

Hier liegt nun die eigentliche Neuheit dieses Programms. Mit einem Memory-Write-Befehl (Listing) wird in die Speicherstelle 257 (\$0101) der Floppy der ASCII-Code von »A« (65) geschrieben. In dieser Speicherstelle wird bei jedem Initialisieren einer Diskette deren Formatkennzeichen im ASCII-Format abgelegt. Ist nun dieser Inhalt gleich 65, so schreibt die Floppy wieder ohne Probleme auf die Diskette, und genau dies wird in dem Programm ausgenutzt.

Nun einige Bemerkungen zur Bedienung des Programms: Nach dem Starten mit »RUN« wird das Formatkennzeichen der Diskette eingelesen und ausgedruckt. Daraufhin gibt man das Format-Zeichen ein, das die Diskette erhalten soll. Drückt man nur RETURN, so wird das vorgegebene Zeichen übernommen und es kommt immer zu einer Änderung des Formatkennzeichens. Zum Schluß wird dann der aktuelle Zustand der Diskette angezeigt.

Achtung! Unmittelbar nach dem Programmlauf ist der Schreibschutz nicht aktiv, da in der Speicherstelle 257 noch immer 65 steht. Erst nach dem Initialisieren einer geschützten Diskette ist dieser Schutz auch wirksam. Auch gegen versehentliches Formatieren, bei dem ja alle Programme gelöscht werden, schützt dieses Programm nicht, da bei diesem Befehl sicherlicherweise kein Formatkennzeichen gelesen wird.

In allen anderen Fällen ist der Schutz jedoch voll wirksam und macht Schreibschutz-Etiketten überflüssig.

(Manfred Lins/ev)

Disk-Tester

Mit diesem Programm lassen sich Disketten schneller, komfortabler und schonender auf defekte Sektoren untersuchen, als das mit dem entsprechenden Programm auf der Test/De-modiskette der Fall ist.

Das Programm ist insbesondere bei Verwendung älterer Disketten oder bei Nutzung der Disketten-Rückseite zu empfehlen. Es überprüft in knapp 10 Minuten die gesamte Diskette auf schadhafte Stellen und markiert die eventuell gefundenen unbrauchbaren Sektoren als belegt, so daß sie von der 1541 nicht mehr zur Datenspeicherung verwendet werden. So ist es möglich, auch teilweise beschädigte Disketten noch zur Datenspeicherung heranzuziehen, ohne daß es zu WRITE- oder READ-ERRORS kommt.

Programmbeschreibung:

Nach dem Starten des Programms (siehe Listing) muß man entscheiden, ob man nur die vollen Blocks (durch Lesen) oder alle Blocks (durch Lesen und Beschreiben) testen will, oder ob man die schon bei vorherigen Tests als fehlerhaft erkannten Blocks nach einem VALIDATE wieder belegen will. Danach wird gefragt, ob der Befehl »VALIDATE« an die Floppy gesendet werden soll. Dies ist nur bei Disketten sinnvoll, auf denen die Daten öfters geändert werden und die keine relativen Dateien, ISAM-Dateien oder ähnliche, im Direktzugriff arbeitenden Speicherformate, enthalten. Bei Disketten mit den oben genannten Dateien, von denen man nicht sicher weiß, ob die beschriebenen Blöcke auch in der BAM (Block Availability Map, Sektorbelegungstabelle) als belegt gekennzeichnet sind, sollte die Option »Leere Blocks testen« auf keinen Fall angewandt werden, da sonst Daten verlorengehen können.

Es folgt nach kurzer Lese- und Decodierzeit die symbolische Belegung der Diskette auf dem Bildschirm, auf der die beleg-

ten Sektoren als »-« und die nicht belegten als »*« gekennzeichnet sind. Wie kommt es dazu? Nun, jede Spur belegt in der BAM (Spur 18, Block 0, Positionen 4-143) 4 Byte. Zuerst kommt die Anzahl der freien Blöcke auf der Spur, im Programm bezeichnet mit FR\$. Dann kommt nacheinander die Belegung der Sektoren 7-0, 15-8 und 23-16. Ein 1-Bit kennzeichnet dabei einen freien Block, ein 0-Bit einen besetzten. Dies wird nun in der Routine »Analysieren der BAM« aufgelöst und in B%(T,S) festgehalten, wobei T die Spur und S den Sektor enthält. Die darauffolgende Testroutine beschränkt sich jetzt nur darauf, die Spuren und Sektoren einzeln durchzuzählen und in Abhängigkeit von B%(T,S) und der Variable TS (0, wenn alle, 1, wenn nur volle Blocks getestet werden sollen) entweder den Block zu lesen, oder ihn mit dem einmal aufgefüllten Puffer zu beschreiben. Danach wird der Fehlerkanal auf die Fehlernummer abgefragt, da dies schneller geht, als immer die ganze Fehlermeldung zu holen. Tritt ein Fehler auf, so wird dieser angezeigt und auf dem Monitor als roter Punkt in der BAM markiert. Gleichzeitig wird die Variable FE erhöht. Tritt kein Fehler auf, so wird die Kennzeichnung für den gerade getesteten Block auf dem Monitor gelöscht. Sind alle Blocks getestet, werden die eventuell vorhandenen defekten Blocks auf der Diskette belegt und in dem Format »Spur,chr\$(13),Block,chr\$(13)« in dem USR-File »DISKFEHLER« gespeichert. Das erste Byte entspricht in diesem Fall der Anzahl aller gefundenen Fehler.

Die Routine »Kaputte Blocks belegen« hat nun nichts weiter zu tun, als dieses File zu lesen und die Blocks zu belegen.

Änderungen für andere Computer

Für einen der »großen« CBMs muß die Zeile 135 und sämtliche POKEs in den Zeilen 150 und 470 gelöscht werden, da sie keine Farbdarstellungen besitzen. Die Zeile 140 muß so geändert werden:

```
140 BI=32768
```

Für einen VC 20 mit 40-Zeichen-Bildschirm müssen nur die farbgebenden POKEs in Zeile 150, die Strings in Zeile 135 und die Variablen in Zeile 140 angepaßt werden. Für die Besitzer eines VC 20 mit 22-Zeichen-Bildschirm sind größere Änderungen erforderlich, da die BAM nicht angezeigt werden kann und sie sich somit mit einer numerischen Darstellung des Testablaufs zufrieden geben müssen. Es werden die freien und die belegten Blocks sowie der zuletzt getestete Sektor angezeigt. Die Zeilen 140, 470 und 635 ff. müssen vollständig gelöscht werden. Folgende Zeilen müssen geändert werden:

```
355 PRINTL$T$" BESETZTE BLOCKS:" O$683-FR:-
PRINTT$" FREIE BLOCKS": "O$FR:D$D$
450 INPUT #15,F:IFF < 20GOTO480
480 PRINTG$" T: " T$Z$S;Z$" "U
```

Zehn Minuten für das Überprüfen einer Diskette mögen zwar immer noch recht lang anmuten, wenn man jedoch bedenkt, wieviel Ärger (READ-ERROR, WRITE-ERROR) ein fehlerhafter Sektor verursachen kann, dann rentiert es sich bestimmt, diese Zeit zu investieren. (Ulrich Langler/ev)

```
100 REM*DISKTESTER                                <068>
105 REM*ULI LANG*                                  <153>
110 REM*TALSTR.10*                                  <167>
115 REM*8609 BISCHBERG*                             <015>
120 REM*TEL.0951/67389*                             <141>
125 :                                                <101>
130 L$=" {CLR}":D$=" {DOWN}":U$=" {UP}":Z$=" {
LEFT}":REM*LOESCHEN,CRSR DOWN,CRSR UP,
CRSR BACK*                                <215>
```

Listing »Disk-Tester«. Bitte beachten Sie die Eingabehinweise auf Seite 6.

```

135 G$=" {GREEN}":O$=" {ORANGE}":B$=" {LIG.BL
    UE}":T$=" {CYAN}":R$=" {LIG.RED}":REM*GR
    UEN,ORANGE,HELLBLAU,TUERKIS,ROT* <199>
140 FARBSPEICHER=55296:BILDSCHIRMSPEICHER=
    1024 <242>
145 DIM T%(100),S%(100):REM*FEHLERHAFT BL
    OCKS* <033>
150 PRINT L$G$D$SPC(15)"DISKTESTER"D$:POKE
    53280,0:POKE 53281,0:REM*SCWARZ* <079>
155 PRINT O$D$"<"B$"A"O$"> ALLE(4SPACE)BLO
    CKS TESTEN" <221>
160 PRINT D$"<"B$"B"O$"> BELEGTE BLOCKS TE
    STEN" <237>
165 PRINT D$"<"B$"K"O$"> KAPUTTE BLOCKS BE
    LEGEN" <027>
170 GET A$ <124>
175 IF A$="A"THEN TS=1:GOTO 210 <250>
180 IF A$="B"THEN TS=0:GOTO 210 <096>
185 IF A$="K"THEN 575 <007>
190 GOTO 170 <230>
195 REM***** <076>
200 REM*DISKETTE TESTEN* <025>
205 REM***** <086>
210 DIM B%(35,23),F$(8):K$=",";N$=CHR$(0):
    FR=0:FE=0:FT=0 <085>
215 FOR I=0 TO 8:READ F$(I):NEXT:REM*FEHL
    ERMELDUNGEN* <102>
220 GOSUB 610:REM*AUF SPACE WARTEN* <179>
225 OPEN 15,8,15,"I" <024>
230 PRINT L$"VALIDATE?"B$"J/N" <214>
235 GET A$ <189>
240 IF A$="N"THEN 255 <164>
245 IF A$="J"THEN PRINT L$T$D$"VALIDATING.
    ..":PRINT#15,"V":GOTO 255 <060>
250 GOTO 235 <138>
255 OPEN 3,8,3,"#":IF TS THEN OPEN 5,8,5,"
    #" <216>
260 PRINT L$T$"BAM WIRD GELESEN(9SPACE)"U$ <248>
265 PRINT#15,"U1:"3;0;18;0:GET#3,A$,A$,A$ <240>
270 IF A$<>CHR$(65)THEN PRINT D$"KEIN DOS
    2.6 ODER 2.5!":GOTO 550 <096>
275 GET#3,A$ <206>
280 REM***** <167>
285 REM*ANALYSIEREN DER BAM * <230>
287 REM*AUFGELOEST IN B%(T,S)* <119>
290 REM***** <177>
295 FOR T=1 TO 35 <025>
300 REM*FREIE BLOCKS,SEKTOR 7-0,SEKTOR 1
    5-8,SEKTOR 23-16* <072>
305 GET#3,FR$:FR=FR+ASC(FR$+N$):S=0 <226>
310 IF FR THEN 320:REM*NICHT ALLE BLOCKS
    BELEGT* <058>
315 FOR S=0 TO 23:B%(T,S)=0:NEXT S:GOTO
    345 <091>
320 A=1:GET#3,BA$:BA=ASC(BA$+N$) <086>
325 FOR S=S TO S+7 <102>
330 B%(T,S)=(BA AND A)/A:A=A+A:FT=FT+B
    %(T,S) <015>
335 NEXT S <245>
340 IF S<23 THEN 320 <149>
345 NEXT T <007>
350 IF FR<>FT THEN PRINT L$R$TAB(14)"!BAM-
    FEHLER!":GOTO 550 <183>
355 GOSUB 635 <037>
360 REM***** <247>
365 REM*START DER TESTROUTINE* <078>
370 REM***** <001>
375 IF TS=0 THEN FR=683-FR:REM*VOLLE BLOCK
    S* <228>
380 IF TS=0 THEN 400 <012>
385 REM*170=%10101010 , 85=%01010101* <012>
390 PRINT#15,"B-P:"5;1:FOR I=1 TO 128:PRIN
    T#5,CHR$(170)CHR$(85);:NEXT <148>
395 REM*PUFFER BRAUCHT NUR EINMAL GEFUELLT
    ZU WERDEN* <074>
400 FOR T=1 TO 35:REM*SPURZAEHLER* <138>
405 IF T<18 THEN MS=20:GOTO 425 <089>
410 IF T<25 THEN MS=18:GOTO 425 <053>
415 IF T<31 THEN MS=17:GOTO 425 <073>
420 MS=16 <083>
425 FOR S=0 TO MS:REM*BLOCKZAEHLER* <161>
430 ON TS*2+B%(T,S)GOTO 485,435,440 <140>
435 PRINT#15,"U1:"3;0;T;S:GOTO 445:REM
    *BLOCK LESEN* <031>
440 PRINT#15,"B-W:"5;0;T;S:REM*BUFFER
    AUF DISK SCHREIBEN* <197>
445 AO=40*(20-S)+T+3:REM*ADRESSOFFSET* <156>
450 INPUT#15,F:IF F<20 THEN P=32:GOTO
    480 <224>
455 IF F<29 THEN F=F*(F-20):GOTO 465 <078>
460 F$="" <255>
465 PRINT R$F;F$ T:"T;Z$" S:"S;Z$" "U
    $ <224>
470 POKE FA+AO,10:P=81+6*B%(T,S):REM*R
    OTES"0,X" <123>
475 T%(FE)=T:S%(FE)=S:FE=FE+B%(T,S) <040>
480 POKE BI+AO,P <254>
485 NEXT S <139>
490 NEXT T:CLOSE 5:CLOSE 3 <191>
495 IF FE=0 THEN PRINT "{6SPACE}!KEINE ZU B
    ELENDEN FEHLER!";:GOTO 550 <007>
500 REM***** <002>
505 REM*BELEGEN DER KAPUTTEN BLOCKS* <069>
510 REM***** <012>
515 PRINT L$T$" TEST ZUENDE, BELEGEN UND K
    ENNZEICHNEN" <220>
520 PRINT TAB(8)"DER FEHLERHAFTEN BLOCKS" <170>
525 FOR I=0 TO FE-1 <062>
530 PRINT#15,"B-A:"0;T%(I);S%(I):REM*IN
    BAM BELEGEN* <216>
535 NEXT <037>
540 OPEN 3,8,3,"@:.DISKFEHLER,U,W":PRINT#3
    ,CHR$(FE-1); <099>
545 FOR I=0 TO FE-1:PRINT#15,T%(I)CHR$(13)
    S%(I):NEXT <120>
550 CLOSE 3:CLOSE 15:END <029>
555 : <023>
560 REM***** <106>
565 REM*WIEDERBELEGUNG NACH VALIDATE* <221>
570 REM***** <116>
575 GOSUB 610 <067>
580 PRINT L$T$"WIEDERBELEGEN DER SCHADHAFT
    EN BLOCKS" <143>
585 OPEN 15,8,15,"I":OPEN 5,8,5,".DISKFEHL
    ER,U,R" <190>
590 GET#5,A$:FE=ASC(A$) <151>
595 FOR I=0 TO FE:INPUT#5,T%(I),S%(I):PRIN
    T R$T:"T;T%(I),"S:"S%(I):NEXT:CLOSE 5 <124>
600 FOR I=0 TO FE:PRINT#15,"B-A:"0;T%(I);S
    %(I):NEXT:CLOSE 15:END <087>
605 : <073>
610 PRINT L$TAB(9)O$"TESTDISKETTE {2SPACE}E
    INLEGEN":PRINT TAB(17)"<"B$"SPACE"O$">
    " <021>
615 GET A$:IF A$<>" THEN 615 <216>
620 RETURN <170>
625 : <093>
630 REM*BAM DARSTELLEN* <210>
635 PRINT L$; <240>
640 FOR S=20 TO 0 STEP-1 <173>
645 PRINT O$S;TAB(4); <038>
650 FOR T=1 TO 35 <126>
655 IF T<18 THEN MS=20:GOTO 675 <091>
660 IF T<25 THEN MS=18:GOTO 675 <055>
665 IF T<31 THEN MS=17:GOTO 675 <075>
670 MS=16 <079>
675 IF S>MS THEN 690 <066>
680 IF B%(T,S)=0 THEN PRINT B$"-";:GOT
    O 690 <183>
685 PRINT G$"*"; <173>
690 NEXT T:PRINT <056>
695 NEXT S:PRINT TAB(4); <214>
700 FOR T=1 TO 35:PRINT RIGHT$(STR$(T),1);
    :NEXT <078>
705 PRINT:RETURN <228>
710 REM*FEHLERMELDUNGEN* <008>
715 DATA"KEIN HEADER {6SPACE}","KEIN SYNC C
    HAR {3SPACE}","KEIN BLOCK VORH. " <119>
720 DATA"PRUEFSUMME DATEN ","BITMUSTERFEHL
    ER {2SPACE}","SCHREIBFEHLER {4SPACE}" <254>
725 DATA"SCHREIBSCHUTZ {4SPACE}","PRUEFSUMM
    E HEADER","BLOCK ZU LANG {4SPACE}" <253>

```

Listing »Disk-Tester« (Schluß)