

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

```
166 ; Nach dem beenden von GEOS wird ein
167 ; Hinweis angezeigt, das der Computer
168 ; jetzt abgeschaltet werden kann oder
169 ; Leertaste drücken um in den BASIC-
170 ; Modus zu wechseln.
171 ; Diese TopDesk-Routine verwendet eine
172 ; Vielzahl an Kernal-Routinen und ist
173 ; komplexer als die GEOS-Routine.
174 ; Der Verzicht auf den BASIC-Hinweis
175 ; spart zudem Speicherplatz.
176 ;
177 ; Standard:
178 ; ENABLE_BASICINFO=FALSE: Deaktiviert
179 ;
180 ; Alternativ:
181 ; ENABLE_BASICINFO=TRUE : Aktiviert
182 ;
183 .ENABLE_BASICINFO    = FALSE
184
185 ; Fenster optimieren:
186 ;
187 ; * EXPERIMENTEL *
188 ;
189 ; TopDesk kann Fenster nur teilweise
190 ; neu zeichnen. Die Routine FrontWindow
191 ; testen auf sich überlappende Fenster
192 ; und zeichnet ggf. nur einen Teil des
193 ; Fensters neu. Die Routine wurde aber
194 ; nie komplettiert, daher werden die
195 ; Fenster immer komplett gezeichnet.
196 ; Funktioniert nur im Icon-Modus!
197 ;
198 ; Standard:
199 ; ENABLE_OPTWIN=FALSE: Nicht optimieren
200 ;
201 ; EXPERIMENTEL:
202 ; ENABLE_OPTWIN=TRUE : Fens. optimieren
203 ;
204 :ENABLE_OPTWIN        = FALSE
205
```

```
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 ;
241 ; SourceCode: opt.DT64.Date.s
242 ;
243
244 ; Build-Datum für DTopDesk64.
245 b "Build: 20241028"
246
247 ;
248 ; SourceCode: opt.DT64DE.Build.s
249 ;
250
251 ; GEOS-Klasse für DTopDesk64.
252 b "DT64-R02-DE V1.04"
253
254 ;
255 ; SourceCode: opt.DT64DE.Class.s
256 ;
257
258 ; GEOS-Klasse für DTopDesk64.
259 c "DT64-R02-DE V1.04"
260
261 ;
262 ; SourceCode: opt.DT64EN.Build.s
263 ;
264
265 ; GEOS-Klasse für DTopDesk64.
266 b "DT64-R02-EN V1.04"
267
268 ;
269 ; SourceCode: opt.DT64EN.Class.s
270 ;
271
272 ; GEOS-Klasse für DTopDesk64.
273 c "DT64-R02-EN V1.04"
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_MENUUSE = TRUE
321 :MENU_VMODE        = CONSTRAINED
322 endif
323 if ENABLE_MENUUSE = FALSE
324 :MENU_VMODE        = UN_CONSTRAINED
325 endif
326
327 ; Horizontal immer UN_CONSTRAINED!
328 :MENU_HMODE        = UN_CONSTRAINED
329

```



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

```

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

```

```

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

```

```

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

```

```
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
```

```

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

```

```
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.

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

1269      jsr    FetchRAM
1270
1271      pla                      ;Zeiger auf
1272      asl                      ;Modul-Adresse.
1273      tay
1274      dey
1275
1276      dey
1277      lda    (r0),y
1278      sta    r1L
1279      iny
1280      lda    (r0),y
1281      sta    r1H
1282      iny
1283
1284      lda    (r0),y            ;Größe Modul
1285      sec                      ;berechnen.
1286      sbc    r1L
1287      sta    r2L
1288      iny
1289      lda    (r0),y
1290      sbc    r1H
1291      sta    r2H
1292
1293      jsr    FetchRAM          ;Modul einlesen.
1294      clc
1295      rts
1296
1297 ; VLIR-Mod. aus Datei laden.
1298 ; Rückgabe: C=0: OK.
1299 ; TD13: GetModule
1300 :Z077F      pha
1301
1302      jsr    S07D6              ;Systemlaufwerk.
1303      bcc    :3                ; => Ok, weiter...
1304
1305      pla                      ;Modul-Nr. verwerfen.
1306 ;      sec
1307      bcs    Z07B7              ;Lfwk. zurücksetzen.
1308
1309 ::3      lda    #< C0400        ;Dateiname TopDesk.
1310      sta    r0L
1311      lda    #> C0400
1312      sta    r0H
1313      jsr    OpenRecordFile
1314
1315      pla
1316      sta    V0422              ;Aktuelles VLIR-Mod.
1317      jsr    PointRecord
1318
1319      lda    V0425 +0           ;VLIR-Adressbereich.
1320      sta    r7L                ; => BASE_MOD_A/B.
1321      lda    V0425 +1
1322      sta    r7H
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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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:

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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          ldy    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

```

```

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

```

```

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

```

```

2369 :Z0E17      jsr   ReadBlock
2370           txa
2371           bne   :done
2372
2373           jsr   Z0EA0           ;Weitere Daten?
2374           beq   :done           ; => Nein, Ende...
2375
2376           ldy   r4H
2377           iny
2378           cpy   MHB04AC         ;HB Ende CopyBuf.
2379           bcc   Z0E15
2380
2381 ; Speicher voll oder Fehler.
2382 ::done      rts
2383
2384 ; Sektorkette auf Disk schreiben.
2385 ; Übergabe: r9 = Nächster Block.
2386 ;           r4 = Zeiger auf Daten.
2387 ; Rückgabe: r1 = Tr/Se des nächsten Block.
2388 ;           r4 = Zeiger auf letzten Block.
2389 ; TD13: WriteChain
2390 :Z0E34      lda   r9L
2391           sta   r3L
2392           sta   r1L
2393           lda   r9H
2394           sta   r3H
2395           sta   r1H
2396
2397           dec   r4H
2398
2399 ::1         inc   r4H
2400           jsr   SetNextFree
2401           txa
2402           bne   :done
2403
2404           ldy   #0
2405           lda   (r4),y           ;Weitere Daten?
2406           beq   :2               ; => Nein, Ende...
2407
2408           lda   r3L
2409           sta   (r4),y
2410           iny
2411           lda   r3H
2412           sta   (r4),y
2413
2414 ::2         jsr   WriteBlock      ;Block schreiben.
2415           txa
2416           bne   :done
2417
2418           lda   r3L
2419           sta   r1L
2420           sta   r9L
2421           lda   r3H
2422           sta   r1H
2423           sta   r9H

```

```

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

```

```
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
```

```

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

```

```

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

```

```

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)

```

```

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

```



```

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

```

```

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 ; Datei löschen.
3016 ::file      lda    fileHeader +0
3017             ldx    fileHeader +1
3018             jmp    S0B34          ;Datei löschen.
3019
3020 ; Auf Datei/Ordner/Verzeichnis testen.
3021 :Z10F0      ldx    #%10000000    ;$80 = NM/Dir.
3022
3023             lda    dirEntryBuf    ;Dateityp einlesen.
3024             and    #FTYPE_MODES
3025             cmp    #FTYPE_DIR     ;CMD-Verzeichnis?
3026             beq    :1             ; => Ja, Ende...
3027
3028             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.

```

```

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

```

```

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



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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 herstell.
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

```



```

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

```

```

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

```

```

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

```

```

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...

```

```

4019          sta    r3L
4020
4021 ::5          lda    r3L
4022          clc
4023          adc    r4L
4024          sta    r4L
4025          lda    r3H
4026          adc    r4H
4027          sta    r4H
4028
4029          lda    mouseYPos
4030          sec
4031          sbc    V16B2
4032
4033          ldy    V16B8          ;Nur ganze Cards?
4034          bne    :6          ; => Nein, weiter...
4035
4036          and    #%11111000          ;Abrunden...
4037 ::6          sta    r2L
4038          clc
4039          adc    r2H
4040          sta    r2H
4041
4042          ldy    #(r4H - r2L)
4043 ::7          lda    r2,y
4044          sta    V16AC,y
4045          dey
4046          bpl    :7
4047
4048          jsr    Z2515          ;Fens.bereich zeigen.
4049
4050          lda    mouseXPos +0          ;Mausposition
4051          sta    V1763 +0          ;zwischenspeichern.
4052          lda    mouseXPos +1
4053          sta    V1763 +1
4054
4055          lda    mouseYPos
4056          sta    V1765
4057
4058          plp
4059          rts
4060
4061 :V1763      w $0000          ;Letzte X-Position
4062 :V1765      b $00          ;Letzte Y-Position
4063
4064 ; Rahmengröße ändern über Mauszeiger-Koordinaten.
4065 ; TD13: DoRubber (Initialisierung)
4066 :Z1766      lda    r0L          ;Zeiger auf
4067          ldx    r0H          ;Verschieberoutine.
4068          jsr    Z16A0          ;Neue Routine setzen.
4069
4070          php
4071          sei
4072
4073          ldy    #(r4H - r2L)

```

```

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.

```

```

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

```

```

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.

```



```

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

```

```

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

```

```

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?

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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.

```

```

4789         clc
4790         adc    #< (8 -1)
4791         sta    r4L
4792         lda    r3H
4793         adc    #> (8 -1)
4794         sta    r4H
4795
4796         jsr    IsMseInRegion      ;Diskname ändern?
4797         beq    :1                 ; => Nein, weiter...
4798
4799 ; Klick Titelzeile: Diskname ändern.
4800         jsr    Z0680              ;Lfwk. = Native?
4801         bne    :0                 ; => Nein, weiter...
4802
4803         ldx    C042A              ;Oberstes Fenster.
4804         lda    V04F0,x            ;Dir.T. für Fenster.
4805         ora    V04F4,x            ;Dir.S. für Fenster.
4806         beq    :0
4807
4808         jsr    OpenRootDir        ;Hauptverzeichnis.
4809
4810         ldx    C042A              ;Oberstes Fenster.
4811         stx    WIN042F            ;Aktives Fenster.
4812
4813 ;         ldx    C042A              ;Oberstes Fenster.
4814         jsr    S09B7              ;Tr/Se SubDir = $00.
4815
4816         jsr    S5842              ;mark.Dat. anzeigen.
4817         jsr    S524C              ;Dat.-Ausw. aufheben.
4818
4819         jsr    S44E8              ;Fens. erneut öffnen.
4820
4821 ::0         jmp    MR39F9          ;Neuer Diskname.
4822
4823 ; Klick auf Statuszeile?
4824 ::1         jsr    Z1ABF          ;Fens.koord. Statusz.
4825
4826         lda    r3L
4827         clc
4828         adc    #SUP -1
4829         jsr    Z1B7E              ;Dateien wählen?
4830         beq    :3                 ; => Nein, weiter...
4831
4832 ; Hinweis:
4833 ; Klick Statuszeile: Dateien wählen.
4834 ; Kurzer Klick: Seite wählen.
4835 ; Langer Klick: Alle wählen.
4836         inc    V04AE              ;Klick auf Gadget.
4837
4838         jsr    S48B6              ;Langer Mausklick?
4839         bcc    :2                 ; => Ja, weiter...
4840
4841         jmp    MR58A0              ;Seite wählen.
4842 ::2         jmp    MR589D          ;Alle Dateien wählen.
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.

```

```

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

```

```

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

```

```

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

```

```

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
5120 ::3          dec    r0H
5121             ldy    r0H
5122             bpl    :1
5123
5124             lda    r1H          ;Fenster verdeckt?
5125             beq    :4          ; => Nein, weiter...
5126
5127             plp
5128             sec              ;Fens. neu zeichnen.
5129             php
5130
5131 ::4          ldx    r1L          ;Neues FrontWindow.
5132             ldy    V1C9F        ;Startfenster.
5133
5134 if ENABLE_OPTWIN = TRUE
5135             pha              ;Anz. verd. Fenster.
5136 endif
5137
5138             jsr    Z1CA0        ;Fens. sortieren.
5139
5140 if ENABLE_OPTWIN = TRUE
5141             pla              ;Anz. verd. Fenster.
5142 endif
5143
5144             plp              ;Fesn. neu zeichnen?
5145
5146             bit    V04AE        ;Klick auf Gadget?
5147             bmi    :7          ; => Ja, Ende...
5148
5149             bcc    :6          ; => N. neu zeichnen.
5150
5151 ; Fenster verdeckt, neu zeichnen.
5152 ::5
5153 if ENABLE_OPTWIN = TRUE
5154             cmp    #2          ;Mehr als ein Fens.?
5155             bcc    :5a        ; => Nein, weiter...
5156 endif
5157
5158 ; Fenster immer neu zeichnen.
5159             inc    V18AD        ;Fens.-Aufbau-Status.
5160 ::5c          ldx    C042A        ;Oberstes Fenster.
5161             jmp    S141D        ;Fenster anzeigen.
5162
5163 if ENABLE_OPTWIN = TRUE
5164 ; Fenster durch ein Fenster verdeckt.
5165 ::5a          lda    V18AD        ;Fens.-Aufbau-Status.
5166             bne    :5c        ; => Neu zeichnen.
5167
5168             ldx    V1CB2        ;Verd. Fenster.
5169             jsr    S11A9        ;a0=Fensterdaten.
5170
5171             ldy    #6 -1        ;Fenstergrenzen für
5172 ::5b          lda    (a0),y      ;Berechnung der
5173             sta    windowTop,y  ;Schnittfläche.

```

```

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.

```

```

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

```

```
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.
```

```

5339 ; TD13: DrawShadow/RemoveShadow
5340 :Z1D7A      lda    #$00                ;Akt.Laufwerk zeigen.
5341           b $2c
5342 :Z1D7D      lda    #$01                ;Kein Lfwk. aktiv.
5343           sta    V1DE1
5344
5345 ; Markierung für alle Laufwerke löschen.
5346           lda    #0                    ;S/W-Modus:
5347           jsr    SetPattern            ;Anzeige löschen.
5348
5349           jsr    i_Rectangle
5350           b PANEL_Y0 +$08
5351           b PANEL_Y0 +$08 +4*$18 -1
5352           w PANEL_X1 -$07
5353           w PANEL_X1 -$01
5354
5355 ; Hinweis:
5356 ; Wird durch i_Rectangle gesetzt.
5357 ;           lda    #PANEL_Y0 +$08
5358 ;           sta    r2L
5359 ;           lda    #PANEL_Y0 +$08 +$68 -1
5360 ;           sta    r2H
5361
5362           lda    COL043A                ;Farbe für alle
5363           jsr    Z1DED                  ;Lfwk. löschen.
5364
5365           ldx    C042A                  ;Oberstes Fenster.
5366           lda    C0448,x                ;Fenster geöffnet?
5367           beq    :done                  ; => Nein, weiter...
5368
5369 ; Aktives Laufwerk anzeigen.
5370 ::open      lda    a0H                  ;Zeiger auf Fenster-
5371           pha                    ;daten speichern.
5372           lda    a0L
5373           pha
5374
5375 ; Titelzeile invertieren.
5376 ;           ldx    C042A                  ;Oberstes Fenster.
5377           jsr    Z11A3                  ;Fens.koord. r2-r4.
5378
5379           lda    r2L
5380           clc
5381           adc    # T0U
5382           sta    r2H
5383
5384 ;           lda    r2L
5385 ;           clc
5386 ;           adc    # T00
5387 ;           sta    r2L
5388           inc    r2L
5389
5390           lda    #< T0L
5391           clc
5392           adc    r3L
5393           sta    r3L

```

```

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                ; Zeiger auf Fens.Dat.
5415         sta     a0L                ; zurücksetzen.
5416         pla
5417         sta     a0H
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                ; Pos. für Anzeige
5435         adc     #6                ; aktives Lfwk.
5436         sta     r1H
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     #""                ; Aktives Laufwerk.
5446         jsr     SmallPutChar
5447
5448         jmp     Z13D3                ; Textgrenzen DeskTop.

```

```

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

```

```

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

```



```

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

```

```

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)

```

```

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

```

```

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

```

```

5779      eor    #$07
5780      tax
5781      lda    (r5),y
5782      and    V2048,x
5783      pha
5784
5785      lda    #$00
5786      ldx    r13H
5787      ldy    r11H
5788      jsr    Z1FB0          ;Bits verschieben.
5789
5790      ldy    r11L
5791      sta    r11L
5792      pla
5793      ora    r11L
5794      jsr    Z1FDA          ;Bits schreiben.
5795      sty    r11L
5796      rts
5797
5798 ; WORD rotieren.
5799 ; Übergabe: A = Highbyte
5800 ;           X = Lowbyte
5801 ;           Y = Anzahl Schritte
5802 ; TD13: Rotieren
5803 :Z1FB0      pha
5804      tya          ;Verschieben = 0 ?
5805      beq    :4          ; => Ja, Ende...
5806
5807      pla
5808      sty    V1FD8          ;Anzahl Schritte.
5809      stx    V1FD9          ;Lowbyte.
5810
5811 ::1        lsr          ;Highbyte um y
5812      ror    V1FD9          ;Schritte nach rechts
5813      bcc    :2          ;rotieren.
5814      ora    #%10000000
5815 ::2        dey
5816      bne    :1
5817
5818      pha
5819
5820      lda    #8          ;Lowbyte um (8-y)
5821      sec          ;nach links schieben,
5822      sbc    V1FD8
5823      tay
5824      lda    V1FD9
5825 ::3        lsr
5826      dey
5827      bne    :3
5828
5829      tax
5830
5831 ::4        pla
5832      rts
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

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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-

```

```

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

```

```

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.

```

```

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

```

```

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

```



```

6439 :V23B1          b > 0*40*8, > 1*40*8, > 2*40*8, > 3*40*8
6440                b > 4*40*8, > 5*40*8, > 6*40*8, > 7*40*8
6441                b > 8*40*8, > 9*40*8, >10*40*8, >11*40*8
6442                b >12*40*8, >13*40*8, >14*40*8, >15*40*8
6443                b >16*40*8, >17*40*8, >18*40*8, >19*40*8
6444                b >20*40*8, >21*40*8, >22*40*8, >23*40*8
6445                b >24*40*8
6446
6447 ; Offset Grafikspalte.
6448 :V23CA          b < 0*8, < 1*8, < 2*8, < 3*8, < 4*8
6449                b < 5*8, < 6*8, < 7*8, < 8*8, < 9*8
6450                b <10*8, <11*8, <12*8, <13*8, <14*8
6451                b <15*8, <16*8, <17*8, <18*8, <19*8
6452                b <20*8, <21*8, <22*8, <23*8, <24*8
6453                b <25*8, <26*8, <27*8, <28*8, <29*8
6454                b <30*8, <31*8, <32*8, <33*8, <34*8
6455                b <35*8, <36*8, <37*8, <38*8, <39*8
6456
6457 :V23F2          b > 0*8, > 1*8, > 2*8, > 3*8, > 4*8
6458                b > 5*8, > 6*8, > 7*8, > 8*8, > 9*8
6459                b >10*8, >11*8, >12*8, >13*8, >14*8
6460                b >15*8, >16*8, >17*8, >18*8, >19*8
6461                b >20*8, >21*8, >22*8, >23*8, >24*8
6462                b >25*8, >26*8, >27*8, >28*8, >29*8
6463                b >30*8, >31*8, >32*8, >33*8, >34*8
6464                b >35*8, >36*8, >37*8, >38*8, >39*8
6465
6466 ; Offset Farbzeile.
6467 :V241A          b < 0*40, < 1*40, < 2*40, < 3*40
6468                b < 4*40, < 5*40, < 6*40, < 7*40
6469                b < 8*40, < 9*40, <10*40, <11*40
6470                b <12*40, <13*40, <14*40, <15*40
6471                b <16*40, <17*40, <18*40, <19*40
6472                b <20*40, <21*40, <22*40, <23*40
6473                b <24*40
6474
6475 :V2433          b > 0*40, > 1*40, > 2*40, > 3*40
6476                b > 4*40, > 5*40, > 6*40, > 7*40
6477                b > 8*40, > 9*40, >10*40, >11*40
6478                b >12*40, >13*40, >14*40, >15*40
6479                b >16*40, >17*40, >18*40, >19*40
6480                b >20*40, >21*40, >22*40, >23*40
6481                b >24*40
6482
6483 ; Ermittlung einer Stringlänge in Zeichen und in Pixel.
6484 ; Übergabe: r0 = Zeiger auf Sstring, max. 256 Zeichen.
6485 ;           Ende mit NULL oder $A0.
6486 ; Rückgabe: X  = Anzahl Zeichen.
6487 ;           r1 = Breite in Pixel.
6488 ; TD13: StringLen
6489 .S244C          lda    #$00
6490                sta    r1L
6491                sta    r1H
6492                sta    r13H
6493                sta    r13L

```

```

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

```

```

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

```

```

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:

```

```

6769 ; r3   - Zeiger auf Speicher für
6770 ;       Dateieinträge.
6771 ; r10L - Ordernummer.
6772 ;       Bit%7=1: Alle Ordner.
6773 ; r11L - max. Anzahl Dateien im
6774 ;       Verzeichnis einlesen.
6775 ; r11H - Anzahl Einträge überlesen.
6776 ; r12L - Bit%76=11: Ignorieren von
6777 ;       gelöschten/leeren Einträgen.
6778 ;       Bit%76=10: Gelöschte einlesen,
6779 ;       leere Einträge ignorieren.
6780 ;       Bit%76=00: Dateien, gelöschte
6781 ;       und leere Einträge einlesen.
6782 ; r12H - GEOS-Dateityp oder
6783 ;       Bit%7=1: Alle Dateitypen.
6784 ; r13L - Anzahl der Bytes aus einem
6785 ;       Dateieintrag einlesen.
6786 ; r13H - Anzahl von Bytes im Eintrag
6787 ;       überlesen.
6788 ;
6789 ; Rückgabe:
6790 ; r11L - Anzahl Dateien eingelesen.
6791 ; r14   - Anzahl Dateien ab Offset.
6792 ; r15L - Flag: Überlauf aufgetreten.
6793 ; r15H - Kopie Offset für Berechnung
6794 ;       Gesamtanzahl Dateien.
6795 ; TD13: FindDirFiles
6796 .S2603          lda    r3L                ;Zeiger auf Speicher
6797                 sta    V2668 +0          ;Dateieinträge.
6798                 lda    r3H
6799                 sta    V2668 +1
6800
6801                 lda    r11L
6802                 sta    V2666 +0
6803                 lda    r11H
6804                 sta    V2666 +1
6805
6806                 lda    #$00              ;Temp.-Datei löschen.
6807                 sta    V2667
6808
6809 :Z260D           lda    V2668 +0
6810                 sta    r3L
6811                 lda    V2668 +1
6812                 sta    r3H
6813
6814                 lda    V2666 +0
6815                 sta    r11L
6816                 lda    V2666 +1
6817                 sta    r11H
6818
6819 ; Hinweis:
6820 ; Die Original-Routine liest mehr als
6821 ; 255 Dateien obwohl Offset + Anzahl
6822 ; nicht >255 sein dürfen (Überlauf).
6823                 lda    #$00              ;Anzahl Dateien ab

```

```

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 ;

```



```

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

```

```

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

```

```

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

```

```

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...

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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.

```

```

7429 ; Wird vor dem laden einer Anwendung
7430 ; oder eines Dokuments gesetzt.
7431 ;
7432 ; TODO:
7433 ; Wozu ist das noch erforderlich?
7434 ; Wenn nach dem beenden nicht mehr zum
7435 ; TopDesk zurückgekehrt wird, dann
7436 ; bleibt die Anzahl in numDrives ggf.
7437 ; bei 2, auch wenn 4 vorhanden.
7438 ;
7439 ; Anwendungen die numDrives auswerten
7440 ; werden damit auf max. zwei Laufwerke
7441 ; reduziert.
7442 ;
7443 ; TD13: TestNumDrives
7444 :Z291E          lda    numDrives
7445                cmp    #2
7446                bcs    :1
7447
7448                lda    #1
7449                b $2c
7450 ::1            lda    #2
7451                sta    numDrives
7452                rts
7453
7454 ; Datei öffnen/ausführen.
7455 ; Übergabe: r6    = Zeiger auf Dateiname.
7456 ;             r0    = Zeiger auf Ordnerliste
7457 ;             r1L   = PrintFlag (Negativ = Drucken!)
7458 ; Rückgabe: X     = Fehler.
7459 ;             r10L  = Nur Ordner: Ordner-Nr.
7460 ; TD13: NewGetFile
7461 .S2931          lda    r1L
7462                sta    V25CF                ;PrintFlag speichern.
7463
7464                lda    r6L
7465                sta    r15L
7466                lda    r6H
7467                sta    r15H
7468                jsr    FindFile                ;Datei suchen.
7469                txa
7470                bne    :2
7471
7472                jsr    Z10F0                ;Ordner/Verz./Datei?
7473                txa
7474                bpl    :1                ; => Kein Verz.
7475
7476                jsr    S09BC                ;1.Block > r1L/r1H.
7477
7478                ldx    #CMD_SUBDIR                ;Fehler: Verzeichnis.
7479                rts
7480
7481 ::1            ldy    dirEntryBuf +22
7482                sty    V25D0                ;Dateityp für öffnen.
7483                cpy    #16                ;GEOS-Dateityp < 16

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

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

```

```

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

```

```

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

```



```

8158          asl
8159          asl
8160          ldy    r1L
8161          cmp    (a9),y
8162          beq    :3
8163          sta    (a9),y
8164
8165          lda    #$01
8166          sta    V04AF          ;Neue PadColorPref.
8167
8168 ::3          pla
8169          and    #%11110000
8170          inc    r1L
8171          ldy    r1L
8172          cmp    (a9),y
8173          beq    :4
8174          sta    (a9),y
8175
8176          lda    #$01
8177          sta    V04AF          ;Neue PadColorPref.
8178
8179 ::4          inc    r1L
8180          inx
8181          cpx    #8
8182          bne    :2
8183
8184 ::exit       rts
8185
8186 ; Preferences suchen/laden.
8187 ; TD13: GetPrefs
8188 .S327F       lda    #< F32C5          ;"Preferences"
8189             ldx    #> F32C5
8190             jsr    Z325D          ;Datenblock laden.
8191             bcs    :exit          ; => Fehler, Ende...
8192
8193             ldy    #3 -1
8194 ::2          lda    diskBlkBuf +2,y
8195             sta    maxMouseSpeed,y
8196             dey
8197             bpl    :2
8198
8199             jsr    InitForIO
8200
8201             lda    diskBlkBuf +5
8202             ora    diskBlkBuf +6
8203             sta    screencolors      ;GEOS-Farben.
8204
8205             lda    diskBlkBuf +5
8206             sta    $d021          ;Hintergrundfarbe.
8207             lda    diskBlkBuf +7
8208             sta    $d027          ;Spritefarbe.
8209             lda    diskBlkBuf +71
8210             sta    $d020          ;Rahmenfarbe.
8211
8212             jsr    DoneWithIO

```

```

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

```

8268		b \$5e,\$00,\$62,\$00,\$66,\$00,\$68,\$00
8269		b \$6b,\$00,\$6e,\$00,\$72,\$00,\$75,\$00
8270		b \$79,\$00,\$7d,\$00,\$81,\$00,\$85,\$00
8271		b \$89,\$00,\$8d,\$00,\$91,\$00,\$95,\$00
8272		b \$9a,\$00,\$9e,\$00,\$a0,\$00,\$a3,\$00
8273		b \$a7,\$00,\$ab,\$00,\$b1,\$00,\$b6,\$00
8274		b \$ba,\$00,\$be,\$00,\$c2,\$00,\$c6,\$00
8275		b \$ca,\$00,\$ce,\$00,\$d2,\$00,\$d6,\$00
8276		b \$dc,\$00,\$e0,\$00,\$e4,\$00,\$e8,\$00
8277		b \$ec,\$00,\$f0,\$00,\$f4,\$00,\$f8,\$00
8278		b \$fd,\$00,\$00,\$01,\$04,\$01,\$08,\$01
8279		b \$0b,\$01,\$0f,\$01,\$13,\$01,\$16,\$01
8280		b \$1a,\$01,\$1e,\$01,\$20,\$01,\$23,\$01
8281		b \$27,\$01,\$29,\$01,\$2f,\$01,\$33,\$01
8282		b \$37,\$01,\$3b,\$01,\$3f,\$01,\$42,\$01
8283		b \$46,\$01,\$49,\$01,\$4d,\$01,\$51,\$01
8284		b \$57,\$01,\$5b,\$01,\$5f,\$01,\$63,\$01
8285		b \$67,\$01,\$6b,\$01,\$6f,\$01,\$74,\$01
8286		b \$74,\$01
8287		
8288	::line1	b \$0a,\$94,\$4c,\$90,\$94,\$a8,\$00,\$00
8289		b \$45,\$98,\$5c,\$9c,\$88,\$00,\$03,\$19
8290		b \$31,\$b3,\$b9,\$95,\$2a,\$88,\$a4,\$98
8291		b \$98,\$dd,\$55,\$55,\$57,\$55,\$52,\$02
8292		b \$04,\$02,\$04,\$11,\$28,\$80,\$00,\$00
8293		b \$01,\$00,\$00,\$00,\$0a,\$aa,\$e0,\$00
8294		
8295	::line2	b \$0a,\$be,\$6d,\$24,\$a2,\$71,\$00,\$02
8296		b \$ac,\$44,\$d1,\$05,\$54,\$05,\$d0,\$a2
8297		b \$aa,\$2a,\$22,\$15,\$2a,\$8d,\$b5,\$55
8298		b \$55,\$09,\$55,\$55,\$51,\$27,\$05,\$01
8299		b \$36,\$26,\$48,\$d8,\$0a,\$b4,\$c4,\$c6
8300		b \$4d,\$aa,\$aa,\$aa,\$e6,\$40,\$90,\$00
8301		
8302	::line3	b \$08,\$14,\$c2,\$1e,\$22,\$fb,\$87,\$04
8303		b \$a4,\$89,\$59,\$88,\$8d,\$28,\$09,\$3b
8304		b \$b2,\$2b,\$ba,\$dd,\$2c,\$8a,\$ad,\$59
8305		b \$58,\$89,\$55,\$52,\$22,\$55,\$50,\$00
8306		b \$55,\$4a,\$ad,\$55,\$2c,\$aa,\$aa,\$aa
8307		b \$99,\$2a,\$aa,\$4a,\$4a,\$aa,\$e0,\$00
8308		
8309	::line4	b \$00,\$3e,\$65,\$a4,\$22,\$71,\$00,\$08
8310		b \$a5,\$05,\$c5,\$49,\$44,\$05,\$d0,\$3a
8311		b \$aa,\$2a,\$22,\$55,\$2a,\$88,\$a5,\$51
8312		b \$54,\$49,\$55,\$55,\$24,\$75,\$50,\$00
8313		b \$55,\$4a,\$c9,\$55,\$2c,\$aa,\$aa,\$aa
8314		b \$8d,\$2a,\$aa,\$4a,\$8a,\$aa,\$90,\$00
8315		
8316	::line5	b \$08,\$14,\$c9,\$9c,\$22,\$a8,\$10,\$50
8317		b \$45,\$d8,\$58,\$88,\$89,\$20,\$01,\$0a
8318		b \$b1,\$b3,\$a1,\$d5,\$4a,\$e8,\$a4,\$90
8319		b \$d5,\$88,\$c8,\$a5,\$27,\$52,\$30,\$00
8320		b \$36,\$26,\$68,\$d5,\$2a,\$aa,\$a4,\$c6
8321		b \$98,\$99,\$14,\$a6,\$e6,\$46,\$e0,\$00
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
8379 ::line3          b $08,$14,$c2,$1e,$22,$fb,$87,$04
8380                b $a4,$89,$59,$88,$8d,$28,$09,$35
8381                b $d9,$15,$dd,$6e,$96,$45,$56,$ac
8382                b $ac,$44,$aa,$a9,$29,$22,$10,$00
8383                b $55,$4a,$ad,$55,$2c,$aa,$aa,$aa
8384                b $99,$2a,$aa,$4a,$48,$42,$a0,$00
8385
8386 ::linbe4          b $00,$3e,$65,$a4,$22,$71,$00,$08
8387                b $a5,$05,$c5,$49,$44,$05,$d0,$3d
8388                b $55,$15,$11,$2a,$95,$44,$52,$a8
8389                b $aa,$24,$aa,$aa,$92,$21,$10,$00
8390                b $55,$4a,$c9,$55,$2c,$aa,$aa,$aa
8391                b $8d,$2a,$aa,$4a,$84,$44,$00,$00
8392
8393 ::line5            b $08,$14,$c9,$9c,$22,$a8,$10,$50
8394                b $45,$d8,$58,$88,$89,$20,$01,$21
8395                b $58,$d9,$d0,$ea,$a5,$74,$52,$48
8396                b $6a,$c4,$64,$52,$93,$a0,$90,$00
8397                b $36,$26,$68,$d5,$2a,$aa,$a4,$c6
8398                b $98,$99,$14,$a6,$e4,$44,$00,$00
8399
8400 ::line6            b $00,$00,$40,$00,$14,$00,$20,$00
8401                b $00,$00,$00,$00,$00,$40,$00,$18
8402                b $00,$00,$00,$00,$00,$00,$00,$00
8403                b $20,$00,$00,$00,$00,$30,$30,$78
8404                b $00,$00,$01,$80,$40,$00,$00,$82
8405                b $00,$00,$00,$04,$02,$48,$00,$00
8406 endif
8407
8408 ; Zeiger auf Speicher Laufwerk-Icons.
8409 .V72DD            w V34D6
8410                w V3526
8411                w V3576
8412                w V35C6
8413
8414 ; Speicher für Laufwerk-Icons.
8415 :V34D6            s DRVICON_SIZE
8416 :V3526            s DRVICON_SIZE
8417 :V3576            s DRVICON_SIZE
8418 :V35C6            s DRVICON_SIZE
8419
8420 ; DoIcons-Tabelle
8421 ; TD13: IconTab
8422 .MENU_ICONS = 3 ;Icons für Drucker/TrashBox/TaskMan.
8423 .DI3616          b $00
8424                w $0000
8425                b $00
8426
8427                w GR3692                ;Trash-Icon
8428                b (PANEL_X0+$10)/8,PANEL_Y0+$70
8429                b $03,$18
8430                w IR5A9C
8431
8432                w GR36DB                ;Drucker-Icon

```

```

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

```

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

```

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

```



```

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

```

```

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

```

```

8708             jsr    S38DB                ;Fensterdat. sichern.
8709
8710 ; Hinweis:
8711 ; Beim speichern der TD-Konfig. wird
8712 ; nur StashRAM verwendet.
8713 ;   => Verändert: r0 bis r3L
8714             jsr    Z09E5                ;TD-Konfig. => REU.
8715
8716             ldx    #(r2H - r0L)
8717 ::2          lda    :data,x
8718             sta    r0,x
8719             dex
8720             bpl    :2
8721
8722             lda    C0427                ;64K-Bank RAMDesk.
8723             sta    r3L
8724             jsr    StashRAM            ;TD.WIN => REU.
8725
8726             ldx    #0
8727 ::3          pla
8728             sta    r0,x
8729             inx
8730             cpx    #(r3L - r0L) +1
8731             bcc    :3
8732
8733 ::exit       rts
8734
8735 ::data       w BASE_TDWIN
8736             w DACC_TDWIN
8737             w SIZE_TDWIN
8738
8739 ; Speicherübersicht $7A00-$7DF2:
8740 ; (siehe auch src.mod#10!)
8741 ;--- TD.WIN
8742 ; $7A00 - $7AFF  Ordnerliste #1-#4
8743 ; $7B40 - $7B43  Fenster-Status
8744 ; $7B44 - $7B49  Größe Fenster #1
8745 ; $7B4A - $7B4F  Größe Fenster #2
8746 ; $7B50 - $7B55  Größe Fenster #3
8747 ; $7B56 - $7B5B  Größe Fenster #4
8748 ; $7B5C - $7B5F  Fensterstapel
8749 ; $7C00 - $7D3F  Fenstertitel #1-#4
8750 ;               4 x 80Byte = $0140.
8751 ; $7DA0 - $7DAB  Aktiver Ordner und
8752 ;               8Byte-Ordnerliste.
8753 ; $7DB0 - $7DB3  Fensterposition.
8754 ; $7DB4 - $7DBB  Lwfk./Part. Fenster #1-#4.
8755 ; $7DF1         Anzeige Block/Kb.
8756 ; $7DF2         Icon-/Text-Modus.
8757 ;---
8758 ; Laufwerk/Fensterdaten zwischenspeichern.
8759 .S38DB       ldy    #0
8760 ::1          ldx    #0
8761 ::2          lda    tdwin_data,y
8762             sta    r0,x

```

```

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

```

```

8928             pha
8929
8930             jsr    S09DD             ;Konfig in REU/Datei.
8931             jsr    Z37DC             ;TopDesk.win -> REU.
8932
8933             ldx    #VMod02
8934             pla
8935             jmp    Z378F             ;VLIR-Modul laden.
8936
8937 ;*** Menü-Routinen.
8938 :MR39DF             ldx    #VMod12
8939             lda    #VJmp1             ;Uhrzeit ändern.
8940             jmp    Z378F             ;VLIR-Modul laden.
8941
8942 ; Quicklink (kurzer Mausklick).
8943 :Z39AE             lda    #$ff             ;GEOS64.Editor.
8944             sta    V4191
8945
8946 ;*** Menü-Routinen.
8947 :Z39B6             ldx    #VMod09             ;Quicklink öffnen.
8948             b $2c
8949 :MR39E9             ;Disk/formatieren.
8950
8951 ;*** Shortcut.
8952 :KB39EC             ldx    #VMod12             ;Disk/formatieren.
8953             lda    #VJmp2
8954             jmp    Z378F             ;VLIR-Modul laden.
8955
8956 ;*** Menü-Routinen.
8957 :MR39F9             lda    #VJmp1             ;Disk/umbenennen.
8958             ldx    #VMod04
8959             jmp    Z377B             ;VLIR-Modul laden.
8960
8961 :Z3A00             lda    #VJmp0             ;Laufwerk-Icons.
8962             b $2c
8963 :Z3A03             lda    #VJmp1             ;TopDesk-Reset.
8964             b $2c
8965 :R3A06             lda    #VJmp2             ;Treiber laden.
8966             ldx    #VMod07
8967             jmp    Z378F             ;VLIR-Modul laden.
8968
8969 ; Laufwerke tauschen.
8970 :Z3A0D             stx    V3A1A             ;Quell-Laufwerk.
8971             sta    V3A1B             ;Ziel-Laufwerk.
8972
8973             ldx    #VMod07
8974             lda    #VJmp3             ;Laufwerke tauschen.
8975             jmp    Z378F             ;VLIR-Modul laden.
8976
8977 ; GEOS-Laufwerke für "Laufwerk tauschen".
8978 ; Wird nur in mod.#07 verwendet.
8979 .V3A1A             b $00 ;Quell-Laufwerk.
8980 .V3A1B             b $00 ;Ziel-Laufwerk.
8981
8982 ;*** Menü-Routinen.

```

```

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

```



```

9038
9039 ; CMD-/64Net-Partition wählen.
9040 ; Übergabe: Y = Laufwerk 8-11.
9041 ::part          tya
9042                jsr    S06AB          ;Laufwerk öffnen.
9043
9044                jsr    Z3A89          ;Partition wählen.
9045
9046                bit    V3A4B          ;64Net-Part. wählen?
9047                bpl    :done          ; => Nein, Ende...
9048
9049 ; 64Net ändert ggf. den Laufwerkstyp,
9050 ; daher Laufwerke aktualisieren.
9051                jsr    Z3A00          ;Laufwerk-Icons.
9052                jsr    Z1514          ;Panel/Icons zeigen.
9053                jsr    Z1DBE          ;Farbe Lfwk. setzen.
9054
9055                lda    C042A          ;Oberstes Fenster.
9056                sta    WIN042F        ;Aktives Fenster.
9057
9058                lda    #WN_ACTIVATE
9059                sta    WIN042E        ;Fenster-Befehl.
9060                jmp    Z4923          ;Window-Handler.
9061
9062 ::done          rts
9063
9064 ; CMD-Partition suchen.
9065 :Z3A83          stx    V3A94          ;Fenster-Nr.
9066
9067                lda    #VJmp2
9068                b $2c
9069
9070 ; CMD-Partition wählen.
9071 :Z3A89          lda    #VJmp0        ;Neue Partition.
9072                b $2c
9073
9074 ; Vorherige CMD-Partition für Fenster öffnen.
9075 :Z3A8C          lda    #VJmp1        ;Part./Fens. öffnen.
9076                ldx    #VMod11
9077                jmp    Z378F          ;VLIR-Modul laden.
9078
9079 ;*** Menü-Routinen:
9080 if ENABLE_GEOHELP = TRUE
9081 :MR3A95          ldx    #VMod12        ;Hilfesystem starten.
9082                lda    #VJmp3
9083                jmp    Z378F          ;VLIR-Modul laden.
9084 endif
9085
9086 ;*** Icon-Routine: TaskManager starten.
9087 :IR3A9F          lda    #$ff
9088                sta    V046F
9089                rts
9090
9091 ; Shortcut und DoIcons.
9092 ; TD13: OpenD8 bis OpenD11

```

```

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_DRVGHOST = FALSE
9129                ldx    #< GR3652 +1      ;Disk-Icon für DnD.
9130                ldy    #> GR3652 +1
9131 endif
9132 if ENABLE_DRVGHOST = 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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```
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
```



```

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

```

```

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

```

```

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

```

```

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.

```

```

10138          lda    :data +0,y
10139          sta    r0L
10140          lda    :data +1,y
10141          sta    r0H
10142
10143          ldy    #$00
10144          lda    #""
10145          sta    (r0),y          ;Modus aktivieren.
10146
10147          txa
10148          bne    :1          ;Icon-/Text-Modus?
10149                               ; => Text-Modus.
10150          jmp    S37B2          ;Anzeige: Icons.
10151 ::1          jmp    Z37AF          ;Anzeige: Text.
10152
10153 ; Tabelle mit Zeigern auf Anzeige-Modi.
10154 ::data          w MT3DB9
10155                  w MT3DC6
10156                  w MT3DD3
10157                  w MT3DE0
10158                  w MT3DED
10159                  w MT3DFA
10160
10161 ;--- Hauptmenü/Fenster:
10162 ; TD13: WindowMenü
10163 if LANG = LANG_DE
10164 :mnu5_of = $0078
10165 :mnu5_xw = $0090
10166 endif
10167 if LANG = LANG_EN
10168 :mnu5_of = $0070
10169 :mnu5_xw = $0080
10170 endif
10171 :mnu5_x0 = mnu0_x0 +mnu5_of
10172 :mnu5_x1 = mnu5_x0 +mnu5_xw -1
10173
10174 if ENABLE_ANIMATE = FALSE
10175 :mnu5_o1 = 0
10176 else
10177 :mnu5_o1 = 1
10178 endif
10179 if ENABLE_DIFFICOL = FALSE
10180 :mnu5_o2 = 0
10181 endif
10182 if ENABLE_DIFFICOL = TRUE
10183 :mnu5_o2 = 1
10184 endif
10185 :mnu5_me = 7 +mnu5_o1 +mnu5_o2
10186
10187 :mnu5_y0 = $10
10188 :mnu5_y1 = (mnu5_y0 +mnu5_me*14)!%00000111
10189
10190 :mnu5_t1 = mnu5_x1 -22
10191 :DS3E86          jsr    Z3BBE          ;Grafik in REU/Buf2.
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
10249 :MT3EDB          b "Alle Dateien wählen"
10250          b GOTOX
10251          w mnu5_t1
10252          b $80,BOLDON,"W",PLAINTEXT,0
10253
10254 :MT3EF6          b "Dateien im Fenster wählen"
10255          b GOTOX
10256          w mnu5_t1
10257          b $80,BOLDON,"X",PLAINTEXT,0
10258
10259 :MT3F19          b "Einstellungen speichern"
10260          b GOTOX
10261          w mnu5_t1
10262          b $80,BOLDON,"I",PLAINTEXT,0
10263
10264 :MT3F38          b " "
10265          b " Automatisch platzieren"
10266          b GOTOX
10267          w mnu5_t1
10268          b $80,BOLDON,"L",PLAINTEXT,0
10269 endif
10270 if LANG!ENABLE_DIFFICOL = LANG_DE!FALSE
10271 .MT3F58          b " "
10272          b " Verschiedene Farben",0
10273 endif
10274 if LANG!ENABLE_DIFFICOL = LANG_DE!TRUE
10275 .MT3F58          b " "
10276          b " Verschiedene Fensterfarben",0
10277 .MT3F59          b " "
10278          b " Verschiedene Iconfarben",0
10279 endif
10280 if LANG!ENABLE_ANIMATE = LANG_DE!TRUE
10281 :MT4048          b " "
10282          b " Fensteranimation",0
10283 endif
10284
10285 if LANG = LANG_EN
10286 :MT3EB3          b "Arrange windows"
10287          b GOTOX
10288          w mnu5_t1
10289          b $80,BOLDON,"S",PLAINTEXT,0
10290
10291 :MT3EC5          b "Close all windows"
10292          b GOTOX
10293          w mnu5_t1
10294          b $80,BOLDON,"C",PLAINTEXT,0
10295
10296 :MT3EDB          b "Select all files"
10297          b GOTOX
10298          w mnu5_t1
10299          b $80,BOLDON,"W",PLAINTEXT,0
10300
10301 :MT3EF6          b "Select page files"
10302          b GOTOX

```

```

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

```



```

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

```

```

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

```

```

10468 :mnu6_t1 = mnu6_x1 -20
10469 :DS3F9E          lda    #" "
10470                ldx    C0428                ;RAM-TopDesk?
10471                beq    :1a                ; => Nein, weiter...
10472                lda    #"*"
10473 ::1a             sta    MT3FFC                ;Menü anpassen.
10474
10475                lda    #" "
10476                ldx    C04B5                ;Kopieren-Dialog?
10477                beq    :1b                ; => Nein, weiter...
10478                lda    #"*"
10479 ::1b             sta    MT3FFD                ;Menü anpassen.
10480
10481                jsr    Z3BBE                ;Grafik in REU/Buf2.
10482
10483                b mnu6_y0,mnu6_y1
10484                w mnu6_x0,mnu6_x1
10485
10486 ::_x3FB1         b mnu6_nr ! VERTICAL ! MENU_VMODE
10487
10488 ::_x3FB2
10489
10490 if ENABLE_GEOHELP = TRUE
10491                w MT4031                ;Hilfe aufrufen
10492                b MENU_ACTION
10493                w MR3A95
10494 endif
10495
10496                w MT4017                ;Reset
10497                b MENU_ACTION
10498                w MR432B
10499
10500                w MT3FDA                ;Neuer Ordner
10501                b MENU_ACTION
10502                w MR39C8
10503
10504                w MT56FE                ;Neues Verzeichnis
10505                b MENU_ACTION
10506                w MR39B9
10507
10508                w MT3FEE                ;Uhrzeit ändern
10509                b MENU_ACTION
10510                w MR39DF
10511
10512                w MT4040                ;Farben ändern
10513                b MENU_ACTION
10514                w MR3A2F
10515
10516                w MT3FFD                ;Kopieren-Dialog
10517                b MENU_ACTION
10518                w MR4024
10519
10520                w MT3FFC                ;RAM-TopDesk
10521                b MENU_ACTION
10522                w MR3A3E

```

```

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

```

```

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

```

```

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:

```

```

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

```



```

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

```

```

10908 ; neu anzeigen.
10909             jsr    S5CF3             ;Fehler ausgeben.
10910 ::ok         jmp    Z15CB             ;Oberstes Fens. neu.
10911 else
10912             bcs    x42CE             ; => Fehler...
10913             jmp    Z15CB             ;Oberstes Fens. neu.
10914
10915 ; Hinweis:
10916 ; Wenn die Dateien nicht eingelesen
10917 ; wurden, dann ist der Dateispeicher
10918 ; leer -> Leeres Fenster.
10919 ; Fehler ausgeben, Fenster schließen.
10920 ;
10921 ; Benötigt Fenster-Nr. in WIN042F!
10922 :x42CE       jmp    Z5CEB             ;Fehler ausgeben und
10923                                     ;Fenster schließen.
10924 endif
10925
10926 ; DnD-Modus beenden.
10927 ; Sprite#1 abschalten, Fenstergrenzen zurücksetzen.
10928 ; TD13: EndGhost
10929 ;
10930 ; Hinweis:
10931 ; Darf Inhalt von Akku nicht verändern!
10932 ; Die Routine ":Handler" testet den
10933 ; Inhalt des Akku auf Fensterbefehle.
10934 :Z42DB       pha
10935
10936             jsr    Z1962             ;Fenstergrenzen.
10937
10938             ldx    #$00
10939             stx    V0423             ;DnD deaktivieren.
10940             inx
10941 ;             ldx    #1
10942             stx    r3L
10943
10944             jsr    DisablSprite
10945
10946             pla
10947             rts
10948
10949 ; Kopiere Name für Datei-Nr. in DIRBUF in Speicher.
10950 ; Übergabe: A = Dateinummer 0-254
10951 ; Rückgabe: r0 = Zeiger auf Eintrag in DIRBUF.
10952 ;             r1 = Zeiger auf Dateiname, Ende NULL-Byte.
10953 ; TD13: GetName2
10954 :Z42EC       sta    r0L             ;Datei-Nr.
10955             lda    #$00
10956             sta    r0H
10957
10958 ;             lda    #> 18             ;16Z. + Tr/Se für
10959             sta    r1H             ;Infoblock = 18.
10960             lda    #< 18
10961             sta    r1L
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 ;          lda    #$00
11020          sta    C0448,x          ;Fenster geschlossen.
11021          sta    C04F8,x          ;Lfwk. für Fenster.
11022          beq    :3
11023
11024 ::1      pla
11025          pha
11026          tax          ;Fenster-Nr.
11027          jsr    Z4331          ;Prefs./PadCol laden.
11028
11029          pla
11030          pha
11031          tax
11032          inc    V18AD          ;Fens.-Aufbau-Status.
11033          jsr    S141D          ;Fenster anzeigen.
11034
11035 ::2      pla          ;Fenster-Nr.
11036
11037 ::3      pla
11038          tax          ;Fenster-Eintrag.
11039
11040          dex          ;Alle Fenster?
11041          bpl    :loop          ; => Nein, weiter...
11042
11043          ldx    C042A          ;Oberstes Fenster.
11044          lda    C0448,x          ;Fenster geöffnet?
11045          bne    :exit          ; => Ja, weiter...
11046
11047          jmp    Z18AE          ;Fenster schließen.
11048
11049 ; Verschiedene Farben/Prefs laden.
11050 ; Übergabe: X = Fenster-Nr.
11051 :Z4330      lda    #" "
11052            b $2c
11053 :Z4331      lda    #"*"
11054
11055 if ENABLE_DIFFICOL = FALSE
11056          cmp    MT3F58          ;Versch. Farben?
11057          bne    :skip          ; => Ja, weiter...
11058 endif
11059 if ENABLE_DIFFICOL = TRUE
11060          cmp    MT3F59          ;Versch. Iconfarben?
11061          bne    :skip          ; => Ja, weiter...
11062 endif
11063
11064 ;          ldx    #$ff          ;Fenster-Nr.
11065          jsr    Z31FD          ;Prefs./PadCol laden.
11066
11067 ::skip      rts
11068
11069 ; Menüzeile vorbereiten.
11070 ; TD13: RedrawHead
11071 .S437D      jsr    Z13D3          ;Textgrenzen Desktop.
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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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.

```

```

11458 ; TD13: OpenNextNr
11459 :Z453C          sta    C051B,x          ;Aktive Ordner-Nr.
11460
11461              txa          ;Fenster-Nr.
11462              pha
11463              jsr    Z4576          ;Dat. neu einlesen.
11464              bcc    :1
11465
11466              pla
11467              jmp    S5CF3          ;Fehler ausgeben.
11468
11469 ::1            pla
11470              pha
11471              tax          ;Fenster-Nr.
11472              jsr    Z31FD          ;Prefs./PadCol laden.
11473
11474              lda    MT3F38          ;Autom. platzieren?
11475              cmp    #""
11476              bne    :2            ; => Nein, weiter...
11477
11478 ; Fenster neu und alle platzieren.
11479              pla
11480              tax          ;Fenster-Nr.
11481              jsr    Z1863          ;Fens.aktivieren und
11482              jmp    Z5655          ;Fens.platzieren.
11483
11484 ; Fenster neu ohne platzieren.
11485 ::2            pla
11486
11487 if ENABLE_ANIMATE = TRUE
11488 ; Hinweis:
11489 ; Animation nur beim Aufbau eines neuen
11490 ; Fenster anzeigen!
11491 ; Wenn ein Ordner geöffnet wird, die
11492 ; Partition gewechselt od. der Diskname
11493 ; geändert wird, dann keine Animation.
11494              bit    animFlag          ;Neues Fenster?
11495              bpl    :4            ; => Nein, weiter...
11496
11497              pha          ;Fenster-Nr.
11498              jsr    AnimateOpen        ;Animation zeigen.
11499 ;              lda    #FALSE
11500 ;              sta    animFlag
11501              inc    animFlag          ;Std.=Animation aus.
11502              pla
11503 endif
11504
11505 ::4            tax          ;Fenster-Nr.
11506              jsr    Z1863          ;Fens.aktivieren.
11507
11508 if ENABLE_DIFFICOL = FALSE
11509              lda    MT3F58          ;Versch. Farben?
11510              cmp    #""
11511              beq    :3            ; => Ja, weiter...
11512 endif

```

```

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:

```

```

11568 ; Im Textmodus wird der Bereich für
11569 ; VLIR-Module überschrieben!
11570 :Z4587          txa
11571                pha
11572                jsr   Z5CB3          ;Diskette öffnen.
11573                bcc   :1
11574                pla
11575                rts
11576
11577 ::1            pla
11578
11579 if FALSE
11580 ; Hinweis:
11581 ; Prefs/PadColorPref erst suchen, wenn
11582 ; die Dateien eingelesen wurden.
11583 ; Ausserdem nur im Icon-Modus suchen.
11584 ;
11585                pha
11586                tax          ;Neue Fenster-Nr.
11587                jsr   Z31FD          ;Prefs./PadCol laden.
11588                pla
11589 endif
11590
11591                tax
11592                jsr   S599A          ;Zeiger auf DIRBUF.
11593                pha
11594
11595                lda   r1H
11596                pha
11597                lda   r1L
11598                pha
11599
11600                lda   #< SIZE_DIRNAM ;Speicher für
11601                sta   r0L          ;Dateinamen löschen.
11602                lda   #> SIZE_DIRNAM
11603                sta   r0H
11604                jsr   ClearRam
11605
11606                pla
11607                sta   r3L
11608                pla
11609                sta   r3H
11610
11611                pla
11612                pha
11613
11614                tax
11615                lda   C051B,x      ;Aktive Ordner-Nr.
11616                sta   r10L          ;Dat./Ordner suchen.
11617                lda   C050B,x      ;Fensterposition.
11618                sta   r11H          ;Dateien überlesen.
11619
11620                lda   #MAXWINFILES ;Max. 16 Dateien.
11621                sta   r11L
11622                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
11676          beq    :11          ; Fehler?
11677          ; => Nein, weiter...

```



```

11678 ; Hinweis:
11679 ; Bei sortierter Text-Anzeige ist im
11680 ; Speicher Platz für ca.150 Dateien.
11681 ; Sind mehr Dateien auf Disk, dann
11682 ; können diese nicht sortiert werden
11683 ; und das Fenster wird geschlossen.
11684         cpx    #BFR_OVERFLOW
11685         bne     :4
11686
11687         jsr     S5CE1                ;Fehler ohne Refresh.
11688
11689         ldx     #NO_ERROR
11690 ::4      pla
11691         tay
11692         lda     #$00
11693         sta     C0448,y              ;Fenster geschlossen.
11694         txa
11695         sec
11696         rts
11697
11698 ; Fensterdaten aktualisieren.
11699 ::11      lda     r15L                ;BFR_OVERFLOW?
11700         beq     :12                ; => Nein, weiter...
11701
11702         pla
11703         pha
11704         tax
11705         lda     C050B,x              ;Offset = 0?
11706         bne     :12                ; => Nein, weiter...
11707
11708 ; Hinweis:
11709 ; Der Aufruf einer Dialogbox kann die
11710 ; Werte von r0 bis r11 verändern. Im
11711 ; folgenden wird aber r11L noch genutzt
11712 ; um die Anzahl Dateien zu ermitteln.
11713 ; Register zwischenspeichern.
11714 ;
11715 ; Korrektur: r11L/r14L sichern.
11716         lda     r11L                ;Max. Anzahl an
11717         pha
11718         lda     r14L                ;Dateien einlesen.
11719         pha
11720         ;Anzahl Dateien ab
11721         ;Offset auf Disk.
11721         ldx     #BFR_OVERFLOW
11722         jsr     S5CE1                ;Fehler ohne Refresh.
11723
11724         pla
11725         sta     r14L
11726         pla
11727         sta     r11L
11728
11729 ;         lda     #MAXFILES            ;Nicht notwendig,
11730 ;         sta     r14L                ;da immer < MAXFILES.
11731
11732 ::12      pla

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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?

```



```

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

```

```

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

```

```

12228          lda    #< inputDevName
12229          sta    r0L
12230          lda    #> inputDevName
12231          sta    r0H
12232          jsr    PutString          ;InputDrv. ausgeben.
12233
12234 ; Druckertreiber ausgeben.
12235          jsr    i_PutString
12236          w PANEL_X0 +2
12237          b PANEL_Y1 -2
12238          b NULL
12239          lda    #< PrntFilename
12240          sta    r0L
12241          lda    #> PrntFilename
12242          sta    r0H
12243          jsr    PutString          ;PrintDrv ausgeben.
12244
12245          jmp    Z13D3              ;Fenster = DeskTop.
12246
12247 ; y-Koordinaten für Laufwerksbuchstaben.
12248 if ENABLE_DRVMODE = FALSE
12249 :V48B2          b PANEL_Y0 +$08 +18
12250                b PANEL_Y0 +$20 +18
12251                b PANEL_Y0 +$38 +18
12252                b PANEL_Y0 +$50 +18
12253 endif
12254
12255 ; Laufwerksmodi.
12256 if ENABLE_DRVMODE = TRUE
12257 :V48B2          b PANEL_Y0 +$08 +5
12258                b PANEL_Y0 +$20 +5
12259                b PANEL_Y0 +$38 +5
12260                b PANEL_Y0 +$50 +5
12261 :V48B3          b "??417181N PC???"
12262 endif
12263
12264 ; Testen of Maustaste gehalten oder nicht.
12265 ; Rückgabe: C = 0: Langer Mausklick.
12266 ;          C = 1: Kurzer Mausklick.
12267 ; TD13: CheckKlick
12268 .S48B6          lda    #20
12269                sta    dblClickCount
12270 ::1            lda    dblClickCount
12271                beq    :2
12272                lda    mouseData
12273                bpl    :1
12274
12275 ; Kurzer Mausklick.
12276                sec
12277                rts
12278
12279 ; Langer Mausklick.
12280 ::2            clc
12281                rts
12282

```

```

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.

```

```

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

```

```
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
```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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:

```

```

12888          sbc    r6L          ;Geamthöhe - H.Balken
12889          sta    r5L          ;Resthöhe speichern.
12890          sta    V4C53         ;Komplette Höhe?
12891          beq     :4           ; => Ja, weiter...
12892
12893          ldx     WIN042F       ;Aktives Fenster.
12894          lda     C050B,x       ;Fensterposition.
12895          sta     r10L
12896
12897          ;          lda     #$00          ;Highbyte wird durch
12898          ;          sta     r5H          ;BBMult gelöscht.
12899          ;          sta     r10H         ; => Nicht verwendet.
12900
12901          ldx     #r5L          ;Oberen Versatz des
12902          ldy     #r10L         ;Balkens berechnen.
12903          jsr     BBMult
12904
12905          ldx     WIN042F       ;Aktives Fenster.
12906          lda     V0517,x       ;Anz.Dat. ab Offset.
12907          sta     r10L
12908
12909          ldx     #r5
12910          ldy     #r10
12911          jsr     Ddiv
12912
12913          lda     r2L
12914          clc
12915          adc     r5L
12916          sta     r2L          ;Obere Pos. Balken.
12917          adc     r6L
12918          sta     r2H          ;Untere Pos. Balken.
12919          ::4          rts
12920
12921          :V4C53          b $00    ;Resthöhe Füllstand ohne Balken.
12922
12923          ; Verschieberoutine für Fenster.
12924          :Z4D6D          dec     V16B8
12925
12926          ldx     WIN042F       ;Aktives Fenster.
12927          lda     V4C53         ;Resthöhe > 0?
12928          beq     Z4DB7        ; => Ja, zum Anfang.
12929
12930          ; Neue Fensterposition berechnen.
12931          lda     r2L
12932          pha
12933          jsr     Z2491         ;Koord. Pos.-Anzeige.
12934          pla
12935          sec
12936          sbc     r2L
12937          sta     r0L
12938
12939          ldx     WIN042F       ;Aktives Fenster.
12940          lda     V0517,x       ;Anz.Dat. ab Offset.
12941          sta     r10L
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 ;
12990 ; SourceCode: inc.mod#00c.s
12991 ;
12992
12993 ;*** Icon-Routine: Drucker wechseln/Dateien drucken.
12994 ; - Dateien per DnD drucken.
12995 ; - Kurzer Klick: Drucker wählen.
12996 ; - Langer Klick: Eingabegerät wählen.
12997 :IR4E1C          lda    V0423          ;DnD aktiv?
12998                  bne    MR4E31          ; => Ja, weiter...
12999
13000                  jsr    S48B6          ;Langer Mausklick?
13001                  bcc    :1              ; => Ja, weiter...
13002
13003 ; Der AKKU enthält ab hier einen Index
13004 ; auf die Dateityp-Tabelle in mod.#09.
13005                  lda    #OFF_OPEN_PRNT ;Drucker wechseln.
13006                  b $2c
13007 ::1              lda    #OFF_OPEN_INPT ;Input wechseln.
13008                  jsr    Z40C1          ;Datei wählen/öffnen.
13009                  jmp    S3B9B
13010
13011 ; Menü: Datei -> drucken
13012 ; TD13: DateiDrucken
13013 :MR4E31          lda    #$ff          ;Datei/drucken.
13014                  b $2c
13015
13016 ; Menü: Datei -> öffnen
13017 ; TD13: Dateiöffnen
13018 :MR4E34          lda    #$00          ;Datei/öffnen.
13019                  sta    V4C54          ;PrintFlag setzen.
13020
13021                  jsr    Z4E67          ;Test MultiFile.
13022                  bcs    :exit          ; => Abbruch...
13023
13024 ; Nur eine Datei markiert.
13025 ::1              jsr    S5842          ;mark.Dat. anzeigen.
13026                  jsr    S5262          ;Suche mark.Datei.
13027                  jsr    S46EB          ;Name mark.Datei.
13028                  bcs    :exit          ; => Nicht gefunden.
13029
13030                  lda    V4172          ;Quicklink-Modus?
13031                  bne    :2              ; => Ja, weiter...
13032
13033                  jmp    Z5017          ;Datei öffnen.
13034
13035 ::2              ldx    #NO_ERROR
13036                  clc
13037
13038 ; Bei Abbruch ist hier das C-Flag gesetzt!
13039 :exit            rts
13040
13041 ; Test auf Multifile.
13042 :Z4E67           jsr    GotoFirstMenu

```

```

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

```

```

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

```

```

13208          ora    V04F4,x          ;Dir.S für Fenster.
13209          beq    exitCLWin         ; => Kein SubDir...
13210
13211 ;          ldx    C042A              ;Oberstes Fenster.
13212          jsr    S09B7              ;Tr/Se SubDir = $00.
13213
13214 ;          lda    #$00
13215          sta    C050B,x            ;Fensterposition.
13216
13217          jsr    S524C              ;Dat.-Ausw. aufheben.
13218          jsr    Z42CD              ;Oberstes Fens. neu.
13219
13220 :exitCLWin    rts
13221
13222 ;*** Shortcut: C= + <-
13223 :KB4EDA      lda    #WN_CLOSE
13224          sta    WIN042E            ;Fenster-Befehl.
13225          ldx    C042A              ;Oberstes Fenster.
13226          stx    WIN042F            ;Aktives Fenster.
13227
13228          lda    C0448,x            ;Fenster geöffnet?
13229          beq    exitCLWin         ; => Nein, Ende...
13230
13231 ; Fenster schließen.
13232 ; Übergabe: X = Fenster-Nr.
13233 :Z4EEF      jsr    S524C            ;Dat.-Ausw. aufheben.
13234          jsr    S5255            ;Mark.Dat=0/Anzeige.
13235
13236          jsr    S48B6              ;Langer Mausklick?
13237
13238          lda    #$00              ;Vorgabe: Kein
13239          sta    V04AE              ;Klick auf Gadget.
13240          sta    C050B,x            ;Fensterposition.
13241
13242 ; Langer Klick=Hauptverzeichnis öffnen?
13243          bcc    :root              ; => Ja, weiter...
13244
13245          ldx    WIN042F            ;Aktives Fenster.
13246          lda    C051B,x            ;Aktive Ordner-Nr.
13247          bne    :exit_sub         ; => Ordner...
13248
13249 ; Kein Ordner geöffnet.
13250          lda    C04F8,x            ;Lfwk. für Fenster.
13251          jsr    SetDevice
13252
13253 ; Hinweis:
13254 ; OpenDisk hier nicht aufrufen, da
13255 ; bei NativeMode ":OpenRootDir" oder
13256 ; ":OpenSubDir" ausgeführt wird.
13257 ; Nur bei nicht-NativeMode ausführen.
13258 ;          jsr    OpenDisk
13259
13260          jsr    Z0680              ;Lfwk. = Native?
13261          bne    :std               ; => Nein, weiter...
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
13319 ; Auswertung einer Datei-Auswahl.
13320 ; Übergabe: X = Nr. der Datei in der Darstellung (0-15)
13321 ; TD13: File_Selected
13322 :Z4F1D          lda    dblClickCount      ;Doppelklick-Status
13323                sta    a9L                ;einlesen.
13324
13325                txa
13326                sta    a1L                ;Datei-Nr. 0-15.
13327
13328                ldx    C042A              ;Oberstes Fenster.
13329 ;                lda    a1L                ;Datei-Nr. 0-15.
13330                cmp    V050F,x            ;Anz.Dat. im Fenster.
13331                bcs    Z4F2E              ; => Ungültig.
13332
13333                lda    V0423              ;DnD aktiv?
13334                bne    :dnd              ; => Ja, weiter...
13335
13336                ldx    a1L                ;Datei-Nr. 0-15.
13337                jsr    S51CE              ;C=-Taste gedrückt?
13338                bcs    :1                ; => Ja, weiter...
13339
13340 ; Single-Klick.
13341                lda    a9L                ;Doppelklick?
13342                bne    :2                ; => Ja, weiter...
13343
13344                txa
13345                ldx    C042A              ;Oberstes Fenster.
13346                jsr    Z57D8              ;Datei markiert?
13347                bcs    :2                ; => Ja...
13348
13349                jmp    Z4F9D              ;Pause Doppelklick.
13350
13351 ; C= + Single-Klick.
13352 ::1            jmp    Z5201              ;Mehrfach-Auswahl.
13353
13354 ; Auf Doppelklick testen.
13355 ::2            jsr    S5842              ;mark.Dat. anzeigen.
13356                jsr    S524C              ;Dat.-Ausw. aufheben.
13357                jsr    Z51E5              ;Dateipos. ermitteln.
13358
13359                pha                      ;Datei-Nr. speichern.
13360
13361                jsr    Z5216              ;mark.Datei in Tab.
13362                jsr    S5842              ;mark.Dat. anzeigen.
13363
13364                lda    a9L                ;Doppelklick aktiv?
13365                bne    :3                ; => Ja, weiter...
13366
13367                pla                      ;Neue Datei für
13368                sta    V4F89              ;Doppelklick setzen.
13369
13370                lda    #20                ;Doppelklick starten.
13371                sta    dblClickCount
13372 ::exit        rts

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

13758 :Z525E          bit   pressFlag          ;Taste gedrückt?
13759                bpl   S5262                ; => Nein, weiter...
13760
13761                jsr   S524C                ;Dat.-Ausw. aufheben.
13762
13763 ; Erste markierte Datei holen.
13764 ; Übergabe: A = Datei-Nr,
13765 ; Rückgabe: X = Aktives Fenster.
13766 ;           A = 0-254: mark.Datei.
13767 ;           $FF : Ende.
13768 ;           C = CLC : mark.Datei gefunden.
13769 ;           SEC : Ende.
13770 ; TD13: GetMark
13771 .S5262          lda   V634F                ;Tab. mark.Dateien.
13772                cmp   #$ff                ;Datei markiert?
13773                beq   :1                ; => Nein, Ende...
13774
13775                pha
13776                tax
13777                jsr   Z5223                ;mark.Dat. abwählen.
13778
13779                ldx   C042A                ;Oberstes Fenster.
13780                pla
13781
13782                clc                        ;Datei gefunden.
13783                rts
13784
13785 ::1             sec                        ;Keine Datei.
13786                rts
13787
13788 ; Neues Verzeichnis laden.
13789 ; Rückgabe : X = Fehler.
13790 ; TD13: NewDirLoad
13791 :Z5276          ldx   C042A                ;Oberstes Fenster.
13792                lda   #$00
13793                sta   C050B,x                ;Fensterposition.
13794
13795                jsr   Z4576                ;Dat. neu einlesen.
13796                bcs   :1
13797
13798                jsr   Z15CB                ;Oberstes Fens. neu.
13799
13800                ldx   #NO_ERROR
13801 ::1             rts
13802
13803 ; Dialogbox: Dateien überschreiben?
13804 :D5289          b   %10000001
13805
13806                b   DBTXTSTR,$08,$0e
13807                w   :1
13808                b   DBTXTSTR,$08,$18
13809                w   :2
13810
13811                b   YES      , $11,$08
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
13869          b OK          , $11, $28
13870
13871          b DBTXTSTR, $08, $52
13872          w T04F5
13873
13874          b CANCEL      , $11, $48
13875
13876          b NULL
13877
13878 if LANG = LANG_DE
13879 ::1          b "Markierte Datei(en) auf", 0
13880 ::2          b "die Ziel-Disk "
13881          b BOLDON, "kopieren"
13882          b PLAINTEXT, " ?", 0
13883
13884 ::3          b "Datei(en) "
13885          b BOLDON, "verschieben"
13886          b PLAINTEXT, " ?", 0
13887 ::4          b "(auf Original-Disk löschen)", 0
13888 endif
13889 if LANG = LANG_EN
13890 ::1          b BOLDON, "Copy"
13891          b PLAINTEXT, " selected file(s)", 0
13892 ::2          b "to destination disk ?", 0
13893
13894 ::3          b BOLDON, "Move"
13895          b PLAINTEXT, " selected file(s) ?", 0
13896 ::4          b "(delete source files)", 0
13897 endif
13898
13899 ; Dialogbox: Datei mit Schreibschutz löschen?
13900 :D5BF2          b %10000001
13901
13902          b DBTXTSTR, $12, $38
13903          w V0470
13904          b DBTXTSTR, $08, $38
13905          w T29D3
13906
13907          b DBTXTSTR, $08, $10
13908          w T29CF
13909
13910          b DBTXTSTR, $08, $20
13911          w :1
13912          b DBTXTSTR, $08, $2a
13913          w T04F3
13914
13915          b YES          , $01, $48
13916          b NO           , $11, $48
13917
13918          b NULL
13919
13920 if LANG = LANG_DE
13921 ::1          b PLAINTEXT
13922          b "Wirklich löschen?"

```

```

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

```

```

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

```

```

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.

```

```

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

```

```

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

```



```

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

```

```

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

```

```

14418 ; $00/$ff in Byte #1/2, da ein Ordner
14419 ; keine Daten beinhaltet.
14420 if TRUE
14421         jsr    Z10F0                ;Ordner/Verz./Datei?
14422         txa
14423         beq    :file                ; => Datei.
14424 else
14425         ldx    dirEntryBuf +1        ;Daten vorhanden?
14426         beq    :no_file              ; => Kein Ordner.
14427
14428         lda    dirEntryBuf           ;Dateityp auswerten.
14429         and    #FTYPE_MODES
14430         cmp    #FTYPE_DIR
14431         bne    :file                ; => Verzeichnis.
14432
14433 ;         lda    dirEntryBuf +$16
14434 ;         cmp    #TDFOLDER           ;TopDesk-Ordner?
14435 ;         bne    :file                ; => Nein, weiter...
14436 ;
14437 ;         lda    dirEntryBuf +21
14438 ;         cmp    #VLIR               ;VLIR-Datei?
14439 ;         beq    :file                ; => Ja, weiter...
14440 endif
14441
14442 ; Ordner/Verzeichnis kopieren/verschieben.
14443 ::no_file      stx    V10CF          ;Dateityp speichern.
14444               bpl    :no_dir        ; => Ordner...
14445
14446               ldx    WIN042F        ;Ziel-Fenster.
14447               ldy    C04F8,x        ;Lfwk. für Fenster,
14448
14449               ldx    #STRUCT_MISMAT
14450               lda    RealDrvMode -8,y
14451               and    #SET_MODE_SUBDIR
14452               beq    :copyErr        ; => Kein Native...
14453
14454 ::no_dir       lda    #0             ;Datei auf Ziel
14455               sta    r9L             ;suchen/löschen.
14456
14457               ldx    #r11
14458               jsr    Z0F30           ;Ziel-Datei suchen.
14459               cpx    #DIR_NOT_EMPTY ;Verz. n. gelöscht.
14460               beq    :cont
14461               cpx    #FILE_EXISTS    ;0./Verz. n. ersetzt.
14462               beq    :4
14463               txa
14464               bne    :copyErr        ; Fehler?
14465                                       ; => Ja, Abbruch...
14466 ::4            bit    V10CF          ;Ordner/Verzeichnis?
14467               bpl    :5              ; => Ordner.
14468
14469 ; Hinweis:
14470 ; Verzeichnis kopieren nicht aufrufen
14471 ; wenn Ziel nicht gelöscht wurde.
14472               txa                    ;File exists?

```

```

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.

```

```

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

```

```

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.

```

```

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

```

```

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

```



```

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

```

```

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

```

```

14968      jsr    i_Rectangle
14969      b SLCT_Y0 +1,SLCT_Y1 -1
14970      w SLCT_X0 +1,SLCT_X1 -1
14971
14972      lda    V04B7                ;Anzahl mark.Dateien.
14973      sta    r0L
14974      lda    #$00                ;Highbyte löschen.
14975      sta    r0H
14976
14977      lda    #< SLCT_X0 +2
14978      sta    r11L
14979      lda    #> SLCT_X0 +2
14980      sta    r11H
14981
14982      lda    #SLCT_Y1 -6
14983      sta    r1H
14984
14985      lda    #SET_LEFTJUST ! SET_SUPRESS
14986      jsr    PutDecimal
14987
14988      ldy    #0
14989 ::2      pla
14990      sta    windowTop,y
14991      iny
14992      cpy    #6
14993      bne    :2
14994
14995      pla
14996      tax                ;Evtl. Fehlerstatus.
14997
14998      rts
14999
15000 ; Markierte Dateien im Fenster anzeigen.
15001 ; Übergabe: WIN042F = Fenster-Nr.
15002 ; TD13: DispMarking
15003 .S5842      jsr    Z4214                ;X/Y für erstes Icon.
15004
15005      lda    #$00
15006 ::1      pha
15007
15008      ldx    WIN042F                ;Aktives Fenster.
15009      jsr    Z57D8                ;Datei markiert?
15010      bcs    :2                ; => Nein...
15011
15012      ldx    WIN042F                ;Aktives Fenster.
15013      pla
15014      cmp    V050F,x                ;Anz.Dat. im Fenster.
15015      bcs    :3
15016
15017      pha
15018      jsr    A2BB2                ;Rechteck f.Eintrag.
15019      bcs    :2                ; => Kein Eintrag...
15020
15021      jsr    InvertRectangle
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
15079             pha
15080             jsr   GotoFirstMenu      ;Hauptmenü.
15081             pla           ;Seite/Alle wählen.
15082
15083             inc    V04AE              ;Klick zurücksetzen.
15084
15085 ::1           b $2c
15086
15087 ; Erste Zeile wählen.
15088 ; TD13: n.v.
15089 :KB58A3      lda    #2              ;Erste Zeile wählen.
15090             sta    V5909            ;Markierungsmodus.
15091
15092             jsr    A2BB5            ;Anz.Ze/Sp und Icons.
15093
15094             ldx    C042A            ;Oberstes Fenster.
15095             lda    C0448,x          ;Fenster geöffnet?
15096             bne    :1              ; => Ja, weiter...
15097             rts
15098
15099 ::1           stx    WIN042F         ;Aktives Fenster.
15100             jsr    Z4214            ;X/Y für erstes Icon.
15101             jsr    S5842            ;mark.Dat. anzeigen.
15102
15103             lda    #0              ;Dateizähler.
15104 ::loop        pha
15105
15106             ldx    C042A            ;Oberstes Fenster.
15107
15108             ldy    V5909            ;Markierungsmodus.
15109             beq    :2              ; => Seite wählen.
15110             cpy    #2
15111             beq    :3              ; => Abwählen.
15112
15113 ; Alle Dateien wählen.
15114             cmp    V0513,x          ;Anz.Dat. auf Disk.
15115             bcs    :next
15116
15117             jsr    Z57F4            ;Datei markiert?
15118             bcc    :next           ; => Ja...
15119             bcs    :5              ; => Nein...
15120
15121 ; Dateien auf Seite wählen.
15122 ::2           cmp    V050F,x         ;Anz.Dat. im Fenster.
15123             bcs    :next           ; => Überspringen.
15124             bcc    :4              ; => Markieren.
15125
15126 ; Erste Zeile wählen.
15127 ::3           cmp    V4CC7,x         ;Anz.Dat. in Zeile.
15128             bcs    :next           ; => Überspringen.
15129
15130 ::4           jsr    Z57D8            ;Datei markiert?
15131             bcc    :next           ; => Ja, weiter...
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
15189         lda    r0L
15190         ora    r0H
15191         bne    :1
15192
15193         ldy    #$ff
15194 ::2        iny
15195         pla
15196         sta    V5973,y
15197         bne    :2
15198
15199         pla
15200         bpl    :3
15201
15202         lda    #< V5973
15203         sta    r0L
15204         lda    #> V5973
15205         sta    r0H
15206         bne    :5
15207
15208 ::3        pha
15209
15210         lda    #< V5973
15211         sta    r0L
15212         lda    #> V5973
15213         sta    r0H
15214         jsr    S244C                ;Hole Stringlänge.
15215
15216         pla
15217         and    #%00111111
15218         clc
15219         adc    r11L
15220         sta    r11L
15221         bcc    :4
15222         inc    r11H
15223
15224 ::4        lda    r11L
15225         sec
15226         sbc    r1L
15227         sta    r11L
15228         lda    r11H
15229         sbc    r1H
15230         sta    r11H
15231
15232 ::5        pla
15233         sta    r1H
15234
15235         jmp    S14C8                ;String ausgeben.
15236
15237 :V5973      s 6    ;Zwischensp. Zahl im Textformat.
15238
15239 ; Ermittlung des Fenstertitels.
15240 ; Übergabe: A = Fenster-Nr.
15241 ; Rückgabe: r1 = Zeiger auf String
15242 ;           Y = Zeiger auf letztes Zeichen.

```

```

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

```

```

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.

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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.

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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.

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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
17019      adc  #TDFOLDER_POS0 -1 ;Sonderbehandlung.
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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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?

```



```

17387             bne      :restart          ; => Alle Dateien.
17388
17389             ldx      v7140              ;Fehlercode.
17390             jsr      S5CF3              ;Fehler ausgeben.
17391
17392 ; Zurück zur "Datei retten"-Dialogbox.
17393 ::reset          jsr      S54F0          ;Status löschen.
17394
17395 ; Hinweis:
17396 ; Hier Anzahl an Dateien zurücksetzen,
17397 ; da durch Status von "Datei retten"
17398 ; die Anzahl verändert wurde.
17399             jsr      S524C              ;Dat.-Ausw. aufheben.
17400
17401 ; "Dateien retten" neu starten.
17402             lda      #> S3997 -1
17403             pha
17404             lda      #< S3997 -1
17405             pha
17406 ;
17407 ; Hinweis:
17408 ; Im Textmodus wird der Bereich für
17409 ; VLIR-Module überschrieben!
17410             jmp      S4296              ;Update akt.Fenster.
17411
17412 ; Dateityp festlegen.
17413 ::undel          ldx      %%10000000 ! PRG
17414             ldy      #$15
17415             lda      (r5),y            ;VLIR-Datei?
17416             beq      :21              ; => Nein, weiter...
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

```

```

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
17498 ; Mausabfrage für Dateifenster.
17499 :r707F          bit    mouseData
17500                bmi    :exit
17501
17502                jsr     z72A5                ;Größe Dateifenster.
17503
17504                jsr     IsMseInRegion        ;Klick Dateifenster?
17505                beq     z70E0                ; => Nein, weiter...
17506
17507                jsr     z7292                ;Datei invertieren.
17508
17509                jsr     z72A5                ;Größe Dateifenster.
17510
17511                lda     mouseYPos            ;Zeiger auf Eintrag
17512                sec                                ;in Liste berechnen.
17513                sbc     r2L
17514                sta     r0L
17515                lda     #$00
17516                sta     r0H
17517                sta     r1H
17518
17519                lda     #ENTRY_YH            ;12 Pixel Höhe eines
17520                sta     r1L                ;Dateieintrags.
17521
17522                ldx     #r0
17523                ldy     #r1
17524                jsr     Ddiv
17525
17526                lda     r0L                ;Datei angeklickt?
17527                clc
17528                adc     v72EA                ;Erste Datei / Liste.
17529                cmp     v72EB                ;Anzahl Dateien.
17530                bcc     :1                ; => Ja, weiter...
17531
17532                ldx     v72EB                ;Anzahl Dateien -1
17533                dex                                ; = Letzte Datei.
17534                txa
17535                sta     v72E9                ;Markierte Datei.
17536
17537                jsr     z7292                ;Datei invertieren.
17538
17539                lda     #10
17540                sta     dblClickCount
17541
17542                lda     #%100000000
17543                sta     mouseData
17544 ::wait          bit     mouseData
17545                bpl     :dblclk
17546                lda     dblClickCount
17547                bne     :wait
17548 ::exit          rts
17549
17550 ::dblclk        lda     v72EB                ;Anzahl Dateien.
17551                beq     :exit                ; => Keine, Ende...

```

```

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.

```

```

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

```

```

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
17718             lda    mouseData           ;Taste gedrückt?
17719             bmi    :5                   ; => Nein, Ende...
17720
17721             inc    v71F2                 ;Dauerfunktion?
17722             jmp    z7128                 ;Nav. wiederholen.
17723
17724 ::5           jmp    z7292                 ;Datei invertieren.
17725
17726 ; Dateiliste ausgeben.
17727 :z71F3        lda    v72EB                 ;Dateien gefunden?
17728             bne    :1                   ; => Ja, weiter...
17729
17730             lda    #< x74C1              ;Hinweis ausgeben und
17731             sta    r0L                   ;Ausgabe beenden.
17732             lda    #> x74C1
17733             sta    r0H
17734             jmp    PutString
17735
17736 ::1           lda    rightMargin +1
17737             pha
17738             lda    rightMargin +0
17739             pha
17740
17741             jsr    z72A5                 ;Größe Dateifenster.
17742
17743             lda    #$00                 ;Füllmuster.
17744             jsr    S140C                 ;Rechteck zeichnen.
17745
17746             lda    #$00
17747             sta    r15L
17748
17749             jsr    z72BE                 ;Koord. mark. Datei.
17750
17751             lda    r4L
17752             sta    rightMargin +0
17753             lda    r4H
17754             sta    rightMargin +1
17755
17756             ldx    #FB0X_FILES
17757             lda    #FB0X_FILES
17758             sec
17759             sbc    v72EB                 ;Anzahl Dateien.
17760             bcc    :2                   ; => Mehr Dateien.
17761             ldx    v72EB                 ;Anzahl Dateien.
17762 ::2           stx    r15H
17763
17764             lda    v72EA                 ;Erste Datei / Liste.
17765             sta    r0L
17766
17767             lda    #17                   ;Größe Dateieintrag.
17768             sta    r1L
17769
17770             ldy    #r0
17771             ldx    #r1

```

```

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?

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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.

```



```

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

```

```

18377      sta    curDirHead +0,y
18378      lda    #$ff
18379      sta    curDirHead +1,y
18380      sta    curDirHead +2,y
18381
18382      lda    r1H
18383      sec
18384      sbc    #$10
18385      tax
18386      lda    v7AA9 -1,x
18387      sta    curDirHead +3,y
18388      iny
18389      iny
18390      iny
18391      iny
18392      inc    r1L
18393      cpy    #$90
18394      bcc    :21
18395
18396      dec    curDirHead +$48
18397
18398      lda    #$fe
18399      sta    curDirHead +$49
18400
18401      pla
18402      cmp    #$02
18403      bne    :23
18404
18405      lda    curDirHead +$03
18406      beq    :23                      ; => 1541-BAM...
18407
18408      jsr    i_FillRam
18409      w 105
18410      w dir2Head
18411      b $ff
18412
18413      jsr    i_FillRam
18414      w 256 -105
18415      w dir2Head +105
18416      b $00
18417
18418      lda    #$24
18419      sta    r1L
18420
18421      lda    #$02
18422      sta    r0H
18423
18424      ldy    #$dd
18425  ::22      sty    r0L
18426      jsr    z7A31                      ;max.Sek./Track.
18427
18428      ldy    r0L
18429      lda    r1H
18430      sta    curDirHead,y
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
18488 ;          jsr  EnterTurbo          ;TurboDOS ein.
18489 ;          txa
18490 ;          bne  :err                ; => Fehler, Abbruch.
18491
18492          jsr  InitForIO              ;I/O aktivieren.
18493
18494 ; Inhalt Native-BAM-Block $01/$xx löschen.
18495          ldy  #$00
18496          lda  #$ff
18497 ::33      sta  diskBlkBuf,y
18498          iny
18499          bne  :33
18500
18501          lda  #$01                  ;Zweiter BAM-Block.
18502          sta  r1L
18503          lda  #$03
18504          sta  r1H
18505
18506          jsr  z74AF                  ;r4 = ":diskBlkBuf".
18507 ::34      jsr  WriteBlock            ;Block speichern.
18508          txa
18509          beq  :35                    ; => OK, weiter...
18510
18511 ; Fehler, Abbruch.
18512          pha
18513          jsr  DoneWithIO            ;I/O abschalten.
18514          pla
18515          tax                          ;Fehlerstatus.
18516 ::err     rts
18517
18518 ::35      inc  r1H
18519          lda  r1H
18520          cmp  #$22                  ;BAM gelöscht?
18521          bcc  :34                    ; => Nein, weiter...
18522
18523          jsr  DoneWithIO            ;I/O abschalten.
18524
18525          jsr  GetDirHead             ;BAM einlesen.
18526          txa
18527          bne  :err                  ; => Fehler, Abbruch.
18528
18529          lda  #$3f
18530          sta  dir2Head +$24
18531
18532          jmp  z760C                  ;Verzeichnis belegen.
18533
18534 ; Unterverzeichnisse überprüfen.
18535 ;
18536 ; Code stammt aus GeoDOS64.
18537 :z77BD    jsr  z794E                  ;Anfang Verzeichnis.
18538
18539          lda  #< fileHeader
18540          sta  r4L
18541          lda  #> fileHeader

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

19341      b $64
19342      b PLAINTEXT
19343      b "or press the space bar to enter the",0
19344      jsr  i_PutString
19345      w $0050
19346      b $72
19347      b PLAINTEXT
19348      b "BASIC mode of your computer.",0
19349  endif
19350
19351  if ENABLE_BASICINFO = TRUE
19352      lda  #< r6FB6
19353      sta  appMain +0
19354      lda  #> r6FB6
19355      sta  appMain +1
19356      jmp  ClearMouseMode
19357
19358  ; Tastaturabfrage für RESET-Modus.
19359  :r6FB6      lda  keyData
19360              cmp  #" "                ;Leertaste gedrückt?
19361              beq  :1                ; => Ja, Ende...
19362              rts
19363
19364  ::1          lda  curDrive
19365              jsr  SetDevice
19366              jsr  PurgeTurbo
19367
19368              sei
19369              lda  CPU_DATA
19370              and  #%11000000
19371              ora  #KRNL_BAS_IO_IN
19372              sta  CPU_DATA
19373
19374              lda  $e395                ;"inivec"-Routine
19375              sta  R801C +1            ;zum initialisieren
19376              lda  $e396                ;von $0300-$030B.
19377              sta  R801C +2
19378
19379              ldy  #0
19380  ::2          lda  R8000,y
19381              sta  $8000,y
19382              iny
19383              bne  :2
19384
19385              jmp  ($fffc)
19386
19387  ; RESET-Routine (ab $8000)
19388  :CBMRESET
19389  ; ---
19390  :R8000      w  $8009
19391              w  $8009
19392              b  $c3,$c2,$cd,$38,$30
19393
19394  :R8009      sei
19395              ldx  #$ff

```

```

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.

```

```

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

```

```

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

```



```

19561 ; markierte Datei behandelt.
19562 ; => Hier wird dann eine andere Datei
19563 ; dupliziert als zuvor markiert.
19564 ;
19565 ; Fix:Nur einzelne Dateien duplizieren.
19566 ; (Wird im Hauptmodul getestet)
19567 :N71D1          lda    #$00
19568                sta    V04A9                ;Modus: Kopieren.
19569
19570 ; Hinweis:
19571 ; SubDir-Info nur nach dem ersten
19572 ; Programmstart löschen.
19573 ;                sta    V5596                ;SubDir-Info löschen.
19574
19575                lda    #$ff                ;Modus: Duplizieren.
19576                b      $2c
19577
19578 ; 1: Datei umbenennen.
19579 :N71E0          lda    #$00                ;Modus: Umbenennen.
19580
19581                ldx    V04B7                ;Anzahl mark.Dateien.
19582                bne    :1                ; => OK, weiter...
19583 ::exit          rts
19584
19585 ::1             sta    V71CF                ;Umbenenn./Dupliz.
19586
19587                ldx    C042A                ;Oberstes Fenster.
19588                lda    V0513,x            ;Anz.Dat. auf Disk.
19589                sta    V7213                ;Aktuelle Dateizahl.
19590
19591                lda    C04F8,x            ;Lfwk. für Fenster.
19592                sta    V0A72                ;Lfwk. Source-Disk.
19593                sta    V0A73                ;Lfwk. Target-Disk.
19594                stx    WIN042F            ;Aktives Fenster.
19595
19596                jsr    S5842                ;mark.Dat. anzeigen.
19597
19598                jsr    S5C89                ;Hole Disk/Fenster.
19599                tax
19600                beq    x7214
19601
19602                jsr    S5245                ;Dateien invertieren.
19603                cpx    #CANCEL_ERR
19604                beq    :exit
19605
19606 ::err           jmp    S5CF3                ;Fehler ausgeben.
19607
19608 :x7214          lda    #< V634F            ;Zeiger Tabelle mit
19609                sta    r2L                ;markierten Dateien.
19610                lda    #> V634F
19611                sta    r2H
19612
19613                lda    #$00                ;$00=Noch kein Name.
19614                sta    V71D0
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
19672 ; *TDFOLDERFIX*
19673 ;
19674 ; Der VLIR-Test funktioniert hier nicht
19675 ; zuverlässig. Ein Ordner hat als Tr/Se
19676 ; $00/$ff in Byte #1/2, da ein Ordner
19677 ; keine Daten beinhaltet.
19678             lda    dirEntryBuf +1           ;Daten vorhanden?
19679             bne    z7295                     ; => Kein Ordner.
19680 ;
19681             lda    dirEntryBuf +21
19682             bne    z7295                     ; => VLIR, weiter...
19683 ; Ordner kann nicht dupliziert werden.
19684             ldx    #STRUCT_MISMATCH         ;Fehler für einen
19685             bne    z72DE                     ;TD-Ordner anzeigen.
19686
19687 ; CMD-Verzeichnis duplizieren.
19688 ;
19689 ; TODO:
19690 ; Wenn CMD-Verzeichnisse nicht kopiert
19691 ; werden können sondern nur ein leeres
19692 ; Verzeichnis erstellt wird, dann kann
19693 ; Duplizieren direkt beendet werden:
19694 ; Es ist einfacher Verzeichnisse nicht
19695 ; zu kopieren und den Anwender neue
19696 ; Verzeichnisse erstellen zu lassen.
19697 ::subdir      jmp    S39C5                   ;Verz. duplizieren.
19698
19699 ; 3: Fortsetzung: Verzeichnis duplizieren.
19700 :N7221        lda    #$01                   ;$01=Name eingegeben.
19701             sta    V71D0                     ;=SubDir dupliziert.
19702
19703             lda    V04AB
19704             beq    :1
19705
19706             sec
19707             rts
19708
19709 ::1           clc
19710             rts
19711
19712 ; Datei umbenennen/duplizieren.
19713 :z7295        lda    #NULL                   ;Dialogbox
19714             sta    v73A0                     ;vorbereiten.
19715
19716             ldy    #$00
19717 ::1           lda    V0470,y                 ;Dateiname als
19718             sta    V0496,y                 ;Vorgabe kopieren.
19719             beq    x72A7
19720             iny
19721             bne    :1
19722
19723 ; Dialogbox: Name eingeben.
19724 :x72A7        lda    #< V0496               ;Neuer Dateiname.
19725             sta    a1L

```

```

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

```

```

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

```

```

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

```

```

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

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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.

```



```

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

```

```

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

```

```

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

```

```

21460          lda    #< d7014
21461          ldx    #> d7014
21462          jsr    S2090                ;DBox ohne Recover.
21463
21464          pla
21465          sta    C0L043C                ;Farbe Dialogbox.
21466
21467          pla
21468          sta    r5L
21469          pla                                ;Zeiger innerhalb
21470          sta    r5H                ;Verzeichnisblock.
21471
21472          pla
21473          sta    r1L
21474          pla                                ;Zeiger auf Tr/Se
21475          sta    r1H                ;Verzeichnisblock.
21476
21477          ldx    v6EF6                ;Daten geändert?
21478          beq    :done                ; => Nein, Ende...
21479
21480          ldy    #$00
21481          lda    dirEntryBuf            ;Dateityp mit
21482          sta    (r5),y                ;Schreibschutz.
21483          sta    fileHeader +68        ;Infoblock.
21484
21485 ; Verzeichnisblock speichern.
21486 ;          lda    #< diskBlkBuf        ;Zeiger auf
21487 ;          sta    r4L                ;Zwischenspeicher.
21488 ;          lda    #> diskBlkBuf
21489 ;          sta    r4H
21490          jsr    PutBlock_dskBuf
21491          txa
21492          bne    :done                ; => Fehler, Abbruch.
21493
21494 ; GEOS-Datei: Infoblock schreiben.
21495          lda    dirEntryBuf +22
21496          beq    :done                ;Keine GEOS-Datei.
21497
21498          jsr    S09C0                ;Infoblock > r1L/r1H.
21499
21500          lda    #< fileHeader
21501          sta    r4L
21502          lda    #> fileHeader
21503          sta    r4H
21504          jsr    PutBlock                ;Infoblock.
21505
21506 ::done          rts
21507
21508 :v6EF6          b $00    ;>0 = Daten geändert.
21509
21510 ; Dialogbox: Datei-Info.
21511 :d7014          b %00000001
21512                b IB0X_Y0,IB0X_Y1
21513                w IB0X_X0,IB0X_X1
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
21571 ; Autor (wenn verfügbar).
21572             b DBVARSTR, IB_DATA_T1, IB_DATA_Y3
21573             b a3L
21574
21575 ; Dateistruktur.
21576             b DBVARSTR, IB_DATA_T1, IB_DATA_Y4
21577             b a4L
21578
21579 ; Datei/Größe.
21580             b DB_USR_ROUT
21581             w r70B0
21582
21583 ; Datei/Datum.
21584             b DB_USR_ROUT
21585             w r7117
21586
21587 ; Schreibschutz.
21588             b DB_USR_ROUT
21589             w r7197
21590
21591 ; Schreibschutz ändern.
21592             b DBOPVEC
21593             w r71C0
21594
21595 ; Texteditor.
21596             b DB_USR_ROUT
21597             w r71F7
21598
21599 ; CLOSE-Icon.
21600             b DBUSRICON, CICON_X0, CICON_Y0
21601             w v776B
21602             b NULL
21603
21604 ; "CLOSE"-Icon.
21605 :GR774D             b $80 +2*16
21606
21607             b %00000000,%00000000
21608             b %01111111,%11111110
21609             b %01000000,%00000010
21610             b %01000000,%00000010
21611             b %01000000,%00000010
21612             b %01000000,%00000010
21613             b %01000111,%11100010
21614             b %01000111,%11100010
21615
21616             b %01000111,%11100010
21617             b %01000111,%11100010
21618             b %01000000,%00000010
21619             b %01000000,%00000010
21620             b %01000000,%00000010
21621             b %01000000,%00000010
21622             b %01111111,%11111110
21623             b %00000000,%00000000
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.

```



```

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

```

```

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

```

```

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
21901         ldy    a4L
21902         lda    v7192,y
21903         beq    :1
21904
21905         lda    r0L                ;Zwei Stellen
21906         cmp    #10                ;ausgeben wenn Zahl
21907         bcs    :1                ; < 10.
21908
21909         lda    #"0"                ;Führende "0".
21910         jsr    PutChar
21911
21912 ::1        lda    #SET_LEFTJUST!SET_SUPRESS
21913         jsr    PutDecimal
21914
21915         ldy    a4L                ;Zahlentrenner.
21916         lda    v718D,y
21917         beq    :exit              ; => Ende...
21918
21919         jsr    PutChar
21920
21921 ; TODO:
21922 ; Jahrtausend über :millenium ausgeben.
21923         inc    a4L
21924         lda    a4L
21925         cmp    #2                ;Jahr ausgeben?
21926         bne    :5                ; => Nein, weiter...
21927
21928         lda    dirEntryBuf +24
21929         beq    :2
21930         lda    dirEntryBuf +23
21931         cmp    #80
21932         bcc    :3
21933
21934         ldx    #"1"
21935         lda    #"9"
21936         bne    :4
21937
21938 ::2        ldx    #"0"                ;Kein Datum, Jahr
21939         b $2c                ;als "0000" zeigen.
21940 ::3        ldx    #"2"
21941         lda    #"0"
21942
21943 ::4        pha                ;Jahrtausend.
21944         txa
21945         jsr    SmallPutChar
21946         pla
21947         jsr    SmallPutChar
21948
21949 ::5        ldy    a4L
21950         bne    :loop
21951
21952 ::exit     lda    #PLAINTEXT
21953         jmp    PutChar
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

```

```

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

```



```

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

```

```

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

```

```

22230          beq    :exit
22231
22232          lda     mouseXPos +0
22233          sta     r11L
22234          lda     mouseXPos +1
22235          sta     r11H
22236
22237          lda     mouseYPos
22238          sta     r1H
22239
22240 ; Textposition setzen.
22241 :z7378      jsr     z74AC
22242 if FALSE
22243          bcc     x739B          ; => C-Flag immer 0!
22244
22245 ; Hinweis:
22246 ; Der Code hier wird nie ausgeführt!
22247 ;---
22248          ldx     v760A          ;Aktuelle Position.
22249          beq     x739B
22250          dex
22251          dex
22252          stx     v760A          ;Neue Position.
22253
22254 ; TODO:
22255 ; Debugging-Code ?
22256 ; Ist so auch noch in TopDesk V1.3
22257 ; enthalten (SetTextPtr/EditText).
22258          jsr     InitForIO
22259          inc     $d020
22260          jsr     DoneWithIO
22261
22262          jsr     z73C4          ;Cursor setzen.
22263
22264          lda     #KEY_RIGHT
22265          sta     keyData
22266
22267          jmp     r7269          ;Tastaturabfrage.
22268 ;---
22269 endif
22270 :x739B      jmp     z73C4          ;Cursor setzen.
22271
22272 ; Text anzeigen.
22273 ; Berechnet alle Zeiger neu und gibt
22274 ; alle Zeilen ab der aktuellen aus.
22275 :z739E      jsr     z7435          ;Text überprüfen.
22276
22277          jsr     z754D          ;Akt.Zei.Pos. testen.
22278
22279          lda     v760B          ;Aktuelle Zeile.
22280          pha
22281
22282          lda     v760B          ;Aktuelle Zeile.
22283          beq     :1
22284

```

```

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

```

```

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

```

```

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

```

```

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

```


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

```

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

```

```

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

```

```

22776 ; Füllmuster gefüllt.
22777 ;---
22778 ::x2BF9          jsr    S11D0          ;Textgrenzen r5-r7.
22779                jsr    S1D1C          ;Schnittfläche?
22780                bcs    z2C2E          ; => Nicht vorhanden.
22781
22782                lda    r2L
22783                clc
22784                adc    #ICON_OFF_Y0
22785                sta    r2L
22786
22787                sec
22788                lda    r2H
22789                sbc    #ICON_OFF_YU
22790                sta    r2H
22791
22792                lda    r4L
22793                sec
22794                sbc    #< ICON_OFF_XR
22795                sta    r4L
22796                lda    r4H
22797                sbc    #> ICON_OFF_XR
22798                sta    r4H
22799
22800                lda    #< ICON_OFF_XL
22801                clc
22802                adc    r3L
22803                sta    r3L
22804                lda    #> ICON_OFF_XL
22805                adc    r3H
22806                sta    r3H
22807
22808                lda    #0                ;Füllmuster.
22809 ;                lda    #2                ;Testmuster.
22810                jsr    S140C          ;Rechteck zeichnen.
22811
22812 ; Hinweis:
22813 ; r2 bis r4 sind jetzt verändert!
22814 ; (Textausgabe nur im mark.Bereich)
22815 ;---
22816 endif
22817
22818 :z2C2E          ldx    WIN042F          ;Aktives Fenster.
22819                jsr    S61E9          ;a9=Fensterfarben.
22820
22821 if ENABLE_DIFFICOL = FALSE
22822 ::4            ldy    #$10
22823                lda    (a9),y          ;Fensterfarbe.
22824 endif
22825 if ENABLE_DIFFICOL = TRUE
22826 ::4            lda    V621A          ;Fensterfarbe.
22827 endif
22828
22829                jsr    S1561          ;Screencolor setzen.
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
22887 ;          txa
22888          pha
22889          jsr    S524C          ;Dat.-Ausw. aufheben.
22890          pla
22891          tax
22892          rts
22893
22894 ; Ausgabe Datei-Icon.
22895 :x2C89      lda    a4L
22896          sta    r3L
22897          clc
22898          adc    #< 24
22899          sta    r4L
22900          lda    a4H
22901          sta    r3H
22902          adc    #> 24
22903          sta    r4H
22904
22905          lda    a3H
22906          sta    r2L
22907          clc
22908          adc    #21
22909          sta    r2H
22910
22911          jsr    S11D0          ;Textgrenzen r5-r7.
22912
22913 ; Hinweis:
22914 ; Hier nur der rechte Rand für das
22915 ; Fenster benötigt.
22916          lda    r4L          ;Nur rechten Rand
22917          sta    v2EB9 +0      ;zwischenspeichern.
22918          lda    r4H
22919          sta    v2EB9 +1
22920
22921          jsr    S1D1C          ;Schnittfläche?
22922          bcc    :2            ; => Ja, weiter...
22923
22924          lda    V18AD          ;Fens.-Aufbau-Status.
22925          bne    :4            ; => Fenster-Aufbau.
22926          inc    v2EB8          ;Geschl.Fenster +1.
22927
22928 ::2          lda    a0H
22929          pha
22930          lda    a0L
22931          pha
22932          ldx    WIN042F        ;Aktives Fenster.
22933          jsr    S11A9          ;a0=Fensterdaten.
22934          jsr    S11C0          ;Fens.daten > r5-r7.
22935          pla
22936          sta    a0L
22937          pla
22938          sta    a0H
22939
22940          lda    r7L          ;Rechten Rand

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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

```

```

23216             dey
23217             bpl     :1
23218
23219 ; Hinweis:
23220 ; Hier wird der GEOS-Dateityp mit einem
23221 ; Bit markiert. Dient als Kennung für
23222 ; "Icon vorhanden". Die unteren 7 Bit
23223 ; werden zur Farbwahl für das Icon
23224 ; verwendet, >=16 Standardfarbe.
23225             lda     diskBlkBuf +69
23226             ldy     #$00
23227             ora     #%10000000
23228             sta     (a2),y             ;$80!GEOS-Dateityp.
23229
23230             ldx     #NO_ERROR
23231 ::err        rts
23232
23233 ; Keine GEOS-Datei.
23234 ::2          iny
23235             lda     (a1),y             ;Infoblock?
23236             beq     :5                 ; => Nein, weiter...
23237
23238 ; Hinweis:
23239 ; Die Routine Z26CD setzt bei einem
23240 ; Verzeichnis den Tr/Se für Infoblock
23241 ; wie bei einem Ordner auf $00/$FF.
23242 ; Da hier bereits eine GEOS-Datei
23243 ; ausgeschlossen wurde, muss es bei
23244 ; $00/$FF ein Verzeichnis sein.
23245             lda     #< GR2FFA         ;Verzeichnis-Icon.
23246             ldx     #> GR2FFA
23247             bne     :3
23248
23249 ::5          lda     #< GR2FBA         ;Commodore-Icon.
23250             ldx     #> GR2FBA
23251
23252 ::3          sta     r4L
23253             stx     r4H
23254
23255             ldy     #64 -1             ;Icon in DIRBUF
23256 ::4          lda     (r4),y             ;kopieren.
23257             sta     (a2),y
23258             dey
23259             bne     :4
23260
23261 ; Hinweis:
23262 ; Bit%7 setzen = Icon vorhanden.
23263             lda     #%10000000 ! NOT_GEOS
23264             sta     (a2),y
23265
23266             ldx     #NO_ERROR
23267             rts
23268
23269 :v2EB8       b $00
23270

```

```

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

```

```

23326          lda    r2L
23327          sta    a2L
23328
23329          lda    a5L
23330          sta    a3L
23331          lda    a5H
23332          sta    a3H
23333
23334          lda    #0
23335 ::1         jsr    A2BB2          ;Rechteck f.Eintrag.
23336          bcs    :2          ; => Kein Eintrag...
23337          pha
23338          jsr    IsMseInRegion
23339          bne    :3
23340          pla
23341 ::2         clc
23342          adc    #1
23343          cmp    a8H          ;Wird verändert.
23344          bne    :1
23345
23346          ldx    #$ff          ;Nicht gefunden.
23347          rts
23348
23349 ::3         pla
23350          tax          ;Icon 0-(MAXFILES-1).
23351          rts
23352
23353 ; 4: Rechteck für Dateieintrag ermitteln.
23354 ; Übergabe: A = Datei-Nr.
23355 ;          a2L,a3 = linke/obere Ecke erstes Icon.
23356 ; TD13: _GetFileRect
23357 :M2F20      pha
23358          sta    r3L
23359          lda    #$00
23360          sta    r3H
23361
23362          ldy    WIN042F      ;Aktives Fenster.
23363          lda    v2EC0,y      ;Anzahl Icons/Zeile.
23364          sta    r4L
23365          lda    #$00
23366          sta    r4H
23367
23368          ldx    #r3
23369          ldy    #r4
23370          jsr    Ddiv
23371
23372          lda    r3L
23373          sta    r2L
23374          lda    r3H
23375          sta    r2H
23376
23377          lda    r8L
23378          sta    r3L
23379          lda    r8H
23380          sta    r3H

```

```

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

```



```

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

```

```

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:

```

```

23546      sta    a3L                      ;$bf + 63 Byte Daten.
23547      lda    #> 64
23548      sta    a3H
23549
23550      ldx    #a2
23551      ldy    #a3
23552      jsr    DMult
23553
23554      ldx    C042A                      ;Oberstes Fenster.
23555      jsr    S599A                      ;Zeiger auf DIRBUF.
23556
23557      lda    #< 16*18 +1                ;Zeiger auf erstes
23558      clc                                ;Byte hinter dem
23559      adc    r1L                        ;Icon-Kopfbyte $bf.
23560      sta    r1L
23561      lda    #> 16*18 +1
23562      adc    r1H
23563      sta    r1H
23564
23565      lda    r1L                        ;Zeiger auf Datei-
23566      clc                                ;Icon berechnen.
23567      adc    a2L
23568      tax
23569      lda    r1H
23570      adc    a2H
23571      tay
23572      rts
23573
23574 ; Commodore-Icon (Nicht-GEOS-Dateien).
23575 :GR2FBA      b $bf
23576      b %00000000,%00000000,%111111100
23577      b %00000000,%00000001,%00000010
23578      b %11111111,%11111110,%00000001
23579      b %10000000,%00000000,%00000001
23580      b %10000000,%111111100,%00000001
23581      b %10000001,%111111100,%00000001
23582      b %10000011,%10000011,%11100001
23583      b %10000011,%10000011,%11000001
23584      b %10000011,%10000000,%00000001
23585      b %10000011,%10000011,%11000001
23586      b %10000011,%10000011,%11100001
23587      b %10000001,%111111100,%00000001
23588      b %10000000,%111111100,%00000001
23589      b %10000000,%00000000,%00000001
23590      b %10000001,%11000001,%00000001
23591      b %10000010,%00000011,%00000001
23592      b %10000011,%11000101,%00000001
23593      b %10000010,%00101111,%10000001
23594      b %10000001,%11000001,%00000001
23595      b %10000000,%00000000,%00000001
23596      b %11111111,%11111111,%11111111
23597
23598 ; Verzeichnis-Icon.
23599 :GR2FFA      b $bf
23600      b %00000000,%00000001,%11111110

```

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

```

```

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

```

```

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

```



```

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

```

```

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.

```

```

23978
23979 ; Zahl 0-9 linksbündig ausgeben.
23980 ; TD13: :num
23981 :z2D60          sta    r0L
23982              lda    #$00
23983              sta    r0H
23984              lda    #SET_LEFTJUST!SET_SUPRESS
23985              jmp     S590A          ;Zahl ausgeben.
23986
23987 ; TODO:
23988 ; Die Liste mit Datei-Information
23989 ; zusammenlegen, Ausnahme für TopDesk-
23990 ; Ordner hinzufügen.
23991 ; TD13: TypTab
23992 :v2D79          w t2DA7  ;$00 = Nicht-GEOS.
23993              w t2DB2  ;$01 = BASIC.
23994              w t2DB8  ;$02 = Assembler.
23995              w t2DC2  ;$03 = Daten.
23996              w t2DC7  ;$04 = Systemdatei.
23997              w t2DD3  ;$05 = Hilfsmittel.
23998              w t2DE1  ;$06 = Anwendung.
23999              w t2DEB  ;$07 = Dokument.
24000              w t2DF4  ;$08 = Zeichensatz.
24001              w t2E05  ;$09 = Druckertreiber.
24002              w t2E14  ;$0A = Eingabetreiber.
24003              w t2E28  ;$0B = Laufwerkstreiber/Ordner.
24004              w t2E32  ;$0C = Startprogramm.
24005              w t2E40  ;$0D = Temporär.
24006              w t2E49  ;$0E = Selbstausführend.
24007              w t2E5A  ;$0F = Eingabetreiber (C128).
24008 :GTDIR          w t2E6F  ;$10 = Unterverzeichnis.
24009              w t2E80  ;$11 = gateWay-Dokument.
24010              w t2EA0  ;$12 = Unbekannt.
24011              w t2EA0  ;$13 = Unbekannt.
24012 :GTERR          w t2EA0  ;$14 = Unbekannt.
24013              w t2E88  ;$15 = GeoShell-Befehl.
24014              w t2E91  ;$16 = GeoFax-Drucker.
24015              w t2E9A  ;$17 = TD-Ordner / TDFOLDER_EXT.
24016 :GTYPESEND
24017 :GTYPESEMAX     = (GTYPESEND - v2D79)/2
24018 :GTYPE_ERR      = (GTERR - v2D79)/2
24019 :GTYPE_DIR       = (GTDIR - v2D79)/2
24020
24021 if LANG = LANG_DE
24022 :t2DA7          b "Nicht-GEOS",0
24023 :t2DB2          b "BASIC",0
24024 :t2DB8          b "Assembler",0
24025 :t2DC2          b "Daten",0
24026 :t2DC7          b "Systemdatei",0
24027 :t2DD3          b "Hilfsprogramm",0
24028 :t2DE1          b "Anwendung",0
24029 :t2DEB          b "Dokument",0
24030 :t2DF4          b "Zeichensatz",0
24031 :t2E05          b "Druckertreiber",0
24032 :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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

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

24601 ; Endadresse testen:

```
24602          g BASE_MOD_B + SIZE_FSERVICE
```

24603

```

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

```

```

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.

```



```

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

```

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	

```

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

```

```

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

```

```

24934 :v7664          b $00
24935 :v7665          b $00
24936
24937 ; 2: Drucker-/Eingabetreiber wechseln.
24938 ; Übergabe: r15  = Zeiger auf Dateiname.
24939 ;           V25D0 = GEOS-Dateityp.
24940 :N6F30          bit   firstBoot          ;GEOS-Boot?
24941                bpl   :skip              ; => Ja, weiter...
24942
24943                jsr   S5C89                ;Hole Disk/Fenster.
24944                tax
24945                beq   :1
24946                cpx   #CANCEL_ERR
24947                beq   :exit
24948
24949                jmp   S5CF3                ;Fehler ausgeben.
24950
24951 ::1             lda   V25CF                ;Aufruf aus Quicklink
24952                cmp   #$01                ;öffnen (mod.#09)?
24953                beq   :skip              ; => Ja, weiter...
24954
24955 ; Aufruf aus Datei/öffnen.
24956                lda   #< t6FED            ;Druckertreiber.
24957                ldx   #> t6FED
24958
24959                ldy   V25D0                ;GEOS-Dateityp.
24960                cpy   #PRINTER            ;Drucker wechseln?
24961                beq   :setdev             ; => Ja, weiter...
24962
24963                lda   #< t6FFC            ;Eingabetreiber.
24964                ldx   #> t6FFC
24965
24966 ; Drucker-/Eingabetreiber installieren.
24967 ; TD13: InstallDriver
24968 ::setdev        sta   r14L
24969                stx   r14H
24970
24971                lda   r15H
24972                pha
24973                lda   r15L
24974                pha
24975
24976                lda   #< d6FCB            ;Treiber wählen.
24977                ldx   #> d6FCB
24978                jsr   S2090                ;DBox ohne Recover.
24979
24980                pla
24981                sta   r15L
24982                pla
24983                sta   r15H
24984
24985                lda   r0L
24986                cmp   #YES                 ;Treiber inst.?
24987                bne   :exit               ; => Nein, Ende...
24988

```

```

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

```

```

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

```

```

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.

```



```

25154 ;---
25155
25156 ; Variante#2:
25157 ; PadColorPref als Vorgabe auf alle
25158 ; Fenster anwenden, aber nur wenn die
25159 ; Option "Verschiedene Fensterfarben"
25160 ; deaktiviert ist. Es besteht das
25161 ; Risiko das einige Iconfarben gleich
25162 ; sind wie die Fensterfarben.
25163 ;--- Var2:
25164 if ENABLE_DIFFICOL = TRUE
25165         lda    MT3F58                ;Versch. Fens.Farben?
25166         cmp    #""
25167         beq    :diffcol             ; => Ja, weiter...
25168
25169         lda    V04AF                ;Neue PadColorPref?
25170         beq    :diffcol             ; => Nein, weiter...
25171
25172         ldy    #15                  ;Farben auf alle
25173 ::copy      lda    V620B,y           ;Fenster anwenden.
25174             sta    V621C,y
25175             sta    V622D,y
25176             sta    V623E,y
25177             dey
25178             bpl    :copy
25179 endif
25180 ;---
25181
25182 ; Verschiedene Fensterfarben:
25183 ;--- Var1+Var2:
25184 ::diffcol   jsr    S327F             ;Prefs. laden.
25185 ;---
25186
25187 ::cont      ldx    #%00000000
25188             lda    C0438             ;TopDesk in Farbe?
25189             beq    :1                ; => Nein, weiter...
25190             ldx    #%10000000       ;Dialogbox (System)
25191 ::1         stx    Flag_SetColor     ;in Farbe anzeigen.
25192
25193 ; Hinweis:
25194 ; Systemfarben nicht verändern.
25195 ;         lda    C0L043C             ;Farbe Dialogbox.
25196 ;         sta    C_DBoxBack
25197
25198         lda    C0L0439             ;GEOS-Hintergrund.
25199         sta    C_GEOS_BACK
25200         sta    screencolors
25201
25202         ldy    #< ResetScreen
25203         ldx    #> ResetScreen
25204
25205         lda    C0438             ;TopDesk in Farbe?
25206         beq    :2                ; => Nein, weiter...
25207
25208         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".

```

```

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

```

```

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

```

```

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

```

```

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.

```



```

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

```

```

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?

```

```

25704      beq    :fd2                ; => Ja, weiter...
25705      cmp    #"4"                ;FD4000?
25706      bne    :err                ; => Ungültig...
25707
25708  ::fd4      ldy    #OFF_FD4
25709      b $2c
25710  ::fd2      ldy    #OFF_FD2
25711
25712  if ENABLE_DRVMODE = FALSE
25713      ldx    curDrive
25714      lda    driveType -8,x
25715      and    #ST_DMODES
25716      cmp    #DrvNative
25717      bcc    :1
25718      beq    :nm
25719
25720  ::dos      iny                ;FD/DOS.
25721  ::nm      iny                ;FD/Native.
25722  endif
25723
25724  ::1      tya
25725
25726      jsr    z7170                ;Icon erzeugen.
25727
25728      sec
25729      rts                ;Laufwerk erkannt.
25730
25731  ::err      clc                ;Nicht erkannt.
25732      rts
25733
25734  ; Auf CMD-HD testen.
25735  :z7166     ldx    curDrive
25736      lda    RealDrvType -8,x
25737      bmi    :err                ; => RAMDisk...
25738      and    #DrvCMD
25739      cmp    #DrvHD                ;CMD-HD?
25740      bne    :err                ; => Nein, Ende...
25741
25742      jsr    z7163                ;Laufwerksmodus.
25743      bcs    :err                ; => Ungültig...
25744      cmp    #DrvNative +1
25745      bcs    :err                ; => Ungültig...
25746
25747      ldy    #OFF_HD
25748
25749  if ENABLE_DRVMODE = FALSE
25750      ldx    curDrive
25751      lda    driveType -8,x
25752      and    #ST_DMODES
25753      cmp    #DrvNative
25754      bne    :1
25755
25756      iny
25757  endif
25758

```

```

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
25810          rts          ;Laufwerk erkannt.
25811
25812 ::err         clc          ;Nicht erkannt.
25813          rts

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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_CBMPc       = _id02d - v7172

```

```

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

```

```

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

```

```

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

```

```
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
```

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

```

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

```



```

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

```

```
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
```

```

26584          b %00111010,%00110001,%00101110
26585 endif
26586
26587 ; Kennung "CMD-HD".
26588 :GR79AE          b %00111000,%01010110,%00001110
26589          b %00111000,%01010101,%00001110
26590          b %00111000,%01110101,%00001110
26591          b %00111000,%01010101,%00001110
26592          b %00111000,%01010101,%00001110
26593          b %00111000,%01010110,%00001110
26594
26595 ; Kennung "CMD-HD/Native".
26596 if ENABLE_DRVMODE = FALSE
26597 :GR79C3          b %00111010,%10110000,%00001110
26598          b %00111010,%10101001,%00101110
26599          b %00111011,%10101001,%10101110
26600          b %00111010,%10101001,%01101110
26601          b %00111010,%10101001,%00101110
26602          b %00111010,%10110001,%00101110
26603 endif
26604
26605 ; Kennung "DOS".
26606 :GR7A6B          b %00111001,%10001000,%11001110
26607          b %00111001,%01010101,%00001110
26608          b %00111001,%01010100,%10001110
26609          b %00111001,%01010100,%01001110
26610          b %00111001,%01010100,%01001110
26611          b %00111001,%10001001,%10001110
26612
26613 ; RAMDisk
26614 if ENABLE_DRVMODE = FALSE
26615 :GR77D9          b $80 +DRVICON_X *DRVICON_Y
26616          b %01111111,%11111111,%11111110
26617          b %01000000,%00000000,%00000010
26618          b %01011111,%11111111,%11111010
26619          b %01000000,%00000000,%00000010
26620          b %01011111,%11111111,%11111010
26621          b %01000000,%00000000,%00000010
26622          b %01011111,%11111110,%00000010
26623          b %01000000,%00000000,%10001010
26624          b %01011100,%10010010,%10101010
26625          b %01010101,%01010110,%10101010
26626          b %01010101,%01011010,%11101010
26627          b %01011101,%11010010,%00101010
26628          b %01010101,%01010010,%00101010
26629          b %01010101,%01010010,%00101010
26630          b %01000000,%00000000,%00000010
26631          b %01111111,%11111111,%11111110
26632          b %00010000,%00010000,%00000000
26633          b %00010111,%11010000,%00000000
26634          b %00010000,%00010000,%00000000
26635          b %00011111,%11111000,%00000000
26636
26637 ; Kennung "RAM1541".
26638 ;:GR7817          b %01000000,%00000000,%10001010

```

```
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
```

```

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

```

```

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,%111111110
26791      b %01101110,%11111111,%111111110
26792      b %01101010,%11111111,%111111110
26793      b %01100010,%11111111,%111111100
26794      b %01111010,%11111111,%111111100
26795      b %01111010,%11111111,%111111110
26796      b %01111111,%11111111,%111111110
26797      b %01111111,%11000011,%111111110
26798      b %01111111,%10011001,%111111110
26799      b %01111111,%10100101,%111111110
26800      b %01111111,%10011001,%111111110
26801      b %01111111,%11000011,%111111110
26802      b %01111111,%11111111,%111111110
26803      b %01110110,%111111110,%00100010

```

```

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

```

```

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:

```



```
26914 ; Der Bereich ab $7900 wird beim laden
26915 ; eines Druckertreibers verändert.
26916 ; Daher sollte die Initialisierung von
26917 ; TopDesk am Ende des Moduls liegen,
26918 ; da der Code nur zu Beginn von TopDesk
26919 ; verwendet wird.
26920 ; Aktuell wird nur der Eingabetreiber
26921 ; beim Treiberwechsel geladen, beim
26922 ; Druckertreiber wird nur der Name des
26923 ; Treibers im Kernal gesetzt.
26924 ;
26925 ; Endadresse testen:
26926         g OS_VARS
26927
```

```

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

```

```

27093 ; Nächstes Ziel-Laufwerk suchen.
27094         jsr    z734E                ;Nächstes Laufwerk =
27095         stx    v6EED                ;Lfwk. Target-Disk.
27096
27097 ; Kopieren nur mit RAMDisk nicht möglich.
27098         lda    #PLAINTEXT           ;Text "Laufwerke"
27099         sta    t7302                ;als Vorgabe.
27100
27101         cpy    #2                    ;Mehrere Laufwerke?
27102         bcs    :4                    ; => Ja, weiter...
27103
27104         lda    #NULL                ;
27105         sta    t7302                ;Keine "Laufwerke".
27106
27107 ;
27108         lda    #$00                 ;
27109         tax                                     ;Keine Laufwerks-
27110         sta    v7351,x              ;auswahl möglich.
27111         inx
27112         cpx    #4
27113         bcc    :3
27114
27115         bit    v6EEA                ;Source = RAMDisk?
27116         bpl    :4                    ; => Nein, weiter...
27117
27118         dey
27119         beq    :4                    ;Zusätzl. Laufwerke?
27120         ; => Ja, weiter...
27121
27122         ldx    #CANCEL_ERR
27123         rts
27124
27125 ; Target-Disk wählen.
27126 ::4         lda    v6EED
27127         jsr    SetDevice            ;Laufwerk aktivieren.
27128         jsr    z73A3                ;Neue Diskette.
27129         txa
27130         bne    :err                ;Fehler?
27131         ; => Ja, Abbruch...
27132
27133 ; Source-Disk.
27134         lda    v6EEC                ;Lfwk. Source-Disk.
27135         clc
27136         adc    #"A" -8
27137         sta    t7360                ;IDD_SDr.
27138
27139 ; Target-Disk.
27140 :x7249      lda    v6EED                ;Lfwk. Target-Disk.
27141         clc
27142         adc    #"A" -8
27143         sta    t7362                ;IDD_DDr.
27144
27145 ; Dialogbox anpassen: Diskwechsel.
27146         ldy    #DISK                ;Diskwechsel.
27147         ldx    v6EED                ;Target-Disk.
27148         lda    driveType -8,x        ;RAMDisk?
27149         bpl    :1                    ; => Nein, weiter...
27150         ldy    #NULL                ;Kein Diskwechsel.

```

```

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

```

```

27203 ; Dialogbox: OK.
27204 ::ok          lda    curDirHead +$bd
27205              cmp    #"P"                ;Hauptdiskette?
27206              beq    :sysdisk
27207              cmp    #"B"                ;Systemdiskette?
27208              beq    :sysdisk
27209
27210              jmp    z7586                ;Disk kopieren.
27211
27212 ; Fehler: Target-Disk ist System-/Hauptdiskette.
27213 ::sysdisk      lda    #< d730F
27214              ldx    #> d730F
27215              jsr    S2090                ;DBox ohne Recover.
27216
27217 ::cancel       ldx    #CANCEL_ERR
27218 ::err          rts
27219
27220 ; Laufwerksliste initialisieren.
27221 ; Rückgabe: Y = Anzahl Ziel-Laufwerke.
27222 :z7350         ldy    #0                ;Anzahl Laufwerke.
27223              ldx    #8
27224 ::loop         lda    driveType -8,x
27225              sta    v7351 -8,x          ;driveType speichern.
27226              beq    :next              ; => 0, weiter...
27227
27228              bit    v6EEA                ;driveType Source.
27229              bpl    :1                  ; => Keine RAMDisk...
27230
27231              cpx    v6EEC                ;Laufwerk = Quelle?
27232              beq    :4                  ; => Ja, ungültig.
27233              bne    :2                  ; => Lfwk. testen...
27234
27235 ::1            cpx    v6EEC                ;Laufwerk = Quelle?
27236              beq    :3                  ; => Ja, weiter...
27237
27238 ::2            and    #ST_DM0DES          ;driveMode Laufwerk
27239              cmp    v6EE8                ; = driveMode Source?
27240              beq    :3                  ; => Ja, gültig.
27241
27242              cmp    #Drv1571              ;Laufwerk = 1571?
27243              bne    :4                  ; => Ja, ungültig.
27244              lda    v6EE8                ;driveMode Source.
27245              cmp    #Drv1541              ;Source = 1541?
27246              bne    :4                  ; => Nein, ungültig.
27247
27248 ::3            iny                      ;Laufwerke +1.
27249              bne    :next
27250
27251 ::4            lda    #$00                ;Laufwerk löschen.
27252              sta    v7351 -8,x
27253
27254 ::next         inx
27255              cpx    #12                  ;Alle Laufwerke?
27256              bcc    :loop                ; => Nein, weiter...
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  #%00000001
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

```



```

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

```

```

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
27424 ; Hinweis:
27425 ; Hier wurde ":GetDirHead" aufgerufen,
27426 ; daher ist Sektor $01/$02 bereits in
27427 ; ":dir2Head" im Speicher.
27428         lda    dir2Head +8
27429         sta    v774A           ;Anz. Tracks Source.
27430         sta    v7122          ;Anz. Tracks Copy.
27431         clc
27432         adc    #$01
27433         sta    v7749           ;Anz.Tracks übrig.
27434
27435         lda    v774B +4        ;Anz. Tracks Target.
27436         cmp    v774A          ;Anz. Tracks Source.
27437         bcs    :2
27438
27439 ; Fehler, nur DiskCopy NativeMode,
27440 ; wenn Disk-Größe unterschiedlich.
27441         jsr    S54F0           ;Hauptmenü starten.
27442         jmp    z7260           ;Zieldisk wählen.
27443
27444 ::2         ldy    #0
27445 ::3         lda    curDirHead +$ab,y
27446         sta    v7750,y
27447         iny
27448         cpy    #18             ;Diskname kopieren.
27449         bcc    :3
27450
27451         jsr    z7704           ;BAM-Block $01/$02
27452         jsr    GetBlock        ;einlesen.
27453         txa
27454         bne    :err
27455
27456         lda    v6EED           ;Lfwk. Target-Disk.
27457         jsr    SetDevice
27458         jsr    OpenDisk
27459         txa
27460         bne    :err
27461
27462         ldy    #0              ;Disk-ID übernehmen.
27463 ::4         lda    v774B,y
27464         sta    DCOPYSTART +4,y
27465         iny
27466         cpy    #5
27467         bcc    :4
27468
27469         jsr    z7704           ;BAM-Block $01/$02
27470         jsr    PutBlock        ;auf Disk schreiben.
27471         txa
27472         bne    :err
27473
27474         jsr    z7709           ;BAM-Block $01/$01
27475         jsr    GetBlock        ;einlesen.
27476         txa
27477         bne    :err

```

```

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

```

```

27533 ; Hinweis:
27534 ; Ersten Sektor setzen.
27535         ldx    #1                ;1541/71/81.
27536         ldy    #0
27537         bit    v6EEF             ;NativeMode?
27538         bpl    :3                ; => Nein, weiter...
27539 ;         ldx    #1                ;NativeMode.
27540         ldy    #3
27541 ::3      stx    r1L               ;Aktueller Sektor
27542         sty    r1H               ;Quelldisk.
27543
27544         stx    r11L              ;Aktueller Sektor
27545         sty    r11H              ;Zieldisk.
27546
27547 ; Sektoren kopieren.
27548 ::loop   lda    #< ReadBlock
27549         sta    r0L
27550         lda    #> ReadBlock
27551         sta    r0H
27552         jsr    z6F75
27553         cpx    #INV_TRACK
27554         beq    :11
27555         txa
27556         bne    :err
27557
27558 ::11     lda    r1L               ;Tr/Se Source-Disk.
27559         sta    r10L
27560         lda    r1H
27561         sta    r10H
27562
27563         ldy    #$01              ;Target-Disk.
27564         lda    #$01              ;Keine neue Disk...
27565         jsr    z6FEA             ;Disk wechseln.
27566         txa
27567         bne    :err
27568
27569         lda    r11L
27570         sta    r1L
27571         lda    r11H
27572         sta    r1H
27573
27574         lda    #< z6FB9
27575         sta    r0L
27576         lda    #> z6FB9
27577         sta    r0H
27578         jsr    z6F75
27579         cpx    #INV_TRACK
27580         beq    :done
27581         txa
27582         bne    :err
27583
27584         ldy    #$00              ;Source-Disk.
27585         lda    #$01              ;Keine neue Disk...
27586         jsr    z6FEA             ;Disk wechseln.
27587         txa

```

```

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

```

```

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

```

```

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

```

```

27863          lda    r8L
27864          sta    v70D9          ;Erster Sektor.
27865
27866 ::ok          ldx    #NO_ERROR
27867
27868 ::exit        pla
27869          sta    r1H
27870          rts
27871
27872 :v70D6        b $00    ;Akt. Track/Status.
27873 :v70D7        b $00    ;Aktueller Sektor.
27874 :v70D8        b $00    ;Sektor per Track.
27875 :v70D9        b $00    ;Erster Sektor.
27876
27877 ; Auf letzten Track testen, letzten Sektor ermitteln.
27878 ; Rückgabe: r1H = Letzter Block/Track.
27879 ;          X   = $00, OK.
27880 ;          $02, Ende erreicht.
27881 ; TD13: GetSectors
27882 :z70DA        lda    v7748          ;Typ Source-Disk.
27883          and    #ST_DM0DES
27884          beq    :err
27885          cmp    #DrvPCDOS
27886          bcs    :err
27887
27888          tay
27889
27890          lda    r1L          ;Track = $00?
27891          beq    :err          ; => Invalid Track.
27892
27893          dey          ;1541?
27894          bne    :1571        ; => Nein, weiter...
27895
27896 ::1541        cmp    #TRACK41 +1
27897          bcc    :getsek
27898          bcs    :err
27899
27900 ::1571        dey          ;1571?
27901          bne    :1581        ; => Nein, weiter...
27902
27903          cmp    #70 +1
27904          bcs    :err          ; => Invalid Track.
27905
27906 ::getsek      ldy    #$07
27907 ::1          cmp    v7123,y
27908          bcs    :2
27909          dey
27910          bpl    :1
27911          bmi    :err          ; => Invalid Track.
27912
27913 ::2          tya
27914          and    #$03
27915          tay
27916          lda    v712B,y
27917          bne    :setsek

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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          ; => Ja, Vorgabewert.
28381
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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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

```

```

28853      jsr    UNTALK
28854
28855      lda    curDrive
28856      jsr    LISTEN
28857      lda    #$ff
28858      jsr    SECOND
28859
28860      ldy    #0                ;"B-P"-Befehl
28861  :::3      tya                ;an Laufwerk senden.
28862      pha
28863      lda    v7A26,y
28864      jsr    BSOUT
28865      pla
28866      iny
28867      cpy    #v7A26len
28868      bcc    :3
28869
28870      jsr    UNLSN            ;Befehl ausführen.
28871
28872      lda    #IEC_DEV_CHAN
28873      jsr    CLOSE
28874
28875      jsr    z7968            ;Datenk. schließen.
28876
28877      lda    v7AF0
28878      inc    v7AF0
28879      cmp    v7AEF            ;Alle Tracks?
28880      bcc    :loop            ; => Nein, weiter...
28881
28882      ldx    #NO_ERROR
28883      rts
28884
28885  ; DiskImage >16Mb, Fehler...
28886  :::err      jsr    z7968            ;Datenk. schließen.
28887
28888      ldx    #INV_TRACK
28889  :::exit      rts
28890
28891  ; P-Befehl für SD2IEC um Pointer auf
28892  ; Byte-Position zu setzen.
28893  :v7A26      b    "P"
28894      b    IEC_DATA_LFN
28895  :v7A28      s    4    ;Pointer für Position in DiskImage.
28896      b    CR
28897  :v7A2X
28898  :v7A26len    = (v7A2X - v7A26)
28899
28900  ; Anzahl Sektoren im Track ermitteln.
28901  ; Übergabe: v7AF1 = Laufwerkstyp 1-4.
28902  ; Rückgabe: X    = Fehler.
28903  ;           v7B78 = Anz. Sektoren/Track.
28904  :z7A2D      ldx    v7AF1
28905      beq    :unknown            ;Type = $00.
28906      dex
28907      beq    :2                ;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
28936                lda    v7AF0                ; D64
28937                cmp    #TRACK41 +1
28938                bcc    :4
28939
28939 ::unknown            ldx    #INV_TRACK
28940                rts
28941
28942 ::3                dey
28943                bne    :unknown            ;1571?
28944                                ; => Nein, Abbruch...
28945
28946                lda    v7AF0                ; D71
28947                cmp    #TRACK71 +1
28948                bcs    :unknown
28949
28949 ::4                ldy    #$07                ;Max. Anzahl
28950 ::5                cmp    v7123,y            ;Sektoren/Track
28951                                ;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

```

```

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

```

```

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

```

```

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

```

```

29183
29184 ;
29185 ; SourceCode: src.mod#09.s
29186 ;
29187
29188 ;
29189 ; Modul-Info:
29190 ; 0 - Neues Verzeichnis
29191 ; 1 - Verzeichnis löschen
29192 ; 2 - Quicklink/Programm starten
29193 ; 3 - Hilfsmittel/Anwendungen/Dokumente/Drucker/Eingabe
29194 ; 4 - Verzeichnis kopieren.
29195 ; 5 - Verzeichnis duplizieren
29196 ;      (Aufruf nur durch mod.#02)
29197 ; 6 - Ordner kopieren
29198 ; 7 - Neuen Ordner erstellen
29199 ;
29200 if .p
29201             t "TopSym"
29202             t "TopSym.MP3"
29203             t "TopSym.ROM"
29204             t "DTopDeskI18N.ext"
29205             t "src.DTopDesk.ext"
29206
29207 :SIZE_DBUF      = 128
29208 endif
29209
29210             o BASE_MOD_A
29211             n "obj.#09"
29212
29213 ::ModEntry_0      jmp     N747E    ;Neues Verzeichnis.
29214 ::ModEntry_1      jmp     N76B7    ;Verzeichnis löschen.
29215 ::ModEntry_2      jmp     N7237    ;Quicklink öffnen.
29216 ::ModEntry_3      jmp     N72D7    ;Appl./DA/Dok./Print/Input.
29217 ::ModEntry_4      jmp     N73A9    ;Verzeichnis kopieren.
29218 ::ModEntry_5      jmp     N745D    ;Verzeichnis duplizieren.
29219 ::ModEntry_6      jmp     N7062    ;Ordner kopieren.
29220 ::ModEntry_7      jmp     N6EC4    ;Neuen Ordner erstellen.
29221
29222 ; 2: Quicklink-Programm starten.
29223 ; Übergabe: V4191:
29224 ; $00 = geoHDscsi
29225 ; $11 = F2/Quicklink #1
29226 ; $22 = F4/Quicklink #2
29227 ; $33 = F6/Quicklink #3
29228 ; $44 = F8/Quicklink #4
29229 ; $FF = GEOS.Editor/GD.CONFIG
29230 :N7237            ldx     #8                ;Erste RAMDisk
29231 ::find            lda     driveType -8,x    ;suchen.
29232                  bmi     :1
29233                  inx
29234                  cpx     #12
29235                  bcc     :find
29236
29237                  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:

```

```

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

```

```

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              ld y    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

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

29843          sta    v71BC          ; zwischenspeichern.
29844
29845          inc     v04B0          ; Zähler Instanz +1.
29846
29847          lda     v0A72          ; Lfwk. Source-Disk.
29848          jsr     s06AB          ; Source-Disk öffnen.
29849
29850          ldy     #16 -1
29851 ::11          lda     v0470,y
29852          pha
29853          dey
29854          bpl     :11
29855
29856          jsr     z7070          ; Ordner kopieren.
29857
29858          ldy     #$00          ; => Datei/Ordner.
29859          sty     v745C          ; Dateityp setzen.
29860
29861 ;          ldy     #0
29862 ::12          pla
29863          sta     v0470,y
29864          iny
29865          cpy     #16
29866          bcc     :12
29867
29868          txa
29869          beq     z715C          ; => OK...
29870          cpx     #FILE_EXISTS
29871          bne     z7151          ; => Fehler...
29872          beq     x71B6          ; => Ende...
29873
29874 ; Ordner/Verzeichnis kopieren fortsetzen.
29875 :z715C          lda     v0A72          ; Lfwk. Source-Disk.
29876          jsr     s06AB          ; Source-Disk öffnen.
29877          txa
29878          bne     z714E
29879
29880          ldx     v04A9          ; Kop.od.verschieben?
29881          beq     x71B6          ; => Kopieren.
29882
29883 ; Modus: Verschieben.
29884          ldx     v04AB          ; Abbruch?
29885          bne     x71B6          ; => Ja, Ende...
29886
29887 ; Quelle löschen.
29888          lda     v745C          ; Verz. kopieren?
29889          beq     :21          ; => Nein, weiter...
29890
29891 ; -> Verzeichnis löschen.
29892 ;
29893 ; Hinweis:
29894 ; Systemroutine zum löschen verwenden,
29895 ; damit Verz. mit Schreibschutz nicht
29896 ; gelöscht wird.
29897 ;          jsr     n76B7          ; Verzeichnis löschen.

```

```

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.

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```



```

30393 :z7550      lda    #$00
30394          sta    r10L
30395          jsr    GetFreeDirBlk
30396          txa
30397          beq    :1
30398          rts
30399
30400 ::1         lda    r1L          ;Tr/Se für Block mit
30401          sta    v76A3          ;neuem Eintrag.
30402          lda    r1H
30403          sta    v76A4
30404          sty    v76A5          ;Offset im Block.
30405
30406          ldx    WIN042F        ;Aktives Fenster.
30407          lda    C051B,x        ;Aktive Ordner-Nr.
30408          pha
30409
30410          tya          ;Offset auf Eintrag.
30411          cmp    #2            ;Eintrag #0?
30412          bne    :2            ; => Nein, weiter...
30413
30414          clc          ;Sonderbehandlung.
30415          adc    #TDFOLDER_POS0 -1
30416
30417 ::2         tax
30418          dex          ;Offset Ordner-Nr.
30419
30420          pla
30421          sta    diskBlkBuf,x    ;Ordner-Nr. setzen.
30422
30423 ; Verzeichniseintrag erzeugen.
30424 ; Übergabe: Y = Offset auf Eintrag.
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

```

```

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

```

```

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

```

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

```

30676
30677 ;
30678 ; SourceCode: src.mod#10.s
30679 ;
30680
30681 ;
30682 ; Modul-Info:
30683 ; 0 - RAM-TopDesk
30684 ; 1 - TopDesk.win einlesen
30685 ; 2 - TopDesk.win speichern
30686 ;
30687 if .p
30688         t "TopSym"
30689         t "TopSym.MP3"
30690         t "TopSym.ROM"
30691         t "DTopDeskI18N.ext"
30692         t "src.DTopDesk.ext"
30693
30694 :VMOD_BASE      = $1000 ;Temp. Ladeadresse VLIR-Modul.
30695 :VMOD_SIZE      = $2000 ;Max. Größe VLIR-Modul.
30696 :VMOD_MAX       = $f000 ;Max. Speicher für VLIR-Module.
30697 :BASE_ENTERDT   = $7e00 ;Startadr. EnterDesktop-Routine.
30698 endif
30699
30700         o BASE_MOD_A
30701         n "obj.#10"
30702
30703 ::ModEntry_0     jmp     N6EB5 ;RAM-TopDesk ein/aus.
30704 ::ModEntry_1     jmp     N70C3 ;TopDesk.win einlesen.
30705 ::ModEntry_2     jmp     N6E00 ;TopDesk.win speichern.
30706
30707 ; 2: TopDesk.win speichern.
30708 ; TD13: SaveWindowStat
30709 :N6E00           jsr     S38DB ;Fensterdat. sichern.
30710
30711 if PANEL_RIGHT = FALSE
30712         jsr     z7120 ;Fens.pos. anpassen.
30713 endif
30714
30715         jsr     S07D3 ;Systemlaufwerk.
30716         txa
30717         bne     :err
30718
30719         lda     #< F3956 ;"TopDesk.win".
30720         ldx     #> F3956
30721         jsr     S0B34 ;Datei löschen.
30722 ;         txa ;Z-Flag ist gesetzt.
30723         beq     :1
30724         cpx     #FILE_NOT_FOUND
30725         bne     :err
30726
30727 ::1           lda     #< V385E
30728         sta     r9L
30729         lda     #> V385E
30730         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

```



```

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

```

```

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

```

```

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
31007             cpx    #INV_RECORD
31008             beq     :ok
31009             bne     :err                ; => Fehler...
31010
31011 ::21             lda     curRecord          ;Max.Mod. eingelesen?
31012             cmp     #TD_MAX_VLIR
31013             bcc     :loop                ; => Weiter...
31014
31015             lda     #OUT_OF_RECORDS
31016             b $2c
31017 ::ok             lda     #NO_ERROR
31018             b $2c
31019 ::buf_full       lda     #BFR_OVERFLOW
31020 ::err            pha
31021             tax
31022             bne     :31                ;Fehler?
31023                                     ; => Ja, weiter...
31024             jsr     z7011                ;Endadr. speichern.
31025
31026 ::31             jsr     z70A4                ;APP_RAM zurückholen.
31027             jsr     z709C                ;Reset REU-MoveData.
31028
31029             jsr     S07C0                ;Lfwk. zurücksetzen.
31030
31031             pla
31032             tax
31033             beq     :32                ; => Nein, weiter...
31034
31035             jsr     S5CE1                ;Fehler ohne Refresh.
31036             jmp     z6EE2                ;RAM-TD abschalten.
31037
31038 ::32             ldx     #(r2H - r0L)
31039 ::33             lda     :data,x
31040             sta     r0,x
31041             dex
31042             bpl     :33
31043
31044             lda     C0427                ;64K-Bank RAM-TD.
31045             sta     r3L
31046
31047             jsr     StashRAM                ;VLIR-Info speichern.
31048
31049 ; Original EnterDeskTop-Routine speichern.
31050             jsr     SetADDR_EnterDT
31051             jsr     FetchRAM
31052
31053             lda     #< DACC_DTBAK
31054             sta     r1L
31055             lda     #> DACC_DTBAK
31056             sta     r1H
31057
31058             lda     C0427                ;64K-Bank RAM-TD.
31059             sta     r3L
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 + RAMDTOPOFF,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 :RAMDTOPOFF     = RAMDTPDAT - newEnterDT
31110
31111 :RAMDTCHKLOW    = (chkDTLow - newEnterDT)
31112 :RAMDTCHKHIGH   = (chkDTHigh - newEnterDT)
31113 ;---
31114
31115 ; VLIR-Info in Tabelle speichern.

```

```

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

```

```

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

```



```

31226          w :t2
31227          b DBTXTSTR,$0a,$2a
31228          w :t3
31229 if LANG = LANG_DE
31230          b DBTXTSTR,$0a,$34
31231          w :t4
31232 endif
31233
31234          b DBTXTSTR,$0a,$44
31235          w :t5
31236
31237          b OK      , $11,$4b
31238          b NULL
31239
31240 if LANG = LANG_DE
31241 ::t1          b BOLDON
31242          b "RAM-TopDesk Installation",0
31243
31244 ::t2          b PLAINTEXT
31245          b "Es ist keine 64K-Speicherbank frei",0
31246 ::t3          b "in welcher RAM-TopDesk installiert",0
31247 ::t4          b "werden könnte.",0
31248 ::t5          b "Bitte GEOS-Konfiguration anpassen.",0
31249 endif
31250 if LANG = LANG_EN
31251 ::t1          b BOLDON
31252          b "Install RAM-TopDesk",0
31253
31254 ::t2          b PLAINTEXT
31255          b "There is no free 64Kb memory bank in",0
31256 ::t3          b "which RAM-TopDesk could be installed.",0
31257 ::t5          b "You must edit the GEOS configuration.",0
31258 endif
31259
31260 :v7346          b $00                      ;Bankzeiger.
31261
31262 :v7347          b %00000011
31263          b %00001100
31264          b %00110000
31265          b %11000000
31266
31267 :v734B          b %00111111
31268          b %11001111
31269          b %11110011
31270          b %11111100
31271
31272 ; Speicherübersicht TopDesk.win $7A00-$7DF2:
31273 ; (siehe auch src.mod#00b!)
31274 ;--- TD.WIN
31275 ; $7A00 - $7AFF  Ordnerliste #1-#4
31276 ; $7B40 - $7B43  Fenster-Status
31277 ; $7B44 - $7B49  Größe Fenster #1
31278 ; $7B4A - $7B4F  Größe Fenster #2
31279 ; $7B50 - $7B55  Größe Fenster #3
31280 ; $7B56 - $7B5B  Größe Fenster #4

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

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

```

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.

```

```

31732
31733 ; CMD-Laufwerk (1541/71/81 bereits ausgeschlossen).
31734 ::1          jsr    z6F7B          ;Speicher auslagern.
31735
31736          ldy    curDrive
31737          lda    V04B8 -8,y          ;Geräteinfo einlesen.
31738          and    #%00000011          ;RAMLink/RAMDrive?
31739          beq    :fd_hd              ; => Nein, weiter...
31740
31741 ; CMD-RL/RD.
31742 ::rl_rd      lda    #$00            ;Zeiger auf erste
31743          sta    r15L                ;Partition.
31744
31745 ::11         inc    r15L
31746          lda    r15L
31747          cmp    #32                  ;Part. durchsucht?
31748          beq    :err                ; => Ja, Fehler.
31749
31750          jsr    z733C                ;Partitionsdaten.
31751          txa
31752          bne    :err                ; => Fehler, Abbruch.
31753
31754          ldy    curDrive
31755          lda    ramBase -8,y
31756          cmp    v40EF +20            ;Part.Adr.MiddleByte.
31757          bne    :11                  ; => Nein, weiter...
31758
31759          lda    r15L                ;Partition gefunden.
31760          bne    :done
31761
31762 ; CMD-FD/HD.
31763 ::fd_hd      lda    #$ff            ;Aktive Partition.
31764          jsr    z733C                ;Partitionsdaten.
31765          txa
31766          bne    :err
31767
31768          lda    v40EF +2              ;Partitions-Nr.
31769          bne    :done
31770
31771 ; Partition nicht gefunden.
31772 ::err        jsr    z6F7C            ;Speicher einlesen.
31773
31774          ldx    #PART_ERR            ;Partitionsfehler.
31775          rts
31776
31777 ; CMD-Partition gefunden.
31778 ::done       pha
31779          jsr    z6F7C                ;Speicher einlesen.
31780          pla
31781
31782 ; 64Net-/CMD-Partition speichern.
31783 ::save       ldx    V3A94
31784          sta    C04FC,x              ;Part. für Fenster.
31785
31786          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

```

```

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

```

```

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

```

```

31952         ldy    curDrive
31953
31954 if DEBUG_64NET = TRUE
31955         cpy    #10                ; Laufwerk #10 ist
31956         beq    :0                ; immer 64Net.
31957 endif
31958
31959         lda    V04B8 -8,y         ; Geräteinfo einlesen.
31960         bmi    :0                ; => 64Net.
31961
31962 ; Sonderbehandlung CMD-Laufwerk.
31963 ;
31964         lda    driveType -8,y
31965         and    #ST_DM0DES
31966         tax
31967
31968         lda    :modes,x           ; Laufwerksmodus.
31969         cmp    #"x"              ; Modus gültig?
31970         beq    :err              ; => Nein, Ende...
31971
31972         sta    v7657 +6           ; CMD-Part.-Format.
31973         bne    :1                ; => CMD.
31974
31975 ; Tabelle mit gültigen Partitionsformaten.
31976 ::modes          b "x478Nxxx"
31977
31978 ; Sonderbehandlung 64Net-Laufwerk.
31979 ::0              dec    V3A4B     ; 64Net-Part. wählen.
31980
31981         jsr    z76B4              ; Auf 64Net testen.
31982         txa
31983         bne    :err
31984
31985         jsr    z79FF              ; 64Net-Lfwk. suchen.
31986
31987 ; Neue Partiton wählen.
31988 ::1              jsr    z6F7B     ; Speicher auslagern.
31989
31990         jsr    z6F9B              ; Eintrag auswählen.
31991         txa
31992         bne    :err              ; Abbruch?
31993         ; => Ja, Ende...
31994 ; Hinweis:
31995 ; Bei keinem Fehler ist der Speicher
31996 ; hier noch nicht zurückgesetzt.
31997         lda    #< z6FEC          ; CMD-Part. öffnen.
31998         ldx    #> z6FEC
31999
32000         bit    V3A4B              ; CMD oder 64Net?
32001         bpl    :2                ; => CMD.
32002
32003         lda    #< z76B8          ; 64Net-Part. öffnen.
32004         ldx    #> z76B8
32005
32006 ; Hinweis:

```

```

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
32063 ::err          rts
32064
32065 ; Auswahl CMD-Partition beenden.
32066 ::done          ldx    C042A          ;Oberstes Fenster.
32067                stx    WIN042F        ;Aktives Fenster.
32068                jsr    S09B7          ;Tr/Se SubDir = $00.
32069
32070 ;                lda    #$00
32071                sta    C0448,x          ;Fenster geschlossen.
32072
32073                jsr    S5842          ;mark.Dat. anzeigen.
32074                jsr    S524C          ;Dat.-Ausw. aufheben.
32075
32076                lda    curDrive
32077                jsr    S06AB          ;Laufwerk öffnen.
32078
32079                ldx    C042A          ;Oberstes Fenster.
32080                jmp    S44E8          ;Fens. erneut öffnen.
32081
32082 ; Speicher in REU auslagern.
32083 :z6F7B          bit    Flag_ExtRAMinUse
32084                bpl     :1
32085
32086                brk     ;Panic! Speicher bereits belegt!
32087
32088 ::1             lda    Flag_ExtRAMinUse
32089                ora     #%10000000
32090                sta    Flag_ExtRAMinUse
32091                bne     z6F7E
32092
32093 ; Speicher aus REU einlesen.
32094 :z6F7C          bit    Flag_ExtRAMinUse
32095                bmi     :1
32096
32097                brk     ;Panic! Speicher ist freigegeben!
32098
32099 ::1             lda    Flag_ExtRAMinUse
32100                and     #%01111111
32101                sta    Flag_ExtRAMinUse
32102
32103 ; Speicher RAM/REU tauschen.
32104 :z6F7E          lda    #< BASEDATA
32105                sta    r0L
32106                lda    #> BASEDATA
32107                sta    r0H
32108
32109                lda    #$00
32110                sta    r1L
32111                sta    r1H
32112
32113                lda    #< SIZEDATA
32114                sta    r2L
32115                lda    #> SIZEDATA
32116                sta    r2H

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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
32393 ; Koordinaten für Eingabefeld.
32394 ::coord2      b PBOX_Y0
32395              b PBOX_Y1
32396              w PBOX_X0
32397              w PBOX_X1
32398
32399 ; Mausabfrage für Dateifenster.
32400 :r73DF        lda  mouseData
32401              bmi  :exit
32402
32403              jsr  z7609          ;Größe Dateifenster.
32404
32405              jsr  IsMseInRegion  ;Klick Dateifenster?
32406              beq  z7440          ; => Nein, weiter...
32407
32408              jsr  z75F6          ;Eintrag invertieren.
32409
32410              jsr  z7609          ;Größe Dateifenster.
32411
32412              lda  mouseYPos      ;Zeiger auf Eintrag
32413              sec                  ;in Liste berechnen.
32414              sbc  r2L
32415              sta  r0L
32416              lda  #$00
32417              sta  r0H
32418              sta  r1H
32419
32420              lda  #ENTRY_YH      ;12 Pixel Höhe eines
32421              sta  r1L            ;Dateieintrags.
32422
32423              ldx  #r0
32424              ldy  #r1
32425              jsr  Ddiv
32426
32427              lda  r0L            ;Datei angeklickt?
32428              clc
32429              adc  v7653          ;Akt. Pos in Liste.
32430              cmp  v7656          ;Anzahl Einträge.
32431              bcc  :1            ; => Ja, weiter...
32432
32433              ldx  v7656          ;Anzahl Einträge -1
32434              dex                ; = Letzter Eintrag.
32435              txa
32436 ::1          sta  v7652          ;Markierter Eintrag.
32437
32438              jsr  z75F6          ;Eintrag invertieren.
32439
32440              lda  #10
32441              sta  dblClickCount
32442
32443              lda  #%10000000
32444              sta  mouseData
32445 ::wait      lda  mouseData
32446              beq  :dblclk

```

```

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

```



```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```


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

```

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

```

```

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

```

```

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

```

```

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

```

```

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

```

```

33272          ldy  curDrive          ;ramBase setzen.
33273          lda  v40EF +20
33274          sta  ramBase -8,y
33275          lda  v40EF +21          ;Low-Byte wird nicht
33276          sta  driveData +3      ;mehr verwendet.
33277
33278 ;          ldx  #NO_ERROR
33279 ::done      rts
33280
33281 ; CMD-Partition über "C-P"-Befehl setzen.
33282 ; Übergabe: X = Partition 1-254.
33283 :z71D3      dex
33284
33285          lda  v4110,x             ;Part.-Nr aus Tab.
33286          sta  v765F +2           ;Part.-Nr. für "C-P".
33287
33288          jsr  PurgeTurbo
33289          jsr  InitForIO
33290
33291          lda  curDrive            ;Aktuelles Laufwerk
33292          pha                    ;zwischenspeichern.
33293
33294          tay
33295          lda  V04B8 -8,y         ;Geräteinfo einlesen.
33296          and  #%00000011        ;RAMLink/RAMDrive?
33297          beq  :1                ; => Nein, weiter...
33298
33299          lda  C0509              ;RAMLink-Adresse.
33300          sta  curDevice
33301          sta  curDrive
33302
33303 ::1          lda  #< v765F        ;"C-P" + Part.-Nr.
33304          sta  r0L
33305          lda  #> v765F
33306          sta  r0H
33307
33308          jsr  z73AF              ;Befehl senden.
33309          txa
33310          pha
33311
33312          jsr  UNLSN
33313
33314          pla
33315          tax
33316
33317          pla                    ;Laufwerk wieder
33318          sta  curDevice          ;zurücksetzen.
33319          sta  curDrive
33320
33321          txa
33322          beq  :done
33323
33324          ldx  #FILE_NOT_FOUND
33325 ::done      jmp  DoneWithIO
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                ;Nächstes Laufwerk.
33367             cpx    #12
33368             bne    :4
33369             ldx    #8
33370 ::4             cpx    a9L          ;Alle Lfwk. getestet?
33371             bne    :1          ; => Nein, weiter...
33372
33373             txa                ;Laufwerk wieder
33374             jmp    SetDevice      ;zurücksetzen.
33375
33376 ; Auf 64Net-Treiber testen.
33377 :z7A28          ldx    #0
33378 ::1             lda    DiskDrvVersion,x
33379             cmp    v7A3D,x
33380             bne    :err
33381             inx

```



```

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

```

```

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.

```

```

33492 ;          lda    #$00
33493          sta    r1H
33494 ;          lda    #$00
33495          sta    r2L
33496
33497          lda    #< diskBlkBuf      ;Zwischenspeicher
33498          sta    r0L                ;für Systemblock.
33499          lda    #> diskBlkBuf
33500          sta    r0H
33501
33502          jsr    z7A8D                ;Block empfangen.
33503
33504 ; Testen ob neue Partition bereits aktiv.
33505          ldy    #8
33506 ::11      cpy    curDrive
33507          beq    :12
33508
33509          lda    v764D -8,y          ;64Net-Laufwerk?
33510          beq    :12                ; => Nein, weiter...
33511
33512          lda    diskBlkBuf +$78,y
33513          cmp    v7652                ;Partition aktiv?
33514          beq    :fail                ; => Ja, Fehler...
33515
33516 ::12      iny
33517          cpy    #12
33518          bne    :11
33519
33520          ldx    v7652                ;=> Neue 64Net-Part.
33521          jsr    z782B                ;Partition setzen.
33522
33523 if DEBUG_64NET = FALSE
33524          ldx    C042A                ;Oberstes Fenster.
33525          lda    v7652                ;Markierte Datei.
33526          sta    C04FC,x              ;Part. für Fenster.
33527 else
33528          ldy    #10
33529          jsr    :fail                ;Dialogbox testen.
33530 endif
33531
33532          jsr    z6F7C                ;Speicher einlesen.
33533
33534          ldx    #$00
33535 ::err      rts
33536
33537 ; Fehler: Bereits aktiv.
33538 ::fail     tya
33539          clc
33540          adc    #"A" -8
33541          sta    v7788
33542
33543          lda    #< d774A            ;Fehler: Partition
33544          ldx    #> d774A            ;bereits aktiv.
33545          jsr    S2090                ;DBox ohne Recover.
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

```

```

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

```

```

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

```

```

33712          pha
33713          lda    r2L
33714          pha
33715          jsr    z798A          ;Dialogbox testen.
33716          pla
33717          sta    r2L
33718          pla
33719          sta    r1H
33720          pla
33721          sta    r0L
33722          pla
33723          sta    r0H
33724  endif
33725
33726          lda    v4210          ;Anzahl Partitionen.
33727          rts
33728
33729 ; 64Net-Laufwerk formatieren.
33730 ::fail          ldx    #$00
33731          txa
33732 ::11          sta    v4210,x
33733          inx
33734          bne    :11
33735
33736 ;
33737 ::12          lda    v7A3D,x
33738          sta    v4210 +1,x
33739          inx
33740          cpx    #v7A3D_len
33741          bne    :12
33742
33743          lda    #$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 ;          lda    #$00
33765          sta    r1H
33766 ;          lda    #$00

```

```

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

```



```

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

```

```

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
33922                jsr    z7AF8                ;Modus "R" oder "W".
33923                jsr    z7AF8                ;Byte senden.
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.

```

```

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

```

```

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

```

```

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

```

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

```

```

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.

```



```

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

```

```

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.

```

```

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

```

```

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

```

```

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

```

```

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

```

```

34588      b %10000000,%00100011
34589      b %10000000,%10001001
34590      b %10000000,%00100011
34591      b %10000000,%10001001
34592      b %10000000,%00100011
34593      b %10000000,%10001001
34594      b %10000000,%00100011
34595      b %10101010,%11111111
34596      b %11010101,%11111111
34597      b %10101010,%11111111
34598      b %11010101,%11111111
34599      b %10101010,%11111111
34600      b %11010101,%11111111
34601      b %10101010,%11111111
34602      b %11111111,%11111111
34603
34604 :ICON_C_x      = $02
34605 :ICON_C_y      = $10
34606
34607 ; Menüaufbau.
34608 :r70F0      jsr   z73B8      ;Schatten#1 Menü.
34609      jsr   z73B7      ;Schatten#2 Menü.
34610
34611 ;--- TopDesk-Fenster.
34612      ldx   #tdwin_x0      ;Position für
34613      ldy   #tdwin_y0      ;TopDesk-Icon #1.
34614      jsr   z7375      ;TDesk-Icon anzeigen.
34615
34616      ldx   #tdwin_x0      ;Position für
34617      ldy   #tdwin_y1      ;TopDesk-Icon #2.
34618      jsr   z7375      ;TDesk-Icon anzeigen.
34619
34620      ldx   #tdwin_x0      ;Position für
34621      ldy   #tdwin_y2      ;TopDesk-Icon #3.
34622      jsr   z7375      ;TDesk-Icon anzeigen.
34623
34624      ldx   #tdwin_x0      ;Position für
34625      ldy   #tdwin_y3      ;TopDesk-Icon #4.
34626      jsr   z7375      ;TDesk-Icon anzeigen.
34627
34628 ;--- Dialogbox.
34629      jsr   z73B6      ;Schatten#1 DBox.
34630      jsr   z73B5      ;Schatten#2 DBox.
34631
34632      jsr   i_FrameRectangle
34633      b dbstd_y0,dbstd_y1
34634      w dbstd_x0,dbstd_x1
34635      b %11111111
34636
34637 ;--- Fehlerbox.
34638      jsr   z73B4      ;Schatten#1 Fehler.
34639      jsr   z73B3      ;Schatten#2 Fehler.
34640
34641      jsr   i_FrameRectangle
34642      b dberr_y0,dberr_y1

```

```

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.

```



```

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

```

```
34808      b $82
34809      b $15,$80
34810      b $05
34811      b $00
34812      b $a5
34813      b $15,$87,$e1,$f8,$7e,$1f,$80,$11
34814      b $84,$21,$08,$42,$10,$80,$11,$94
34815      b $21,$08,$42,$10,$80,$11,$84,$21
34816      b $08,$42,$10,$80,$11,$87,$e1,$f8
34817      b $7e,$1f,$80,$11,$80
34818      b $05
34819      b $00
34820      b $81
34821      b $11
34822      b $07
34823      b $ff
34824      b $81
34825      b $80
34826      b $05
34827      b $00
34828      b $89
34829      b $11,$83,$5c,$80,$01,$ae,$00,$15,$80
34830      b $05
34831      b $00
34832      b $81
34833      b $11
34834      b $07
34835      b $ff
34836  endif
34837
34838 ; DTopDesk-Fenster.
34839 :GR72EE      b $80 +7*8
34840
34841 b #####,#####,#####,#####,#####,#####,#####
34842 b %0000#00,%0#000000,%00000000,%00000000,%00000000,%00000000,%000#000#
34843 b %0#0#0#0#,%0#00#0##,%0#0###0#,%00000000,%00000000,%00000000,%0#0#0#0#
34844 b %0000#00,%0#000000,%00000000,%00000000,%00000000,%00000000,%000#000#
34845 b #####,#####,#####,#####,#####,#####,#####
34846 b %00000000,%00000000,%00000000,%00000000,%00000000,%00000000,%000#000#
34847 b %00000000,%0#####0,%000#####,%#0000###,%###0000#,%#####000,%000#000#
34848 b %00000000,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#000#
34849
34850      b $80 +7*8
34851
34852 b %00###00,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#000#
34853 b %00000000,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#000#
34854 b %00000000,%0#####0,%000#####,%#0000###,%###0000#,%#####000,%000#0#0#
34855 b %00000000,%00000000,%00000000,%00000000,%00000000,%00000000,%000#0#0#
34856 b %00000000,%00000000,%00000000,%00000000,%00000000,%00000000,%000#0#0#
34857 b %00000000,%0#####0,%000#####,%#0000###,%###0000#,%#####000,%000#0#0#
34858 b %00000000,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#0#0#
34859 b %00###00,%0#0000#0,%000#0000,%#0000#00,%00#0000#,%0000#000,%000#0#0#
34860
34861      b $80 +7*8
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
34897                   beq      :1                ; => Für b $2c...
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.

```

```

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

```



```

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

```

```

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.

```

```

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

```

```

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
35359 ; TODO:
35360 ; Wozu VLIR-Modul aktivieren?
35361 if FALSE
35362         lda    #1
35363         sta    V04B0           ;VLIR-Modul aktiv.
35364 endif
35365
35366 ; TODO:
35367 ; Wozu Echtzeit-Uhr anhalten?
35368 if FALSE
35369         jsr    z7814           ;C64-Uhr anhalten.
35370 endif
35371
35372         jsr    MouseOff       ;Maus abschalten.
35373
35374 ; Hinweis:
35375 ; Der erste Prozess hat die Nr.0!
35376 ; TopDesk nutzt hier die Nr.1=Fehler!
35377 ;
35378         ldx    #$01
35379         ldx    #0             ;Uhr-Prozess
35380         jsr    BlockProcess   ;anhalten.
35381
35382         lda    keyVector +0
35383         sta    v76AD +0
35384         lda    keyVector +1
35385         sta    v76AD +1
35386
35387         lda    #< r76AF
35388         sta    keyVector +0
35389         lda    #> r76AF
35390         sta    keyVector +1
35391
35392         ldx    #$00           ;Cursor-Position
35393         stx    v78C9           ;auf Anfang setzen.
35394
35395         jsr    z77E1           ;max. Taste/Eingabe.
35396
35397         lda    #REV_ON
35398         sta    V6167           ;Cursor ein.
35399
35400         lda    #$00           ;Sekunden löschen.
35401         sta    V61E8
35402
35403         jmp    S6118           ;Datum/Uhr anzeigen.
35404
35405 ; Puffer für Tastatur-Vektor.
35406 :v76AD w $0000
35407 ; Tastatur abfragen.
35408 :r76AF lda    keyData
35409         cmp    v78D7
35410         beq    :1
35411         bcs    :2
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              ;Beenden?
35451          bne     :done                ; => Nein, Ende...
35452
35453          jsr     z77BB              ;Cursor ausschalten.
35454
35455          jsr     z785A                ;Eingaben überprüfen.
35456          beq     :31                ; => OK, weiter...
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

```

```

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

```

```

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.

```



```

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

```

```

35633
35634          lda    #$00                ;AM/PM-Flag löschen.
35635          sta    r15L
35636
35637          lda    hour
35638          cmp    #12
35639          bcc    :1
35640
35641          lda    %%10000000          ;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 ?

```

```

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

```

```

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

```

```

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

```

```

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

```



```

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

```

```

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

```

```

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

```

```
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 ;
36218 ; SourceCode: lnk.DTopDesk.DE.s
36219 ;
36220
36221 ;--- GEOS-Dateiname/Info.
36222         n "DTOPDESK64"
36223
36224         h "A fork of TopDesk64..."
36225         h "C64 and MP64/GDOS64 only!"
36226
36227         m
36228
36229         - "obj.#00"
36230         - "obj.#01"
36231         - "obj.#02"
36232         - "obj.#03"
36233         - "obj.#04"
36234         - "obj.#05"
36235         - "obj.#06"
36236         - "obj.#07"
36237         - "obj.#08"
36238         - "obj.#09"
36239         - "obj.#10"
36240         - "obj.#11"
36241         - "obj.#12"
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 ;
36276 ; SourceCode: ass.DTopDesk.s
36277 ;
36278
36279 *** Systemlabels.
36280 if .p
36281             t "TopSym"
36282 endif
36283
36284             n "ass.DTopDesk"
36285             c "ass.SysFile V1.0"
36286             t "opt.Author"
36287             f 4 ;SYSTEM
36288             z $40 ;GEOS64 oder GEOS128 40/80 Zeichen
36289
36290             o $4000
36291
36292             h "* AutoAssembler system file."
36293             h "Create DTopDesk DE/EN..."
36294
36295 ;--- DTopDesk DE/EN erstellen.
36296 :DTOPDESK      b $f0,"ass.DTopDeskDE.s",$00
36297                b $f0,"ass.DTopDeskEN.s",$00
36298                b $ff
36299
36300 ;--- Erlaubte Dateigröße: 8192 Bytes.
36301 ;      Datenspeicher von $4000-$5fff.
36302                g $5fff
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 ;
36341 ; SourceCode: ass.DTopDeskEN.s
36342 ;
36343
36344 ;*** Systemlabels.
36345 if .p
36346             t "TopSym"
36347 endif
36348
36349             n "ass.DTopDeskEN"
36350             c "ass.SysFile V1.0"
36351             t "opt.Author"
36352             f 4 ;SYSTEM
36353             z $40 ;GEOS64 oder GEOS128 40/80 Zeichen
36354
36355             o $4000
36356
36357             h "* AutoAssembler Systemdatei."
36358             h "Erstellt DTopDesk..."
36359
36360 ;--- DTopDesk löschen.
36361 :DTPDESK_DEL    b $f0,"DTopDeskI18N.EN",$00
36362                t "a01.DTopDeskEN"
36363                t "a01.DTopDesk"
36364
36365 ;--- DTPDESK linken.
36366                b $f5
36367                b $f0,"lnk.DTopDesk.EN",$00
36368                b $f4
36369
36370 ;--- Erlaubte Dateigröße: 8192 Bytes.
36371 ;    Datenspeicher von $4000-$5fff.
36372                g $5fff
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 ;
36396 ; SourceCode: a01.DTopDeskDE.s
36397 ;
36398
36399 ;--- DTopDesk löschen.
36400 :DTOPDESK_DEL_DE    b $f1
36401
36402             jsr    OpenDisk
36403
36404             lda    #< :1                ;DTOPDESK löschen.
36405             sta    r0L
36406             lda    #> :1
36407             sta    r0H
36408             jsr    DeleteFile
36409
36410             lda    #< :next
36411             sta    a0L
36412             lda    #> :next
36413             sta    a0H
36414
36415             rts
36416
36417 ::1         b "DTOPDESK64", $00
36418
36419 ::next
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 ;
36450 ; SourceCode: TopSym.s
36451 ;
36452
36453 ; Systemlabels
36454 ; Version 04.07.1989
36455 ;
36456 ; Revision 26.03.2023:
36457 ; APP_LVAR und APP_LRAM ergänzt.
36458 ;
36459 ; Revision 26.12.2022:
36460 ; sysApplData ergänzt.
36461 ;
36462 ; Revision 03.12.2022:
36463 ; sysVersion ergänzt.
36464 :APPLICATION = $06
36465 :APPL_DATA = $07
36466 :APP_LVAR = $0200
36467 :APP_LRAM = $0334
36468 :APP_RAM = $0400
36469 :APP_VAR = $7f40
36470 :ASSEMBLY = $02
36471 :AUTO_EXEC = $0e
36472 :AllocateBlock = $9048
36473 :AppendRecord = $c289
36474 :BACKSPACE = $08
36475 :BACK_SCR_BASE = $6000
36476 :BASIC = $01
36477 :BBMult = $c160
36478 :BMult = $c163
36479 :BOLDON = $18
36480 :BRKVector = $84af
36481 :BitOtherClip = $c2c5
36482 :BitmapClip = $c2aa
36483 :BitmapUp = $c142
36484 :BldGDirEntry = $c1f3
36485 :BlkAlloc = $c1fc
36486 :BlockProcess = $c10c
36487 :BootGEOS = $c000
36488 :CANCEL = $02
36489 :CBM = $05
36490 :COLOR_MATRIX = $8c00
36491 :CONSTRAINED = $40
36492 :CPU_DATA = $01
36493 :CPU_DDR = $00
36494 :CR = $0d
36495 :CRC = $c20e
36496 :CalcBlksFree = $c1db
36497 :CallRoutine = $c1d8
36498 :ChangeDiskDevice = $c2bc
36499 :ChkDkGEOS = $c1de
36500 :ClearMouseMode = $c19c
36501 :ClearRam = $c178
36502 :CloseRecordFile = $c277
```

```
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
```

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

```
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
```



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

```
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
```

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

```
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
```

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

```
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
```

```
36998 :stringY = $84c0
36999 :sysApplData = $8fe8
37000 :sysDBData = $851d
37001 :sysFlgCopy = $c012
37002 :sysRAMFlg = $88c4
37003 :sysVersion = $c011
37004 :turboFlags = $8492
37005 :usedRecords = $8497
37006 :vdcClrMode = $88be
37007 :version = $c00f
37008 :windowBottom = $34
37009 :windowTop = $33
37010 :year = $8516
37011
```

```

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

```

```

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

```

```

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

```

```

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

```

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

