
31186          asl
31187          bcs    x7258
31188
31189 :x724A          inc    v7346
31190          dec    r0L
31191          bpl    x7244
31192
31193          iny                                ;Max. GEOS-DACC-
31194          cpy    #(4096 /64 /4)             ;Bytes in Tabelle.
31195          bne    x723D
31196
31197 :x7258          ldx    v7346                 ;Bank#0 übergehen.
31198          beq    x724A
31199
31200          lda    #< d727D                   ;Kein Speicher frei.
31201          ldx    #> d727D
31202          jsr    S2090                       ;DBox ohne Recover.
31203
31204          lda    #$00                       ;Kein Speicher frei.
31205          rts
31206
31207 ; 64K-Bank reservieren.
31208 :x726B          asl
31209          bcs    x724A
31210
31211          lda    RamBankInUse,y
31212          ldx    r0L
31213          ora    v7347,x
31214          sta    RamBankInUse,y
31215
31216          lda    v7346
31217          rts
31218
31219 ; Dialogbox: Kein Speicher frei.
31220 :d727D          b    %10000001
31221
31222          b    DBTXTSTR,$0a,$10
31223          w    :t1
31224
31225          b    DBTXTSTR,$0a,$20

```



```

                                w :t2                                31226
                                b DBTXTSTR,$0a,$2a                31227
                                w :t3                                31228
if LANG = LANG_DE                                31229
                                b DBTXTSTR,$0a,$34                31230
                                w :t4                                31231
endif                                              31232
                                b DBTXTSTR,$0a,$44                31233
                                w :t5                                31234
                                b OK      , $11,$4b                31235
                                b NULL                                31236
                                b OK      , $11,$4b                31237
                                b NULL                                31238
                                b OK      , $11,$4b                31239
if LANG = LANG_DE                                31240
::t1                                b BOLDON                        31241
                                b "RAM-TopDesk Installation",0    31242
                                b "RAM-TopDesk Installation",0    31243
::t2                                b PLAINTEXT                    31244
                                b "Es ist keine 64K-Speicherbank frei",0 31245
::t3                                b "in welcher RAM-TopDesk installiert",0 31246
::t4                                b "werden könnte.",0          31247
::t5                                b "Bitte GEOS-Konfiguration anpassen.",0 31248
endif                                              31249
if LANG = LANG_EN                                31250
::t1                                b BOLDON                        31251
                                b "Install RAM-TopDesk",0          31252
                                b "Install RAM-TopDesk",0          31253
::t2                                b PLAINTEXT                    31254
                                b "There is no free 64Kb memory bank in",0 31255
::t3                                b "which RAM-TopDesk could be installed.",0 31256
::t5                                b "You must edit the GEOS configuration.",0 31257
endif                                              31258
                                b "You must edit the GEOS configuration.",0 31259
:v7346                                b $00                        ;Bankzeiger. 31260
                                b $00                        ;Bankzeiger. 31261
:v7347                                b %000000011                31262
                                b %00001100                    31263
                                b %00110000                    31264
                                b %11000000                    31265
                                b %11000000                    31266
:v734B                                b %00111111                31267
                                b %11001111                    31268
                                b %11110011                    31269
                                b %11111100                    31270
                                b %11111100                    31271
; Speicherübersicht TopDesk.win $7A00-$7DF2:    31272
; (siehe auch src.mod#00b!)                    31273
;--- TD.WIN                                    31274
; $7A00 - $7AFF  Ordnerliste #1-#4                31275
; $7B40 - $7B43  Fenster-Status                    31276
; $7B44 - $7B49  Größe Fenster #1                  31277
; $7B4A - $7B4F  Größe Fenster #2                  31278
; $7B50 - $7B55  Größe Fenster #3                  31279
; $7B56 - $7B5B  Größe Fenster #4                  31280

```

```

31281 ; $7B5C - $7B5F Fensterstapel
31282 ; $7C00 - $7D3F Fenstertitel #1-#4
31283 ; 4 x 80Byte = $0140.
31284 ; $7DA0 - $7DAB Aktiver Ordner und
31285 ; 8Byte-Ordnerliste.
31286 ; $7DB0 - $7DB3 Fensterposition.
31287 ; $7DB4 - $7DBB Lwfk./Part. Fenster #1-#4.
31288 ; $7DF1 Anzeige Block/Kb.
31289 ; $7DF2 Icon-/Text-Modus.
31290 ;---
31291
31292 ; 1: TopDesk.win einlesen.
31293 :N70C3 lda C0429 ;System in REU?
31294 beq :file ; => Nein, weiter...
31295
31296 ; TopDesk.win aus REU einlesen.
31297 ::reu ldx #(r2H - r0L)
31298 ::1 lda :data,x
31299 sta r0,x
31300 dex
31301 bpl :1
31302
31303 lda C0427 ;64K-Bank RAM-TD.
31304 sta r3L
31305 jsr FetchRAM
31306
31307 jmp z710D ;TD.WIN übernehmen.
31308
31309 ::data w BASE_TDWIN ;Zwischenspeicher.
31310 w DACC_TDWIN ;Adresse TD.WIN/REU.
31311 w SIZE_TDWIN ;Max. Größe TD.WIN.
31312
31313 ; TopDesk.win aus Datei einlesen.
31314 ::file lda C0424 ;TopDesk/Laufwerk.
31315 jsr SetDevice
31316 jsr OpenDisk
31317 txa
31318 bne :err ; => Fehler...
31319
31320 lda #< F3956 ;"TopDesk.win".
31321 sta r6L
31322 lda #> F3956
31323 sta r6H
31324 jsr FindFile ;Datei suchen.
31325 txa
31326 beq :read ; => Gefunden...
31327
31328 ::err lda #$00
31329 jsr S37B2 ;Anzeige: Icons.
31330
31331 lda C0424 ;TopDesk/Laufwerk.
31332 jmp S3AB0 ;Laufwerk öffnen.
31333
31334 ; TopDesk.win einlesen.
31335 ::read lda #< BASE_TDWIN ;Zeiger auf

```

```

        sta    r7L                ;Zwischenspeicher.      31336
        lda    #> BASE_TDWIN      31337
        sta    r7H                31338
                                   31339
        lda    #< SIZE_TDWIN      ;Max. Größe TD.WIN.    31340
        sta    r2L                31341
        lda    #> SIZE_TDWIN      31342
        sta    r2H                31343
                                   31344
        jsr    S09BC                ;1.Block > r1L/r1H.    31345
                                   31346
        jsr    ReadFile            ;TD.WIN einlesen.      31347
        txa                        31348
        bne    :err                31349
                                   31350
if PANEL_RIGHT = FALSE            31351
        jsr    z7110                ;Fens.pos. anpassen.  31352
endif                              31353
                                   31354
; Einstellungen aus TD.WIN übernehmen. 31355
:z710D        jsr    z7130          ;Fens.daten kopieren. 31356
                                   31357
; Fenster aktualisieren.              31358
        lda    #4                  ;Max. 4 Fenster.      31359
        sta    v7204                ;Stack-Pointer.      31360
                                   31361
; Fenster-Daten übernehmen.          31362
:z718F        dec    v7204          ;Fenster #1 bis #4?  31363
        bpl    :2                  ; => Ja, weiter...     31364
                                   31365
; Hinweis:                          31366
; TopDesk64 testet auf "System in REU", 31367
; der Wert wird aber vor dem Aufruf auf 31368
; $00 gesetzt. Daher startet TopDesk64 31369
; immer nur mit zwei Fenstern.       31370
;         lda    C0429              ;System in REU?      31371
;         beq    :1                  ; => Nein, weiter...  31372
;         lda    C0428              ;RAM-TopDesk?        31373
        bne    :1                  ; => Ja, weiter...     31374
                                   31375
; Hinweis:                          31376
; Ohne RAM-TopDesk max. zwei Fenster  31377
; möglich, #3/#4 ggf. schließen.     31378
;                                     31379
; TODO:                             31380
; Die restlichen Fenster-Eigenschaften 31381
; werden hier aber nicht gelöscht.    31382
        lda    #$00                31383
        ldx    C042A +2            ;Fenster-Nr. #2/#3    31384
        sta    C0448,x            ;schließen.           31385
        ldx    C042A +3            31386
        sta    C0448,x            31387
                                   31388
::1          jmp    S4320          ;Prefs. und Fens.neu.  31389
                                   31390

```

```

31391 ::2          ldx    v7204          ;Stack-Pointer.
31392          lda    C042A,x          ;Fenster-Nr. holen.
31393          tax
31394
31395          lda    C0448,x          ;Fenster geöffnet?
31396          beq    :5              ; => Nein, weiter...
31397
31398          ldy    C04F8,x          ;Lfwk. für Fenster.
31399          lda    RealDrvMode -8,y
31400 ;          and    #SET_MODE_PARTITION
31401 ;          beq    :2a          ; => Kein CMD-Lfwk.
31402          bpl    :2a          ; => Kein CMD-Lfwk.
31403
31404          lda    C04FC,x          ;Partition gesetzt?
31405          bne    :2a          ; => Ja, weiter...
31406
31407          sta    C0448,x          ;Fenster schließen.
31408          beq    :5              ; => Nächstes Fens.
31409
31410 ::2a          lda    C042A          ;Oberstes Fenster.
31411          pha
31412
31413          stx    C042A          ;Oberstes Fenster.
31414
31415 ;          lda    C04F8,x          ;Lfwk. für Fenster.
31416          tya
31417          jsr    S06AB          ;Laufwerk öffnen.
31418
31419          pla
31420          sta    C042A          ;Oberstes Fenster.
31421
31422          txa
31423          bne    :6              ; => Fehler, Ende...
31424
31425          ldx    v7204          ;Stack-Pointer.
31426          lda    C042A,x          ;Fenster-Nr. holen.
31427          jsr    S5979          ;Zeiger Fenstertitel.
31428
31429          lda    #< 2          ;Zeiger auf Diskname
31430          clc                  ;hinter "x:".
31431          adc    r1L
31432          sta    r1L
31433          lda    #> 2
31434          adc    r1H
31435          sta    r1H
31436
31437          ldy    #0              ;Gleiche Disk?
31438 ::3          lda    (r5),y
31439          cmp    (r1),y
31440          bne    :4
31441          iny
31442          cpy    #16
31443          bne    :3
31444          beq    :5              ; => Ja, weiter...
31445

```

::4	cmp	#\$a0	;Ende Diskname?	31446
	bne	:6	; => Nein, weiter...	31447
	lda	(r1),y		31448
	cmp	#TDSPLIT	;Beginn Ordner?	31449
	beq	:5	; => Ja, weiter...	31450
				31451
::6	ldy	v7204	;Stack-Pointer.	31452
	ldx	C042A,y	;Fenster-Nr. holen.	31453
	lda	#\$00	;Kein Ordner aktiv.	31454
	sta	C051B,x	;Aktive Ordner-Nr.	31455
				31456
	jsr	S598F	;Zeiger Ordnerliste.	31457
	jsr	S28A7	;Ordnerliste löschen.	31458
				31459
::5	jmp	z718F	; => Nächstes Fens.	31460
				31461
:v7204	b	\$00	;Stack-Pointer.	31462
				31463
			; Fensterdaten einlesen.	31464
:z7130	ldy	#0		31465
::1	ldx	#0		31466
::2	lda	tdwin_data,y		31467
	sta	r0,x		31468
	iny			31469
	inx			31470
	cpx	#(r2H - r0L) +1		31471
	bcc	:2		31472
				31473
	tya			31474
	pha			31475
	jsr	MoveData		31476
	pla			31477
	tay			31478
				31479
	cpy	#tdwin_data_s		31480
	bcc	:1		31481
				31482
			; Menü-Optionen definieren.	31483
	lda	TDWIN_SIZEMODE		31484
	sta	MT3E11		31485
				31486
	cmp	#""	;KB/Blocks wechseln.	31487
	bne	:3		31488
				31489
	lda	#" "		31490
	b	\$2c		31491
::3	lda	#""		31492
	sta	MT3E05		31493
				31494
	lda	TDWIN_VIEWMODE		31495
	sta	C04B2	;Icon-/Text-Modus.	31496
				31497
	jsr	S3E48	;Modus festlegen.	31498
				31499
	rts			31500

```

31501
31502 ; Daten für TOPDESK.WIN.
31503 :tdwin_data
31504 ::00          w TDWIN_SUBDIR          ;Ordner-Liste.
31505              w C624F
31506              w 4 * MAXDIRSIZE
31507
31508 ::01          w TDWIN_WINDOWS          ;Fensterstatus.
31509              w C0448
31510              w 4
31511
31512 ::02          w TDWIN_WSTACK           ;Fensterstapel.
31513              w C042A
31514              w 4
31515
31516 ::03          w TDWIN_WSIZE1           ;Größe Fenster #1.
31517              w C04BC
31518              w 6
31519
31520 ::04          w TDWIN_WSIZE2           ;Größe Fenster #2.
31521              w C04C7
31522              w 6
31523
31524 ::05          w TDWIN_WSIZE3           ;Größe Fenster #3.
31525              w C04D2
31526              w 6
31527
31528 ::06          w TDWIN_WSIZE4           ;Größe Fenster #4.
31529              w C04DD
31530              w 6
31531
31532 ::07          w TDWIN_TITLE
31533              w START_TITLE_DATA        ;Fenstertitel #1-#4.
31534              w SIZE_TITLE_DATA
31535
31536 ; TODO:
31537 ; Warum wird hier die temp.Ordnerliste
31538 ; in der TD.WIN gespeichert?
31539 ; Die temporäre Ordnerliste wird nur
31540 ; in mod.#09 beim anlegen eines neuen
31541 ; Ordners verwendet.
31542 ::08          w TDWIN_SDINFO           ;Aktive Ordner-Nr.
31543              w C051B
31544              w 4+8 ;Inkl. C051F!
31545
31546 ::09          w TDWIN_WINPOS           ;Fensterposition.
31547              w C050B
31548              w 4
31549
31550 ::10          w TDWIN_DRVPART           ;Lfwk./Part. für
31551              w C04F8                   ;Fenster #1-#4.
31552              w 4+4
31553 :tdwin_data_e
31554 :tdwin_data_s = ( tdwin_data_e - tdwin_data )
31555

```

```

; Hinweis: 31556
; Ist das Panel links, dann Fenster- 31557
; Positionen korrigieren. 31558
if PANEL_RIGHT = FALSE 31559
31560
; Fenster-Pos. an TD v4/v5 anpassen. 31561
:z7120      ldx    #0 31562
::1        lda    TDWIN_WSIZE1 +2,x 31563
          sec 31564
          sbc    #< PANEL_XW 31565
          sta    TDWIN_WSIZE1 +2,x 31566
          lda    TDWIN_WSIZE1 +3,x 31567
          sbc    #> PANEL_XW 31568
          sta    TDWIN_WSIZE1 +3,x 31569
31570
          lda    TDWIN_WSIZE1 +4,x 31571
          sec 31572
          sbc    #< PANEL_XW 31573
          sta    TDWIN_WSIZE1 +4,x 31574
          lda    TDWIN_WSIZE1 +5,x 31575
          sbc    #> PANEL_XW 31576
          sta    TDWIN_WSIZE1 +5,x 31577
31578
          inx 31579
          inx 31580
          inx 31581
          inx 31582
          inx 31583
          inx 31584
          cpx    #4 *6 ;Max. 4 Fenster. 31585
          bcc    :1 31586
          rts 31587
31588
; Fenster-Pos. an Panel/links anpassen. 31589
:z7110      ldx    #0 31590
::1        lda    TDWIN_WSIZE1 +2,x 31591
          clc 31592
          adc    #< PANEL_XW 31593
          sta    TDWIN_WSIZE1 +2,x 31594
          lda    TDWIN_WSIZE1 +3,x 31595
          adc    #> PANEL_XW 31596
          sta    TDWIN_WSIZE1 +3,x 31597
31598
          lda    TDWIN_WSIZE1 +4,x 31599
          clc 31600
          adc    #< PANEL_XW 31601
          sta    TDWIN_WSIZE1 +4,x 31602
          lda    TDWIN_WSIZE1 +5,x 31603
          adc    #> PANEL_XW 31604
          sta    TDWIN_WSIZE1 +5,x 31605
31606
          inx 31607
          inx 31608
          inx 31609
          inx 31610

```

```
31611      inx
31612      inx
31613      cpx   #4 *6           ;Max. 4 Fenster.
31614      bcc   :1
31615      rts
31616 endif
31617
31618 ; Endadresse testen.
31619 ;           g BASE_ENTERDT           ;RAM-TopDesk.
31620           g BASE_TDWIN           ;TopDesk-Config.
31621
```



```

; 31622
; 31623
; SourceCode: src.mod#11.s 31624
; 31625
; 31626
; 31627
; Modul-Info: 31628
; 0 - Neue Partition wählen. 31629
; 1 - Partition für Fenster öffnen. 31630
; 2 - Aktive Partition für CMD-Laufwerk suchen. 31631
; 31632
if .p 31633
    t "TopSym" 31634
    t "TopSym.MP3" 31635
    t "TopSym.ROM" 31636
    t "DTopDeskI18N.ext" 31637
    t "src.DTopDesk.ext" 31638
    31639
; Debug: 64Net: 31640
; Um den Programmablauf für 64Net zu 31641
; testen auf "TRUE" setzen. 31642
; Laufwerk C: wird dann als 64Net-Lfwk. 31643
; behandelt und es werden alle Fehler- 31644
; meldungen angezeigt. 31645
; Format Laufwerk C: = RL81! 31646
:DEBUG_64NET = FALSE 31647
    31648
; Größe Dialogbox: 31649
:DBOX_X0 = $0040 31650
:DBOX_XW = $00c0 31651
:DBOX_X1 = DBOX_X0 +DBOX_XW -1 31652
:DBOX_Y0 = $20 31653
:DBOX_YH = $70 31654
:DBOX_Y1 = DBOX_Y0 +DBOX_YH -1 31655
    31656
; Größe Dateifenster. 31657
:FBOX_X0 = DBOX_X0 +$08 31658
:FBOX_XW = $0070 31659
:FBOX_X1 = FBOX_X0 +FBOX_XW -1 31660
:FBOX_Y0 = DBOX_Y0 +$08 31661
:FBOX_YH = $48 31662
:FBOX_Y1 = FBOX_Y0 +FBOX_YH -1 31663
:ENTRY_YH = 12 31664
:FBOX_FILES = FBOX_YH / ENTRY_YH 31665
    31666
; Größe Eingabefeld. 31667
:PBOX_Y0 = DBOX_Y0 +$38 31668
:PBOX_YH = $10 31669
:PBOX_Y1 = PBOX_Y0 +PBOX_YH -1 31670
:PBOX_X0 = DBOX_X0 +$08 +FBOX_XW +$10 31671
:PBOX_XW = $30 31672
:PBOX_X1 = PBOX_X0 +PBOX_XW -1 31673
    31674
; Position Navigation: 31675
:POSNAV_X = DBOX_X0+$10 31676

```

```

31677 :POSNAV_Y          = DBOX_Y1 -$08 -$10 +1
31678
31679 ; Zwischenspeicher:
31680 ; Liegt bei TopDesk immer ab $3000 im
31681 ; Speicher, bei DTopDesk ab MAININIT.
31682 ; Bis zum Beginn des Modulspeichers
31683 ; sind damit ca. $3D00 Bytes frei.
31684 ;
31685 ; Tabelle mit Part.-Namen.
31686 ;BASEDATA          = $3000 ;255 x 17 Zeichen.
31687 :BASEDATA          = MAININIT
31688
31689 ; Zwischenspeicher CMD-Partitionsdaten:
31690 ;v40EF              = $40ef ;Nur 31 Bytes erforderlich.
31691 :v40EF              = BASEDATA + 255*17
31692
31693 ; Tabelle mit Partitions-Nr.
31694 ;v4110              = $4110 ;Tabelle mit Part.-Nr.
31695 :v4110              = v40EF +33
31696
31697 ; Symbole für 64Net:
31698 ;v4210              = $4210 ;Speicher für 64Net-Datenblock.
31699 :v4210              = v4110 +256
31700 ;v4310              = $4310 ;Speicher Laufwerkstreiber.
31701 :v4310              = v4210 +256
31702
31703 :END_DATA           = v4310 +14*256
31704
31705 :SIZEDATA           = END_DATA - BASEDATA
31706 endif
31707
31708                 o BASE_MOD_A
31709                 n "obj.#11"
31710
31711 ::ModEntry_0        jmp    N6F21 ;Neue Partition wählen.
31712 ::ModEntry_1        jmp    N7BA7 ;Partition öffnen.
31713 ;:ModEntry_2        jmp    N6EB5 ;Aktive Partition suchen.
31714
31715 ; 2: Aktive Partition für CMD-Laufwerk suchen.
31716 :N6EB5
31717
31718 ; Hinweis:
31719 ; Das Modul überschreibt hier keinen
31720 ; Programmcode, das Modul muss daher
31721 ; nicht immer neu geladen werden.
31722 ;                 lda    #$00                ;Kein VLIR-Modul mehr
31723 ;                 sta    V0422                ;im Speicher!
31724
31725                 ldy     curDrive
31726                 lda     V04B8 -8,y            ;Geräteinfo einlesen.
31727                 bpl     :1                    ; => Kein 64Net
31728
31729 ; 64Net-Laufwerk.
31730 ::64net             lda     #$00                ;Keine Partition.
31731                     beq     :save              ; => Ende.

```

```

; CMD-Laufwerk (1541/71/81 bereits ausgeschlossen).
::1      jsr    z6F7B      ;Speicher auslagern.
        ldy    curDrive
        lda    V04B8 -8,y  ;Geräteinfo einlesen.
        and    #%00000011  ;RAMLink/RAMDrive?
        beq    :fd_hd      ; => Nein, weiter...
; CMD-RL/RD.
::rl_rd   lda    #$00      ;Zeiger auf erste
        sta    r15L        ;Partition.
::11      inc    r15L
        lda    r15L
        cmp    #32         ;Part. durchsucht?
        beq    :err        ; => Ja, Fehler.
        jsr    z733C      ;Partitionsdaten.
        txa
        bne    :err        ; => Fehler, Abbruch.
        ldy    curDrive
        lda    ramBase -8,y
        cmp    v40EF +20   ;Part.Adr.MiddleByte.
        bne    :11        ; => Nein, weiter...
        lda    r15L        ;Partition gefunden.
        bne    :done
; CMD-FD/HD.
::fd_hd   lda    #$ff      ;Aktive Partition.
        jsr    z733C      ;Partitionsdaten.
        txa
        bne    :err
        lda    v40EF +2    ;Partitions-Nr.
        bne    :done
; Partition nicht gefunden.
::err     jsr    z6F7C      ;Speicher einlesen.
        ldx    #PART_ERR   ;Partitionsfehler.
        rts
; CMD-Partition gefunden.
::done    pha
        jsr    z6F7C      ;Speicher einlesen.
        pla
; 64Net-/CMD-Partition speichern.
::save    ldx    V3A94
        sta    C04FC,x     ;Part. für Fenster.
        lda    C042A       ;Oberstes Fenster.

```

```

31787          pha
31788
31789          stx    C042A          ;Oberstes Fenster.
31790          lda    curDrive
31791          jsr    S06AB          ;Partition öffnen.
31792
31793          pla
31794          sta    C042A          ;Oberstes Fenster.
31795
31796          rts
31797
31798 ; Partitionsdaten einlesen.
31799 ; Übergabe: A = Part.-Nr.
31800 :z733C          sta    r4L          ;Nr. der Partition.
31801
31802          jsr    PurgeTurbo
31803          jsr    InitForIO
31804
31805          lda    curDrive
31806          pha          ;Akt.Lfwk. speichern.
31807
31808          tay
31809          lda    V04B8 -8,y      ;Geräteinfo einlesen.
31810          and    #%00000011      ;RAMLink/RAMDrive?
31811          beq    :1              ; => Nein, weiter...
31812
31813          lda    C0509          ;RAMLink-Adresse.
31814          sta    curDevice
31815          sta    curDrive
31816
31817 ::1          lda    #< v7664      ;"G-P"
31818          sta    r0L
31819          lda    #> v7664
31820          sta    r0H
31821
31822          jsr    z73AF          ;Befehl senden.
31823          txa
31824          bne    :exit          ; => Fehler.
31825
31826          lda    r4L          ;Nr. der Partition
31827          jsr    CIOUT          ;an Laufwerk senden.
31828
31829          lda    #CR          ;"G-P"-Befehl
31830          jsr    CIOUT          ;abschließen.
31831
31832          jsr    UNLSN
31833
31834          lda    #$00
31835          sta    STATUS
31836
31837          lda    curDrive          ;Laufwerk auf
31838          jsr    TALK          ;Senden umschalten.
31839          lda    #$ff
31840          jsr    TKSA
31841

```

```

ldy #0 ;Partitionsdaten 31842
::loop jsr ACPTR ;einlesen. 31843
sta v40EF,y 31844
31845
bit STATUS ;Ende der Daten? 31846
bmi :done ; => Ja, weiter... 31847
31848
iny ;"G-P" sendet max. 31849
cpy #31 ;Bytes 0-30. 31850
bcc :loop ; => Nächstes Byte... 31851
31852
::done tya 31853
pha 31854
jsr UNTALK 31855
pla 31856
cmp #31 ;Vorzeitig beendet? 31857
bne :err ; => Ja, Fehler... 31858
31859
lda r4L ;Gesuchte Part.-Nr. 31860
cmp #$ff ;Aktuelle Partition? 31861
beq :ok ; => Ja, Ende... 31862
cmp v40EF +2 ;Stimmt Part-Nr.? 31863
beq :ok ; => Ja, OK... 31864
31865
::err ldx #STRUCT_MISMAT ;Daten ungültig. 31866
b $2c 31867
::ok ldx #NO_ERROR ;Daten OK. 31868
31869
::exit pla ;Akt.Lfwk. setzen. 31870
sta curDrive 31871
sta curDevice 31872
31873
jmp DoneWithIO 31874
31875
; Befehl für Info/Partition abfragen. 31876
:v7664 b "G-P",0 31877
31878
; Befehl an Laufwerk senden. 31879
:z73AF lda #$00 31880
sta STATUS 31881
31882
lda curDrive 31883
jsr LISTEN 31884
lda #$ff ;Sekundäradresse. 31885
jsr SECOND 31886
31887
bit STATUS 31888
bpl :1 31889
31890
jsr UNLSN 31891
31892
ldx #DEV_NOT_FOUND 31893
rts 31894
31895
::1 ldy #$00 31896

```

```

31897 ::loop          lda    (r0),y
31898                beq     :done
31899                jsr     CIOUT
31900                iny
31901                bne     :loop
31902
31903 ::done          ldx     #NO_ERROR
31904                rts
31905
31906 ; 1: Partition für Fenster öffnen.
31907 :N7BA7          ldx     C042A                ;Oberstes Fenster.
31908                lda     C04F8,x            ;Lfwk. für Fenster.
31909                tay
31910                cmp     V21B1,x            ;Lfwk./Fens.geändert?
31911                bne     :clrdat            ; => Ja, weiter...
31912
31913                lda     V5C34,x            ;Lfwk.Typ geändert?
31914                cmp     RealDrvType -8,y
31915                bne     :clrdat            ; => Ja, weiter...
31916
31917                lda     V0728,x            ;Partition setzen?
31918                beq     :exit              ; => Nein, weiter...
31919
31920                sta     C04FC,x            ;Part. für Fenster.
31921                sta     r3H                ;Neue Partition.
31922
31923                lda     #$00
31924                sta     C050B,x            ;Fensterposition.
31925                jsr     OpenPartition      ;Partition öffnen.
31926
31927 ; Fens.pos. löschen und Fenster öffnen.
31928                jsr     :clrdat            ;Fens.dat. löschen.
31929
31930 ; Hinweis:
31931 ; Im Textmodus wird der Bereich für
31932 ; VLIR-Module überschrieben!
31933                jmp     S4296                ;Update akt.Fenster.
31934
31935 ; Fensterdaten löschen.
31936 ::clrdat        lda     #$00
31937                ldx     C042A                ;Oberstes Fenster.
31938                sta     V0728,x            ;Letztes Lfwk.
31939                sta     V5C34,x            ;Letzter Lfwk.-Typ.
31940                sta     C050B,x            ;Fensterposition.
31941 ::exit          rts
31942
31943 ; 0: Neue Partition wählen.
31944 :N6F21          lda     #$00                ;Kein VLIR-Modul mehr
31945                sta     V0422                ;im Speicher!
31946
31947 ; Modus für CMD/64Net für Menüroutine "Partition wechseln".
31948                sta     V3A4B                ;CMD-Part. wählen.
31949
31950                jsr     S5C89                ;Hole Disk/Fenster.
31951

```



```

32007 ; Die Routine muss am Ende den Speicher
32008 ; wieder zurücksetzen!
32009 ::2          jsr    CallRoutine          ;CMD-Part. öffnen.
32010          txa                          ;Part. gewechselt?
32011          bne     :err                  ; => Nein, Ende...
32012
32013 ; Hinweis:
32014 ; Der Speicher ist wieder hergestellt.
32015          bit     V3A4B                  ;CMD oder 64Net?
32016          bpl     :done                  ; => CMD.
32017
32018 ; Auswahl 64Net-Partition beenden.
32019 ::64net      ldx     C042A              ;Oberstes Fenster.
32020          lda     C04F8,x                ;Lfwk. für Fenster.
32021          sta     r0L
32022
32023 ; Andere 64Net-Fenster suchen/schließen.
32024          ldy     #0
32025          ldx     #4 -1
32026 ::loop       lda     C0448,x            ;Fenster geöffnet?
32027          beq     :next                  ; => Nein, weiter...
32028
32029          lda     C04F8,x                ;Lfwk. für Fenster.
32030          cpx     C042A                  ;Oberstes Fenster?
32031          beq     :next                  ; => Ja, weiter...
32032
32033          cmp     r0L                    ;Gleiches Lfwk.?
32034          bne     :next                  ; => And. Laufwerk.
32035
32036          jsr     S09B5                  ;CMD-Info löschen.
32037
32038 ;          lda     #$00
32039          sta     C0448,x                ;Fenster geschlossen.
32040          sta     C04F8,x                ;Lfwk. für Fenster.
32041
32042          iny                          ;Anzahl Fenster +1.
32043
32044 ::next        dex
32045          bpl     :loop
32046
32047          cpy     #$00                  ;Fens. geschlossen?
32048          beq     :ok                    ; => Nein, weiter...
32049
32050 ; Hinweis:
32051 ; Hier werden die verbleibenden Fenster
32052 ; neu geöffnet, nur andere Fenster für
32053 ; das 64Net-Lfwk. werden geschlossen.
32054          jsr     S157E                  ;Alle Fens. anzeigen.
32055
32056 ::ok          ldx     C042A              ;Oberstes Fenster.
32057          lda     C04F8,x                ;Lfwk. für Fenster.
32058          jsr     SetDevice              ;Laufwerk und
32059          jsr     OpenDisk               ;Diskette öffnen.
32060          txa
32061          beq     :done

```



			32062
::err	rts		32063
			32064
; Auswahl CMD-Partition beenden.			32065
::done	ldx	C042A	;Oberstes Fenster.
	stx	WIN042F	;Aktives Fenster.
	jsr	S09B7	;Tr/Se SubDir = \$00.
			32069
;	lda	#\$00	32070
	sta	C0448,x	;Fenster geschlossen.
			32072
	jsr	S5842	;mark.Dat. anzeigen.
	jsr	S524C	;Dat.-Ausw. aufheben.
			32075
	lda	curDrive	32076
	jsr	S06AB	;Laufwerk öffnen.
			32078
	ldx	C042A	;Oberstes Fenster.
	jmp	S44E8	;Fens. erneut öffnen.
			32081
; Speicher in REU auslagern.			32082
:z6F7B	bit	Flag_ExtRAMinUse	32083
	bpl	:1	32084
			32085
	brk	;Panic! Speicher bereits belegt!	32086
			32087
::1	lda	Flag_ExtRAMinUse	32088
	ora	%%10000000	32089
	sta	Flag_ExtRAMinUse	32090
	bne	z6F7E	32091
			32092
; Speicher aus REU einlesen.			32093
:z6F7C	bit	Flag_ExtRAMinUse	32094
	bmi	:1	32095
			32096
	brk	;Panic! Speicher ist freigegeben!	32097
			32098
::1	lda	Flag_ExtRAMinUse	32099
	and	%%01111111	32100
	sta	Flag_ExtRAMinUse	32101
			32102
; Speicher RAM/REU tauschen.			32103
:z6F7E	lda	#< BASEDATA	32104
	sta	r0L	32105
	lda	#> BASEDATA	32106
	sta	r0H	32107
			32108
	lda	#\$00	32109
	sta	r1L	32110
	sta	r1H	32111
			32112
	lda	#< SIZEDATA	32113
	sta	r2L	32114
	lda	#> SIZEDATA	32115
	sta	r2H	32116

```

32117
32118         lda    MP3_64K_DATA
32119         sta    r3L
32120
32121         jmp     SwapRAM
32122
32123 ; Dialogbox: Partition aus Liste auswählen.
32124 :z6F9B         lda    #< z723F           ;Part.-Liste: CMD.
32125                 ldx    #> z723F
32126
32127                 bit     V3A4B           ;CMD/64Net?
32128                 bpl     :1              ; => CMD, weiter...
32129
32130                 lda    #< z77B8         ;Part.-Liste 64Net.
32131                 ldx    #> z77B8
32132
32133 ::1             jsr     CallRoutine      ;Part.-Liste.
32134                 txa
32135                 beq     :dbox           ; => OK, weiter...
32136
32137                 pha
32138                 jsr     z6F7C           ;Speicher einlesen.
32139                 pla
32140                 tax
32141
32142                 jsr     S5CF3           ;Fehler ausgeben.
32143
32144                 ldx    #$ff            ;Abbruch...
32145                 rts
32146
32147 ; Auswahlbox anzeigen.
32148 ::dbox         lda    #< v7072         ;Zeiger auf Speicher
32149                 sta    a2L             ;für Part.-Nr.
32150                 lda    #> v7072
32151                 sta    a2H
32152
32153 if ENABLE_ORIGCOL = FALSE
32154                 lda    COL043C         ;Systemfarbe für
32155                 pha                    ;Dateiauswahlbox
32156                 lda    C_FBoxBack      ;verwenden.
32157                 sta    COL043C
32158 endif
32159
32160                 lda    #< d705C         ;Auswahlbox.
32161                 ldx    #> d705C
32162                 jsr     S2090           ;DBox ohne Recover.
32163
32164 if ENABLE_ORIGCOL = FALSE
32165                 pla
32166                 sta    COL043C         ;Farbe zurücksetzen.
32167 endif
32168
32169                 lda    v7656           ;Anzahl Partitionen.
32170                 beq     :exit           ; => Keine Part.
32171

```

```

        lda    v705B                ;Part./Doppelklick?      32172
        bne    :open                ; => Ja, weiter...       32173
                                           32174
        lda    r0L                  32175
        cmp    #OPEN                ;"OPEN"?                 32176
        beq    :open                ; => Ja, öffnen...       32177
        cmp    #CANCEL              ;"CANCEL"?               32178
        beq    :exit                ; => Ja, Abbruch...      32179
                                           32180
        jsr    z7C0B                ;Eingabe testen.        32181
        txa                               32182
        beq    :open                ; => OK, weiter...       32183
                                           32184
; Partition bereits aktiv/nicht gewechselt.      32185
::exit    jsr    z6F7C                ;Speicher einlesen.   32186
                                           32187
        ldx    #$ff                ;Abbruch...              32188
        rts                               32189
                                           32190
; Auswahl gültig, Ende.                        32191
::open    ldx    #$00                ;OK, Ende...           32192
        rts                               32193
                                           32194
; Auswahlbox: Eingabe Part.-Nr. auswerten.      32195
; Rückgabe: X = $00: Eingabe OK.                32196
;                      $FF: Eingabe ungültig.      32197
:z7C0B    jsr    z77A1                ;Eingabe nach DEZ.     32198
        tax                               32199
        beq    :err                ; => Ungültig...          32200
                                           32201
        bit    V3A4B                ;CMD/64Net?             32202
        bmi    :1                  ; => 64Net, weiter...     32203
                                           32204
; Hinweis:                                     32205
; Bei 64Net Partition von 1-255. Bei             32206
; CMD-Laufwerk 255=Aktuelle Partition.           32207
        inx                               32208
        beq    :err                ; => Ungültig...          32209
                                           32210
::1        ldx    C042A                ;Oberstes Fenster.   32211
        cmp    C04FC,x              ;Bereits aktiv?         32212
        beq    :err                ; => Ja, Ende...          32213
                                           32214
        sta    r0L                  ;Partition 1-254/255.     32215
                                           32216
        lda    #< z7004              ;Eingabe testen:       32217
        ldx    #> z7004              ;CMD-Laufwerk.          32218
                                           32219
        bit    V3A4B                ;CMD/64Net?             32220
        bpl    :2                  ; => CMD, weiter...       32221
                                           32222
        lda    #< z7005              ;Eingabe testen:       32223
        ldx    #> z7005              ;64Net-Laufwerk.        32224
                                           32225
::2        jsr    CallRoutine          32226

```

```

32227          cpx    #255          ;Partition = 255?
32228          beq     :err          ; => Ja, Ende...
32229
32230          stx     v7652          ;Markierte Datei.
32231
32232          ldx     #$00          ;Eingabe OK.
32233          rts
32234
32235 ::err      ldx     #$ff          ;Eingabe ungültig.
32236          rts
32237
32238 ; CMD-Laufwerk: Auswahl testen.
32239 ; Rückgabe: X = 255 : Fehler.
32240 ;          0-253 : Eintrag in Tabelle.
32241 :z7004      ldx     v7656          ;Anzahl Partitionen.
32242          dex
32243 ::1        lda     v4110,x
32244          cmp     r0L
32245          beq     :ok
32246
32247          dex
32248          cpx     #$ff
32249          bne     :1
32250
32251 ;          ldx     #$ff          ;Nicht gefunden.
32252 ::ok      rts
32253
32254 ; 64Net-Laufwerk: Auswahl testen.
32255 ; Rückgabe: X = 255 : Fehler.
32256 ;          0-254 : Eintrag in Tabelle.
32257 :z7005      lda     v7656          ;Anzahl Partitionen.
32258          sec
32259          sbc     r0L          ;Partition gültig?
32260          bcc     :err          ; => Nein, Ende...
32261
32262          ldx     r0L
32263          dex
32264          rts
32265
32266 ::err      ldx     #$ff          ;Ungültig.
32267          rts
32268
32269 ; Zahl von ASCII nach DEZ wandeln.
32270 :z77A1      ldx     #0
32271          stx     r0L
32272          stx     r0H
32273
32274          lda     #10
32275          sta     r1L
32276          stx     r1H
32277
32278 ;          ldx     #0
32279 ::1        lda     v7072,x
32280          beq     :done
32281

```

	pha		32282
	txa		32283
	pha		32284
			32285
	ldx	#r0	32286
	ldy	#r1	32287
	jsr	DMult	32288
			32289
	pla		32290
	tax		32291
	pla		32292
	sec		32293
	sbc	"0"	32294
	bcs	:2	32295
::null	lda	"0"	32296
::2	cmp	#10	32297
	bcs	:null	32298
			32299
	clc		32300
	adc	r0L	32301
	sta	r0L	32302
	bcc	:3	32303
	inc	r0H	32304
			32305
::3	inx		32306
	bne	:1	32307
			32308
::done	lda	r0H	32309
	bne	:err	32310
	lda	r0L	32311
	bne	:ok	32312
			32313
::err	lda	#255	32314
::ok	rts		32315
			32316
; Dialogboxroutine Partition wählen.			32317
:r7124	jsr	z7609	32318
			32319
if ENABLE_ORIGCOL = TRUE			32320
	lda	#DBFILE_COL	32321
else			32322
	lda	C_FBoxFiles	32323
endif			32324
	jsr	S1561	32325
			32326
	dec	r2L	32327
	inc	r2H	32328
			32329
	inc	r4L	32330
	dec	r3L	32331
			32332
	lda	11111111	32333
	jsr	FrameRectangle	32334
			32335
	jsr	i_BitmapUp	32336

```

32337          w NAVICON
32338          b POSNAV_X/8, POSNAV_Y
32339          b NAVICON_X, NAVICON_Y
32340
32341          ldx  #(r4H - r2L)          ;Bereich für
32342 :::1          lda  :coord1,x        ;Navigation.
32343          sta  r2,x
32344          dex
32345          bpl  :1
32346
32347 if ENABLE_ORIGCOL = TRUE
32348          lda  #NAVBAR_COL
32349 else
32350          lda  C_DBoxDIcon          ;Farbe Navigation.
32351 endif
32352          jsr  S1561                ;Screencolor setzen.
32353
32354          ldx  #(r4H - r2L)          ;Bereich für
32355 :::2          lda  :coord2,x        ;Eingabefeld.
32356          sta  r2,x
32357          dex
32358          bpl  :2
32359
32360          lda  #$00                  ;Füllmuster.
32361          jsr  S140C                ;Rechteck zeichnen.
32362
32363          lda  #%11111111          ;Rahmen um
32364          jsr  FrameRectangle        ;Eingabefeld.
32365
32366 if ENABLE_ORIGCOL = TRUE
32367          lda  #PARTNR_COL          ;Farbe Part.Eingabe.
32368 else
32369          lda  C_InputField          ;Farbe Part.Eingabe.
32370 endif
32371          jsr  S1561                ;Screencolor setzen.
32372
32373 ; Mausabfrage installieren.
32374          lda  #< r73DF
32375          sta  otherPressVec +0
32376          lda  #> r73DF
32377          sta  otherPressVec +1
32378
32379          lda  #$00
32380          sta  v7652                ;Markierte Datei.
32381          sta  v705B                ;Vorgabe: Kein Db.kl.
32382          sta  v7653                ;Erste Datei / Liste.
32383
32384          jsr  z7553                ;Update Dateiliste.
32385          jmp  z75F6                ;Eintrag invertieren.
32386
32387 ; Koordinaten für Navigation.
32388 :::coord1      b DBOX_Y1 -$08 -$10 +1
32389              b DBOX_Y1 -$08 -$10 +1 +NAVICON_Y -1
32390              w DBOX_X0 +$10
32391              w DBOX_X0 +$10 +NAVICON_X*8 -1

```

		32392
; Koordinaten für Eingabefeld.		32393
::coord2	b PBOX_Y0	32394
	b PBOX_Y1	32395
	w PBOX_X0	32396
	w PBOX_X1	32397
		32398
; Mausabfrage für Dateifenster.		32399
:r73DF	lda mouseData	32400
	bmi :exit	32401
		32402
	jsr z7609	32403
		32404
	jsr IsMseInRegion	32405
	beq z7440	32406
		32407
	jsr z75F6	32408
		32409
	jsr z7609	32410
		32411
	lda mouseYPos	32412
	sec	32413
	sbc r2L	32414
	sta r0L	32415
	lda #\$00	32416
	sta r0H	32417
	sta r1H	32418
		32419
	lda #ENTRY_YH	32420
	sta r1L	32421
		32422
	ldx #r0	32423
	ldy #r1	32424
	jsr Ddiv	32425
		32426
	lda r0L	32427
	clc	32428
	adc v7653	32429
	cmp v7656	32430
	bcc :1	32431
		32432
	ldx v7656	32433
	dex	32434
	txa	32435
::1	sta v7652	32436
		32437
	jsr z75F6	32438
		32439
	lda #10	32440
	sta dblClickCount	32441
		32442
	lda #%10000000	32443
	sta mouseData	32444
::wait	lda mouseData	32445
	beq :dblclk	32446

```

32447         lda    dblClickCount
32448         bne     :wait
32449  ::exit      rts
32450
32451  ::dblclk     lda     v7656             ;Anzahl Partitionen.
32452             beq     :exit             ; => Kein, Ende...
32453
32454             inc     v705B             ; => Doppelklick.
32455             jmp     RstrFrmDialogue
32456
32457 ; Klick auf Navbar auswerten.
32458 :z7440        lda     #$00
32459             sta     v74A6             ;0=Datei, 1=Seite.
32460             sta     v7552             ;Keine Dauerfunktion.
32461             sta     v7553             ;Zähler auf Anfang.
32462
32463             ldx     #(r4H - r2L)      ;Koordinaten für
32464  ::1          lda     :coord,x        ;Navbar auf Anfang
32465             sta     r2,x             ;setzen.
32466             dex
32467             bpl     :1
32468
32469  ::loop        jsr     IsMseInRegion   ;Icon gewählt?
32470             bne     z7488            ; => Ja, weiter...
32471
32472             lda     r4L               ;Koord. für das
32473             sta     r3L               ;nächste Icon.
32474             clc
32475             adc     #< 16
32476             sta     r4L
32477             lda     r4H
32478             sta     r3H
32479             adc     #> 16
32480             sta     r4H
32481
32482             inc     r3L               ;X-Links korrigieren.
32483             bne     :2
32484             inc     r3H
32485
32486  ::2          inc     v7553
32487             lda     v7553
32488             cmp     #6               ;Alle Icons getestet?
32489             bne     :loop            ; => Nein, weiter...
32490
32491             rts
32492
32493  ::coord       b     POSNAV_Y,POSNAV_Y +15
32494             w     POSNAV_X,POSNAV_X +15
32495
32496 ; Navbar-Icon-Routine aufrufen.
32497 :z7488        lda     v7553
32498             asl
32499             tay
32500             lda     v749A +0,y
32501             ldx     v749A +1,y

```



	jmp	CallRoutine		32502
				32503
:v749A	w r74FE	;Nächste Datei.		32504
	w r74C7	;Vorherige Datei.		32505
	w r74B1	;Zum Ende.		32506
	w r74A7	;Zum Anfang.		32507
	w r74E9	;Nächste Seite.		32508
	w r74C2	;Vorherige Seite.		32509
				32510
; Zum Anfang.				32511
:r74A7	lda	v7552	;Dauerfunktion?	32512
	bne	:1	; => Ja, weiter...	32513
				32514
	ldx	v7653	;Erster Eintrag?	32515
	beq	:2	; => Ja, Ende...	32516
				32517
	ldx	#\$00	;Erster Eintrag.	32518
	jmp	z750D	;Neue Pos. anzeigen.	32519
				32520
::1	jmp	z75F6	;Eintrag invertieren.	32521
::2	rts			32522
				32523
; Zum Ende.				32524
:r74B1	lda	v7552	;Dauerfunktion?	32525
	bne	:1	; => Ja, weiter...	32526
				32527
	lda	v7656	;Anzahl Dateien.	32528
	sec			32529
	sbc	#FB0X_FILES		32530
	bcc	r74A7		32531
				32532
	cmp	v7653	;Erster Eintrag?	32533
	beq	:2	; => Ja, Ende...	32534
				32535
	tax		;Letzte Datei.	32536
	jmp	z750D	;Neue Pos. anzeigen.	32537
				32538
::1	jmp	z75F6	;Eintrag invertieren.	32539
::2	rts			32540
				32541
; Vorherige Seite.				32542
:r74C2	lda	#\$01		32543
	sta	v74A6		32544
				32545
; Vorherige Datei.				32546
:r74C7	ldx	v7653	;Erster Eintrag?	32547
	bne	:1	; => Nein, weiter...	32548
				32549
	lda	v7552	;Dauerfunktion?	32550
	bne	:4	; => Ja, weiter...	32551
	beq	:5	; => Nein, Ende...	32552
				32553
::1	dex			32554
				32555
	lda	v74A6	;Seite zurück?	32556

```

32557          beq    :3          ; => Nein, weiter...
32558
32559          lda    v7653          ;Erster Eintrag.
32560          sec
32561          sbc    #FB0X_FILES    ;Neue Seite möglich?
32562          bcs    :2          ; => Ja, weiter...
32563          lda    #$00          ;Zum Anfang.
32564 ::2          tax
32565 ::3          jmp    z750D      ;Neue Pos. anzeigen.
32566
32567 ::4          jmp    z75F6      ;Eintrag invertieren.
32568 ::5          rts
32569
32570 ; Nächste Seite.
32571 :r74E9      lda    v7653          ;Erster Eintrag.
32572          clc
32573          adc    #FB0X_FILES +FB0X_FILES +1
32574          cmp    v7656          ;Anzahl Einträge.
32575          bcs    r74B1
32576
32577          lda    v7653          ;Erster Eintrag.
32578          clc
32579          adc    #FB0X_FILES
32580          tax
32581          jmp    z750D      ;Neue Pos. anzeigen.
32582
32583 ; Nächste Datei.
32584 :r74FE      ldx    v7653          ;Erster Eintrag.
32585          inx
32586          lda    v7656          ;Anzahl Einträge.
32587          sec
32588          sbc    v7653          ;Erster Eintrag.
32589          cmp    #FB0X_FILES +1
32590          bcs    z750D
32591
32592          lda    v7552          ;Dauerfunktion?
32593          bne    :1          ; => Ja, weiter...
32594          rts
32595
32596 ::1          jmp    z75F6      ;Eintrag invertieren.
32597
32598 ; Neue Position anzeigen.
32599 :z750D      txa
32600          pha
32601
32602          ldy    v7552          ;Dauerfunktion?
32603          bne    :1          ; => Ja, weiter...
32604          jsr    z75F6      ;Eintrag invertieren.
32605
32606 ::1          pla
32607          sta    v7653          ;Erste Datei / Liste.
32608          cmp    v7652          ;Markierte Datei.
32609          bcc    :2
32610          sta    v7652          ;Markierte Datei.
32611

```

::2	clc		32612
	adc	#FBOX_FILES -1	32613
	cmp	v7652 ;Markierte Datei.	32614
	bcs	:3	32615
	sta	v7652 ;Markierte Datei.	32616
			32617
::3	jsr	z7553 ;Liste ausgeben.	32618
			32619
	ldx	#3	32620
::4	jsr	SCPU_Pause	32621
	dex		32622
	bne	:4	32623
			32624
	lda	mouseData ;Maustaste gedrückt?	32625
	bmi	:5 ; => Nein, Ende..	32626
			32627
;	ldx	#\$00	32628
	inx		32629
	stx	v7552 ;Dauerfunktion.	32630
	jmp	z7488 ;Nav. wiederholen.	32631
			32632
::5	jmp	z75F6 ;Eintrag invertieren.	32633
			32634
;	Dateiliste ausgeben.		32635
:z7553	lda	v7656 ;Part. gefunden?	32636
	bne	:1 ; => Ja, weiter...	32637
			32638
	lda	#< v74A7 ;Hinweis ausgeben und	32639
	sta	r0L ;Ausgabe beenden.	32640
	lda	#> v74A7	32641
	sta	r0H	32642
	jmp	PutString	32643
			32644
::1	lda	rightMargin +1	32645
	pha		32646
	lda	rightMargin +0	32647
	pha		32648
			32649
	jsr	z7609 ;Größe Dateifenster.	32650
			32651
	lda	#\$00 ;Füllmuster.	32652
	jsr	S140C ;Rechteck zeichnen.	32653
			32654
	lda	#\$00	32655
	sta	r15L	32656
			32657
	jsr	z7622 ;Koord. mark. Datei.	32658
			32659
	lda	r4L	32660
	sta	rightMargin +0	32661
	lda	r4H	32662
	sta	rightMargin +1	32663
			32664
	ldx	#FBOX_FILES	32665
	lda	#FBOX_FILES	32666

```

32667          sec
32668          sbc    v7656          ;Anzahl Dateien.
32669          bcc    :2
32670          ldx    v7656          ;Anzahl Dateien.
32671  ::2          stx    r15H
32672
32673          lda    v7653          ;Erster Eintrag.
32674          sta    r0L
32675
32676          lda    #17          ;Größe Dateieintrag.
32677          sta    r1L
32678
32679          ldy    #r0
32680          ldx    #r1
32681          jsr    BBMult
32682
32683          lda    r1L          ;Zeiger auf Datei-
32684          clc          ;liste berechnen.
32685          adc    #< BASEDATA
32686          sta    r14L
32687          lda    r1H
32688          adc    #> BASEDATA
32689          sta    r14H
32690
32691  ::loop        lda    r15L
32692          jsr    z7622          ;Koord. mark. Datei.
32693
32694          lda    r3H          ;X-Koordinate für
32695          sta    r11H          ;Dateiname.
32696          lda    r3L
32697          sta    r11L
32698
32699          lda    r2L          ;Y-Koordinate für
32700          clc          ;Dateiname.
32701          adc    #$07
32702          sta    r1H
32703
32704          lda    r14H          ;Dateiname ausgeben.
32705          sta    r0H
32706          lda    r14L
32707          sta    r0L
32708          jsr    PutString
32709
32710          lda    #$00          ;Highbyte immer 0.
32711          sta    r0H
32712
32713          lda    r15L          ;Datei-Nr.
32714          clc          ;Erste Datei = Nr.1!
32715          adc    v7653          ;Erste Datei / Liste.
32716          tax
32717          lda    v4110,x
32718          sta    r0L
32719
32720          lda    #< FBOX_X1 -1 -24
32721          sta    r11L

```

	lda	#> FBOX_X1 -1 -24	32722
	sta	r11H	32723
			32724
	lda	#24 ! SET_RIGHTJUST!SET_SUPRESS	32725
	jsr	PutDecimal	32726
			32727
	lda	#17	32728
	clc		32729
	adc	r14L	32730
	sta	r14L	32731
	bcc	:3	32732
	inc	r14H	32733
			32734
::3	inc	r15L	32735
	lda	r15L	32736
	cmp	r15H ;Liste voll?	32737
	bne	:loop ; => Nein, weiter...	32738
			32739
::done	pla		32740
	sta	rightMargin +0	32741
	pla		32742
	sta	rightMargin +1	32743
	rts		32744
			32745
; Aktuellen Eintrag invertieren.			32746
:z75F6	lda	v7656 ;Anzahl Partitionen.	32747
	bne	:1 ; >0, weiter...	32748
	rts		32749
			32750
::1	lda	v7652 ;Markierte Datei.	32751
	sec		32752
	sbc	v7653 ;Erste Datei / Liste.	32753
	jsr	z7622 ;Koord. mark. Datei.	32754
	jmp	InvertRectangle	32755
			32756
; Größe Dateifenster.			32757
:z7609	ldx	#(r4H - r2L)	32758
::1	lda	:coord,x	32759
	sta	r2,x	32760
	dex		32761
	bpl	:1	32762
	rts		32763
			32764
::coord	b	FBOX_Y0, FBOX_Y1	32765
	w	FBOX_X0, FBOX_X1	32766
			32767
; Koordinaten für markierte Datei berechnen.			32768
:z7622	sta	r0L ;Eintrag im Fenster.	32769
			32770
	lda	#ENTRY_YH ;12 Pixel Höhe für	32771
	sta	r1L ;Dateieintrag.	32772
			32773
	ldy	#r1	32774
	ldx	#r0	32775
	jsr	BBMult	32776

```

32777
32778          jsr    z7609          ;Größe Dateifenster.
32779
32780          lda    r0L
32781          clc
32782          adc    r2L
32783          sta    r2L
32784          clc
32785          adc    #ENTRY_YH
32786          sta    r2H
32787
32788          inc    r2L
32789          dec    r2H
32790          dec    r2H
32791
32792          inc    r3L
32793          bne    :1
32794          inc    r3H
32795
32796 ::1          ldx    #r4
32797          jmp    Ddec
32798
32799 ; Angaben für Dateiauswahlbox.
32800 :v7651          b $00    ;Anzahl Dateien.
32801 :v7652          b $00    ;Markierter Eintrag.
32802 :v7653          b $00    ;Erste Datei / Liste.
32803 :v7656          b $00    ;Anzahl Dateien.
32804 :v705B          b $00    ;$01 = Partition/Doppelklick.
32805 :v7552          b $00    ;Dauerfunktion für Navigation.
32806 :v7553          b $00    ;Nr. Navbar-Icon (0-5).
32807 :v74A6          b $00    ;Datei/Seite zurück.
32808
32809 :v74A7          b GOTOXY
32810              w FBOX_X0 +8
32811              b FBOX_Y0 +8 +8
32812              b PLAINTEXT,BOLDON
32813 if LANG = LANG_DE
32814              b "Keine Partitionen!"
32815 endif
32816 if LANG = LANG_EN
32817              b "No partitions!"
32818 endif
32819              b PLAINTEXT,0
32820
32821 ; Dateiauswahlbox.
32822 :d705C          b %00000001
32823              b DBOX_Y0,DBOX_Y1
32824              w DBOX_X0,DBOX_X1
32825
32826              b DB_USR_ROUT
32827              w r7124
32828
32829              b DBTXTSTR
32830              b PBOX_X0-DBOX_X0 +1
32831              b PBOX_Y0-DBOX_Y0 -10 -5

```

w :t1	32832
	32833
b DBTXTSTR	32834
b PBOX_X0-DBOX_X0 +1	32835
b PBOX_Y0-DBOX_Y0 -5	32836
w :t2	32837
	32838
b DBTXTSTR	32839
b PBOX_X0-DBOX_X0 +3	32840
b PBOX_Y1-DBOX_Y0 -5	32841
w :t3	32842
	32843
; Hinweis:	32844
; Y-Koordinate für Eingabe ist nicht	32845
; die Baseline, sondern Y-oben!	32846
b DBGETSTRING	32847
b PBOX_X0-DBOX_X0 +22	32848
b PBOX_Y0-DBOX_Y0 +\$10 -12	32849
b a2L	32850
b 3	32851
	32852
b OPEN , \$11, \$08	32853
b CANCEL, \$11, \$58	32854
	32855
b NULL	32856
	32857
if LANG = LANG_DE	32858
::t1 b PLAINTEXT	32859
b "Partition", 0	32860
::t2 b "eingeben:", 0	32861
::t3 b "Nr:", 0	32862
endif	32863
if LANG = LANG_EN	32864
::t1 b PLAINTEXT	32865
b "Enter", 0	32866
::t2 b "partition:", 0	32867
::t3 b "No:", 0	32868
endif	32869
	32870
; Eingabefeld für Part.-Nr.	32871
:v7072 s 3	32872
b NULL	32873
	32874
; Breite/Höhe Navbar-Icon.	32875
:NAVICON_X = 12	32876
:NAVICON_Y = 16	32877
	32878
; Navbar-Icon (TopDesk-Version).	32879
if FALSE	32880
:NAVICON	32881
b \$0c	32882
b \$ff	32883
b \$db	32884
b \$80, \$01, \$00, \$01, \$00, \$01, \$00, \$01, \$00, \$01, \$00, \$01	32885
b \$80, \$01, \$00, \$01, \$00, \$01, \$1f, \$e1, \$00, \$01, \$00, \$01	32886

```

32887      b      $83,$01,$03,$01,$03,$01,$00,$01,$06,$c1,$06,$c1
32888      b      $83,$01,$07,$81,$03,$01,$03,$01,$06,$c1,$0e,$e1
32889      b      $83,$01,$0f,$c1,$03,$01,$07,$81,$06,$c1,$1e,$f1
32890      b      $83,$01,$1b,$61,$03,$01,$0f,$c1,$06,$c1,$36,$d9
32891      b      $83,$01,$03,$01,$03,$01,$1b,$61,$06,$c1,$06,$c1
32892      b      $83,$01,$03,$01,$1b,$61,$03
32893      b $cd
32894      b                                $01,$06,$c1,$06,$c1
32895      b      $9b,$61,$03,$01,$0f,$c1,$03,$01,$36,$d9,$06,$c1
32896      b      $8f,$c1,$03,$01,$07,$81,$03,$01,$1e,$f1,$06,$c1
32897      b      $87,$81,$03,$01,$03,$01,$03,$01,$0e,$e1,$06,$c1
32898      b      $83,$01,$03,$01,$00,$01,$03,$01,$06,$c1,$06,$c1
32899      b      $80,$01,$00,$01,$1f,$e1,$00,$01,$00,$01,$00,$01
32900      b      $80,$01,$00,$01,$00,$01,$00,$01,$00,$01,$00,$01
32901      b $0c
32902      b      $ff
32903 endif
32904
32905 ; Navbar-Icon (DTopDesk-Version).
32906 :NAVICON
32907 ::nb_00      = %11111111 +%11111111 *256
32908 ::nb_01      = %10000000 +%00000001 *256
32909 ::nb_02      = %10000000 +%00000001 *256
32910 ::nb_03      = %10000000 +%00000001 *256
32911 ::nb_04      = %10000001 +%10000001 *256
32912 ::nb_05      = %10000001 +%10000001 *256
32913 ::nb_06      = %10000001 +%10000001 *256
32914 ::nb_07      = %10000001 +%10000001 *256
32915 ::nb_08      = %10001111 +%11110001 *256
32916 ::nb_09      = %10000111 +%11110001 *256
32917 ::nb_0A      = %10000011 +%11000001 *256
32918 ::nb_0B      = %10000001 +%10000001 *256
32919 ::nb_0C      = %10000000 +%00000001 *256
32920 ::nb_0D      = %10000000 +%00000001 *256
32921 ::nb_0E      = %10000000 +%00000001 *256
32922 ::nb_0F      = %11111111 +%11111111 *256
32923
32924 ::nb_10      = %11111111 +%11111111 *256
32925 ::nb_11      = %10000000 +%00000001 *256
32926 ::nb_12      = %10000000 +%00000001 *256
32927 ::nb_13      = %10000000 +%00000001 *256
32928 ::nb_14      = %10000001 +%10000001 *256
32929 ::nb_15      = %10000011 +%11000001 *256
32930 ::nb_16      = %10000111 +%11110001 *256
32931 ::nb_17      = %10001111 +%11110001 *256
32932 ::nb_18      = %10000001 +%10000001 *256
32933 ::nb_19      = %10000001 +%10000001 *256
32934 ::nb_1A      = %10000001 +%10000001 *256
32935 ::nb_1B      = %10000001 +%10000001 *256
32936 ::nb_1C      = %10000000 +%00000001 *256
32937 ::nb_1D      = %10000000 +%00000001 *256
32938 ::nb_1E      = %10000000 +%00000001 *256
32939 ::nb_1F      = %11111111 +%11111111 *256
32940
32941 ::nb_20      = %11111111 +%11111111 *256

```



::nb_21	= %10000000 +%00000001 *256	32942
::nb_22	= %10000000 +%00000001 *256	32943
::nb_23	= %10000001 +%10000001 *256	32944
::nb_24	= %10000001 +%10000001 *256	32945
::nb_25	= %10000001 +%10000001 *256	32946
::nb_26	= %10000001 +%10000001 *256	32947
::nb_27	= %10001111 +%11110001 *256	32948
::nb_28	= %10000111 +%11100001 *256	32949
::nb_29	= %10000011 +%11000001 *256	32950
::nb_2A	= %10000001 +%10000001 *256	32951
::nb_2B	= %10000000 +%00000001 *256	32952
::nb_2C	= %10011111 +%11111001 *256	32953
::nb_2D	= %10000000 +%00000001 *256	32954
::nb_2E	= %10000000 +%00000001 *256	32955
::nb_2F	= %11111111 +%11111111 *256	32956
		32957
::nb_30	= %11111111 +%11111111 *256	32958
::nb_31	= %10000000 +%00000001 *256	32959
::nb_32	= %10000000 +%00000001 *256	32960
::nb_33	= %10011111 +%11111001 *256	32961
::nb_34	= %10000000 +%00000001 *256	32962
::nb_35	= %10000001 +%10000001 *256	32963
::nb_36	= %10000011 +%11000001 *256	32964
::nb_37	= %10000111 +%11100001 *256	32965
::nb_38	= %10001111 +%11110001 *256	32966
::nb_39	= %10000001 +%10000001 *256	32967
::nb_3A	= %10000001 +%10000001 *256	32968
::nb_3B	= %10000001 +%10000001 *256	32969
::nb_3C	= %10000001 +%10000001 *256	32970
::nb_3D	= %10000000 +%00000001 *256	32971
::nb_3E	= %10000000 +%00000001 *256	32972
::nb_3F	= %11111111 +%11111111 *256	32973
		32974
::nb_40	= %11111111 +%11111111 *256	32975
::nb_41	= %10000000 +%00000001 *256	32976
::nb_42	= %10000000 +%00000001 *256	32977
::nb_43	= %10011111 +%11111001 *256	32978
::nb_44	= %10010000 +%00001001 *256	32979
::nb_45	= %10010000 +%00001001 *256	32980
::nb_46	= %10010001 +%10001001 *256	32981
::nb_47	= %10010001 +%10001001 *256	32982
::nb_48	= %10010111 +%11101001 *256	32983
::nb_49	= %10010011 +%11001001 *256	32984
::nb_4A	= %10010001 +%10011001 *256	32985
::nb_4B	= %10010000 +%00101001 *256	32986
::nb_4C	= %10011111 +%11111001 *256	32987
::nb_4D	= %10000000 +%00000001 *256	32988
::nb_4E	= %10000000 +%00000001 *256	32989
::nb_4F	= %11111111 +%11111111 *256	32990
		32991
::nb_50	= %11111111 +%11111111 *256	32992
::nb_51	= %10000000 +%00000001 *256	32993
::nb_52	= %10000000 +%00000001 *256	32994
::nb_53	= %10011111 +%11111001 *256	32995
::nb_54	= %10010000 +%00001001 *256	32996

```

32997 ::nb_55          = %10010000 +%00001001 *256
32998 ::nb_56          = %10010001 +%10001001 *256
32999 ::nb_57          = %10010011 +%11001001 *256
33000 ::nb_58          = %10010111 +%11101001 *256
33001 ::nb_59          = %10010001 +%10001001 *256
33002 ::nb_5A          = %10010001 +%10011001 *256
33003 ::nb_5B          = %10010000 +%00101001 *256
33004 ::nb_5C          = %10011111 +%11111001 *256
33005 ::nb_5D          = %10000000 +%00000001 *256
33006 ::nb_5E          = %10000000 +%00000001 *256
33007 ::nb_5F          = %11111111 +%11111111 *256
33008
33009 ; Navbar-Icon-Definition.
33010          b $80 +6*2*2
33011          w :nb_00,:nb_10,:nb_20          ;Zeile #1.
33012          w :nb_30,:nb_40,:nb_50
33013          w :nb_01,:nb_11,:nb_21          ;Zeile #2.
33014          w :nb_31,:nb_41,:nb_51
33015
33016          b $80 +6*2*2
33017          w :nb_02,:nb_12,:nb_22          ;Zeile #3.
33018          w :nb_32,:nb_42,:nb_52
33019          w :nb_03,:nb_13,:nb_23          ;Zeile #4.
33020          w :nb_33,:nb_43,:nb_53
33021
33022          b $80 +6*2*2
33023          w :nb_04,:nb_14,:nb_24          ;Zeile #5.
33024          w :nb_34,:nb_44,:nb_54
33025          w :nb_05,:nb_15,:nb_25          ;Zeile #6.
33026          w :nb_35,:nb_45,:nb_55
33027
33028          b $80 +6*2*2
33029          w :nb_06,:nb_16,:nb_26          ;Zeile #7.
33030          w :nb_36,:nb_46,:nb_56
33031          w :nb_07,:nb_17,:nb_27          ;Zeile #8.
33032          w :nb_37,:nb_47,:nb_57
33033
33034          b $80 +6*2*2
33035          w :nb_08,:nb_18,:nb_28          ;Zeile #9.
33036          w :nb_38,:nb_48,:nb_58
33037          w :nb_09,:nb_19,:nb_29          ;Zeile #10.
33038          w :nb_39,:nb_49,:nb_59
33039
33040          b $80 +6*2*2
33041          w :nb_0A,:nb_1A,:nb_2A          ;Zeile #11.
33042          w :nb_3A,:nb_4A,:nb_5A
33043          w :nb_0B,:nb_1B,:nb_2B          ;Zeile #12.
33044          w :nb_3B,:nb_4B,:nb_5B
33045
33046          b $80 +6*2*2
33047          w :nb_0C,:nb_1C,:nb_2C          ;Zeile #13.
33048          w :nb_3C,:nb_4C,:nb_5C
33049          w :nb_0D,:nb_1D,:nb_2D          ;Zeile #14.
33050          w :nb_3D,:nb_4D,:nb_5D
33051

```

	b \$80 +6*2*2		33052
	w :nb_0E,:nb_1E,:nb_2E	;Zeile #15.	33053
	w :nb_3E,:nb_4E,:nb_5E		33054
	w :nb_0F,:nb_1F,:nb_2F	;Zeile #16.	33055
	w :nb_3F,:nb_4F,:nb_5F		33056
			33057
	; CMD: Partitionsliste erzeugen.		33058
:z723F	lda #< BASEDATA		33059
	sta r6L		33060
	lda #> BASEDATA		33061
	sta r6H		33062
			33063
	lda #\$00		33064
	sta v7656	;Anzahl Dateien.	33065
			33066
	jsr PurgeTurbo		33067
	jsr InitForIO		33068
			33069
	lda #\$00		33070
	sta STATUS		33071
			33072
	lda curDrive		33073
	pha		33074
			33075
	tay		33076
	lda V04B8 -8,y	;Geräteinfo einlesen.	33077
	and #%00000011	;RAMLink/RAMDrive?	33078
	beq :1	; => Nein, weiter...	33079
			33080
	lda C0509	;RAMLink-Adresse.	33081
	sta curDevice		33082
	sta curDrive		33083
			33084
::1	lda curDrive		33085
	jsr LISTEN		33086
	lda #\$f0		33087
	jsr SECOND		33088
			33089
	bit STATUS		33090
	bpl :2		33091
			33092
	jsr UNLSN		33093
	jmp :error		33094
			33095
::2	ldy #0	;Verzeichnis mit	33096
::send	lda v7657,y	;Partitionen abrufen.	33097
	jsr CIOUT		33098
	iny		33099
	cpy #v7657_len		33100
	bne :send		33101
			33102
	jsr UNLSN		33103
			33104
	lda #\$00		33105
	sta STATUS		33106

```

33107
33108          lda    curDrive          ;Laufwerk auf
33109          jsr    TALK              ;Senden umschalten.
33110          lda    #$f0
33111          jsr    TKSA
33112
33113          jsr    ACPTR              ;Daten abrufen.
33114
33115          bit    STATUS             ;Fehler?
33116          bvc    :3                ; => Nein, weiter...
33117
33118          jsr    UNTALK
33119          jmp    :error
33120
33121  ::3          ldy    #31            ;Verzeichnisheader
33122  ::skip1      jsr    ACPTR          ;überlesen.
33123          dey
33124          bne    :skip1
33125
33126  ::loop       jsr    ACPTR
33127          cmp    #$00              ;Ende erreicht?
33128          beq    :end              ; => Ja, Ende...
33129
33130          jsr    ACPTR
33131          jsr    ACPTR
33132
33133          ldx    v7656              ;Anzahl Partitionen.
33134          sta    v4110,x            ;Part.-Nr in Tab.
33135          inc    v7656              ;Anzahl Part. +1.
33136
33137  ::skip3      jsr    ACPTR
33138          cmp    #$22              ;Anfang Name suchen.
33139          bne    :skip3
33140
33141          ldy    #0
33142  ::11         jsr    ACPTR          ;Part.-Name einlesen.
33143          cmp    #$22
33144          beq    :12
33145          sta    (r6),y
33146          iny
33147          bne    :11
33148
33149  ::12         cpy    #17            ;Max. 16 Zeichen.
33150          beq    :13
33151
33152          lda    #$00              ;Name mit $00
33153          sta    (r6),y            ;abschließen.
33154          iny
33155          bne    :12
33156
33157  ::13         lda    r6L            ;Zeiger auf
33158          clc                      ;nächsten Namen.
33159          adc    #< 17
33160          sta    r6L
33161          lda    r6H

```

```

                                adc    #> 17                33162
                                sta    r6H                33163
                                33164
::skip2                        jsr    ACPTR                ;Eintrag bis zum 33165
                                cmp    #$00                ;Ende überlesen. 33166
                                bne    :skip2              33167
                                beq    :loop               33168
                                33169
::end                          jsr    UNTALK              33170
                                33171
                                lda    #$00              33172
                                sta    STATUS             33173
                                33174
                                lda    curDrive           33175
                                jsr    LISTEN             33176
                                lda    #$e0              33177
                                jsr    SECOND             33178
                                33179
                                jsr    UNLSN              33180
                                33181
                                ldx    #NO_ERROR           33182
                                b $2c                    33183
::error                        ldx    #DEV_NOT_FOUND      33184
                                33185
                                pla                      ;Laufwerk wieder 33186
                                sta    curDrive           ;zurücksetzen. 33187
                                sta    curDevice          33188
                                33189
                                jmp     DoneWithIO        33190
                                33191
; Partitionsverzeichnis. 33192
:v7657                        b "$=P:*=8"                33193
:v7657_end                    33194
:v7657_len                    = v7657_end - v7657        33195
                                33196
; CMD-Laufwerk: Partition öffnen. 33197
; 33198
; Hinweis: 33199
; Die Routine muss am Ende in jedem 33200
; Fall den Speicher wieder herstellen! 33201
:z6FEC                        jsr    z71BC                ;Part. wechseln. 33202
                                33203
                                ldx    C042A              ;Oberstes Fenster. 33204
                                lda    v765F +2           ;Part. für "C-P" = 33205
                                cmp    C04FC,x            ;Part. für Fenster. 33206
                                beq    :err               33207
                                33208
; Hinweis: 33209
; Im Text-Modus wurde die aktive 33210
; Partition bisher nicht gespeichert. 33211
; Der Fehler lag aber beim öffnen der 33212
; vorherigen Partition und dem Aufruf 33213
; von S4296 über einen JSR-Befehl. 33214
; 33215
; => Fehler behoben, kann entfallen. 33216

```

```

33217 ;---
33218 ;          lda    C04B2          ;Icon-/Text-Modus.
33219 ;          bne    :1            ; => Text-Modus...
33220 ;---
33221
33222 ; Aktive Partition speichern.
33223 ;          ldx    C042A          ;Oberstes Fenster.
33224 ;          lda    C04F8,x        ;Lfwk. für Fenster.
33225 ;          sta    V21B1,x        ;Letztes Lfwk./Fens.
33226 ;          tay
33227 ;          lda    RealDrvType -8,y
33228 ;          sta    V5C34,x        ;Letzter LTyp./Fens.
33229 ;          lda    C04FC,x        ;Part. für Fenster.
33230 ;          sta    V0728,x        ;Letzte Part./Fens.
33231
33232 ::1          lda    v765F +2      ;Neue Partition =
33233 ;          sta    C04FC,x        ;Part. für Fenster.
33234
33235 ;          jsr    z6F7C          ;Speicher einlesen.
33236 ;          ldx    #$00          ; => Kein Fehler.
33237 ;          rts
33238
33239 ::err        jsr    z6F7C          ;Speicher einlesen.
33240 ;          ldx    #$ff          ; => Fehler, Ende...
33241 ;          rts
33242
33243 ; Befehl für Partitionswechsel.
33244 :v765F        b $43,$d0    ;"cP"
33245 ;          b $00          ;Wird verändert...
33246 ;          b CR,0
33247
33248 ; CMD-Partition wechseln.
33249 :z71BC        ldx    v7656          ;Anzahl Einträge.
33250 ;          bne    :1            ; => Mehr als 1 Part.
33251
33252 ;          ldx    #FILE_NOT_FOUND
33253 ;          rts
33254
33255 ::1          ldx    v7652          ;Markierter Eintrag.
33256 ;          inx              ; = Partition 1-254.
33257
33258 ;          jsr    z71D3          ;"C-P" senden.
33259 ;          txa              ;Fehler?
33260 ;          bne    :done         ; => Ja, Abbruch...
33261
33262 ;          ldy    curDrive
33263 ;          lda    driveType -8,y ;RL/RD oder FD/HD?
33264 ;          bpl    :done         ; => FD/HD, Ende.
33265
33266 ; CMD-RL/RD-Partiton setzen.
33267 ::rl_rd      lda    v765F +2      ;Daten für neue
33268 ;          jsr    z733C          ;Partition einlesen.
33269 ;          txa
33270 ;          bne    :done
33271

```

	ldy	curDrive	;ramBase setzen.	33272
	lda	v40EF +20		33273
	sta	ramBase -8,y		33274
	lda	v40EF +21	;Low-Byte wird nicht	33275
	sta	driveData +3	;mehr verwendet.	33276
				33277
;	ldx	#NO_ERROR		33278
::done	rts			33279
				33280
;			CMD-Partition über "C-P"-Befehl setzen.	33281
;			Übergabe: X = Partition 1-254.	33282
:z71D3	dex			33283
				33284
	lda	v4110,x	;Part.-Nr aus Tab.	33285
	sta	v765F +2	;Part.-Nr. für "C-P".	33286
				33287
	jsr	PurgeTurbo		33288
	jsr	InitForIO		33289
				33290
	lda	curDrive	;Aktuelles Laufwerk	33291
	pha		;zwischenspeichern.	33292
				33293
	tay			33294
	lda	V04B8 -8,y	;Geräteinfo einlesen.	33295
	and	##00000011	;RAMLink/RAMDrive?	33296
	beq	:1	; => Nein, weiter...	33297
				33298
	lda	C0509	;RAMLink-Adresse.	33299
	sta	curDevice		33300
	sta	curDrive		33301
				33302
::1	lda	#< v765F	;"C-P" + Part.-Nr.	33303
	sta	r0L		33304
	lda	#> v765F		33305
	sta	r0H		33306
				33307
	jsr	z73AF	;Befehl senden.	33308
	txa			33309
	pha			33310
				33311
	jsr	UNLSN		33312
				33313
	pla			33314
	tax			33315
				33316
	pla		;Laufwerk wieder	33317
	sta	curDevice	;zurücksetzen.	33318
	sta	curDrive		33319
				33320
	txa			33321
	beq	:done		33322
				33323
	ldx	#FILE_NOT_FOUND		33324
::done	jmp	DoneWithIO		33325
				33326

```

33327 ; Auf 64Net testen.
33328 :z76B4          jsr    z7A42          ;64Net bereit?
33329
33330 if DEBUG_64NET = FALSE
33331             txa
33332             beq     :done              ; => Ja, weiter...
33333 endif
33334
33335             lda     #< d7B7D          ;Fehler: 64Net ist
33336             ldx     #> d7B7D          ;nicht bereit.
33337             jsr     S2090              ;DBox ohne Recover.
33338
33339 if DEBUG_64NET = FALSE
33340             ldx     #DEV_NOT_FOUND
33341 else
33342             ldx     #NO_ERROR
33343 endif
33344 ::done      rts
33345
33346 ; 64Net-Laufwerke suchen/erkennen.
33347 :z79FF        lda     curDrive
33348             sta     a9L
33349             bne     :2
33350
33351 ::1           lda     driveType -8,x
33352             beq     :3                  ; => Kein Laufwerk.
33353             txa
33354             jsr     SetDevice          ;Lfwk. aktivieren.
33355
33356 ::2           jsr     z7A28            ;Auf 64Net testen.
33357
33358             ldx     curDrive
33359 if DEBUG_64NET = TRUE
33360             cpx     #10                ;Laufwerk #10 ist
33361             bne     :3                  ;immer 64Net.
33362             lda     #$ff
33363 endif
33364 ::3           sta     v764D -8,x        ;$FF: 64Net-Treiber.
33365
33366             inx
33367             cpx     #12
33368             bne     :4
33369             ldx     #8
33370 ::4           cpx     a9L                ;Alle Lfwk. getestet?
33371             bne     :1                  ; => Nein, weiter...
33372
33373             txa
33374             jmp     SetDevice          ;Laufwerk wieder
33375                                     ;zurücksetzen.
33376
33376 ; Auf 64Net-Treiber testen.
33377 :z7A28        ldx     #0
33378 ::1           lda     DiskDrvVersion,x
33379             cmp     v7A3D,x
33380             bne     :err
33381             inx

```



	cpx	#v7A3D_len		33382
	bne	:1		33383
				33384
	lda	#\$ff	;64Net-Treiber.	33385
	rts			33386
				33387
::err	lda	#\$00	;Kein 64Net-Treiber.	33388
	rts			33389
				33390
; 64Net: Partitionsliste erzeugen.				33391
:z77B8	jsr	z7A42	;Init 64Net.	33392
	txa			33393
	beq	:init	; => Ja, weiter...	33394
				33395
	ldx	#DEV_NOT_FOUND		33396
	rts			33397
				33398
::init	jsr	z7907	;Anz. Part. holen.	33399
	sta	v7651	;Anzahl Partitionen.	33400
	tax		;Part. vorhanden?	33401
	beq	:done	; => Nein, Ende...	33402
				33403
	lda	#< BASEDATA		33404
	sta	r6L		33405
	lda	#> BASEDATA		33406
	sta	r6H		33407
				33408
;	lda	#< v4210	;r0 bereits gesetzt	33409
;	sta	r0L	;durch z7907.	33410
;	lda	#> v4210		33411
;	sta	r0H		33412
				33413
	lda	#\$00		33414
	sta	v7656	;Zähler Partitionen.	33415
				33416
::loop	ldx	v7656	;Aktuelle Partition.	33417
	inx		;Nr. 1-255.	33418
	stx	r1L		33419
				33420
;	lda	#\$00	;r1H/r2L sind bereits	33421
;	sta	r1H	;\$00 durch z7907.	33422
;	lda	#\$00		33423
;	sta	r2L		33424
				33425
	jsr	z7A8D	;Block empfangen.	33426
				33427
	lda	v4210 +\$80	;Neue Partition vom	33428
	cmp	#3	;Typ 0(41)-2(81)?	33429
	bcs	:next	; => Nein, weiter...	33430
				33431
	ldy	#0	;Diskname	33432
::1	lda	v4210 +6,y	;einlesen.	33433
	sta	(r6),y		33434
	beq	:2		33435
	iny			33436

```

33437         bne      :1
33438
33439 ::2         lda      #NULL           ;Mit NULL-Bytes
33440 ::3         sta      (r6),y         ;auffüllen.
33441         iny
33442         cpy      #17
33443         bne      :3
33444
33445         lda      r6L               ;Zeiger auf
33446         clc                               ;nächsten Namen.
33447         adc      #< 17
33448         sta      r6L
33449         lda      r6H
33450         adc      #> 17
33451         sta      r6H
33452
33453 ::next      inc      v7656           ;Anzahl Dateien +1.
33454         lda      v7656           ;Anzahl Dateien.
33455         cmp      v7651
33456         bne      :loop
33457
33458         ldx      #$00             ;Tabelle mit
33459         ldy      #$01             ;Partitions-Nr. von
33460 ::11        tya                               ;1-x erzeugen.
33461         sta      v4110,x
33462         iny
33463         inx
33464         cpx      v7656           ;Anzahl Dateien.
33465         bne      :11
33466
33467         ldx      #$00
33468 ::done      rts
33469
33470 ; 64Net-Laufwerk: Partition öffnen.
33471 ;
33472 ; Hinweis:
33473 ; Die Routine muss am Ende in jedem
33474 ; Fall den Speicher wieder herstellen!
33475 :z76B8      ldx      v7652           ;=> Neue 64Net-Part.
33476
33477         inx                               ;Daten für Partition
33478         stx      r1L             ;einlesen (1-255).
33479         lda      #$00
33480         sta      r1H
33481         sta      r2L
33482
33483         lda      #< v4210         ;Zwischenspeicher
33484         sta      r0L             ;Für Part.-Info.
33485         lda      #> v4210
33486         sta      r0H
33487
33488         jsr      z7A8D           ;Block empfangen.
33489
33490         lda      #$00           ;Systemblock
33491         sta      r1L           ;einlesen.

```

		lda	#\$00		33492
		sta	r1H		33493
		lda	#\$00		33494
		sta	r2L		33495
					33496
		lda	#< diskBlkBuf	;Zwischenspeicher	33497
		sta	r0L	;für Systemblock.	33498
		lda	#> diskBlkBuf		33499
		sta	r0H		33500
					33501
		jsr	z7A8D	;Block empfangen.	33502
					33503
					33504
		ldy	#8		33505
		cpy	curDrive		33506
		beq	:12		33507
					33508
		lda	v764D -8,y	;64Net-Laufwerk?	33509
		beq	:12	; => Nein, weiter...	33510
					33511
		lda	diskBlkBuf +\$78,y		33512
		cmp	v7652	;Partition aktiv?	33513
		beq	:fail	; => Ja, Fehler...	33514
					33515
		iny			33516
		cpy	#12		33517
		bne	:11		33518
					33519
		ldx	v7652	;=> Neue 64Net-Part.	33520
		jsr	z782B	;Partition setzen.	33521
					33522
					33523
		ldx	C042A	;Oberstes Fenster.	33524
		lda	v7652	;Markierte Datei.	33525
		sta	C04FC,x	;Part. für Fenster.	33526
					33527
		ldy	#10		33528
		jsr	:fail	;Dialogbox testen.	33529
					33530
					33531
		jsr	z6F7C	;Speicher einlesen.	33532
					33533
		ldx	#\$00		33534
		rts			33535
					33536
					33537
		tya			33538
		clc			33539
		adc	#"A" -8		33540
		sta	v7788		33541
					33542
		lda	#< d774A	;Fehler: Partition	33543
		ldx	#> d774A	;bereits aktiv.	33544
		jsr	S2090	;DBox ohne Recover.	33545
					33546

```

33547 if DEBUG_64NET = FALSE
33548         jsr    z6F7C                ;Speicher einlesen.
33549 endif
33550
33551         ldx    #$ff
33552         rts
33553
33554 ; 64Net-Partition aktivieren, ggf. Treiber wechseln.
33555 ; Übergabe: AKKU = Part.-Nr.
33556 ;           v4210 = Part.-Info.
33557 :z782B      sta    v7888                ;64Net-Partition-Nr.
33558
33559         ldy    curDrive
33560         lda    driveType -8,y
33561         sta    v7889                ;64Net-Lfwk.-Modus.
33562
33563         ldx    v4210 +$80            ;Partitionsformat
33564         inx                      ;0-2 nach 41-81
33565         stx    v788A                ;wandeln.
33566
33567 if DEBUG_64NET = FALSE
33568         txa
33569         sta    driveType -8,y        ;Lfwk. konfigurieren.
33570         sta    RealDrvType -8,y
33571 endif
33572
33573         lda    #< v4210
33574         sta    r0L
33575         lda    #> v4210
33576         sta    r0H
33577
33578         lda    #$00                ;Systemblock.
33579         sta    r1L
33580 ;         lda    #$00
33581         sta    r1H
33582 ;         lda    #$00
33583         sta    r2L
33584
33585         jsr    z7A8D                ;Block empfangen.
33586
33587         ldy    curDrive
33588         lda    v7888                ;64Net-Partition-Nr.
33589         sta    v4210 +$80 -8,y
33590
33591 if DEBUG_64NET = FALSE
33592         lda    #$00                ;Nur 64Net:
33593         sta    $9054                ;Systemblock geändert.
33594 endif
33595
33596         jsr    z7AC4                ;Block senden.
33597
33598         lda    v7889                ;64Net-Lfwk.-Modus.
33599         cmp    #Drv1581            ;1541/71?
33600         bcc    :41_71              ; => Ja, weiter...
33601

```

```

::81          lda    v788A          ;Neue Part.1581.          33602
               cmp    #Drv1581      ;Bereits aktiv?          33603
               beq    :done         ; => Ja, Ende...         33604
               ;                                     33605
::set         jmp    z788B          ;64Net-Treiber laden.    33606
               ;                                     33607
::41_71       lda    v788A          ;Neue Prt. 1541/71.    33608
               cmp    #Drv1581      ;Bereits aktiv?          33609
               beq    :set          ; => Nein, Treiber.        33610
               ;                                     33611
::done        rts                  33612
               ;                                     33613
; 64Net-Treiber 1541/71/81 laden. 33614
:z788B        lda    #< v4310       ;Ablagebereich für      33615
               sta    r0L           ;Laufwerkstreiber.        33616
               lda    #> v4310       33617
               sta    r0H           33618
               ;                                     33619
::41_71       lda    #$00           ;Startadr. Treiber      33620
               sta    r1L           ;64Net für 1541/71.        33621
               lda    #$01           33622
               sta    r1H           33623
               lda    #$00           33624
               sta    r2L           33625
               ;                                     33626
               ldy    curDrive       33627
               lda    driveType -8,y 33628
               cmp    #Drv1581      ;1541/71?              33629
               bcc    :1             ; => Ja, weiter...        33630
               ;                                     33631
::81          lda    #$0e           ;Startadr. Treiber      33632
               sta    r1L           ;64Net für 1581.          33633
               lda    #$01           33634
               sta    r1H           33635
               lda    #$00           33636
               sta    r2L           33637
               ;                                     33638
::1           lda    #$00           ;Blockzähler löschen. 33639
               sta    v7906         33640
               ;                                     33641
::loop        jsr    z7A8D          ;Block empfangen.      33642
               ;                                     33643
               inc    v7906         ;Blockzähler +1.        33644
               ;                                     33645
               inc    r0H           ;Highb. RAM-Adresse.    33646
               inc    r1L           ;Blockadresse +1.        33647
               ;                                     33648
               lda    v7906         ;Mind. $0d80 Bytes      33649
               cmp    #14           ; = 14 Blocks?          33650
               bne    :loop         ; => Nein, weiter...        33651
               ;                                     33652
if DEBUG_64NET = FALSE 33653
               lda    #< v4310       ;Treiber in REU        33654
               sta    r0L           ;speichern.              33655
               lda    #> v4310       33656

```

```

33657          sta    r0H
33658
33659          ldx    curDrive
33660          lda    DskDrvBaseL -8,x
33661          sta    r1L
33662          lda    DskDrvBaseH -8,x
33663          sta    r1H
33664
33665          lda    #< DISK_SIZE
33666          sta    r2L
33667          lda    #> DISK_SIZE
33668          sta    r2H
33669
33670          lda    #$00
33671          sta    r3L
33672
33673          jsr    StashRAM
33674
33675          jsr    i_MoveData          ;Treiber aktivieren.
33676          w v4310
33677          w DISK_BASE
33678          w DISK_SIZE
33679 endif
33680
33681          rts
33682
33683 ; Anzahl Partition einlesen.
33684 :z7907          lda    #< v4210
33685          sta    r0L
33686          lda    #> v4210
33687          sta    r0H
33688
33689          lda    #$00          ;Systemblock.
33690          sta    r1L
33691 ;          lda    #$00
33692          sta    r1H
33693 ;          lda    #$00
33694          sta    r2L
33695
33696          jsr    z7A8D          ;Block empfangen.
33697
33698          ldx    #0
33699 ::1          lda    v7A3D,x          ;64Net-Kennung.
33700          cmp    v4210 +1,x
33701          bne    :fail
33702          inx
33703          cpx    #v7A3D_len
33704          bne    :1
33705
33706 if DEBUG_64NET = TRUE
33707          lda    r0H
33708          pha
33709          lda    r0L
33710          pha
33711          lda    r1H

```

```

        pha                                33712
        lda    r2L                        33713
        pha                                33714
        jsr    z798A                      ;Dialogbox testen. 33715
        pla                                33716
        sta    r2L                        33717
        pla                                33718
        sta    r1H                        33719
        pla                                33720
        sta    r0L                        33721
        pla                                33722
        sta    r0H                        33723
endif                                     33724
                                           33725
        lda    v4210                      ;Anzahl Partitionen. 33726
        rts                                33727
                                           33728
; 64Net-Laufwerk formatieren.           33729
::fail        ldx    #$00                 33730
              txa                        33731
::11          sta    v4210,x              33732
              inx                        33733
              bne     :11                 33734
                                           33735
;                                           33736
::12          ldx    #0                   33737
              lda    v7A3D,x              33738
              sta    v4210 +1,x           33739
              inx                        33740
              cpx    #v7A3D_len           33741
              bne     :12                 33742
                                           33743
              lda    #$ff                 33744
              ldx    #$00                 33745
::13          sta    v4210 +$80,x         33746
              inx                        33747
              cpx    #$10                 33748
              bne     :13                 33749
                                           33750
              lda    #$1b                  ;Ende Systembereich. 33751
              sta    v4210 +$7c           33752
              lda    #$01                 33753
              sta    v4210 +$7d           33754
              lda    #$00                 33755
              sta    v4210 +$7e           33756
                                           33757
              lda    #< v4210             33758
              sta    r0L                  33759
              lda    #> v4210             33760
              sta    r0H                  33761
                                           33762
              lda    #$00                  ;Systemblock.      33763
              sta    r1L                  33764
;                                           33765
              lda    #$00                  33766

```

```

33767          sta    r2L
33768
33769          jsr     z7AC4          ;Block senden.
33770          jsr     z7A8D          ;Block empfangen.
33771
33772          ldx     #0
33773 :::21      lda     v7A3D,x        ;Kennung 64Net.
33774          cmp     v4210 +1,x
33775          bne     z798A          ; => Fehler.
33776          inx
33777          cpx     #v7A3D_len
33778          bne     :21
33779
33780          lda     v4210          ;Anzahl Partitionen.
33781          rts
33782
33783 ; Fehler: Nicht lizenziert.
33784 :z798A      lda     #< d799A
33785          ldx     #> d799A
33786          jsr     S2090          ;DBox ohne Recover.
33787
33788          ldx     #DEV_NOT_FOUND
33789          rts
33790
33791 ; Block von 64Net empfangen.
33792 ; $0000      = Systemblock:
33793 ;           $xx00      = Anzahl Partitionen.
33794 ;           $xx01-$xx06 = "64NET"-Kennung.
33795 ;           $xx7C-$xx7E = Low/Mid/High-Byte = $00:011B
33796 ;           für Ende Systembereich.
33797 ;           $xx80-$xx83 = Aktive Part. für Lfwk. 8-11.
33798 ;           $xx84-$xx8F = $FF.
33799 ; $0001-$00FF = Daten für Partition 1-255:
33800 ;           $xx06-$xx15 =16Z. Diskname.
33801 ;           $xx80      = 0(41), 1(71) oder 2(81).
33802 ; $0100-$010D = Treiber 1541/71.
33803 ; $010E-$011B = Treiber 1581.
33804 :z7A8D      php
33805          sei
33806
33807 if DEBUG_64NET = FALSE
33808          ldx     CPU_DATA
33809 ;          lda     #$b5
33810          lda     #%10110101      ;Wozu Bit%7?
33811          sta     CPU_DATA
33812
33813          lda     #"R"          ;Modus "R" = Read.
33814          jsr     z7AF0          ;Befehl senden.
33815
33816          ldy     #$00
33817 :::1      jsr     z7B4F          ;Byte empfangen.
33818          sta     (r0),y
33819          iny
33820          bne     :1
33821

```



```

        jsr    z7B4F                ;Byte empfangen.          33822
                                           33823
        stx    CPU_DATA              33824
else                                           33825
; Hinweis:                                   33826
; Im Debug-Modus sind der Systemblock       33827
; und Part.-Info vordefiniert.              33828
        lda    r1H                  ;Laufwerk-Treiber?       33829
        beq    :11                  ; => Nein, weiter...       33830
                                           33831
        ldy    #0                   ;Block löschen.          33832
        tya                                           33833
::1      sta    (r0),y              33834
        iny                                           33835
        bne    :1                   33836
                                           33837
        plp                                           33838
        rts                                           33839
                                           33840
::11     lda    r1L                  ;Systemblock?            33841
        bne    :21                  ; => Nein, weiter...       33842
                                           33843
        ldy    #0                   33844
::12     lda    sdata,y              33845
        sta    (r0),y              33846
        iny                                           33847
        bne    :12                  33848
                                           33849
        plp                                           33850
        rts                                           33851
                                           33852
::21     ldy    #0                   ;Part.-Info.             33853
::22     lda    pdata,y              33854
        sta    (r0),y              33855
        iny                                           33856
        bne    :22                  33857
                                           33858
        lda    driveType +10 -8      33859
        and    #ST_DM0DES            33860
        tax                                           33861
        dex                                           33862
        stx    ptype                33863
                                           33864
        plp                                           33865
        rts                                           33866
                                           33867
; Daten für Systemblock:                    33868
:sdata   b 2                          33869
        b "64NET"                    33870
        e sdata +$7c                 33871
        b $1b,$01,$00                33872
        e sdata +$80                 33873
        b $ff,$ff,$ff,$ff,$ff,$ff,$ff,$ff,$ff 33874
        b $ff,$ff,$ff,$ff,$ff,$ff,$ff,$ff,$ff 33875
        e sdata +256                 33876

```

```

33877
33878 ; Part.-Information.
33879 :pdata          e pdata +$06
33880                b "64Net-Partition",0
33881                e pdata +$80
33882 :ptype          b 2
33883                e pdata +256
33884 endif
33885
33886                plp
33887                rts
33888
33889 ; Block an 64Net senden.
33890 :z7AC4          php
33891                sei
33892
33893 if DEBUG_64NET = FALSE
33894                ldx    CPU_DATA
33895 ;                lda    #$b5
33896                lda    #%10110101          ;Wozu Bit%7?
33897                sta    CPU_DATA
33898
33899                lda    #"W"                ;Modus "W" = Write.
33900                jsr    z7AF0                ;Befehl senden.
33901
33902                ldy    #$00
33903 ::1            lda    (r0),y
33904                jsr    z7AF8                ;Byte senden.
33905                iny
33906                bne    :1
33907
33908                stx    CPU_DATA
33909 endif
33910
33911                plp
33912                rts
33913
33914 ; Befehl Lesen/Schreiben an Server senden.
33915 ; Übergabe: A = "R": Block lesen, "W": Block schreiben.
33916 :z7AF0          pha
33917                lda    #"X"                ;Kennbyte.
33918                jsr    z7AF8                ;Byte senden.
33919                lda    #"L"
33920                jsr    z7AF8                ;Byte senden.
33921                pla                ;Modus "R" oder "W".
33922                jsr    z7AF8                ;Byte senden.
33923
33924                lda    r1L                ;Low-Byte.
33925                jsr    z7AF8                ;Byte senden.
33926                lda    r1H                ;Mid-Byte.
33927                jsr    z7AF8                ;Byte senden.
33928                lda    r2L                ;High-Byte.
33929 ;                jsr    z7AF8                ;Byte senden.
33930
33931 ; Byte an 64Net senden.

```

```

; Übergabe: A = Datenbyte.                                     33932
; Rückgabe: -                                                 33933
; Ändert : A, r2H                                             33934
:z7AF8      sta    r2H                                         33935
            and    #%00000011                                   33936
            ora    #%00000100           ;Neues Datenbyte.      33937
            sta    $dd01                                         33938
::1         lda    $dd01                                         33939
            and    #%01110000                                   33940
            cmp    #%01010000                                   33941
            bne    :1                                           33942
                                                    33943
            lda    r2H                                         33944
            lsr                                           33945
            lsr                                           33946
            sta    r2H                                         33947
            and    #%00000011                                   33948
            sta    $dd01                                         33949
::2         lda    $dd01                                         33950
            and    #%01110000                                   33951
            cmp    #%00100000                                   33952
            bne    :2                                           33953
                                                    33954
            lda    r2H                                         33955
            lsr                                           33956
            lsr                                           33957
            sta    r2H                                         33958
            and    #%00000011                                   33959
            ora    #%00000100           ;Bit%2=1:Mehr Daten.    33960
            sta    $dd01                                         33961
::3         lda    $dd01                                         33962
            and    #%01110000                                   33963
            cmp    #%01100000                                   33964
            bne    :3                                           33965
                                                    33966
            lda    r2H                                         33967
            lsr                                           33968
            lsr                                           33969
            and    #%00000011           ;Bit%7-%2 bereits 0.    33970
            sta    $dd01                                         33971
::4         lda    $dd01                                         33972
            and    #%01110000                                   33973
            cmp    #%00010000                                   33974
            bne    :4                                           33975
                                                    33976
            lda    #%00000011           ;Datenbyte komplett.    33977
            sta    $dd01                                         33978
            rts                                           33979
                                                    33980
; Byte von 64Net empfangen.                                     33981
; Übergabe: -                                                 33982
; Rückgabe: A = Datenbyte.                                     33983
; Ändert : A, r2H                                             33984
:z7B4F      lda    #%00000100           ;High-Nibble holen.    33985
            sta    $dd01                                         33986

```

```

33987 ::1          lda    $dd01
33988             and    #%00001000
33989             beq     :1
33990
33991             lda    $dd01
33992             lsr
33993             lsr
33994             lsr
33995             lsr
33996 ;             and    #%00001111          ;Bit%7-%4 bereits 0.
33997             sta    r2H
33998
33999             lda    #%00000000          ;Low-Nibble holen.
34000             sta    $dd01
34001 ::2          lda    $dd01
34002             and    #%00001000
34003             bne     :2
34004
34005             lda    $dd01
34006             and    #%11110000
34007             ora    r2H
34008             rts
34009
34010 ; 64Net-Server testen.
34011 :z7A42        cli                      ;IRQ freigeben, da
34012                                     ;hier dblClickCount
34013                                     ;verwendet wird!
34014
34015             ldx     CPU_DATA
34016
34017 if DEBUG_64NET = FALSE
34018 ;             lda    #$b5
34019             lda    #%10110101          ;Wozu Bit%7?
34020             sta    CPU_DATA
34021
34022             lda    #%00000111
34023             sta    $dd03
34024             lda    #%00000000
34025             sta    $dd01
34026
34027             lda    #%00001000
34028             sta    $dd01
34029
34030             lda    #50
34031             sta    dblClickCount
34032
34033 ::wait1        lda    $dd01
34034             and    #%11110000
34035             cmp     #%10000000
34036             beq     :1
34037
34038             lda    dblClickCount
34039             bne     :wait1
34040             beq     :err
34041

```

```

::1          lda    #%00000000          34042
              sta    $dd01              34043
::wait2      lda    $dd01              34044
              and    #%11110000        34045
              cmp    #%00000000        34046
              bne    :ok                34047
              lda    dblClickCount      34048
              bne    :wait2            34049
              lda    dblClickCount      34050
              bne    :wait2            34051
::err        stx    CPU_DATA            34052
              ldx    #$ff              34053
              rts                      34054
endif        34055
              34056
::ok         stx    CPU_DATA            34057
              ldx    #$00              34058
              rts                      34059
              34060
; Kennung für 64Net.                    34061
:v7A3D       b "64NET"                 34062
:v7A3D_end   34063
:v7A3D_len   = v7A3D_end - v7A3D       34064
              34065
:v7906       b $00 ;Blockzähler für Laufwerkstreiber. 34066
:v7888       b $00 ;64Net-Partition-Nr. 34067
:v7889       b $00 ;64Net-Lfwk.-Modus. 34068
:v788A       b $00 ;GEOS-Lfwk.-Modus. 34069
:v764D       s 4 ;$FF: 64Net-Treiber. 34070
              34071
; Dialogbox: 64Net nicht bereit.        34072
:d7B7D       b %10000001              34073
              34074
              b DBTXTSTR,$0c,$10      34075
              w T29CF                 34076
              34077
              b DBTXTSTR,$0c,$20      34078
              w :t1                   34079
              34080
              b CANCEL ,,$11,$48      34081
              34082
              b NULL                  34083
              34084
if LANG = LANG_DE                       34085
::t1         b "64Net-Server ist nicht bereit.",0 34086
endif        34087
if LANG = LANG_EN                       34088
::t1         b "64Net server not ready.",0 34089
endif        34090
              34091
; Dialogbox: 64Net nicht lizenziert.    34092
:d799A       b %10000001              34093
              34094
              b DBTXTSTR,$0c,$10      34095
              w T29CF                 34096

```

```
34097
34098          b DBTXTSTR,$0c,$20
34099          w :t1
34100          b DBTXTSTR,$0c,$2a
34101          w :t2
34102
34103          b CANCEL  , $11,$48
34104
34105          b NULL
34106
34107 if LANG = LANG_DE
34108 ::t1          b "Sie können die Funktion nur mit einer",0
34109 ::t2          b "registrierten 64Net-Version verwenden!",0
34110 endif
34111 if LANG = LANG_EN
34112 ::t1          b "You can only use this function with a",0
34113 ::t2          b "registered version of 64Net!",0
34114 endif
34115
34116 ; Dialogbox: Partition bereits aktiv.
34117 :d774A          b %10000001
34118
34119          b DBTXTSTR,$0c,$10
34120          w T29CF
34121
34122          b DBTXTSTR,$0c,$20
34123          w :t1
34124          b DBTXTSTR,$0c,$2a
34125          w :t2
34126
34127          b CANCEL  , $11,$48
34128
34129          b NULL
34130
34131 if LANG = LANG_DE
34132 ::t1          b "Die 64Net-Partition ist bereits",0
34133 ::t2          b "aktiv in Laufwerk "
34134 endif
34135 if LANG = LANG_EN
34136 ::t1          b "The 64Net partition is already",0
34137 ::t2          b "active on drive "
34138 endif
34139
34140          b BOLDON
34141 :v7788          b "x:"
34142          b PLAINTEXT
34143          b " !",0
34144
34145 ;*** Endadresse testen:
34146          g OS_VARS
34147
```

```

34148
; 34149
; SourceCode: src.mod#12.s 34150
; 34151
34152
; 34153
; Modul-Info: 34154
; 0 - Farben ändern 34155
; 1 - Uhrzeit ändern 34156
; 2 - Disk formatieren 34157
; 3 - Hilfesystem starten (Optional) 34158
; 34159
if .p 34160
    t "TopSym" 34161
    t "TopSym.MP3" 34162
    t "TopSym.ROM" 34163
    t "DTopDeskI18N.ext" 34164
    t "src.DTopDesk.ext" 34165
endif 34166
34167
    o BASE_MOD_A 34168
    n "obj.#12" 34169
34170
::ModEntry_0    jmp    N6F89    ;Farben ändern. 34171
::ModEntry_1    jmp    N7674    ;Uhrzeit ändern. 34172
::ModEntry_2    jmp    N7A06    ;Disk formatieren. 34173
34174
if ENABLE_GEOHELP = TRUE 34175
;:ModEntry_3    jmp    N6FF8    ;Hilfesystem starten. 34176
34177
; 3: Hilfesystem starten. 34178
:N6FF8          lda    curDrive    ;Akt. Laufwerk 34179
                sta    a9L        ;zwischenspeichern. 34180
34181
                jsr    z6EF3        ;GeoHelpView suchen. 34182
                bcs    :exit        ; => Nicht gefunden. 34183
34184
                lda    #< v6EE7    ;Zeiger Dateiname. 34185
                sta    r6L        34186
                lda    #> v6EE7    34187
                sta    r6H        34188
                jmp    S2931        ;Datei öffnen. 34189
34190
; Hilfesystem nicht gefunden. 34191
::exit          lda    a9L        ;Laufwerk wieder 34192
                jsr    SetDevice    ;zurücksetzen. 34193
34194
                ldx    v6EE6        ;Part. gewechselt? 34195
                beq    :1          ; => Nein, weiter... 34196
34197
; Hinweis: 34198
; S0723 ruft OpenPartition auf, dabei 34199
; wird auch OpenDisk ausgeführt. 34200
                ldx    v6EE6        ;Part. für Fenster. 34201
                jmp    S0723        ;Partition öffnen. 34202

```

```

34203
34204 ::1          jmp    OpenDisk          ;Disk öffnen, Ende.
34205
34206 ; Hilfesystem suchen.
34207 :z6EF3        lda    #$00              ;Part. für Fenster
34208              sta    v6EE6              ;nicht verändert.
34209
34210              lda    C050A              ;TopDesk/Partition.
34211              beq    :2                  ; => Nicht definiert.
34212
34213              ldx    C042A              ;Oberstes Fenster.
34214              lda    C0448,x            ;Fenster geöffnet?
34215              beq    :1                  ; => Nein, weiter...
34216              lda    C04FC,x            ;Part. für Fenster.
34217              sta    v6EE6
34218
34219 ::1          lda    C0424              ;TopDesk/Laufwerk.
34220              jsr    SetDevice
34221
34222 ; Hinweis:
34223 ; S0723 ruft OpenPartition auf, dabei
34224 ; wird auch OpenDisk ausgeführt.
34225 ;              jsr    OpenDisk
34226
34227              ldx    C050A              ;TopDesk/Partition.
34228              jsr    S0723              ;Partition öffnen.
34229
34230              lda    curDrive
34231              bne    :3
34232
34233 ::2          lda    C0424              ;TopDesk/Laufwerk.
34234 ::3          sta    v6F68              ;Als 1.Lfwk. setzen.
34235              jsr    SetDevice          ;Lfwk. aktivieren.
34236
34237 ; Hilfesystem auf akt. Lfwk. suchen.
34238 ::loop        jsr    OpenDisk          ;Diskete öffnen.
34239              txa
34240              bne    :next              ; => Fehler...
34241
34242              lda    #< v6EE7
34243              sta    r6L
34244              lda    #> v6EE7
34245              sta    r6H
34246              jsr    FindFile            ;GeoHelpView suchen.
34247              txa
34248              bne    :next              ; => Nicht gefunden.
34249
34250              clc
34251              rts
34252
34253 ::next        ldx    numDrives          ;Alle Laufwerke
34254              dex                        ;durchsucht?
34255              beq    :err                ; => Ja, Ende...
34256
34257              lda    curDrive            ;Nächstes Laufwerk.

```



```

tax 34258
sec 34259
sbc #$07 34260
cmp numDrives 34261
bne :4 34262
ldx #$07 34263
::4 inx 34264
txa 34265
pha 34266
jsr SetDevice ;Neues Laufwerk. 34267
pla 34268
cmp v6F68 ;Erstes Laufwerk? 34269
bne :loop ; => Nein, weiter... 34270
34271
::err lda #< d6F69 34272
ldx #> d6F69 34273
jsr S2090 ;DBox ohne Recover. 34274
34275
sec 34276
rts 34277
34278
:v6F68 b $08 ;Erstes Laufwerk für Suche. 34279
:v6EE6 b $00 ;Partition für Fenster. 34280
34281
:v6EE7 b "GeoHelpView",0 34282
34283
; Dialogbox: Hilfesystem nicht gefunden. 34284
:d6F69 b %10000001 34285
34286
b DBTXTSTR,$08,$10 34287
w T29CF 34288
34289
b DBTXTSTR,$08,$20 34290
w :t1 34291
34292
b OK , $11,$48 34293
34294
b NULL 34295
endif 34296
34297
if ENABLE_GEOHELP!LANG = TRUE!LANG_DE 34298
::t1 b "GeoHelp-System nicht gefunden!",0 34299
endif 34300
if ENABLE_GEOHELP!LANG = TRUE!LANG_EN 34301
::t1 b "GeoHelp system not found!",0 34302
endif 34303
34304
; Menükoordinaten für Farbe ändern: 34305
if PANEL_RIGHT = TRUE 34306
:colwin_y0 = $18 34307
:colwin_yh = $a0 34308
:colwin_y1 = colwin_y0 +colwin_yh -1 34309
:colwin_x0 = $0018 34310
:colwin_xw = $00e0 34311
:colwin_x1 = colwin_x0 +colwin_xw -1 34312

```

```

34313 else
34314 :colwin_y0      = $18
34315 :colwin_yh      = $a0
34316 :colwin_y1      = colwin_y0 +colwin_yh -1
34317 :colwin_x0      = $0040
34318 :colwin_xw      = $00e0
34319 :colwin_x1      = colwin_x0 +colwin_xw -1
34320 endif
34321
34322 if .p
34323 :tdwin_y0        = (colwin_y0 +$10)
34324 :tdwin_x0        = (colwin_x0 +$08)/8
34325 :tdwin_yof       = $20
34326 :tdwin_y1        = tdwin_y0 +tdwin_yof
34327 :tdwin_y2        = tdwin_y1 +tdwin_yof
34328 :tdwin_y3        = tdwin_y2 +tdwin_yof
34329
34330 :dbstd_y0        = colwin_y0 +$10
34331 :dbstd_yh        = $18
34332 :dbstd_y1        = dbstd_y0 +dbstd_yh -1
34333 :dbstd_x0        = colwin_x0 +$0058
34334 :dbstd_xw        = $0038
34335 :dbstd_x1        = dbstd_x0 +dbstd_xw -1
34336
34337 :dberr_y0        = colwin_y0 +$10 +$28*1
34338 :dberr_yh        = $18
34339 :dberr_y1        = dberr_y0 +dberr_yh -1
34340 :dberr_x0        = colwin_x0 +$0058
34341 :dberr_xw        = $0038
34342 :dberr_x1        = dberr_x0 +dberr_xw -1
34343
34344 :dbinf_y0        = colwin_y0 +$10 +$28*2
34345 :dbinf_yh        = $18
34346 :dbinf_y1        = dbinf_y0 +dbinf_yh -1
34347 :dbinf_x0        = colwin_x0 +$0058
34348 :dbinf_xw        = $0038
34349 :dbinf_x1        = dbinf_x0 +dbinf_xw -1
34350
34351 :dbicon_y0       = colwin_y0 +$10
34352 :dbicon_yh       = $10
34353 :dbicon_yof      = $18
34354 :dbicon_x0       = colwin_x0 +$00a8
34355 :dbicon_xw       = $0030
34356 :dbicon_y1       = dbicon_y0 +dbicon_yof
34357 :dbicon_y2       = dbicon_y1 +dbicon_yof
34358 :dbicon_y3       = dbicon_y2 +dbicon_yof +8
34359 :dbicon_y4       = dbicon_y3 +dbicon_yof
34360
34361 :drive_y0        = DRIVE_Y0 +8 +2*DRIVE_YH
34362 :drive_y1        = drive_y0 +DRIVE_YH -1
34363 :drive_x0        = DRIVE_X0 +16
34364 :drive_x1        = drive_x0 +DRIVE_XW -1
34365 endif
34366
34367 ; 0: Farben ändern.

```

:N6F89	lda	C0438	;TopDesk in Farbe	34368
	sta	v6FBE	;zwischenspeichern.	34369
				34370
	jsr	S14F8	;Bildschirm löschen.	34371
	jsr	z73B9	;Desktop löschen.	34372
				34373
	jsr	z7442	;Farben einlesen.	34374
				34375
	lda	#< d6FBF	;Farbmenü starten.	34376
	ldx	#> d6FBF		34377
	jsr	S2090	;DBox ohne Recover.	34378
				34379
	lda	r0L		34380
	cmp	#CANCEL	;Abbruch?	34381
	beq	:cancel	; => Ja, Ende...	34382
				34383
	lda	C0438	;TopDesk in Farbe	34384
	sta	v6FBE	;zwischenspeichern.	34385
				34386
	jsr	z73D7	;Farben übernehmen.	34387
	jsr	S09DD	;Konfig in REU/Datei.	34388
				34389
::cancel	lda	v6FBE	;Modus für	34390
	sta	C0438	;TopDesk in Farbe.	34391
	bne	:on		34392
				34393
::off	lda	##00000000		34394
	b	\$2c		34395
::on	lda	##10000000	;Dialogboxen (System)	34396
	sta	Flag_SetColor	;mit Farbe anzeigen.	34397
				34398
	jsr	S14F8	;Bildschirm löschen.	34399
	jmp	S3B9B	;UI neu zeichnen.	34400
				34401
:v6FBE	b	\$00	;Zwischensp. TopDesk-Farbmodus.	34402
				34403
; Dialogbox: Farbmenü.				34404
:d6FBF	b	%00100010	;Bit%5=keine Farbe.	34405
			;Bit%1=Schatten wie	34406
			;DeskTop-Muster.	34407
				34408
	b	colwin_y0,colwin_y1	;Größe Dialogbox.	34409
	w	colwin_x0,colwin_x1		34410
				34411
	b	DB_USR_ROUT	;Farbe darstellen.	34412
	w	z7559		34413
				34414
	b	DB_USR_ROUT	;Grafiken anzeigen.	34415
	w	r70F0		34416
				34417
	b	OK		34418
	b	(dbicon_x0 - colwin_x0)/8		34419
	b	dbicon_y3 - colwin_y0		34420
	b	CANCEL		34421
	b	(dbicon_x0 - colwin_x0)/8		34422

```

34423      b dbicon_y4 - colwin_y0
34424
34425      b DBUSRICON          ;Standard.
34426      b (dbicon_x0 - colwin_x0)/8
34427      b dbicon_y1 - colwin_y0
34428      w v6FE2
34429      b DBUSRICON          ;Farbe-Ein/Aus.
34430      b (dbicon_x0 - colwin_x0)/8
34431      b dbicon_y2 - colwin_y0
34432      w v703C
34433
34434      b DBTXTSTR
34435      b dbicon_x0 - colwin_x0
34436      b dbicon_y1 - colwin_y0 -16
34437      w :t1a
34438      b DBTXTSTR
34439      b dbicon_x0 - colwin_x0
34440      b dbicon_y1 - colwin_y0 -6
34441      w :t1b
34442
34443      b DBTXTSTR
34444      b dbicon_x0 - colwin_x0 +18
34445      b dbicon_y1 - colwin_y0 +12
34446      w :t1c
34447      b DBTXTSTR
34448      b dbicon_x0 - colwin_x0 +18
34449      b dbicon_y2 - colwin_y0 +12
34450      w :t1d
34451
34452  if LANG = LANG_DE
34453      b DBTXTSTR
34454      b dbstd_x0 - colwin_x0 +2
34455      b dbstd_y0 - colwin_y0 +14
34456      w :t4
34457      b DBTXTSTR
34458      b dberr_x0 - colwin_x0 +2
34459      b dberr_y0 - colwin_y0 +14
34460      w :t6
34461      b DBTXTSTR
34462      b dbinf_x0 - colwin_x0 +6
34463      b dbinf_y0 - colwin_y0 +14
34464      w :t5
34465  endif
34466  if LANG = LANG_EN
34467      b DBTXTSTR
34468      b dbstd_x0 - colwin_x0 +6
34469      b dbstd_y0 - colwin_y0 +14
34470      w :t4
34471      b DBTXTSTR
34472      b dberr_x0 - colwin_x0 +3
34473      b dberr_y0 - colwin_y0 +14
34474      w :t6
34475      b DBTXTSTR
34476      b dbinf_x0 - colwin_x0 +4
34477      b dbinf_y0 - colwin_y0 +14

```

	w :t5	34478
endif		34479
		34480
if LANG = LANG_DE		34481
	b DBTXTSTR,\$36,\$0b	34482
	w :t0	34483
endif		34484
if LANG = LANG_EN		34485
	b DBTXTSTR,\$3c,\$0b	34486
	w :t0	34487
endif		34488
		34489
	b DBTXTSTR,\$08,\$91	34490
	w :t2a	34491
	b DBTXTSTR,\$78,\$91	34492
	w :t2b	34493
	b DBTXTSTR,\$08,\$9b	34494
	w :t3a	34495
	b DBTXTSTR,\$78,\$9b	34496
	w :t3b	34497
		34498
	b DBOPVEC ;Mausabfrage.	34499
	w r74BB	34500
		34501
	b NULL	34502
		34503
if LANG = LANG_DE		34504
; Dialogbox-Titel.		34505
::t0	b BOLDON	34506
	b OUTLINEON	34507
	b "Farbeinstellungen"	34508
	b PLAINTEXT,0	34509
		34510
::t1a	b BOLDON	34511
	b "Desktop-",0	34512
::t1b	b "Farben:"	34513
	b PLAINTEXT,0	34514
::t1c	b "Reset",0	34515
::t1d	b "Ein/Aus",0	34516
		34517
::t3a	b "C= + "	34518
::t2a	b "Linke Maustaste",0	34519
::t2b	b "= Hintergrundfarbe",0	34520
::t3b	b "= Vordergrundfarbe",0	34521
		34522
::t4	b BOLDON	34523
	b "DialogBox",0	34524
::t6	b "FehlerBox",0	34525
::t5	b "InfoBox"	34526
	b PLAINTEXT,0	34527
endif		34528
if LANG = LANG_EN		34529
; Dialogbox-Titel.		34530
::t0	b BOLDON	34531
	b OUTLINEON	34532

```

34533      b "Color settings"
34534      b PLAINTEXT,0
34535
34536  ::t1a      b BOLDON
34537      b "Desktop ",0
34538  ::t1b      b "colors:"
34539      b PLAINTEXT,0
34540  ::t1c      b "Reset",0
34541  ::t1d      b "On/Off",0
34542
34543  ::t3a      b "C= + "
34544  ::t2a      b "Left mouse button",0
34545  ::t2b      b "= Background color",0
34546  ::t3b      b "= Foreground color",0
34547
34548  ::t4      b BOLDON
34549      b "Dialogue",0
34550  ::t6      b "Error box",0
34551  ::t5      b "Info box"
34552      b PLAINTEXT,0
34553  endif
34554
34555  ; Daten für Reset-Icon.
34556  :v6FE2      w GR6FEA
34557      b $00,$00,ICON_R_x,ICON_R_y
34558      w r7392
34559
34560  ; Daten für Farbe-Ein/Aus.
34561  :v703C      w GR704C
34562      b $00,$00,ICON_C_x,ICON_C_y
34563      w r73AA
34564
34565  :GR6FEA      b $80 +16*2
34566      b %11111111,%11111111
34567      b %10000000,%00000001
34568      b %10011000,%00000101
34569      b %10011100,%00011001
34570      b %10001110,%00110001
34571      b %10000111,%01100001
34572      b %10000011,%11000001
34573      b %10000001,%11000001
34574      b %10000011,%11000001
34575      b %10000111,%01100001
34576      b %10001110,%00110001
34577      b %10011100,%00011001
34578      b %10111000,%00000101
34579      b %10111000,%00000001
34580      b %10000000,%00000001
34581      b %11111111,%11111111
34582
34583  :ICON_R_x    = $02
34584  :ICON_R_y    = $10
34585
34586  :GR704C      b $80 +16*2
34587      b %11111111,%11111111

```

	b %10000000,%00100011	34588
	b %10000000,%10001001	34589
	b %10000000,%00100011	34590
	b %10000000,%10001001	34591
	b %10000000,%00100011	34592
	b %10000000,%10001001	34593
	b %10000000,%00100011	34594
	b %10101010,%11111111	34595
	b %11010101,%11111111	34596
	b %10101010,%11111111	34597
	b %11010101,%11111111	34598
	b %10101010,%11111111	34599
	b %11010101,%11111111	34600
	b %10101010,%11111111	34601
	b %11111111,%11111111	34602
		34603
:ICON_C_x	= \$02	34604
:ICON_C_y	= \$10	34605
		34606
; Menüaufbau.		34607
:r70F0	jsr z73B8	;Schatten#1 Menü.
	jsr z73B7	;Schatten#2 Menü.
		34609
		34610
;--- TopDesk-Fenster.		34611
	ldx #tdwin_x0	;Position für
	ldy #tdwin_y0	;TopDesk-Icon #1.
	jsr z7375	;TDesk-Icon anzeigen.
		34615
	ldx #tdwin_x0	;Position für
	ldy #tdwin_y1	;TopDesk-Icon #2.
	jsr z7375	;TDesk-Icon anzeigen.
		34619
	ldx #tdwin_x0	;Position für
	ldy #tdwin_y2	;TopDesk-Icon #3.
	jsr z7375	;TDesk-Icon anzeigen.
		34623
	ldx #tdwin_x0	;Position für
	ldy #tdwin_y3	;TopDesk-Icon #4.
	jsr z7375	;TDesk-Icon anzeigen.
		34627
;--- Dialogbox.		34628
	jsr z73B6	;Schatten#1 DBox.
	jsr z73B5	;Schatten#2 DBox.
		34631
	jsr i_FrameRectangle	
	b dbstd_y0,dbstd_y1	
	w dbstd_x0,dbstd_x1	
	b %11111111	
		34636
;--- Fehlerbox.		34637
	jsr z73B4	;Schatten#1 Fehler.
	jsr z73B3	;Schatten#2 Fehler.
		34639
		34640
	jsr i_FrameRectangle	
	b dberr_y0,dberr_y1	
		34642

```

34643          w dberr_x0,dberr_x1
34644          b %11111111
34645
34646 ;--- Infobox.
34647          jsr  z73B2          ;Schatten#1 Infobox.
34648          jsr  z73B1          ;Schatten#2 Infobox.
34649
34650          jsr  i_FrameRectangle
34651          b dbinf_y0,dbinf_y1
34652          w dbinf_x0,dbinf_x1
34653          b %11111111
34654
34655 ;--- Statuszeile.
34656          lda  #$00
34657          jsr  SetPattern
34658          jsr  i_Rectangle
34659          b STATUS_Y0+1,STATUS_Y1-1
34660          w STATUS_X0+STATUS_XW/2,STATUS_X1 -1
34661
34662          lda  v749B
34663          jsr  S1561          ;Screencolor setzen.
34664
34665          jsr  i_PutString
34666          w MENU_X0+MENU_XW/2 -4
34667          b STATUS_Y1 -6
34668          b PLAINTEXT
34669 if LANG = LANG_DE
34670          b "  Status:"
34671 endif
34672 if LANG = LANG_EN
34673          b "  Status:"
34674 endif
34675          b NULL
34676
34677          ldx  #(r4H - r3L)    ;Trennlinie zeichnen.
34678 ::1          lda  :coord,x
34679          sta  r3,x
34680          dex
34681          bpl  :1
34682
34683          lda  #%01010101    ;Linienmuster.
34684          jmp  VerticalLine
34685
34686 ::coord      b STATUS_Y0+1
34687          b STATUS_Y1-1
34688          w STATUS_X0+STATUS_XW/2
34689
34690 ; TDesk-Icon anzeigen.
34691 ; Übergabe: X/Y = X/Y-Koord. für Icon.
34692 :z7375      lda  #< GR72EE    ;Zeiger TDesk-Icon.
34693          sta  r0L
34694          lda  #> GR72EE
34695          sta  r0H
34696
34697          stx  r1L            ;X/Y-Position.

```



	sty	r1H		34698
				34699
	lda	#tdwin_xw	;Breite TDesk-Icon.	34700
	sta	r2L		34701
	lda	#tdwin_yh	;Höhe TDesk-Icon.	34702
	sta	r2H		34703
				34704
	jmp	BitmapUp	;TDesk-Icon anzeigen.	34705
				34706
			; Schatten-Koordinaten setzen.	34707
:z73B1	ldy	#6 *1	;Schatten #2 Infobox.	34708
	b	\$2c		34709
:z73B2	ldy	#6 *2	;Schatten #1 Infobox.	34710
	b	\$2c		34711
:z73B3	ldy	#6 *3	;Schatten #2 Fehler.	34712
	b	\$2c		34713
:z73B4	ldy	#6 *4	;Schatten #1 Fehler.	34714
	b	\$2c		34715
:z73B5	ldy	#6 *5	;Schatten #2 DBox.	34716
	b	\$2c		34717
:z73B6	ldy	#6 *6	;Schatten #1 DBox.	34718
	b	\$2c		34719
:z73B7	ldy	#6 *7	;Schatten #2 Menü.	34720
	b	\$2c		34721
:z73B8	ldy	#6 *8	;Schatten #1 Menü.	34722
	jsr	z73C0		34723
				34724
	lda	#\$01	;Füllmuster.	34725
	jmp	S140C	;Rechteck zeichnen.	34726
				34727
			; DeskTop löschen.	34728
:z73B9	ldy	#6 *9	;Desktop.	34729
	jsr	z73C0		34730
				34731
	lda	screencolors	;Bildschirmfarben.	34732
	jsr	DirectColor		34733
				34734
	lda	#\$02	;Füllmuster.	34735
	jmp	S140C	;Rechteck zeichnen.	34736
				34737
			; Grafik-Koordinaten setzen.	34738
:z73B0	ldy	#6 *0	;Bildschirm.	34739
				34740
:z73C0	ldx	#0		34741
::1	lda	:coord,y		34742
	sta	r2,x		34743
	iny			34744
	inx			34745
	cpx	:(r4H - r2L) +1		34746
	bcc	:1		34747
	rts			34748
				34749
::coord	b	SCREEN_Y0, SCREEN_Y1		34750
	w	SCREEN_X0, SCREEN_X1		34751
				34752

```

34753      b dbinf_y0 +4, dbinf_y1 +4
34754      w dbinf_x1 +1, dbinf_x1 +4
34755
34756      b dbinf_y1 +1, dbinf_y1 +4
34757      w dbinf_x0 +8, dbinf_x1
34758
34759      b dberr_y0 +4, dberr_y1 +4
34760      w dberr_x1 +1, dberr_x1 +4
34761
34762      b dberr_y1 +1, dberr_y1 +4
34763      w dberr_x0 +8, dberr_x1
34764
34765      b dbstd_y0 +4, dbstd_y1 +4
34766      w dbstd_x1 +1, dbstd_x1 +4
34767
34768      b dbstd_y1 +1, dbstd_y1 +4
34769      w dbstd_x0 +8, dbstd_x1
34770
34771      b colwin_y0 +8, colwin_y1
34772      w colwin_x1 +1, colwin_x1 +4
34773
34774      b colwin_y1 +1, colwin_y1 +4
34775      w colwin_x0 +8, colwin_x1 +4
34776
34777      b DESK_Y0, DESK_Y1
34778      w DESK_X0, DESK_X1
34779
34780 ; TopDesk-Fenster (Original).
34781 if FALSE
34782 :GR72EE      b $07
34783              b $ff
34784              b $81
34785              b $82
34786              b $05
34787              b $00
34788              b $89
34789              b $11,$ba,$5a,$e8,$00,$00,$00,$15,$82
34790              b $05
34791              b $00
34792              b $81
34793              b $11
34794              b $07
34795              b $ff
34796              b $81
34797              b $80
34798              b $05
34799              b $00
34800              b $a5
34801              b $11,$87,$e1,$f8,$7e,$1f,$80,$15
34802              b $84,$21,$08,$42,$10,$80,$15,$94
34803              b $21,$08,$42,$10,$80,$15,$84,$21
34804              b $08,$42,$10,$80,$15,$87,$e1,$f8
34805              b $7e,$1f,$80,$15,$80
34806              b $05
34807              b $00

```

```

    b $82 34808
    b $15,$80 34809
    b $05 34810
    b $00 34811
    b $a5 34812
    b $15,$87,$e1,$f8,$7e,$1f,$80,$11 34813
    b $84,$21,$08,$42,$10,$80,$11,$94 34814
    b $21,$08,$42,$10,$80,$11,$84,$21 34815
    b $08,$42,$10,$80,$11,$87,$e1,$f8 34816
    b $7e,$1f,$80,$11,$80 34817
    b $05 34818
    b $00 34819
    b $81 34820
    b $11 34821
    b $07 34822
    b $ff 34823
    b $81 34824
    b $80 34825
    b $05 34826
    b $00 34827
    b $89 34828
    b $11,$83,$5c,$80,$01,$ae,$00,$15,$80 34829
    b $05 34830
    b $00 34831
    b $81 34832
    b $11 34833
    b $07 34834
    b $ff 34835
endif 34836
34837
; DTopDesk-Fenster. 34838
:GR72EE b $80 +7*8 34839
34840
b #####,#####,#####,#####,#####,#####,##### 34841
b %0000#00,%0#000000,%00000000,%00000000,%00000000,%00000000#,%000#000# 34842
b %0#0#0#0#,%0#00#0##,%0#0###0#,%00000000,%00000000,%00000000#,%0#0#0#0# 34843
b %0000#00,%0#000000,%00000000,%00000000,%00000000,%00000000#,%000#000# 34844
b #####,#####,#####,#####,#####,#####,##### 34845
b %00000000,%00000000,%00000000,%00000000,%00000000,%00000000,%000#000# 34846
b %00000000,%0#####0,%000#####,%#0000###,%###0000#,%#####000,%000#000# 34847
b %00000000,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#000# 34848
34849
    b $80 +7*8 34850
34851
b %00###00,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#000# 34852
b %00000000,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#000# 34853
b %00000000,%0#####0,%000#####,%#0000###,%###0000#,%#####000,%000#0#0# 34854
b %00000000,%00000000,%00000000,%00000000,%00000000,%00000000,%000#0#0# 34855
b %00000000,%00000000,%00000000,%00000000,%00000000,%00000000,%000#0#0# 34856
b %00000000,%0#####0,%000#####,%#0000###,%###0000#,%#####000,%000#0#0# 34857
b %00000000,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#0#0# 34858
b %00###00,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#0#0# 34859
34860
    b $80 +7*8 34861
34862
```

```

34863 b %#0000000,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#0#0#
34864 b %#0000000,%0#####0,%000#####,%#0000###,%###0000#,%#####000,%000#000#
34865 b %#0000000,%00000000,%00000000,%00000000,%00000000,%00000000,%000#000#
34866 b %#####,%#####,%#####,%#####,%#####,%#####,%#####,%#####
34867 b %#0000#0#,%00#00000,%00000000,%00000000,%00000000,%00000000,%000#000#
34868 b %#0##0#00,%#0#00##0,%#0###00#,%00000000,%00000##0,%#0###000,%000#0#0#
34869 b %#0000#0#,%00#00000,%00000000,%00000000,%00000000,%00000000,%000#000#
34870 b %#####,%#####,%#####,%#####,%#####,%#####,%#####,%#####
34871
34872 :tdwin_xw          = 7      ; .x
34873 :tdwin_yh          = 3*8    ; .y
34874
34875 ; Standardfarben setzen.
34876 :r7392             ldy      #MAXCOLS -1
34877 ::1               lda      v74AC,y
34878                   sta      v749D,y
34879                   dey
34880                   bpl      :1
34881
34882                   lda      %#00000000
34883                   sta      C0438
34884
34885 ; Farbe ein/aus.
34886 :r73AA             lda      C0438                      ;TopDesk mit/ohne
34887                   and      %#00000001
34888                   eor      %#00000001                  ;Farbe anzeigen.
34889                   sta      C0438
34890                   bne      :on
34891
34892 ::off             lda      %#00000000
34893                   b $2c
34894 ::on              lda      %#10000000                  ;Dialogboxen (System)
34895                   sta      Flag_SetColor                ;mit Farbe anzeigen.
34896                   tax                                      ; => Für b $2c...
34897                   beq      :1
34898
34899                   jmp      z7559                        ;Farben anzeigen.
34900
34901 ::1               jsr      z73B0                        ;Koord. Bildschirm.
34902
34903                   lda      v749D
34904                   sta      COL0439                      ;Hintergrundfarbe
34905                   sta      C_GEOS_BACK                  ;für GEOS setzen.
34906                   sta      screencolors
34907
34908                   jmp      S1561                        ;Screencolor setzen.
34909
34910 ; Farben in System übernehmen.
34911 :z73D7             ldy      #13 -1
34912 ::1               lda      v749D,y
34913                   sta      BASE_COLCFG +1,y
34914                   dey
34915                   bpl      :1
34916
34917 ;                   lda      v749D                      ;Bildschirmfarben.

```

```

sta COL0439 34918
sta C_GEOS_BACK 34919
sta screencolors 34920
34921
; Hinweis: 34922
; Systemfarben nicht verändern. 34923
; lda v74A0 34924
; sta C_DBoxBack ;Farbe Dialogbox. 34925
34926
lda v74A8 34927
sta V621B ;TDesk-Fenster #1. 34928
lda v74A9 34929
sta V622C ;TDesk-Fenster #2. 34930
lda v74AA 34931
sta V623D ;TDesk-Fenster #3. 34932
lda v74AB 34933
sta V624E ;TDesk-Fenster #4. 34934
34935
rts 34936
34937
; Systemfarben für Farbeditor einlesen. 34938
:z7442 ldy #13 -1 34939
::1 lda BASE_COLCFG +1,y 34940
sta v749D,y 34941
dey 34942
bpl :1 34943
34944
lda V621B ;TDesk-Fenster #1. 34945
sta v74A8 34946
lda V622C ;TDesk-Fenster #2. 34947
sta v74A9 34948
lda V623D ;TDesk-Fenster #3. 34949
sta v74AA 34950
lda V624E ;TDesk-Fenster #4. 34951
sta v74AB 34952
34953
rts 34954
34955
; Aktuelle Desktop-Farben: 34956
; Entspricht der Reihenfolge ab 34957
; mod.#00a/COL0439 bis COL0445! 34958
:v749D b $00 ;GEOS-Farbe. 34959
:v749F b $00 ;Panel/Laufwerk. 34960
:v749C b $00 ;Aktuelles Laufwerk. 34961
:v74A0 b $00 ;Dialogbox. 34962
:v74A1 b $00 ;GEOS-Icon. 34963
:v74A2 b $00 ;Fehlerbox. 34964
:v74A3 b $00 ;Infobox. 34965
:v74A5 b $00 ;Trash-Icon. 34966
:v74A7 b $00 ;Drucker-Icon. 34967
:v74A4 b $00 ;gew.Dateien. 34968
:v74A6 b $00 ;Datum/Uhrzeit. 34969
:v749E b $00 ;Hauptmenü. 34970
:v749B b $00 ;Statuszeile. 34971
34972

```

```

34973 :v74A8          b $00 ;TDesk-Fenster #1.
34974 :v74A9          b $00 ;TDesk-Fenster #2.
34975 :v74AA          b $00 ;TDesk-Fenster #3.
34976 :v74AB          b $00 ;TDesk-Fenster #4.
34977
34978 :ENDCOLS
34979 :MAXCOLS          = (ENDCOLS - v749D)
34980
34981 ; Tabelle mit Standard-Farben.
34982 :v74AC
34983 if ENABLE_ORIGCOL = TRUE
34984         b $bf ;GEOS-Farbe.
34985         b $0c ;Panel/Laufwerk.
34986         b $7c ;Aktuelles Laufwerk.
34987         b $0d ;Dialogbox.
34988         b $04 ;GEOS-Icon (Original).
34989         b $12 ;Fehlerbox.
34990         b $0e ;Infobox.
34991         b $6c ;Trash-Icon (Original).
34992         b $1c ;Drucker-Icon (Original).
34993         b $07 ;gew.Dateien.
34994         b $0d ;Datum/Uhrzeit.
34995         b $03 ;Hauptmenü.
34996         b $12 ;Statuszeile.
34997 else
34998 ; Hinweis:
34999 ; Werden hier Farben geändert, dann
35000 ; auch Werte in src.mod#00a/BASE_COLCFG
35001 ; anpassen!
35002         b $bf ;GEOS-Farbe.
35003         b $0c ;Panel/Laufwerk.
35004         b $dc ;Aktuelles Laufwerk.
35005         b $03 ;Dialogbox.
35006         b $03 ;GEOS-Icon.
35007         b $12 ;Fehlerbox.
35008         b $0e ;Infobox.
35009         b $0c ;Trash-Icon.
35010         b $0c ;Drucker-Icon.
35011         b $07 ;gew.Dateien.
35012         b $0d ;Datum/Uhrzeit.
35013         b $03 ;Hauptmenü.
35014         b $07 ;Statuszeile.
35015 endif
35016
35017         b $0f ;TDesk-Fenster #1.
35018         b $0d ;TDesk-Fenster #2.
35019         b $0a ;TDesk-Fenster #3.
35020         b $0e ;TDesk-Fenster #4.
35021
35022 ; Mausabfrage.
35023 :r74BB          bit   mouseData
35024                 bmi   :done
35025
35026                 lda    C0438                ;TopDesk in Farbe?
35027                 bne    :1                    ; => Ja, weiter...

```

```

35028
35029      lda    #MAXAREA -1          ;Start: S/W.
35030      b $2c
35031  ::1      lda    #$00              ;Start: Farbe.
35032      sta    r15L
35033
35034  ::loop   lda    r15L
35035      jsr    z7490                ;Zeiger Farb-Bereich.
35036
35037      lda    r14H
35038      ora    r14L
35039      beq    :done                ; => Ende...
35040
35041      lda    C0438                ;TopDesk in Farbe?
35042      beq    :edit                ; => Nein, weiter...
35043
35044      ldy    #(r4H - r2L)
35045  ::2      lda    (r14),y
35046      sta    r2,y
35047      dey
35048      bpl    :2
35049
35050      jsr    IsMseInRegion        ;Mausklick?
35051      bmi    :edit                ; => Ja, auswerten...
35052
35053      inc    r15L
35054      bne    :loop
35055
35056  ::done   rts
35057
35058 ; Farbe für gewählten Bereich ändern.
35059  ::edit   lda    r15L              ;Icon angeklickt?
35060      cmp    #CHKCOL1
35061      bcc    :11
35062      cmp    #CHKCOL2
35063      bcc    :done                ; => Ja, Ende...
35064
35065  ::11     ldy    #6                ;Zeiger auf Farbwert
35066      lda    (r14),y              ;einlesen.
35067      sta    r13L
35068      iny
35069      lda    (r14),y
35070      sta    r13H
35071
35072      jsr    S51CE                ;C=-Taste gedrückt?
35073      bcs    :31                  ; => Ja, weiter...
35074
35075      lda    r15L              ;Icon angeklickt?
35076      cmp    #CHKCOL3
35077      bcc    :20
35078      cmp    #CHKCOL4
35079      bcs    :20                  ; => Ja, Ende...
35080
35081      lda    #< v749F
35082      sta    r13L

```

```

35083          lda    #> v749F
35084          sta    r13H
35085
35086 ; Hintergrundfarbe.
35087 ::20          ldy    #$00          ;Hintergrundfarbe.
35088          lda    (r13),y
35089 ::21          tay
35090          and    #%00001111
35091          cmp    #%00001111
35092          bne    :22
35093
35094          tya
35095          and    #%11110000
35096          clv
35097          bvc    :23
35098
35099 ::22          tya
35100          clc
35101          adc    #$01
35102 ::23          tax
35103          asl
35104          asl
35105          asl
35106          asl
35107          sta    r15H          ;Vordergrund- mit
35108          txa          ;Hintergrundfarbe
35109          and    #%11110000          ;vergleichen.
35110          cmp    r15H          ;Unterschiedlich?
35111          bne    :24          ; => Ja, weiter...
35112
35113          txa
35114          clv
35115          bvc    :21          ;Wert korrigieren.
35116
35117 ::24          txa
35118          jmp    z7555          ;Farbwert speichern.
35119
35120 ; Vordergrundfarbe.
35121 ::31          ldy    #$00
35122          lda    (r13),y
35123 ::32          clc
35124          adc    #$10
35125          tax
35126          lsr
35127          lsr
35128          lsr
35129          lsr
35130          sta    r15H          ;Vordergrund- mit
35131          txa          ;Hintergrundfarbe
35132          and    #%00001111          ;vergleichen.
35133          cmp    r15H          ;Unterschiedlich?
35134          bne    :33          ; => Ja, weiter...
35135
35136          txa
35137          clv

```



	bvc	:32	;Wert korrigieren.	35138
				35139
::33	txa			35140
;	jmp	z7555	;Farbwert speichern.	35141
				35142
			; Neuen Farbwert speichern.	35143
:z7555	ldy	#\$00		35144
	sta	(r13),y		35145
				35146
			; Farbe Panel-Icons an Panel-Hintergrund anpassen.	35147
	lda	v749F		35148
	and	##00001111		35149
	sta	r15H		35150
				35151
	lda	v749C		35152
	and	##11110000		35153
	ora	r15H		35154
	sta	v749C		35155
				35156
	lda	v74A7		35157
	and	##11110000		35158
	ora	r15H		35159
	sta	v74A7		35160
				35161
	lda	v74A5		35162
	and	##11110000		35163
	ora	r15H		35164
	sta	v74A5		35165
				35166
			; Anzeige aktualisieren.	35167
:z7559	lda	C0438	;TopDesk in Farbe?	35168
	bne	:1	; => Ja, weiter...	35169
				35170
	lda	v749D	;Bildschirmfarbe.	35171
	sta	C_GEOS_BACK		35172
	sta	screencolors		35173
				35174
::1	lda	#MAXAREA -1	;Alle Farbenbereiche	35175
	sta	r15L	;zeichnen.	35176
				35177
::loop	lda	r15L		35178
	jsr	z7490	;Zeiger Farb-Bereich.	35179
				35180
;	lda	r14H		35181
	ora	r14L		35182
	beq	:done		35183
				35184
	ldy	##(r4H - r2L)	;Farbbereich	35185
::2	lda	(r14),y	;setzen.	35186
	sta	r2,y		35187
	dey			35188
	bpl	:2		35189
				35190
	ldy	#6	;Zeiger auf	35191
	lda	(r14),y	;Farbwert einlesen.	35192

```

35193          sta    r13L
35194          iny
35195          lda     (r14),y
35196          sta     r13H
35197
35198          ldy     #0
35199          lda     (r13),y          ;Farbe einlesen.
35200          jsr     S1561          ;Screencolor setzen.
35201
35202          dec     r15L          ;Menü aktualisiert?
35203          bpl     :loop          ; => Nein, weiter...
35204
35205 ::done      bit     mouseData
35206          bpl     :done
35207          rts
35208
35209 ; Zeiger auf Farb-Bereich einlesen.
35210 :z7490      asl
35211          tay
35212          lda     v759F +0,y
35213          sta     r14L
35214          lda     v759F +1,y
35215          sta     r14H
35216          rts
35217
35218 ; Tabelle mit Abfrage-Bereichen.
35219 ;
35220 ; Hinweis:
35221 ; Beim Bildaufbau werden die Bereiche
35222 ; von hinten nach vorne abgearbeitet!
35223 ; D.h. Farbbereiche die weiter vorne
35224 ; in der Tabelle stehen überschreiben
35225 ; Farben am Ende der Tabelle.
35226 :v759F      w v75CB  ;TDesk-Fenster #1.
35227          w v75D3  ;TDesk-Fenster #2.
35228          w v75DB  ;TDesk-Fenster #3.
35229          w v75E3  ;TDesk-Fenster #4.
35230
35231          w v75EB  ;Dialogbox.
35232          w v75F3  ;Fehlerbox.
35233          w v75FB  ;Infobox.
35234
35235 :skip1      w v7603  ;"Reset"-Icon.
35236          w v760B  ;"Farbe ein/aus"-Icon #1.
35237          w v7613  ;"Farbe ein/aus"-Icon #2.
35238          w v7614  ;"Farbe ein/aus"-Icon #3.
35239          w v7615  ;"Farbe ein/aus"-Icon #4.
35240
35241          w v761B  ;"OK"-Icon.
35242          w v7623  ;"Abbruch"-Icon.
35243
35244          w v762B  ;Farbmenü-Fenster.
35245
35246 :skip2      w v7633  ;Hauptmenü.
35247          w v7634  ;Statuszeile.

```

	w v7663	;Task-Icon.	35248
	w v763B	;gew.Dateien.	35249
	w v7643	;Datum/Uhrzeit.	35250
			35251
:skip3	w v764B	;Trash-Icon.	35252
	w v7653	;Drucker-Icon.	35253
	w v765A	;Aktuelles Laufwerk.	35254
:skip4	w v765B	;Panel.	35255
			35256
	w v766B	;Desktop.	35257
			35258
	w NULL		35259
			35260
; Bei Mausabfrage diese Bereiche übergehen.			35261
:CHKCOL1 = (skip1 -v759F)/2			35262
:CHKCOL2 = (skip2 -v759F)/2			35263
:CHKCOL3 = (skip3 -v759F)/2			35264
:CHKCOL4 = (skip4 -v759F)/2			35265
			35266
; Farb-Bereiche:			35267
:v75CB	b tdwin_y0,tdwin_y0 +tdwin_yh -1		35268
	w tdwin_x0*8,(tdwin_x0 +tdwin_xw)*8 -1		35269
	w v74A8	;TDesk-Fenster #1.	35270
:v75D3	b tdwin_y1,tdwin_y1 +tdwin_yh -1		35271
	w tdwin_x0*8,(tdwin_x0 +tdwin_xw)*8 -1		35272
	w v74A9	;TDesk-Fenster #2.	35273
:v75DB	b tdwin_y2,tdwin_y2 +tdwin_yh -1		35274
	w tdwin_x0*8,(tdwin_x0 +tdwin_xw)*8 -1		35275
	w v74AA	;TDesk-Fenster #3.	35276
:v75E3	b tdwin_y3,tdwin_y3 +tdwin_yh -1		35277
	w tdwin_x0*8,(tdwin_x0 +tdwin_xw)*8 -1		35278
	w v74AB	;TDesk-Fenster #4.	35279
			35280
:v75EB	b dbstd_y0,dbstd_y1		35281
	w dbstd_x0,dbstd_x1		35282
	w v74A0	;Dialogbox.	35283
:v75F3	b dberr_y0,dberr_y1		35284
	w dberr_x0,dberr_x1		35285
	w v74A2	;Fehlerbox.	35286
:v75FB	b dbinf_y0,dbinf_y1		35287
	w dbinf_x0,dbinf_x1		35288
	w v74A3	;Infobox.	35289
			35290
:v7603	b dbicon_y1 ,dbicon_y1 +15		35291
	w dbicon_x0 ,dbicon_x0 +15		35292
	w C_DBoxDIcon	;"Reset"-Icon.	35293
:v760B	b dbicon_y2 ,dbicon_y2 +7		35294
	w dbicon_x0 ,dbicon_x0 +7		35295
	w v766F +0	;"Farbe an/aus"-Icon.	35296
:v7613	b dbicon_y2 ,dbicon_y2 +7		35297
	w dbicon_x0 +8,dbicon_x0 +15		35298
	w v766F +1	;"Farbe an/aus"-Icon.	35299
:v7614	b dbicon_y2 +8,dbicon_y2 +8 +7		35300
	w dbicon_x0 ,dbicon_x0 +7		35301
	w v766F +2	;"Farbe an/aus"-Icon.	35302

```

35303 :v7615      b dbicon_y2 +8,dbicon_y2 +8 +7
35304      w dbicon_x0 +8,dbicon_x0 +15
35305      w v766F +3      ;"Farbe an/aus"-Icon.
35306
35307 :v761B      b dbicon_y3,dbicon_y3 +dbicon_yh -1
35308      w dbicon_x0,dbicon_x0 +dbicon_xw -1
35309      w C_DBoxDIcon      ;"OK"-Icon.
35310 :v7623      b dbicon_y4,dbicon_y4 +dbicon_yh -1
35311      w dbicon_x0,dbicon_x0 +dbicon_xw -1
35312      w C_DBoxDIcon      ;"Abbruch"-Icon.
35313
35314 :v762B      b colwin_y0,colwin_y1
35315      w colwin_x0,colwin_x1
35316      w v74A0      ;Farbmenü/Dialogbox.
35317
35318 :v7633      b MENU_Y0,MENU_Y1      ;Hauptmenü.
35319      w MENU_X0,MENU_X1-MENU_XW/2
35320      w v749E
35321 :v7634      b STATUS_Y0,STATUS_Y1      ;Statuszeile.
35322      w STATUS_X0+STATUS_XW/2,STATUS_X1
35323      w v749B
35324 :v7663      b TASK_Y0,TASK_Y1      ;Task-Icon.
35325      w TASK_X0,TASK_X1
35326      w v74A1
35327 :v763B      b SLCT_Y0,SLCT_Y1      ;gew.Dateien.
35328      w SLCT_X0,SLCT_X1
35329      w v74A4
35330 :v7643      b TIME_Y0,TIME_Y1      ;Datum/Uhrzeit.
35331      w TIME_X0,TIME_X1
35332      w v74A6
35333
35334 :v764B      b TRASH_Y0,TRASH_Y1      ;Trash-Icon.
35335      w TRASH_X0,TRASH_X1
35336      w v74A5
35337 :v7653      b PRINT_Y0,PRINT_Y1      ;Drucker-Icon.
35338      w PRINT_X0,PRINT_X1
35339      w v74A7
35340 :v765A      b drive_y0,drive_y1      ;Aktuelles Laufwerk.
35341      w drive_x0,drive_x1
35342      w v749C
35343 :v765B      b PANEL_Y0,PANEL_Y1      ;Panel.
35344      w PANEL_X0,PANEL_X1
35345      w v749F
35346
35347 :v766B      b DESK_Y0,DESK_Y1      ;Desktop.
35348      w DESK_X0,DESK_X1
35349      w v749D
35350 :ENDAREA
35351 :MAXAREA      = (ENDAREA - v75CB)/8
35352
35353 ; Farben für "Farbe ein/aus"-Icon.
35354 :v766F      b $06,$02,$07,$00
35355
35356 ; 1: Uhrzeit ändern.
35357 :N7674

```

```

35358
; TODO: 35359
; Wozu VLIR-Modul aktivieren? 35360
if FALSE 35361
    lda    #1 35362
    sta    V04B0 ;VLIR-Modul aktiv. 35363
endif 35364
35365
; TODO: 35366
; Wozu Echtzeit-Uhr anhalten? 35367
if FALSE 35368
    jsr    z7814 ;C64-Uhr anhalten. 35369
endif 35370
35371
    jsr    MouseOff ;Maus abschalten. 35372
35373
; Hinweis: 35374
; Der erste Prozess hat die Nr.0! 35375
; TopDesk nutzt hier die Nr.1=Fehler! 35376
; 35377
    ldx    #$01 35377
    ldx    #0 ;Uhr-Prozess 35378
    jsr    BlockProcess ;anhalten. 35379
35380
    lda    keyVector +0 35381
    sta    v76AD +0 35382
    lda    keyVector +1 35383
    sta    v76AD +1 35384
35385
    lda    #< r76AF 35386
    sta    keyVector +0 35387
    lda    #> r76AF 35388
    sta    keyVector +1 35389
35390
    ldx    #$00 ;Cursor-Position 35391
    stx    v78C9 ;auf Anfang setzen. 35392
35393
    jsr    z77E1 ;max. Taste/Eingabe. 35394
35395
    lda    #REV_ON 35396
    sta    V6167 ;Cursor ein. 35397
35398
    lda    #$00 ;Sekunden löschen. 35399
    sta    V61E8 35400
35401
    jmp    S6118 ;Datum/Uhr anzeigen. 35402
35403
; Puffer für Tastatur-Vektor. 35404
:v76AD w $0000 35405
35406
; Tastatur abfragen. 35407
:r76AF lda    keyData 35408
      cmp    v78D7 35409
      beq    :1 35410
      bcs    :2 35411
35412

```

```

35413 ::1          cmp    #"0"          ;Zifferntaste?
35414          bcc     :2              ; => Nein, weiter...
35415
35416          pha
35417          jsr     z77BB            ;Cursor ausschalten.
35418          pla
35419          sta     V6168,x          ;Text Datum/Uhrzeit.
35420          clv
35421          bvc     :crsr_r          ;Cursor rechts.
35422
35423 ::2          cmp    #KEY_RIGHT
35424          bne     :other1
35425
35426          jsr     z77BB            ;Cursor ausschalten.
35427
35428 ; Cursor nach rechts.
35429 ::crsr_r     iny
35430          cpy     #12              ;Am rechten Rand?
35431          bne     :11              ; => Nein, weiter...
35432
35433          ldy     #$00              ;Cursor auf Anfang.
35434 ::11         jmp     z77C7          ;Cursor einschalten.
35435
35436 ::other1     cmp    #KEY_DELETE
35437          beq     :crsr_l
35438          cmp    #KEY_LEFT
35439          bne     :other2
35440
35441 ; Cursor nach links.
35442 ::crsr_l     jsr     z77BB            ;Cursor ausschalten.
35443
35444          dey
35445          bpl     :21              ;Am linken Rand?
35446          ; => Nein, weiter...
35447
35447          ldy     #12 -1            ;Cursor auf Ende.
35448 ::21         jmp     z77C7          ;Cursor einschalten.
35449
35450 ::other2     cmp    #CR
35451          bne     :done              ;Beenden?
35452          ; => Nein, Ende...
35453
35453          jsr     z77BB            ;Cursor ausschalten.
35454
35454          jsr     z785A            ;Eingaben überprüfen.
35455          beq     :31              ; => OK, weiter...
35456
35457
35458 ; Hinweis:
35459 ; Wozu Fehlerton bei falscher Eingabe?
35460 if FALSE
35461          tya
35462          pha
35463
35464          jsr     InitForIO
35465          sta     $dc08            ;Uhrzeit starten.
35466          jsr     DoneWithIO
35467

```

```

        lda    #$02                                35468
        jsr    S24C0                                ;Fehlerton.          35469
                                                35470
        jsr    InitForIO                            35471
        sta    $dc0b                                ;Uhrzeit anhalten.        35472
        jsr    DoneWithIO                            35473
                                                35474
        pla                                         35475
        tay                                         ;Neue Cursor-Pos.      35476
endif                                           35477
                                                35478
        jmp    z77C7                                ;Cursor einschalten.    35479
                                                35480
::31      jsr    InitForIO                            35481
                                                35482
        jsr    z7737                                ;Eingaben übernehmen.   35483
                                                35484
        lda    #$00                                ;Sek. und 1/10Sek.      35485
        sta    $dc09                                ;löschen und Uhr       35486
        sta    $dc08                                ;wieder starten.       35487
                                                35488
        jsr    DoneWithIO                            35489
                                                35490
; Hinweis:                                       35491
; Cursor abschalten/Uhrzeit anzeigen.          35492
        lda    #$00                                ;Sekunden löschen.     35493
        sta    seconds                            35494
                                                35495
        jsr    S6118                                ;Datum/Uhr anzeigen.   35496
                                                35497
        jsr    MouseUp                            35498
                                                35499
; Hinweis:                                       35500
; Der erste Prozess hat die Nr.0!              35501
; TopDesk nutzt hier die Nr.1=Fehler!         35502
;                                                35503
        ldx    #$01                                35503
        ldx    #0                                  ;Uhr-Prozess          35504
        jsr    UnblockProcess                      ;fortsetzen.          35505
                                                35506
        lda    v76AD +0                            35507
        sta    keyVector +0                        35508
        lda    v76AD +1                            35509
        sta    keyVector +1                        35510
                                                35511
; TODO:                                       35512
; Wozu VLIR-Modul deaktivieren?               35513
if FALSE                                       35514
        lda    #0                                  35515
        sta    V04B0                                ;VLIR-Modul inaktiv.   35516
endif                                           35517
                                                35518
::done      rts                                    35519
                                                35520
; Eingaben auslesen und Uhr setzen.            35521
:z7737      lda    V6168 +OFF_DAY +0              35522

```

```

35523      ldx  V6168 +OFF_DAY +2
35524      jsr  z77F7          ;ASCII nach Dezimal.
35525      sta  day            ;Tag speichern.
35526
35527      lda  V6168 +OFF_MONTH +0
35528      ldx  V6168 +OFF_MONTH +2
35529      jsr  z77F7          ;ASCII nach Dezimal.
35530      sta  month          ;Monat speichern.
35531
35532      lda  V6168 +OFF_YEAR +0
35533      ldx  V6168 +OFF_YEAR +2
35534      jsr  z77F7          ;ASCII nach Dezimal.
35535      sta  year            ;Jahr speichern.
35536
35537      lda  V6168 +OFF_MILLENIU +0
35538      ldx  V6168 +OFF_MILLENIU +2
35539      jsr  z77F7          ;ASCII nach Dezimal.
35540      sta  millenium      ;Jahrtausend (MP3).
35541
35542      lda  #$00
35543      sta  r15L            ;AM/PM-Flag.
35544
35545      lda  V6168 +OFF_HOUR +0
35546      ldx  V6168 +OFF_HOUR +2
35547      jsr  z77F7          ;ASCII nach Dezimal.
35548      cmp  #24
35549      bne  :1
35550      lda  #0
35551  ::1      sta  hour        ;Stunde speichern.
35552
35553      cmp  #12
35554      bcc  :2
35555  ;      sec
35556      sbc  #12
35557
35558      ldx  ##10000000
35559      stx  r15L            ;AM/PM-Flag.
35560
35561  ::2      jsr  z783B          ;Dezimal nach BCD.
35562      clc
35563      adc  r15L
35564      sta  $dc0b            ;Uhr/Stunde setzen.
35565
35566      lda  V6168 +OFF_MINUTES +0
35567      ldx  V6168 +OFF_MINUTES +2
35568      jsr  z77F7          ;ASCII nach Dezimal.
35569      sta  minutes          ;Minute speichern.
35570
35571      jsr  z783B          ;Dezimal nach BCD.
35572      sta  $dc0a            ;Uhr/Minute setzen.
35573
35574      rts
35575
35576  ; Cursor ausschalten.
35577  :z77BB      ldy  v78C9          ;Cursor-Position.

```



```

        ldx    v78CA,y          35578
        lda    #PLAINTEXT      35579
        sta    V6167,x          ;Text Datum/Uhrzeit. 35580
        rts                     35581
                                   35582
; Cursor einschalten.          35583
:z77C7    ldx    v78CA,y          35584
        lda    #REV_ON          35585
        sta    V6167,x          ;Text Datum/Uhrzeit. 35586
        sty    v78C9            ;Cursor-Position.    35587
                                   35588
if FALSE                          35589
        jsr    InitForIO        35590
                                   35591
        jsr    z7737            ;Eingaben übernehmen. 35592
                                   35593
        jsr    DoneWithIO       35594
endif                          35595
                                   35596
        jsr    z77E1            ;max. Taste/Eingabe.  35597
                                   35598
;                               35599
        jsr    S60DF            ;Uhrzeit einlesen.   35599
        jmp    S6118            ;Datum/Uhr anzeigen.  35600
                                   35601
; max. erlaubte Taste für Eingabe. 35602
:z77E1    ldy    v78C9            ;Cursor-Position.   35603
        lda    :data,y          35604
        sta    v78D7            35605
        rts                     35606
                                   35607
::data    b $33,$39,$31,$39      35608
        b $32,$39,$39,$39      35609
        b $32,$39,$35,$39      35610
                                   35611
; ASCII nach Dezimal 00-99 wandeln. 35612
:z77F7    sec                     35613
        sbc    #"0"             35614
        sta    r15H             35615
        asl                     35616
        asl                     35617
        asl                     35618
        clc                     35619
        adc    r15H             35620
        adc    r15H             35621
        sta    r15H             35622
        txa                     35623
        sec                     35624
        sbc    #"0"             35625
        clc                     35626
        adc    r15H             35627
        rts                     35628
                                   35629
; Timer: Stunde setzen = Uhr anhalten. 35630
if FALSE                          35631
:z7814    jsr    InitForIO        35632

```

```

35633
35634          lda    #$00                ;AM/PM-Flag löschen.
35635          sta    r15L
35636
35637          lda    hour
35638          cmp    #12
35639          bcc    :1
35640
35641          lda    %%100000000          ;PM-Flag setzen.
35642          sta    r15L
35643
35644          lda    hour                ;Stunde von 0-23
35645          sec    ;nach 12-11 wandeln.
35646          sbc    #12
35647
35648 ::1          jsr    z783B            ;Dezimal nach BCD.
35649          clc
35650          adc    r15L
35651          sta    $dc0b                ; -> Uhr anhalten.
35652
35653          jmp    DoneWithIO
35654 endif
35655
35656 ; Dezimal nach BCD wandeln.
35657 :z783B          sta    r0L
35658          lda    #$00
35659          sta    r0H
35660          ldx    #r0
35661
35662          lda    #< 10
35663          sta    r1L
35664          lda    #> 10
35665          sta    r1H
35666          ldy    #r1
35667
35668          jsr    Ddiv
35669
35670          lda    r0L
35671          asl
35672          asl
35673          asl
35674          asl
35675          clc
35676          adc    r8L
35677          rts
35678
35679 ; Eingaben überprüfen.
35680 :z785A          lda    day            ;Tag > 0 ?
35681          bne    :1                ; => Ja, weiter...
35682
35683          ldy    #$01                ;Cursor auf Tag2.
35684          lda    #$ff                ;Fehler.
35685          rts
35686
35687 ::1          cmp    #31 +1          ;Tag =< 31 ?

```

	bcc	:2	; => Ja, weiter...	35688
				35689
	ldy	#\$00	;Cursor auf Tag1.	35690
	lda	#\$ff	;Fehler.	35691
	rts			35692
				35693
	; Monat testen.			35694
::2	lda	month	;Monat > 0 ?	35695
	bne	:4	; => Ja, weiter...	35696
				35697
::3	ldy	#\$03	;Cursor auf Monat2.	35698
	lda	#\$ff	;Fehler.	35699
	rts			35700
				35701
::4	cmp	#12 +1	;Monat =< 12 ?	35702
	bcc	:5	; => Ja, weiter...	35703
				35704
	ldy	#\$02	;Cursor auf Monat1.	35705
	lda	#\$ff	;Fehler.	35706
	rts			35707
				35708
	; Auf Schaltjahr testen.			35709
	;			35710
	; TODO:			35711
	; Sonderbehandlung wenn Jahrhundert:			35712
	; xx00 = Kein Schaltjahr.			35713
	; xx00/400 = Schaltjahr.			35714
::5	cmp	#2	;Monat = Februar?	35715
	bne	:9	; => Nein, weiter...	35716
				35717
	lda	day	;Tag gültig?	35718
	cmp	#29		35719
	beq	:6		35720
	bcs	:3		35721
	bcc	:12	;Uhrzeit testen.	35722
				35723
::6	lda	year	;Auf Schaltjahr	35724
	ror		;testen.	35725
	bcs	:7		35726
	ror			35727
	bcc	:12	;Uhrzeit testen.	35728
				35729
::7	ldy	#\$01	;Cursor auf Tag2.	35730
	lda	#\$ff	;Fehler.	35731
	rts			35732
				35733
::9	cmp	#8	;August-Dezember?	35734
	bcs	:11	; => Ja, weiter...	35735
				35736
	ror		;August-Dezember.	35737
	bcc	:10		35738
	bcs	:12	; => OK, weiter...	35739
				35740
::11	ror		;Januar-Juli.	35741
	bcc	:12	; => OK, weiter...	35742

```

35743
35744 ; Auf 30/31 Tage testen.
35745 ::10          lda    day                ;Monate mit 30 Tagen.
35746             cmp    #31                ;Tag gültig?
35747             beq    :7                  ; => Ja, weiter...
35748
35749 ; Uhrzeit testen.
35750 ::12          lda    hour
35751             cmp    #24                ;Stunde gültig?
35752             bcs    :13                ; => Nein, Fehler...
35753
35754             lda    #$00                ;Eingabe i.O.
35755             rts
35756
35757 ::13          ldy    #$06                ;Cursor auf Stunde1.
35758             lda    #$ff                ;Fehler.
35759             rts
35760
35761 ; Index auf String für Datum/Uhrzeit.
35762 :v78CA        b OFF_DAY +0            , OFF_DAY +2
35763             b OFF_MONTH +0          , OFF_MONTH +2
35764             b OFF_MILLENIUM +0    , OFF_MILLENIUM +2
35765             b OFF_YEAR +0          , OFF_YEAR +2
35766             b OFF_HOUR +0          , OFF_HOUR +2
35767             b OFF_MINUTES +0      , OFF_MINUTES +2
35768             b NULL
35769
35770 :v78C9        b $00    ;Cursor-Position.
35771 :v78D7        b $33    ;Max.Taste/Zahl.
35772
35773 ; 2: Disk formatieren.
35774 :N7A06        ldx    curDrive            ;Aktuelles Laufwerk
35775             stx    v79CF                ;Zwischenspeichern.
35776
35777 ; Echtes Laufwerk suchen.
35778 ;           ldx    curDrive
35779 ::loop        lda    driveType -8,x
35780             beq    :next
35781             bpl    :init
35782
35783 ::next         inx
35784             cpx    #12
35785             bcc    :1
35786             ldx    #8
35787 ::1           cpx    v79CF                ;Akt. Laufwerk?
35788             bne    :loop                ; => Nein, weiter...
35789
35790             jmp    z7ABF                ; => Ende.
35791
35792 ::init         txa
35793             jsr    SetDevice            ;Laufwerk setzen.
35794
35795             lda    curDrive
35796             clc
35797             adc    #"A" -8

```

```

        sta    t7B8D                                35798
                                                    35799
; Formatbox anzeigen.                                35800
::dbox      lda    #NULL                            ;String für Diskname 35801
        sta    V0472                                ;löschen.          35802
                                                    35803
        lda    #< V0472                            ;Zeiger Diskname. 35804
        sta    r5L                                35805
        lda    #> V0472                            35806
        sta    r5H                                35807
                                                    35808
; MP3-Icons für Laufwerke definieren:              35809
        ldy    #0                                    35810
        ldx    #4 -1                                35811
::21        lda    driveType,x                      35812
        pha                                35813
        bmi    :22                                35814
        beq    :23                                35815
        iny                                35816
        bne    :23                                35817
::22        lda    #$00                             35818
        sta    driveType,x                      35819
::23        dex                                35820
        bpl    :21                                35821
                                                    35822
        dey                                35823
        bne    :24                                35824
        lda    #NULL                             35825
        b $2c                                     35826
::24        lda    #DRIVE                          35827
        sta    d7B5D                             35828
                                                    35829
; Dialogbox öffnen.                                35830
        lda    #< d7B3B                            ;Name eingeben. 35831
        ldx    #> d7B3B                            35832
        jsr    S2090                                ;DBox ohne Recover. 35833
                                                    35834
; MP3-Konfiguration zurücksetzen:                  35835
        ldx    #0                                    35836
::31        pla                                35837
        sta    driveType,x                      35838
        inx                                35839
        cpx    #4                                    35840
        bcc    :31                                35841
                                                    35842
; Rückmeldung auswerten:                          35843
        lda    r0L                                35844
        bpl    :41                                35845
                                                    35846
; Laufwerk-Icon.                                  35847
        and    #%00001111                        35848
        clc                                35849
        adc    #"A" -8                            35850
        sta    t7B8D                             35851
        bne    :dbox                             35852

```

```

35853
35854 ; Eingabe oder Abbruch.
35855 ::41          cmp    #CANCEL          ;Abbruch?
35856              beq    z7ABF            ; => Ja, Ende...
35857
35858              jmp    z7AC5            ;Disk formatieren.
35859
35860 ; Abbruch.
35861 :z7ABF          lda    v79CF          ;Laufwerk setzen.
35862              jsr    SetDevice
35863
35864              jmp    S54F0            ;Hauptmenü.
35865
35866 ; Laufwerk formatieren.
35867 :z7AC5          lda    t7B8D
35868              sec
35869              sbc    #"A" -8
35870              jsr    SetDevice
35871
35872              lda    #"1"            ;Modus 1571:
35873              sta    v79D5 +4        ;Double-Sided.
35874
35875 ; Ein-/Doppelseitig nur bei 1571
35876 ; abfragen (bei SD2IEC nicht möglich).
35877              jsr    z78D0            ;1571-Laufwerk?
35878              bcc    :1              ; => Keine 1571...
35879
35880              lda    #< d7C47        ;Doppelseitig?
35881              ldx    #> d7C47
35882              jsr    S2090            ;DBox ohne Recover.
35883
35884              lda    r0L
35885              cmp    #CANCEL
35886              beq    z7ABF
35887              cmp    #YES
35888              beq    :1
35889
35890              lda    #"0"            ;Modus 1571:
35891              sta    v79D5 +4        ;Single-Sided.
35892
35893 ::1              jsr    z78D8            ;Formatieren.
35894              txa                    ;Fehler?
35895              beq    :ok              ; => Nein, weiter...
35896
35897              jmp    S5CF3            ;Fehler ausgeben.
35898
35899 ; Andere Fenster für Laufwerk schließen.
35900 ::ok            ldy    #$00          ;Zähler Fens.neu = 0.
35901
35902              ldx    #$04            ;Max. 4 Fenster.
35903 ::loop          dex                    ;Ende?
35904              bmi    :done            ; => Ja, weiter...
35905
35906              lda    C0448,x          ;Fenster geöffnet?
35907              beq    :loop            ; => Nein, weiter...

```

```

35908
35909      lda    C04F8,x          ;Lfwk. für Fenster.
35910      cmp    curDrive        ;Akt. Laufwerk?
35911      bne    :loop           ; => Nein, weiter...
35912
35913      lda    #$00
35914      sta    C0448,x          ;Fenster schließen.
35915      iny
35916      bne    :loop
35917
35918 ; Alle Fenster neu öffnen, auch wenn
35919 ; nur ein Fenster geöffnet war:
35920 ; Das neue Fenster für das formatierte
35921 ; Laufwerk könnte evtl. an anderer
35922 ; Stelle geöffnet werden und das alte
35923 ; Fenster bleibt sichtbar.
35924 ::done      lda    curDrive    ;Lfwk. für Fenster.
35925             pha
35926
35927             jsr    S157E        ;Alle Fens. anzeigen.
35928
35929             pla
35930             jmp    S3AB0        ;Laufwerk öffnen.
35931
35932 ; Auf 1571 testen.
35933 ; Rückgabe: C=1 : 1571
35934 :z78D0      ldx    curDrive
35935             lda    RealDrvType -8,x
35936             cmp    #Drv1571    ;1571?
35937             bne    :1          ; => Nein, weiter...
35938
35939             lda    RealDrvMode -8,x
35940             and    #SET_MODE_SD2IEC
35941             bne    :1          ; => SD2IEC...
35942
35943             sec
35944             rts
35945
35946 ::1          clc
35947             rts
35948
35949 ; Echtes Laufwerk formatieren.
35950 :z78D8      jsr    PurgeTurbo
35951             jsr    InitForIO
35952
35953             lda    #$01          ;Daten für
35954             ldx    curDrive      ;Befehlskanal.
35955             ldy    #$6f
35956             jsr    SETLFS
35957
35958             lda    #$00
35959             jsr    SETNAM
35960             jsr    OPENCHN
35961
35962             lda    curDrive

```

```

35963      jsr    LISTEN
35964      lda    #$6f                ; => REOPEN Channel.
35965      jsr    SECOND
35966
35967      jsr    z78D0                ;1571-Laufwerk?
35968      bcc    :format            ; => Keine 1571...
35969
35970      lda    #< v79D5            ;"U0>Mx" für
35971      sta    r15L                ;Ein-/Doppelseitig.
35972      lda    #> v79D5
35973      sta    r15H
35974      jsr    z79DE                ;Befehl senden.
35975
35976  ::format      lda    #"N"                ;"N0:" senden.
35977      jsr    CIOUT
35978      lda    #"0"
35979      jsr    CIOUT
35980      lda    #":"
35981      jsr    CIOUT
35982
35983      lda    #< V0472            ;Zeiger Diskname.
35984      sta    r15L
35985      lda    #> V0472
35986      sta    r15H
35987      jsr    z79DE                ;Befehl senden.
35988
35989      lda    #< v79D0            ;",TD" senden.
35990      sta    r15L
35991      lda    #> v79D0
35992      sta    r15H
35993      jsr    z79DE                ;Befehl senden.
35994
35995      jsr    z78D0                ;1571-Laufwerk?
35996      bcc    :2                  ; => Keine 1571...
35997
35998      lda    v79D5 +4            ;Modus 1571:
35999      cmp    #"1"                ;Double-Sided?
36000      beq    :2                  ; => Ja, weiter...
36001
36002      lda    #"1"                ;Zurück in den
36003      sta    v79D5 +4            ;Double-Sided-Modus.
36004
36005      lda    #< v79D5            ;"U0>Mx" für
36006      sta    r15L                ;Ein-/Doppelseitig.
36007      lda    #> v79D5
36008      sta    r15H
36009      jsr    z79DE                ;Befehl senden.
36010
36011  ::2          lda    curDrive
36012      jsr    UNLSN
36013
36014      lda    #$00
36015      sta    STATUS
36016
36017      lda    curDrive            ;Fehlerstatus

```



```

        jsr    TALK                ;abfragen.          36018
        lda    #$6f                36019
        jsr    TKSA                36020
                                   36021
        jsr    ACPTR                36022
        sec                        36023
        sbc    #"0"                36024
        asl                        36025
        sta    r0L                 36026
        asl                        36027
        asl                        36028
        clc                        36029
        adc    r0L                 36030
        sta    r0L                 36031
                                   36032
        jsr    ACPTR                36033
        sec                        36034
        sbc    #"0"                36035
        clc                        36036
        adc    r0L                 36037
        sta    r0L                ;Fehlerstatus.      36038
                                   36039
::skip    jsr    ACPTR                36040
                                   36041
        bit    STATUS              36042
        bvc    :skip              36043
                                   36044
        lda    curDrive            36045
        jsr    UNTALK              36046
                                   36047
        lda    #$01                36048
        jsr    CLOSE              36049
                                   36050
        jsr    DoneWithIO          36051
                                   36052
        ldx    r0L                ;Fehlerstatus = $00? 36053
        bne    :err                ; => Nein, Ende... 36054
                                   36055
        lda    #< $2000            ;Verzögerung bis 36056
        sta    r0L                ;Laufwerk bereit. 36057
        lda    #> $2000            36058
        sta    r0H                36059
                                   36060
::wait    ldx    #r0                36061
        jsr    Ddec                36062
        bne    :wait              36063
                                   36064
        jsr    OpenDisk            ;Diskette öffnen. 36065
        txa                        ;Fehler?          36066
        bne    :err                ; => Ja, Abbruch... 36067
                                   36068
; Hinweis:                        36069
; Hier wurde eine echte Diskette neu 36070
; formatiert => Keine RAMDisk.     36071
;                                ;Bei RAMDisk keine 36072
        lda    curType

```

```

36073 ;          and   #DrvRAM          ;GEOS-Disk erzeugen.
36074 ;          bne   :err
36075
36076          jsr    SetGEOSDisk        ;GEOS-Diskette.
36077
36078 ::err          rts
36079
36080 ; Ein oder mehrere Befehle an Laufwerk senden.
36081 ; Trennung durch CR, Ende mit NULL.
36082 ; Übergabe: r15 = Zeiger auf Befehl.
36083 ; verändert: A,X,Y,r14H
36084 :z79DE          ldy   #$00
36085 ::loop          lda   (r15),y
36086                beq   :done
36087
36088                sty   r14H
36089
36090                pha
36091                jsr   CIOUT
36092                pla
36093                cmp   #CR              ;Ende String?
36094                bne   :next            ; => Nein, weiter...
36095
36096                lda   curDrive          ;Befehl ausführen.
36097                jsr   UNLSN
36098
36099                lda   curDrive          ;Laufwerk zurück
36100                jsr   LISTEN            ;auf Empfang.
36101                lda   #$6f
36102                jsr   SECOND
36103
36104 ::next          ldy   r14H
36105                iny
36106                bne   :loop
36107
36108 ::done          rts
36109
36110 ; Zwischenspeicher für Laufwerk.
36111 :v79CF          b   $00
36112
36113 ; ID für Format.
36114 :v79D0          b   ",TD",CR,0
36115
36116 ; Hinweis:
36117 ; Im Anschluss an den U0-Befehl wird
36118 ; immer auch ein I-Befehl gesendet.
36119 :v79D5          b   "U0>MX",CR
36120                b   "I",CR,0
36121
36122 ; Dialogbox: 1571-Modus.
36123 :d7C47          b   %10000001
36124
36125                b   DBTXTSTR,$08,$10
36126                w   :t1
36127

```

	b DBTXTSTR,\$08,\$20	36128
	w :t2	36129
		36130
	b CANCEL ,,\$11,\$48	36131
	b YES ,,\$01,\$48	36132
	b NO ,,\$09,\$48	36133
		36134
	b NULL	36135
		36136
if LANG = LANG_DE		36137
::t1	b BOLDON	36138
	b "Ziel-Laufwerk ist vom Typ 1571:"	36139
	b PLAINTEXT,0	36140
		36141
::t2	b "Diskette doppelseitig formatieren?",0	36142
endif		36143
if LANG = LANG_EN		36144
::t1	b BOLDON	36145
	b "Target drive is of type 1571:"	36146
	b PLAINTEXT,0	36147
		36148
::t2	b "Format disk double-sided?",0	36149
endif		36150
		36151
; Dialogbox: Disk formatieren.		36152
:d7B3B	b %10000001	36153
		36154
	b DBTXTSTR ,,\$08,\$0a	36155
	w T29CF	36156
		36157
	b DBTXTSTR ,,\$08,\$16	36158
	w t7B72	36159
		36160
	b DBTXTSTR ,,\$08,\$22	36161
	w t7B90	36162
	b DBTXTSTR ,,\$08,\$2c	36163
	w t7BB4	36164
		36165
	b DBTXTSTR ,,\$08,\$38	36166
	w t7BDA	36167
		36168
	b DBTXTSTR ,,\$08,\$42	36169
	w T29D5	36170
	b DBGETSTRING	36171
	b \$10,\$3c	36172
	b r5L,16	36173
		36174
	b CANCEL ,,\$11,\$48	36175
		36176
:d7B5D	b DRIVE ,,\$07,\$48	36177
		36178
	b DBTXTSTR ,,\$08,\$54	36179
	w t7BDB	36180
		36181
	b NULL	36182

```
36183
36184 if LANG = LANG_DE
36185 :t7B72          b PLAINTEXT
36186                b "Diskette in Laufwerk "
36187                b BOLDON
36188 :t7B8D          b ".: "
36189                b PLAINTEXT
36190                b " wird formatiert.",0
36191
36192 :t7B90          b "Alle Daten auf der eingelegten Diskette",0
36193 :t7BB4          b "werden vollständig gelöscht!",0
36194 :t7BDA          b "Bitte neuen Diskettenamen eingeben:",0
36195
36196 :t7BDB          b "Laufwerke:",0
36197 endif
36198 if LANG = LANG_EN
36199 :t7B72          b PLAINTEXT
36200                b "Disk in drive "
36201                b BOLDON
36202 :t7B8D          b ".: "
36203                b PLAINTEXT
36204                b " will be formatted.",0
36205
36206 :t7B90          b "All files on the current disk are",0
36207 :t7BB4          b "completey deleted!",0
36208 :t7BDA          b "Enter new disk name:",0
36209
36210 :t7BDB          b "Drives:",0
36211 endif
36212
36213 ;*** Endadresse testen:
36214                g OS_VARS
36215
```

	36216
;	36217
; SourceCode: lnk.DTopDesk.DE.s	36218
;	36219
	36220
;--- GEOS-Dateiname/Info.	36221
n "DTPDESK64"	36222
	36223
h "A fork of TopDesk64..."	36224
h "C64 and MP64/GDOS64 only!"	36225
	36226
m	36227
	36228
- "obj.#00"	36229
- "obj.#01"	36230
- "obj.#02"	36231
- "obj.#03"	36232
- "obj.#04"	36233
- "obj.#05"	36234
- "obj.#06"	36235
- "obj.#07"	36236
- "obj.#08"	36237
- "obj.#09"	36238
- "obj.#10"	36239
- "obj.#11"	36240
- "obj.#12"	36241
	36242
/	36243
	36244

```
36245
36246 ;
36247 ; SourceCode: lnk.DTopDesk.EN.s
36248 ;
36249
36250 ;--- GEOS-Dateiname/Info.
36251         n "DTOPDESK64-EN"
36252
36253         h "A fork of TopDesk64..."
36254         h "C64 and MP64/GDOS64 only!"
36255
36256         m
36257
36258         - "obj.#00"
36259         - "obj.#01"
36260         - "obj.#02"
36261         - "obj.#03"
36262         - "obj.#04"
36263         - "obj.#05"
36264         - "obj.#06"
36265         - "obj.#07"
36266         - "obj.#08"
36267         - "obj.#09"
36268         - "obj.#10"
36269         - "obj.#11"
36270         - "obj.#12"
36271
36272         /
36273
```

```

36274
; 36275
; SourceCode: ass.DTopDesk.s 36276
; 36277
36278
;*** Systemlabels. 36279
if .p 36280
    t "TopSym" 36281
endif 36282
36283
    n "ass.DTopDesk" 36284
    c "ass.SysFile V1.0" 36285
    t "opt.Author" 36286
    f 4 ;SYSTEM 36287
    z $40 ;GEOS64 oder GEOS128 40/80 Zeichen 36288
36289
    o $4000 36290
36291
    h "* AutoAssembler system file." 36292
    h "Create DTopDesk DE/EN..." 36293
36294
;--- DTopDesk DE/EN erstellen. 36295
:DTOPDESK    b $f0,"ass.DTopDeskDE.s",$00 36296
            b $f0,"ass.DTopDeskEN.s",$00 36297
            b $ff 36298
36299
;--- Erlaubte Dateigröße: 8192 Bytes. 36300
;    Datenspeicher von $4000-$5fff. 36301
            g $5fff 36302
36303

```

```
36304
36305 ;
36306 ; SourceCode: ass.DTopDeskDE.s
36307 ;
36308
36309 ;*** Systemlabels.
36310 if .p
36311             t "TopSym"
36312 endif
36313
36314             n "ass.DTopDeskDE"
36315             c "ass.SysFile V1.0"
36316             t "opt.Author"
36317             f 4 ;SYSTEM
36318             z $40 ;GEOS64 oder GEOS128 40/80 Zeichen
36319
36320             o $4000
36321
36322             h "* AutoAssembler Systemdatei."
36323             h "Erstellt DTopDesk..."
36324
36325 ;--- DTopDesk löschen.
36326 :DTOPDESK_DEL    b $f0,"DTopDeskI18N.DE",$00
36327                 t "a01.DTopDeskDE"
36328                 t "a01.DTopDesk"
36329
36330 ;--- DTOPDESK linken.
36331                 b $f5
36332                 b $f0,"lnk.DTopDesk.DE",$00
36333                 b $f4
36334
36335 ;--- Erlaubte Dateigröße: 8192 Bytes.
36336 ;   Datenspeicher von $4000-$5fff.
36337                 g $5fff
36338
```



```

36339
; 36340
; SourceCode: ass.DTopDeskEN.s 36341
; 36342
36343
;*** Systemlabels. 36344
if .p 36345
    t "TopSym" 36346
endif 36347
36348
    n "ass.DTopDeskEN" 36349
    c "ass.SysFile V1.0" 36350
    t "opt.Author" 36351
    f 4 ;SYSTEM 36352
    z $40 ;GEOS64 oder GEOS128 40/80 Zeichen 36353
36354
    o $4000 36355
36356
    h "* AutoAssembler Systemdatei." 36357
    h "Erstellt DTopDesk..." 36358
36359
;--- DTopDesk löschen. 36360
:DTOPDESK_DEL    b $f0,"DTopDeskI18N.EN",$00 36361
                t "a01.DTopDeskEN" 36362
                t "a01.DTopDesk" 36363
36364
;--- DTOPDESK linken. 36365
                b $f5 36366
                b $f0,"lnk.DTopDesk.EN",$00 36367
                b $f4 36368
36369
;--- Erlaubte Dateigröße: 8192 Bytes. 36370
;    Datenspeicher von $4000-$5fff. 36371
                g $5fff 36372
36373

```

```
36374
36375 ;
36376 ; SourceCode: a01.DTopDesk.s
36377 ;
36378
36379 ;--- DTOPDESK.
36380 b $f0,"src.DTopDesk64",$00
36381 b $f0,"src.mod#01",$00
36382 b $f0,"src.mod#02",$00
36383 b $f0,"src.mod#03",$00
36384 b $f0,"src.mod#04",$00
36385 b $f0,"src.mod#05",$00
36386 b $f0,"src.mod#06",$00
36387 b $f0,"src.mod#07",$00
36388 b $f0,"src.mod#08",$00
36389 b $f0,"src.mod#09",$00
36390 b $f0,"src.mod#10",$00
36391 b $f0,"src.mod#11",$00
36392 b $f0,"src.mod#12",$00
36393
```

```

36394
; 36395
; SourceCode: a01.DTopDeskDE.s 36396
; 36397
36398
;--- DTopDesk löschen. 36399
:DTOPDESK_DEL_DE    b $f1 36400
36401
        jsr    OpenDisk 36402
36403
        lda    #< :1          ;DTOPDESK löschen. 36404
        sta    r0L 36405
        lda    #> :1 36406
        sta    r0H 36407
        jsr    DeleteFile 36408
36409
        lda    #< :next 36410
        sta    a0L 36411
        lda    #> :next 36412
        sta    a0H 36413
36414
        rts 36415
36416
::1      b "DTOPDESK64", $00 36417
36418
::next 36419
36420

```

```
36421
36422 ;
36423 ; SourceCode: a01.DTopDeskEN.s
36424 ;
36425
36426 ;--- DTopDesk löschen.
36427 :DTOPDESK_DEL_EN    b $f1
36428
36429             jsr    OpenDisk
36430
36431             lda    #< :1                ;DTOPDESK löschen.
36432             sta    r0L
36433             lda    #> :1
36434             sta    r0H
36435             jsr    DeleteFile
36436
36437             lda    #< :next
36438             sta    a0L
36439             lda    #> :next
36440             sta    a0H
36441
36442             rts
36443
36444 ::1         b "DTOPDESK64-EN", $00
36445
36446 ::next
36447
```

	36448
;	36449
; SourceCode: TopSym.s	36450
;	36451
	36452
; Systemlabels	36453
; Version 04.07.1989	36454
;	36455
; Revision 26.03.2023:	36456
; APP_LVAR und APP_LRAM ergänzt.	36457
;	36458
; Revision 26.12.2022:	36459
; sysApplData ergänzt.	36460
;	36461
; Revision 03.12.2022:	36462
; sysVersion ergänzt.	36463
:APPLICATION = \$06	36464
:APPL_DATA = \$07	36465
:APP_LVAR = \$0200	36466
:APP_LRAM = \$0334	36467
:APP_RAM = \$0400	36468
:APP_VAR = \$7f40	36469
:ASSEMBLY = \$02	36470
:AUTO_EXEC = \$0e	36471
:AllocateBlock = \$9048	36472
:AppendRecord = \$c289	36473
:BACKSPACE = \$08	36474
:BACK_SCR_BASE = \$6000	36475
:BASIC = \$01	36476
:BBMult = \$c160	36477
:BMult = \$c163	36478
:BOLDON = \$18	36479
:BRKVector = \$84af	36480
:BitOtherClip = \$c2c5	36481
:BitmapClip = \$c2aa	36482
:BitmapUp = \$c142	36483
:BldGDirEntry = \$c1f3	36484
:BlkAlloc = \$c1fc	36485
:BlockProcess = \$c10c	36486
:BootGEOS = \$c000	36487
:CANCEL = \$02	36488
:CBM = \$05	36489
:COLOR_MATRIX = \$8c00	36490
:CONSTRAINED = \$40	36491
:CPU_DATA = \$01	36492
:CPU_DDR = \$00	36493
:CR = \$0d	36494
:CRC = \$c20e	36495
:CalcBlksFree = \$c1db	36496
:CallRoutine = \$c1d8	36497
:ChangeDiskDevice = \$c2bc	36498
:ChkDkGEOS = \$c1de	36499
:ClearMouseMode = \$c19c	36500
:ClearRam = \$c178	36501
:CloseRecordFile = \$c277	36502

```
36503 :CmpFString = $c26e
36504 :CmpString = $c26b
36505 :CopyFString = $c268
36506 :CopyString = $c265
36507 :DATA = $03
36508 :DBGETFILES = $10
36509 :DBGETSTRING = $0d
36510 :DBGRPHSTR = $0f
36511 :DBOPVEC = $11
36512 :DBSYSOPV = $0e
36513 :DBTXTSTR = $0b
36514 :DBUSRICON = $12
36515 :DBVARSTR = $0c
36516 :DB_USR_ROUT = $13
36517 :DESK_ACC = $05
36518 :DISK = $06
36519 :DISK_BASE = $9000
36520 :DMult = $c166
36521 :DSdiv = $c16c
36522 :DShiftLeft = $c15d
36523 :DShiftRight = $c262
36524 :DYN_SUB_MENU = $40
36525 :Dabs = $c16f
36526 :Ddec = $c175
36527 :Ddiv = $c169
36528 :DeleteFile = $c238
36529 :DeleteRecord = $c283
36530 :DisablSprite = $c1d5
36531 :Dnegate = $c172
36532 :DoDlgBox = $c256
36533 :DoIcons = $c15a
36534 :DoInlineReturn = $c2a4
36535 :DoMenu = $c151
36536 :DoPreviousMenu = $c190
36537 :DoRAMOp = $c2d4
36538 :DoneWithIO = $c25f
36539 :DrACurDkNm = $841e
36540 :DrBCurDkNm = $8430
36541 :DrCCurDkNm = $88dc
36542 :DrDCurDkNm = $88ee
36543 :DrawLine = $c130
36544 :DrawPoint = $c133
36545 :DrawSprite = $c1c6
36546 :ESC_GRAPHICS = $10
36547 :ESC_PUTSTRING = $06
36548 :ESC_RULER = $11
36549 :EXP_BASE = $df00
36550 :EnablSprite = $c1d2
36551 :EnableProcess = $c109
36552 :EnterDeskTop = $c22c
36553 :EnterTurbo = $c214
36554 :ExitTurbo = $c232
36555 :FALSE = $00
36556 :FONT = $08
36557 :FORWARDSPACE = $09
```

:FRAME_RECT0 = \$07	36558
:FastDelFile = \$c244	36559
:FetchRAM = \$c2cb	36560
:FillRam = \$c17b	36561
:FindBAMBit = \$c2ad	36562
:FindFTypes = \$c23b	36563
:FindFile = \$c20b	36564
:FirstInit = \$c271	36565
:FollowChain = \$c205	36566
:FrameRectangle = \$c127	36567
:FreeBlock = \$c2b9	36568
:FreeFile = \$c226	36569
:FreezeProcess = \$c112	36570
:GOTOX = \$14	36571
:GOTOXY = \$16	36572
:GOTOY = \$15	36573
:Get1stDirEntry = \$9030	36574
:GetBlock = \$c1e4	36575
:GetCharWidth = \$c1c9	36576
:GetDimensions = \$790c	36577
:GetDirHead = \$c247	36578
:GetFHdrInfo = \$c229	36579
:GetFile = \$c208	36580
:GetFreeDirBlk = \$c1f6	36581
:GetNextChar = \$c2a7	36582
:GetNxtDirEntry = \$9033	36583
:GetPtrCurDkNm = \$c298	36584
:GetRandom = \$c187	36585
:GetRealSize = \$c1b1	36586
:GetScanLine = \$c13c	36587
:GetSerialNumber = \$c196	36588
:GetString = \$c1ba	36589
:GotoFirstMenu = \$c1bd	36590
:GraphicsString = \$c136	36591
:HOME = \$0b	36592
:HORIZONTAL = \$00	36593
:HorizontalLine = \$c118	36594
:INCOMPATIBLE = \$0e	36595
:INPUT_128 = \$0f	36596
:INPUT_DEVICE = \$0a	36597
:IO_IN = \$35	36598
:IRQ_VECTOR = \$fffe	36599
:ITALICON = \$19	36600
:ImprintRectangle = \$c250	36601
:InitForIO = \$c25c	36602
:InitForPrint = \$7900	36603
:InitMouse = \$fe80	36604
:InitProcesses = \$c103	36605
:InitRam = \$c181	36606
:InitTextPrompt = \$c1c0	36607
:InsertRecord = \$c286	36608
:InterruptMain = \$c100	36609
:InvertLine = \$c11b	36610
:InvertRectangle = \$c12a	36611
:IsMseInRegion = \$c2b3	36612

```
36613 :KEY_BPS = $18
36614 :KEY_CLEAR = $13
36615 :KEY_DELETE = $1d
36616 :KEY_DOWN = $11
36617 :KEY_F1 = $01
36618 :KEY_F2 = $02
36619 :KEY_F3 = $03
36620 :KEY_F4 = $04
36621 :KEY_F5 = $05
36622 :KEY_F6 = $06
36623 :KEY_F7 = $0e
36624 :KEY_F8 = $0f
36625 :KEY_HOME = $12
36626 :KEY_INSERT = $1c
36627 :KEY_INVALID = $1f
36628 :KEY_LARROW = $14
36629 :KEY_LEFT = $08
36630 :KEY_RIGHT = $1e
36631 :KEY_RUN = $17
36632 :KEY_STOP = $16
36633 :KEY_UP = $10
36634 :KEY_UPARROW = $15
36635 :KRNL_BAS_IO_IN = $37
36636 :KRNL_IO_IN = $36
36637 :LF = $0a
36638 :LINETO = $02
36639 :LdApplic = $c21d
36640 :LdDeskAcc = $c217
36641 :LdFile = $c211
36642 :LoadCharSet = $c1cc
36643 :MENU_ACTION = $00
36644 :MOUSE_BASE = $fe80
36645 :MOUSE_JMP = $fe80
36646 :MOVEPENTO = $01
36647 :MainLoop = $c1c3
36648 :MouseOff = $c18d
36649 :MouseUp = $c18a
36650 :MoveData = $c17e
36651 :NEWCARDSET = $17
36652 :NEWPATTERN = $05
36653 :NMI_VECTOR = $fffa
36654 :NO = $04
36655 :NOT_GEOS = $00
36656 :NULL = $00
36657 :NewDisk = $c1e1
36658 :NextRecord = $c27a
36659 :NxtBlkAlloc = $c24d
36660 :OK = $01
36661 :OPEN = $05
36662 :OUTLINEON = $1a
36663 :OpenDisk = $c2a1
36664 :OpenRecordFile = $c274
36665 :PAGE_BREAK = $0c
36666 :PEN_XY_DELTA = $0a
36667 :PEN_X_DELTA = $08
```



:PEN_Y_DELTA = \$09	36668
:PLAINTEXT = \$1b	36669
:PRG = \$02	36670
:PRINTBASE = \$7900	36671
:PRINTER = \$09	36672
:Panic = \$c2c2	36673
:PointRecord = \$c280	36674
:PosSprite = \$c1cf	36675
:PreviousRecord = \$c27d	36676
:PrintASCII = \$790f	36677
:PrintBuffer = \$7906	36678
:PrntDiskName = \$8476	36679
:PrntFileName = \$8465	36680
:PrntFilename = PrntFileName	36681
:PromptOff = \$c29e	36682
:PromptOn = \$c29b	36683
:PurgeTurbo = \$c235	36684
:PutBlock = \$c1e7	36685
:PutChar = \$c145	36686
:PutDecimal = \$c184	36687
:PutDirHead = \$c24a	36688
:PutString = \$c148	36689
:RAM_64K = \$30	36690
:RECTANGLETO = \$03	36691
:REL = \$04	36692
:RESET_VECTOR = \$fffc	36693
:REV_OFF = \$13	36694
:REV_ON = \$12	36695
:ReDoMenu = \$c193	36696
:ReadBlock = \$c21a	36697
:ReadByte = \$c2b6	36698
:ReadFile = \$c1ff	36699
:ReadLink = \$904b	36700
:ReadRecord = \$c28c	36701
:RecoverAllMenus = \$c157	36702
:RecoverLine = \$c11e	36703
:RecoverMenu = \$c154	36704
:RecoverRectangle = \$c12d	36705
:RecoverVector = \$84b1	36706
:Rectangle = \$c124	36707
:RenameFile = \$c259	36708
:ResetHandle = \$c003	36709
:RestartProcess = \$c106	36710
:RstrAppl = \$c23e	36711
:RstrFrmDialogue = \$c2bf	36712
:SCREEN_BASE = \$a000	36713
:SEQ = \$01	36714
:SEQUENTIAL = \$00	36715
:SET_BLOCKED = \$40	36716
:SET_BOLD = \$40	36717
:SET_DB_POS = \$00	36718
:SET_FROZEN = \$20	36719
:SET_ICONSON = \$20	36720
:SET_INPUTCHG = \$40	36721
:SET_ITALIC = \$10	36722

```
36723 :SET_KEYPRESS = $80
36724 :SET_LEFTJUST = $80
36725 :SET_MENUON = $40
36726 :SET_MOUSE = $20
36727 :SET_MSE_ON = $80
36728 :SET_NOSUPRESS = $00
36729 :SET_NOTIMER = $10
36730 :SET_OFFBOTTOM = $40
36731 :SET_OFFLEFT = $20
36732 :SET_OFFMENU = $08
36733 :SET_OFFRIGHT = $10
36734 :SET_OFFTOP = $80
36735 :SET_OUTLINE = $08
36736 :SET_PLAINTEXT = $00
36737 :SET_REVERSE = $20
36738 :SET_RIGHTJUST = $00
36739 :SET_RUNABLE = $80
36740 :SET_SUBSCRIPT = $02
36741 :SET_SUPERSCRIPT = $04
36742 :SET_SUPRESS = $40
36743 :SET_UNDERLINE = $80
36744 :SHORTCUT = $80
36745 :SPRITE_PICS = $8a00
36746 :STATUS = $90
36747 :STRUCT_MISMAT = $0a
36748 :ST_FLASH = $80
36749 :ST_INVERT = $40
36750 :ST_LD_AT_ADDR = $01
36751 :ST_LD_DATA = $80
36752 :ST_PR_DATA = $40
36753 :ST_WRGs_FORE = $20
36754 :ST_WR_BACK = $40
36755 :ST_WR_FORE = $80
36756 :ST_WR_PR = $40
36757 :SUB_MENU = $80
36758 :SYSTEM = $04
36759 :SYSTEM_BOOT = $0c
36760 :SaveFile = $c1ed
36761 :SetDevice = $c2b0
36762 :SetGDirEntry = $c1f0
36763 :SetGEOSDisk = $c1ea
36764 :SetMouse = $fe89
36765 :SetNLQ = $7915
36766 :SetNextFree = $c292
36767 :SetPattern = $c139
36768 :Sleep = $c199
36769 :SlowMouse = $fe83
36770 :SmallPutChar = $c202
36771 :StartASCII = $7912
36772 :StartAppl = $c22f
36773 :StartMouseMode = $c14e
36774 :StartPrint = $7903
36775 :StashRAM = $c2c8
36776 :StopPrint = $7909
36777 :StringFaultVec = $84ab
```

:SwapRAM = \$c2ce	36778
:TAB = \$09	36779
:TEMPORARY = \$0d	36780
:TRUE = \$ff	36781
:TestPoint = \$c13f	36782
:ToBasic = \$c241	36783
:ULINEOFF = \$0f	36784
:ULINEON = \$0e	36785
:UN_CONSTRAINED = \$00	36786
:UPLINE = \$0c	36787
:USELAST = \$7f	36788
:USR = \$03	36789
:UnblockProcess = \$c10f	36790
:UnfreezeProcess = \$c115	36791
:UpdateMouse = \$fe86	36792
:UpdateRecordFile = \$c295	36793
:UseSystemFont = \$c14b	36794
:VERTICAL = \$80	36795
:VLIR = \$01	36796
:VerWriteBlock = \$c223	36797
:VerifyRAM = \$c2d1	36798
:VerticalLine = \$c121	36799
:WriteBlock = \$c220	36800
:WriteFile = \$c1f9	36801
:WriteRecord = \$c28f	36802
:YES = \$03	36803
:a0 = \$fb	36804
:a0H = \$fc	36805
:a0L = \$fb	36806
:a1 = \$fd	36807
:a1H = \$fe	36808
:a1L = \$fd	36809
:a2 = \$70	36810
:a2H = \$71	36811
:a2L = \$70	36812
:a3 = \$72	36813
:a3H = \$73	36814
:a3L = \$72	36815
:a4 = \$74	36816
:a4H = \$75	36817
:a4L = \$74	36818
:a5 = \$76	36819
:a5H = \$77	36820
:a5L = \$76	36821
:a6 = \$78	36822
:a6H = \$79	36823
:a6L = \$78	36824
:a7 = \$7a	36825
:a7H = \$7b	36826
:a7L = \$7a	36827
:a8 = \$7c	36828
:a8H = \$7d	36829
:a8L = \$7c	36830
:a9 = \$7e	36831
:a9H = \$7f	36832

```
36833 :a9L = $7e
36834 :alarmSetFlag = $851c
36835 :alarmTmtVector = $84ad
36836 :alphaFlag = $84b4
36837 :appMain = $849b
36838 :baselineOffset = $26
36839 :bkvec = $0316
36840 :bootName = $c006
36841 :c128Flag = $c013
36842 :cardDataPntr = $2c
36843 :curDevice = $ba
36844 :curDirHead = $8200
36845 :curDrive = $8489
36846 :curHeight = $29
36847 :curIndexTable = $2a
36848 :curPattern = $22
36849 :curRecord = $8496
36850 :curSetWidth = $27
36851 :curType = $88c6
36852 :currentMode = $2e
36853 :dataDiskName = $8453
36854 :dataFileName = $8442
36855 :dateCopy = $c018
36856 :day = $8518
36857 :dblClickCount = $8515
36858 :dir2Head = $8900
36859 :dirEntryBuf = $8400
36860 :diskBlkBuf = $8000
36861 :diskOpenFlg = $848a
36862 :dispBufferOn = $2f
36863 :dlgBoxRamBuf = $851f
36864 :driveData = $88bf
36865 :driveType = $848e
36866 :faultData = $84b6
36867 :fileHeader = $8100
36868 :fileSize = $8499
36869 :fileTrScTab = $8300
36870 :fileWritten = $8498
36871 :firstBoot = $88c5
36872 :hour = $8519
36873 :i_BitmapUp = $c1ab
36874 :i_FillRam = $c1b4
36875 :i_FrameRectangle = $c1a2
36876 :i_GraphicsString = $c1a8
36877 :i_ImprintRectangle = $c253
36878 :i_MoveData = $c1b7
36879 :i_PutString = $c1ae
36880 :i_RecoverRectangle = $c1a5
36881 :i_Rectangle = $c19f
36882 :iconSelFlag = $84b5
36883 :inputData = $8506
36884 :inputDevName = $88cb
36885 :inputVector = $84a5
36886 :intBotVector = $849f
36887 :intTopVector = $849d
```

:interleave = \$848c	36888
:irqvec = \$0314	36889
:isGEOS = \$848b	36890
:kernalVectors = \$031a	36891
:keyData = \$8504	36892
:keyVector = \$84a3	36893
:leftMargin = \$35	36894
:maxMouseSpeed = \$8501	36895
:menuNumber = \$84b7	36896
:minMouseSpeed = \$8502	36897
:minutes = \$851a	36898
:month = \$8517	36899
:mouseAccel = \$8503	36900
:mouseBottom = \$84b9	36901
:mouseData = \$8505	36902
:mouseFaultVec = \$84a7	36903
:mouseLeft = \$84ba	36904
:mouseOn = \$30	36905
:mousePicData = \$84c1	36906
:mouseRight = \$84bc	36907
:mouseTop = \$84b8	36908
:mouseVector = \$84a1	36909
:mouseXPos = \$3a	36910
:mouseYPos = \$3c	36911
:msePicPtr = \$31	36912
:nationality = \$c010	36913
:nmivec = \$0318	36914
:numDrives = \$848d	36915
:obj0Pointer = \$8ff8	36916
:obj1Pointer = \$8ff9	36917
:obj2Pointer = \$8ffa	36918
:obj3Pointer = \$8ffb	36919
:obj4Pointer = \$8ffc	36920
:obj5Pointer = \$8ffd	36921
:obj6Pointer = \$8ffe	36922
:obj7Pointer = \$8fff	36923
:otherPressVec = \$84a9	36924
:pressFlag = \$39	36925
:r0 = \$02	36926
:r0H = \$03	36927
:r0L = \$02	36928
:r1 = \$04	36929
:r10 = \$16	36930
:r10H = \$17	36931
:r10L = \$16	36932
:r11 = \$18	36933
:r11H = \$19	36934
:r11L = \$18	36935
:r12 = \$1a	36936
:r12H = \$1b	36937
:r12L = \$1a	36938
:r13 = \$1c	36939
:r13H = \$1d	36940
:r13L = \$1c	36941
:r14 = \$1e	36942

```
36943 :r14H = $1f
36944 :r14L = $1e
36945 :r15 = $20
36946 :r15H = $21
36947 :r15L = $20
36948 :r1H = $05
36949 :r1L = $04
36950 :r2 = $06
36951 :r2H = $07
36952 :r2L = $06
36953 :r3 = $08
36954 :r3H = $09
36955 :r3L = $08
36956 :r4 = $0a
36957 :r4H = $0b
36958 :r4L = $0a
36959 :r5 = $0c
36960 :r5H = $0d
36961 :r5L = $0c
36962 :r6 = $0e
36963 :r6H = $0f
36964 :r6L = $0e
36965 :r7 = $10
36966 :r7H = $11
36967 :r7L = $10
36968 :r8 = $12
36969 :r8H = $13
36970 :r8L = $12
36971 :r9 = $14
36972 :r9H = $15
36973 :r9L = $14
36974 :ramBase = $88c7
36975 :ramExpSize = $88c3
36976 :random = $850a
36977 :rasreg = $d012
36978 :returnAddress = $3d
36979 :rightMargin = $37
36980 :saveFontTab = $850c
36981 :savedmoby2 = $88bb
36982 :scr80colors = $88bd
36983 :scr80polar = $88bc
36984 :screencolors = $851e
36985 :seconds = $851b
36986 :selectionFlash = $84b3
36987 :sidbase = $d400
36988 :spr0pic = $8a00
36989 :spr1pic = $8a40
36990 :spr2pic = $8a80
36991 :spr3pic = $8ac0
36992 :spr4pic = $8b00
36993 :spr5pic = $8b40
36994 :spr6pic = $8b80
36995 :spr7pic = $8bc0
36996 :string = $24
36997 :stringX = $84be
```



```

37012
37013 ;
37014 ; SourceCode: TopSym.MP3.s
37015 ;
37016
37017 ; Symboltabelle für GEOS/MegaPatch3
37018 ; Revision 29.10.2022
37019
37020 ; Erweiterte Systemadressen
37021 :DskDrvBaseL      = $9f7e
37022 :DskDrvBaseH      = $9f82
37023 :doubleSideFlg    = $9f86
37024 :drivePartData    = $9f8a
37025 :RealDrvType       = $9f8e
37026 :RealDrvMode       = $9f92
37027 :RamBankInUse      = $9f96
37028 :RamBankFirst      = $9fa6
37029 ; :GEOS_RAM_TYP    = $9fa8
37030 :MP3_64K_SYSTEM    = $9fa9
37031 :MP3_64K_DATA      = $9faa
37032 :MP3_64K_DISK      = $9fab
37033 ; :Flag_Optimize    = $9fac
37034 :millenium          = $9fad
37035 ; :Flag_LoadPrnt    = $9fae
37036 ; :PrntFileNameRAM  = $9faf
37037 ; :Flag_Spooler     = $9fc0
37038 ; :Flag_SpoolMinB   = $9fc1
37039 ; :Flag_SpoolMaxB   = $9fc2
37040 ; :Flag_SpoolADDR   = $9fc3
37041 ; :Flag_SpoolCount  = $9fc6
37042 ; :Flag_SplCurDok   = $9fc7
37043 ; :Flag_SplMaxDok   = $9fc8
37044 :Flag_TaskAktiv     = $9fc9
37045 :Flag_TaskBank      = $9fca
37046 :Flag_ExtRAMinUse   = $9fcb
37047 ; :Flag_ScrSvCnt    = $9fcc
37048 ; :Flag_ScrSaver    = $9fcd
37049 :Flag_CrsrRepeat    = $9fce
37050 :BackScrPattern     = $9fcf
37051 :Flag_SetColor      = $9fd0
37052 :Flag_ColorDBox    = $9fd1
37053 ; :Flag_IconMinX    = $9fd2
37054 ; :Flag_IconMinY    = $9fd3
37055 ; :Flag_IconDown    = $9fd4
37056 :Flag_DBoxType      = $9fd5 ;Used by kernal only.
37057 ; :Flag_GetFiles    = $9fd6 ;Used by kernal only.
37058 ; :DB_GFileType     = $9fd7
37059 ; :DB_GFileClass    = $9fd8
37060 ; :DB_GetFileEntry  = $9fda
37061 ; :DB_StdBoxSize    = $9fdb
37062 ; :Flag_SetMLine    = $9fe1
37063 ; :Flag_MenuStatus  = $9fe2
37064 ; :DM_LastEntry     = $9fe3
37065 ; :DM_LastNumEntry  = $9fe9
37066 :MP3_COLOR_DATA     = $9fea

```



	37067
<b>; Systemadressen der Farbtabelle</b>	37068
<b>:C_Balken</b> = \$9fea	37069
<b>:C_Register</b> = \$9feb	37070
<b>:C_RegisterOff</b> = \$9fec	37071
<b>:C_RegisterBack</b> = \$9fed	37072
<b>:C_Mouse</b> = \$9fee	37073
<b>:C_DBoxTitel</b> = \$9fef	37074
<b>:C_DBoxBack</b> = \$9ff0	37075
<b>:C_DBoxDIcon</b> = \$9ff1	37076
<b>:C_FBoxTitel</b> = \$9ff2	37077
<b>:C_FBoxBack</b> = \$9ff3	37078
<b>:C_FBoxDIcon</b> = \$9ff4	37079
<b>:C_FBoxFiles</b> = \$9ff5	37080
<b>:C_WinTitel</b> = \$9ff6	37081
<b>:C_WinBack</b> = \$9ff7	37082
<b>:C_WinShadow</b> = \$9ff8	37083
<b>:C_WinIcon</b> = \$9ff9	37084
<b>:C_PullDMenu</b> = \$9ffa	37085
<b>:C_InputField</b> = \$9ffb	37086
<b>:C_InputFieldOff</b> = \$9ffc	37087
<b>:C_GEOS_BACK</b> = \$9ffd	37088
<b>:C_GEOS_FRAME</b> = \$9ffe	37089
<b>:C_GEOS_MOUSE</b> = \$9fff	37090
	37091
<b>; Speichererweiterungen</b>	37092
<b>; :RAM_SCPU</b> = \$10	37093
<b>; :RAM_BBG</b> = \$20	37094
<b>; :RAM_GEORAM</b> = \$20	37095
<b>; :RAM_REU</b> = \$40	37096
<b>; :RAM_RL</b> = \$80	37097
	37098
<b>; Flag_SetColor: Dialogbox-Farben setzen</b>	37099
<b>:SET_DBOXCOL_OFF</b> = \$00	37100
<b>:SET_DBOXCOL_STD</b> = \$40	37101
<b>:SET_DBOXCOL_ON</b> = \$80	37102
	37103
<b>; Zusätzliche Symbole für DoDlgBox</b>	37104
<b>:DBOXCOLON</b> = %01000000	37105
<b>:DBOXCOLOFF</b> = %00000000	37106
<b>:DRIVE</b> = \$07	37107
<b>:DUMMY</b> = \$08	37108
<b>:DBUSRFILES</b> = \$09	37109
<b>:DBSETCOL</b> = \$0a	37110
<b>:DBSELECTPART</b> = %10000000	37111
<b>:DBSETDRVICON</b> = %01000000	37112
<b>:DBOXDRVA</b> = \$88	37113
<b>:DBOXDRVB</b> = \$89	37114
<b>:DBOXDRVC</b> = \$8a	37115
<b>:DBOXDRVD</b> = \$8c	37116
	37117
<b>; Neue Kernal-Routinen</b>	37118
<b>:i_UserColor</b> = \$c0dc	37119
<b>:i_ColorBox</b> = \$c0df	37120
<b>:DirectColor</b> = \$c0e2	37121

```

37122 :RecColorBox      = $c0e5
37123 :GetBackScreen    = $c0e8
37124 :ResetScreen      = $c0eb
37125 :GEOS_InitSystem   = $c0ee
37126 :PutKeyInBuffer    = $c0f1
37127 :SCPU_Pause        = $c0f4
37128
37129 ; SuperCPU-Optimierung für GEOS
37130 ; :SCPU_OptOn        = $c0f7
37131 ; :SCPU_OptOff       = $c0fa
37132 ; :SCPU_SetOpt       = $c0fd
37133
37134 ; Register-Menu
37135 ; :BOX_USER           = $01
37136 ; :BOX_USER_VIEW     = $02
37137 ; :BOX_USEROPT       = $03
37138 ; :BOX_USEROPT_VIEW  = $04
37139 ; :BOX_FRAME         = $05
37140 ; :BOX_ICON          = $06
37141 ; :BOX_ICON_VIEW    = $07
37142 ; :BOX_OPTION        = $08
37143 ; :BOX_OPTION_VIEW   = $09
37144 ; :BOX_STRING        = $0a
37145 ; :BOX_STRING_VIEW   = $0b
37146 ; :BOX_NUMERIC       = $0c
37147 ; :BOX_NUMERIC_VIEW = $0d
37148 ; :NUMERIC_LEFT      = %00000000
37149 ; :NUMERIC_RIGHT     = %10000000
37150 ; :NUMERIC_SETSPC    = %00000000
37151 ; :NUMERIC_SET0      = %01000000
37152 ; :NUMERIC_BYTE      = %00000000
37153 ; :NUMERIC_WORD      = %00100000
37154 ; :USE_COLOR_INPUT   = $ff
37155 ; :USE_COLOR_REG     = $ee
37156 ; :NO_OPT_UPDATE     = $00
37157
37158 ; Registermenü-Routinen
37159 ; :DoRegister         = $6d00
37160 ; :ExitRegisterMenu  = $6d03
37161 ; :RegisterInitMenu  = $6d06
37162 ; :RegisterUpdate    = $6d09
37163 ; :RegisterAllOpt    = $6d0c
37164 ; :RegisterNextOpt   = $6d0f
37165 ; :RegDrawOptFrame   = $6d12
37166 ; :RegClrOptFrame    = $6d15
37167 ; :RegisterSetFont   = $6d18
37168 ; :RegisterAktiv     = $6d1b
37169
37170 ; Systemadressen Registermenü
37171 ; :SetADDR_Register = $cfe6 ;Zeiger auf Registermenü
37172 :LD_ADDR_REGISTER  = $6d00 ;Ladeadresse Registermenü
37173
37174 ; Systemadressen erweiterte Routinen.
37175 :SetADDR_EnterDT   = $cfe3 ;Zeiger auf EnterDeskTop-Routine
37176

```

; Erweiterte Adressen im Laufwerkstreiber		37177
:GetBlock_dskBuf	= \$903c	37178
:PutBlock_dskBuf	= \$903f	37179
:DiskDrvType	= \$904e	37180
:DiskDrvVersion	= \$904f	37181
:OpenRootDir	= \$9050	37182
:OpenSubDir	= \$9053	37183
:GetBAMBlock	= \$9056	37184
:PutBAMBlock	= \$9059	37185
:GetPDirEntry	= \$905c	37186
:ReadPDirEntry	= \$905f	37187
:OpenPartition	= \$9062	37188
:SwapPartition	= \$9065	37189
:GetPTypeData	= \$9068	37190
:SendFloppyCom	= \$906b	37191
:DiskDrvTypeExt	= \$9074	37192
:DDRV_EXT_DATA1	= \$907a	37193
:DDRV_EXT_DATA2	= \$907b	37194
:InitForDskDvOp	= \$907c	37195
:DoneWithDskDvOp	= \$907f	37196
:dir3Head	= \$9c80	37197
		37198
; Verschiedene Symbole		37199
:OS_VARS	= \$8000 ;OS variable base	37200
:MP3_CODE	= \$c014	37201
:DDRV_CODE	= \$906e	37202
:DDX_CODE	= \$9074	37203
; :BASE_AUTO_BOOT	= \$5000 ;Ladeadresse der AutoBoot-Routine.	37204
; :SIZE_AUTO_BOOT	= \$0500 ;max. Größe der AutoBoot-Routine.	37205
		37206
; Erweiterte Diskettenfehlermeldungen		37207
:NO_ERROR	= \$00	37208
:NO_BLOCKS	= \$01	37209
:INV_TRACK	= \$02	37210
; :INSUFF_SPACE	= \$03	37211
:FULL_DIRECTORY	= \$04	37212
:FILE_NOT_FOUND	= \$05	37213
:BAD_BAM	= \$06	37214
; :UNOPENED_VLIR	= \$07	37215
:INV_RECORD	= \$08	37216
:OUT_OF_RECORDS	= \$09	37217
; :STRUCT_MISMAT	= \$0a ;In TopSym definiert.	37218
:BFR_OVERFLOW	= \$0b	37219
:CANCEL_ERR	= \$0c	37220
:DEV_NOT_FOUND	= \$0d	37221
; :INCOMPATIBLE	= \$0e ;In TopSym definiert.	37222
; :HDR_NOT_THERE	= \$20	37223
:NO_SYNC	= \$21	37224
; :DBLK_NOT_THERE	= \$22	37225
; :DAT_CHKSUM_ERR	= \$23	37226
; WR_VER_ERR	= \$25	37227
:WR_PR_ON	= \$26	37228
; :HDR_CHKSUM_ERR	= \$27	37229
; :DSK_ID_MISMAT	= \$29	37230
; :BYTE_DEC_ERR	= \$2e	37231

```

37232 ; :NO_PARTITION      = $30
37233 ; :PART_FORMAT_ERR   = $31
37234 ; :ILLEGAL_PARTITION= $32
37235 ; :NO_PART_FD_ERR    = $33
37236 ; :ILLEGAL_DEVICE    = $40
37237 :NO_FREE_RAM         = $60
37238 :DOS_MISMATCH        = $73
37239
37240 ; Definition der Laufwerkstypen
37241 :ST_DMODES            = %00000111
37242 ; :DRIVE_MODES       = %00000111
37243 :Drv1541              = $01
37244 :Drv1571              = $02
37245 :Drv1581              = $03
37246 :DrvIECBNM           = $04
37247 :DrvSD2IEC           = $04
37248 :DrvNative           = $04
37249 :DrvPCDOS            = $05
37250 :Drv81DOS             = $05
37251
37252 :DrvShadow             = %01000000
37253 :DrvShadow1541        = DrvShadow ! Drv1541
37254 ; :DrvShadow1571     = DrvShadow ! Drv1571
37255 ; :DrvShadow1581     = DrvShadow ! Drv1581
37256 ; :DrvShadowNM       = DrvShadow ! DrvNative
37257
37258 :DrvRAM                 = %10000000
37259 :DrvRAM1541            = DrvRAM ! Drv1541
37260 :DrvRAM1571            = DrvRAM ! Drv1571
37261 :DrvRAM1581            = DrvRAM ! Drv1581
37262 :DrvRAMNM             = DrvRAM ! DrvNative
37263
37264 :DrvCREU                = %10100000
37265 :DrvRAMNM_CREU        = DrvCREU ! DrvNative
37266 :DrvGRAM              = %10110000
37267 :DrvRAMNM_GRAM        = DrvGRAM ! DrvNative
37268 :DrvSCPU               = %11000000
37269 :DrvRAMNM_SCPU        = DrvSCPU ! DrvNative
37270
37271 :DrvFD                  = %00010000
37272 :DrvFD41              = DrvFD ! Drv1541
37273 :DrvFD71              = DrvFD ! Drv1571
37274 :DrvFD81              = DrvFD ! Drv1581
37275 :DrvFD2               = DrvFD
37276 :DrvFD4               = DrvFD
37277 :DrvFDNM              = DrvFD ! DrvNative
37278 :DrvFDDOS              = DrvFD ! DrvPCDOS
37279
37280 :DrvHD                  = %00100000
37281 :DrvHD41              = DrvHD ! Drv1541
37282 :DrvHD71              = DrvHD ! Drv1571
37283 :DrvHD81              = DrvHD ! Drv1581
37284 :DrvHDNM              = DrvHD ! DrvNative
37285
37286 :DrvRAMLink             = %00110000

```

```

:DrvRL41          = DrvRAMLink ! Drv1541      37287
:DrvRL71          = DrvRAMLink ! Drv1571      37288
:DrvRL81          = DrvRAMLink ! Drv1581      37289
:DrvRLNM          = DrvRAMLink ! DrvNative    37290
                                           37291
:DrvCMD           = %00110000                37292
                                           37293
; Definition der Laufwerks-Modi              37294
:SET_MODE_PARTITION = %10000000              37295
:SET_MODE_SUBDIR   = %01000000              37296
:SET_MODE_FASTDISK = %00100000              37297
:SET_MODE_SRAM     = %00010000              37298
:SET_MODE_CRAM     = %00001000              37299
:SET_MODE_GRAM     = %00000100              37300
:SET_MODE_SD2IEC   = %00000010              37301
                                           37302
; CBM-Dateitypen                             37303
:FMODE_CLOSED      = %10000000              37304
:FMODE_WRPROT      = %01000000              37305
:FTYPE_MODES       = %00000111              37306
;FTYPE_DEL         = $00                     37307
;FTYPE_SEQ         = $01                     37308
;FTYPE_PRG         = $02                     37309
;FTYPE_USR         = $03                     37310
;FTYPE_REL         = $04                     37311
;FTYPE_DIR         = $06                     37312
                                           37313
; Sonstige C64-Systemadressen                 37314
:zpage             = $0000                   37315
                                           37316

```

```

37317
37318 ;
37319 ; SourceCode: TopSym.ROM.s
37320 ;
37321
37322 ; Einsprünge im C64-Kernal
37323 ; Revision 29.10.2022
37324
37325 :IOINIT          = $fda3 ;Reset: CIA.
37326 :CINT           = $ff81 ;Reset: Timer, IO, PAL/NTSC, Bildschirm.
37327 ; :IOINIT       = $ff84 ;Reset: CIA.
37328 :SETMSG         = $ff90 ;Dateiparameter definieren.
37329 :SECOND         = $ff93 ;Sekundär-Adresse nach LISTEN senden.
37330 :TKSA          = $ff96 ;Sekundär-Adresse nach TALK senden.
37331 :ACPTR         = $ffa5 ;Byte-Eingabe vom IEC-Bus.
37332 :CIOUT         = $ffa8 ;Byte-Ausgabe auf IEC-Bus.
37333 :UNTALK        = $ffab ;UNTALK-Signal auf IEC-Bus senden.
37334 :UNLSN        = $ffae ;UNLISTEN-Signal auf IEC-Bus senden.
37335 :LISTEN        = $ffb1 ;LISTEN-Signal auf IEC-Bus senden.
37336 :TALK         = $ffb4 ;TALK-Signal auf IEC-Bus senden.
37337 :SETLFS        = $ffba ;Dateiparameter setzen.
37338 :SETNAM        = $ffbd ;Dateiname setzen.
37339 :OPENCHN       = $ffc0 ;Datei öffnen.
37340 :CLOSE         = $ffc3 ;Datei schließen.
37341 :CHKIN         = $ffc6 ;Eingabefile setzen.
37342 :CKOUT         = $ffc9 ;Ausgabefile setzen.
37343 :CLRCHN        = $ffcc ;Standard-I/O setzen.
37344 :BSOUT         = $ffd2 ;Zeichen ausgeben.
37345 :LOAD          = $ffd5 ;Datei laden.
37346 :GETIN         = $ffe4 ;Tastatur-Eingabe.
37347 :CLALL         = $ffe7 ;Alle Kanäle schließen.
37348
37349 ;*** Einsprünge im RAMLink-Kernal.
37350 :EN_SET_REC     = $e0a9 ;Enable RAMLink, set REC page.
37351 :RL_HW_EN      = $e0b1 ;Enable RAMLink, turn off interrupts.
37352 :SET_REC_IMG   = $fe03 ;Set REC page.
37353 :EXEC_REC_REU   = $fe06 ;Execute according to REU register.
37354 :EXEC_REC_SEC   = $fe09 ;Execute according to sector register.
37355 :RL_HW_DIS     = $fe0c ;Disable RAMLink, turn interrupts on.
37356 :RL_HW_DIS2    = $fe0f ;Disable RAMLink, leave interrupts off.
37357 :EXEC_REU_DIS   = $fe1e ;Exec REU, Disable RL, interrupts on.
37358 :EXEC_SEC_DIS   = $fe21 ;Exec sector, Disable RL, interrupts on.
37359

```

37360

