```
3110 L=19: REM  ZAHL LEERE DIR-BLOECKE      <082>
3120 DIM Z(25): REM  ZUSTAND DER BLOCKS     <004>
3130 FOR I=1 TO 3: REM  BYTES SPUR 18       <154>
3140 : GET# 2, W$: REM  BELEGUNGSCODE       <176>
3150 : W=ASC(W$+CHR$(0))                    <011>
3160 : FOR J=1 TO 8: REM  DUALSTELLEN       <207>
3170 :  W=W/2                               <181>
3180 :  IF W=INT(W) THEN Z(B)=1             <203>
3190 :  IF Z(B)=0 AND A$="N" THEN A$="": Z
     (B)=1: REM  EINEN BLOCK FREIHALTEN     <092>
3195 :  IF Z(B)=1 AND B<19 THEN L=L-1       <093>
3200 :  B=B+1                               <071>
3085 :  W=INT(W)                            <134>
3220 : NEXT J                               <076>
3230 NEXT I                                 <010>
3310 IF L<1 THEN 8000                       <076>
3390 PRINT: PRINT: PRINT "ES WERDEN NUN";L
     ;" BLOECKE FREIGESTELLT !": PRINT      <233>
3990 :                                      <156>
4000 REM  ****************************      <202>
4010 REM    FUELLEN DER DIR.- BLOECKE       <251>
4020 REM  ****************************      <222>
4021 :                                      <187>
4025 B=0: REM  BLOCKZAEHLER, SIEHE OBEN     <163>
4027 FOR E=0 TO E(B)-1: REM EINTR/BLOCK     <038>
4030 AT=T(B): REM  ALTER DIR-TRACK          <099>
4035 AS=S(B): NS=AS: REM  ALTER SECTOR      <166>
4040 IF AT=0 AND AS=0 THEN 7050             <004>
4050 I=0                                    <173>
4060 Z=E*32+3: REM  STELLE DES ZEIGERS      <170>
4100 R=0: REM  MARKE DIR.-BLOCK AENDERN     <117>
4150 PRINT# 1, "U1:"2;0;AT;AS: REM  ALT     <111>
4160 PRINT# 1, "B-P:"2,Z: REM  ZEIGER       <151>
4170 GET# 2, T$,S$: REM  ZEIGER FORTS.      <185>
4180 T=ASC(T$+CHR$(0))                      <003>
4190 S=ASC(S$+CHR$(0))                      <200>
4191 IF T=18 THEN 4670: REM  BEARBEITET     <245>
4192 IF T=0 OR T>35 OR S>20 THEN Z=1: GOTO
      4520: REM  FILEENDE                   <211>
4200 FOR I=0 TO 18 STEP 10: REM  SUCHE NAC
     H FREIEM SECTOR IM SECTORABSTAND       <037>
4210 : IF Z(I)=0 THEN 4500: REM  LEER       <196>
4212 : IF I=9 THEN GOTO 4230                <056>
4215 : IF I>8 THEN I=I-9: GOTO 4210         <096>
4220 NEXT I                                 <240>
4230 Z=-1: REM  MARKE: SPUR 18 VOLL         <231>
4300 GOTO 4520: REM  RUECK OHNE AENDER.     <214>
4500 PRINT# 1, "B-P:"2,Z: REM  ZEIGER       <237>
4510 PRINT# 2, CHR$(18);CHR$(I);: REM  ZE
     IGER VERBIEGEN                         <111>
4515 IF AT=18 AND AS=NS THEN R=1: REM  IN
      DEN BLOCK, AUS DEM GELESEN WURDE      <181>
4520 PRINT "{6SPACE}BLOCK";RIGHT$("{2SPACE
     }"+STR$(AT),3);","RIGHT$("{2SPACE}"+S
     TR$(AS),3);                            <252>
4525 IF R=1 THEN PRINT"{2SPACE}ZEIGER";E+1
     ;"GEAENDERT": GOTO 4570                <023>
4530 PRINT "{2SPACE}-->{2SPACE}18,";RIGHT$
     ("{2SPACE}"+STR$(NS),3)                <128>
4550 PRINT# 1, "B-A:"0,18,NS:REM IN BAM     <251>
4560 PRINT# 1, "B-F:"0,AT,AS: REM  ALTEN B
     LOCK FREIGEBEN                         <209>
4570 Z(I)=1: REM BLOCK IN LISTE BELEGEN     <254>
4580 PRINT# 1, "U2:"2;0;18;NS:REM RUEC      <235>
4650 AT=T: AS=S: NS=I: REM  NORM. BLOCK     <035>
4655 IF Z=-1 THEN GOTO 9000: REM  OK        <109>
4660 IF Z<>1 THEN Z=0: GOTO 4100            <234>
4670 NEXT E: REM  NEUER DIR.-EINTRAG        <120>
4680 B=B+1: GOTO 4027: REM  WEITER MIT NAE
     CHSTEM DIRECTORYBLOCK                  <087>
6990 :                                      <108>
7000 REM  ****************************      <154>
7010 REM          FEHLERMELDUNGEN           <130>
7020 REM  ****************************      <174>
7030 :                                      <148>
7050 PRINT: PRINT "ES WAREN NICHT GENUG BE
     LEGTE BLOCKS DA."                      <036>
7060 PRINT " SPUR 18 IST IMMERNOCH TEILWEI
     SE FREI!"                              <099>
7090 GOTO 9000                              <236>
7800 PRINT:PRINT:PRINT"DIE DISKETTE IST LE
     ER!"                                   <146>
7810 GOTO 9000                              <194>
8000 PRINT "TUT MIR LEID, DA IST NICHTS ZU
      MACHEN."                              <162>
8010 PRINT "DIE GESAMTE SPUR 18 IST VOLL B
     ELEGT !"                               <160>
9000 PRINT                                  <212>
9010 CLOSE 1: CLOSE 2: END                  <099>
```

Listing »Disk Füller« (Schluß)

Schreibschutz per Software

Eleganter als das Hantieren mit Klebestreifen ist ein Disketten-Schreibschutz per Software. Hier ist ein Programm, das einen solchen Schreibschutz nicht nur anbringen, sondern ihn auch wieder entfernen kann.

Die Funktion dieses Programms basiert darauf, das Formatkennzeichen (»A«) auf der Diskette zu ändern (hier in »X«). Versucht man nun, Daten auf die Diskette zu schreiben, so reagiert das Floppy-Laufwerk mit der Fehlermeldung »CBM DOS V2.6 1541«. Das Lesen ist jedoch weiterhin problemlos möglich. Damit erspart man sich die zuweilen umständliche Verwendung von Schreibschutz-Aufklebern auf der Diskette. Außerdem ist dieser softwaremäßig realisierte Schreibschutz um einiges sicherer, denn er kann nicht durch Leichtsinn oder Versehen entfernt werden.

Eine Schwierigkeit ergibt sich aber, wenn das neue Formatkennzeichen wieder in »A« geändert werden soll, um zum Beispiel noch ein Programm auf der Diskette zu speichern. Da ja jegliches Schreiben unmöglich ist, hat man selbst mit den Direktzugriffs-Befehlen keine Möglichkeit mehr, wieder ein »A« auf die Diskette zu bringen.

```
1 REM ********************************      <051>
2 REM **                            **      <002>
3 REM **       SCHREIBSCHUTZ        **      <255>
4 REM **                            **      <004>
5 REM **     '85 BY                 **      <128>
6 REM ** MANFRED LINS               **      <031>
7 REM ** REITACKER 1                **      <221>
8 REM ** 6492 SINNTAL-ZUENTERSBACH  **      <140>
9 REM ********************************      <059>
10 PRINT CHR$(147):REM CLEAR/HOME           <146>
20 X=7:Y=2:GOSUB 510                        <017>
30 PRINT"D I S K{2SPACE}-{2SPACE}S C H U T
    Z"                                      <035>
40 X=2:Y=6:GOSUB 510                        <217>
50 PRINT"ALTES FORMATKENNZEICHEN :";        <075>
60 OPEN 1,8,15,"I":OPEN 2,8,2,"#"           <051>
70 PRINT#1,"U1 2 0 18 0"                    <130>
80 PRINT#1,"B-P 2 2"                        <244>
90 GET#2,AF$: REM ALTES FORMATKENNZ.        <169>
100 PRINT AF$                               <188>
110 X=2:Y=9:GOSUB 510                       <034>
120 PRINT"WIRD GEAENDERT IN{3SPACE}";       <012>
130 IF AF$="A"THEN PRINT"X":GOTO 150        <096>
140 PRINT"A"                                <091>
150 X=20:Y=9:GOSUB 510                      <236>
160 INPUT NF$                               <016>
170 PRINT#1,"M-W"CHR$(1)CHR$(1)CHR$(1)CHR$
    (65)                                    <032>
180 PRINT#1,"B-P 2 2"                       <088>
190 PRINT#2,NF$:REM NEUES FORMATKENNZ.      <161>
200 PRINT#1,"U2 2 0 18 0"                   <133>
210 CLOSE 2:CLOSE 1                         <232>
220 IF NF$="A"THEN D$="{RVSON,2SPACE}KEIN
    SCHREIBSCHUTZ !{2SPACE}":GOTO 240       <001>
230 D$="{RVSON,2SPACE}SCHREIBSCHUTZ AKTIVI
    ERT !{2SPACE}"                          <022>
240 X=6:Y=14:GOSUB 510                      <177>
250 PRINT D$                                <196>
260 END                                     <008>
500 REM ROUTINE ZUM CURSORSETZEN            <250>
510 POKE 211,X:POKE 214,Y:SYS 58640:RETURN  <011>
520 REM UNTERPROGRAMMENDE                   <169>
```

Listing »Schreibschutz«. Bitte beachten Sie die Eingabehinweise auf Seite 6.

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 logischerweise 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/Demodiskette 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 belegten 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.