

Checksummer — keine Fehler mehr beim Abtippen von Listings

Das Programm Checksummer 64 ist für all die Leute gedacht, die sich manchmal vor Verzweiflung »die Haare raufen« könnten. Da sitzt man mehrere Stunden, um ein gutes Programm aus dem 64'er Magazin abzutippen, und dann: ein Fehler in der DATA-Zeile oder ein falscher Buchstabe... und schon geht die Fehlersuche los. Hier soll der Checksummer 64 weiterhelfen.

Der Checksummer 64 ist ein kleines Maschinenprogramm, das, wenn es aktiviert ist, Sie sofort davon unterrichtet, ob Sie die jeweilige Programmzeile korrekt eingegeben haben.

1. Tippen Sie den Basic-Lader sorgfältig ein. Es gibt zwei Versionen: eine für den Commodore 64 und eine für den VC 20.

2. Bevor Sie »RUN« eingeben, speichern Sie den Basic-Lader bitte erst ab, denn wenn Sie zum Beispiel einen Fehler bei den eingetippten POKE-Anweisungen gemacht haben, ist es möglich, daß der Rechner aussteigt. Heben Sie sich den abgespeicherten Checksummer 64 auf — Sie werden ihn immer wieder brauchen, wenn Sie ein Basic-Programm aus dem 64'er Sonderheft oder in Zukunft aus dem 64'er Magazin eintippen wollen.

3. Der Checksummer 64 überprüft sich selbst. Wenn Sie einen Fehler in den DATAs gemacht haben, listen Sie die fehlerhafte Zeile einfach, korrigieren sie und starten dann das Programm neu.

4. Nach Initialisierung des Maschinenprogramms ist der Checksummer 64 aktiviert. Er steht innerhalb des Betriebssystems und verbraucht kein einziges Byte Speicherplatz. Es sei hier für Interessierte gesagt, daß selbst alle Sprungvektoren unverändert bleiben, das Programm also mit einer Vielzahl von anderen Programmier-Spracherweiterungen wie etwa Exbasic Level II problemlos zusammenarbeitet. Achten Sie aber darauf, daß bestimmte Spracherweiterungen das hinter dem ROM liegende RAM für Hires-Grafiken benutzen. Wird zum Beispiel eine Hires-Grafik von Simons Basic aus angesprochen, so wird der Checksummer 64 zerstört.

5. Wenn Sie den Checksummer 64 zwischenzeitlich nicht benutzen, können Sie ihn jederzeit mit »POKE 1, 55« desakti-

vieren. Auch durch Drücken der Run-Stop- und der Restore-Taste wird der Checksummer 64 deaktiviert. Wollen Sie, daß der Checksummer 64 auch noch nach Drücken dieser Tastenkombination erhalten bleibt, so geben Sie bei aktiviertem Checksummer 64 »POKE 64982, 53« ein. Der Checksummer 64 ist dann nur durch »POKE 1, 55« abschaltbar.

Wollen Sie den Checksummer 64 wieder einschalten, so geben Sie bitte »POKE 1, 53« ein.

Das Maschinenprogramm bleibt solange erhalten, bis der Computer ausgeschaltet, oder wenn von anderen Programmen auf das hinter dem ROM liegende RAM zugegriffen wird.

6. Eine Checksumme wird nur dann ausgegeben, wenn der Commodore 64 (VC 20) eindeutig erkennt, daß Sie eine Zeile bestehend aus der Zeilennummer und zumindestens einem alphanumerischen Zeichen eingegeben haben. Ansonsten reagiert der Commodore 64 normal.

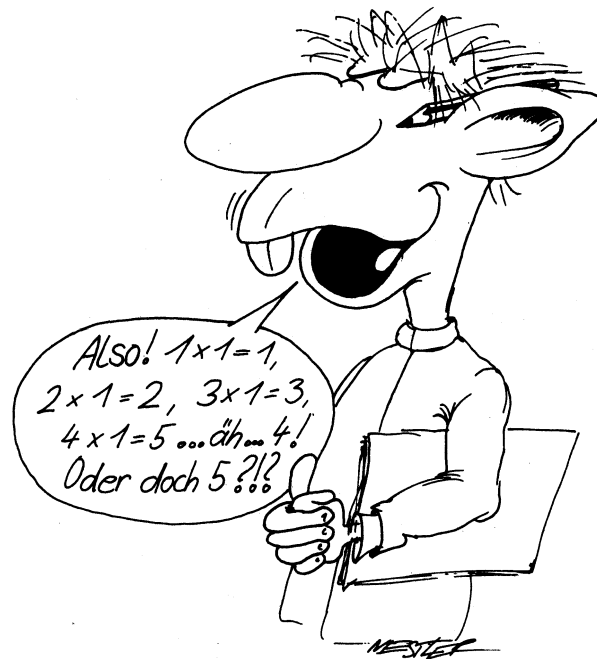
Hinweis: Wenn Sie bei aktiviertem Checksummer 64 ein Programm mit »LOAD« in den Speicher holen, wird auch eine Checksumme ausgegeben. Dies liegt jedoch an rechnerinternen Routinen und hat keine weitere Bedeutung, stellt insbesondere keine Gefahr für das geladene Programm dar, da alle Pointer richtig gesetzt werden.

Nach Eingabe von RUN wird zunächst einmal das ROM in das RAM des Commodore 64 verschoben, wonach der Basic-Interpreter modifiziert wird. Dadurch hat man den Vorteil, trotz einer zusätzlichen Routine das gesamte RAM des Rechners zur Verfügung zu haben. Nach ordnungsgemäsem Ablauf des Programms können Sie sofort mit Eingaben beginnen. Für Maschinensprache-Spezialisten weise ich darauf hin, daß ich ausnutze, daß die Einschaltmeldungen des Rechners nur nach einem Reset generiert wird. Der Textbereich, in dem die Meldung steht, wird von dem erzeugten Maschinenprogramm überschrieben.

Alle veröffentlichten Listings sind mit einer Checksumme versehen, die am Ende jeder Programmzeile steht. Diese Checksumme steht zwischen < und >. Sie wird beim Eintippen des Programms nicht mit eingegeben. Die Zahl zwischen den beiden Zeichen stellt lediglich eine Information für Sie dar. Wenn Sie diese Checksumme dennoch mit eintippen, werden Sie schnell bemerken, daß Sie etwas falsch gemacht haben. Bei aktiviertem Checksummer 64 wird nämlich nach Eingabe einer Basic-Zeile, die mit Return beendet wird, in die linke obere Bildschirmcke die Checksumme eingeblendet, die mit der Summe aus dem veröffentlichten Listing übereinstimmen muß. Ist das nicht der Fall, haben Sie die Zeile anders eingegeben, als sie im Listing dargestellt ist. Vergessen Sie also bitte nicht, daß die am Ende einer Zeile in < und > stehende Prüfsumme nicht mit eingegeben werden darf.

Der Checksummer 64 ist so ausgelegt, daß er abhängig von der Zeilennummer und dem Text der Zeile eine Checksumme ausgibt. Beim Bilden dieser Checksumme werden Spaces (Leertaste) überlesen, was für Sie bedeutet, das es egal ist, wieviel Leerzeichen Sie zwischen den Worten lassen, da Sie für den Programmablauf ohnehin keine Bedeutung haben. Aber manchmal ist das richtige Setzen von Leerzeichen doch wichtig, besonders innerhalb von Strings (Zeichenketten), die gedruckt werden sollen. Seien Sie deshalb besonders genau bei Leerzeichen, die innerhalb von Anführungszeichen stehen, denn meistens ermöglichen nur die richtig gesetzten Spaces eine sinnvolle Textausgabe auf dem Bildschirm.

Beachten Sie auch, daß es durchaus erlaubt ist, Abkürzungen für die Commodore-Befehlsörter zu verwenden. So führt die Eingabe von »?« als Kurzschreibweise für »PRINT« nicht etwa zu einem Checksummer-Fehler, sondern wird korrekt verarbeitet und dementsprechend die Checksumme generiert. Nachdem Sie ein Listing eingegeben haben, sollten Sie es aus Sicherheitsgründen vor dem Starten abspeichern. Sie brau-



chen hierfür jedoch nicht den Checksummer 64 zu deaktivieren.

Hinweise zum Lesen von Listings

Die Listings haben sich ein wenig im Ausdruckformat verändert, um Ihnen das Eingeben von Programmen wesentlich zu erleichtern.

— Cursorsteuerzeichen und andere Steuerzeichen, die schwer zu lesen sind, werden von nun an in Klartext in speziellen Klammern gesetzt.

Tritt mehrmals hintereinander dasselbe Steuerzeichen auf, so wird diese Steuerzeichen-Sequenz zusammengefasst, indem zuerst das Steuerzeichen und dann die Anzahl der Wiederholungen dieses Steuerzeichens in Klartext ausgegeben wird.

— alle Commodore-Grafikzeichen, die über Shift zu erreichen sind, werden nicht mehr als Grafikzeichen, sondern als Klartextzeichen dargestellt. Dabei wird aus dem Zeichen, das Sie auf dem Bildschirm sehen, wenn Sie die Tastenkombination Shift und »A« ansprechen, wieder ein »A«. Um dieses »A« vom normalen »A« unterscheiden zu können, ist es etwas kleiner als das gewöhnliche »A« und ist außerdem mit einem Unterstreichungszeichen versehen. Diese Vereinbarung gilt auch für sämtliche andere Commodore-Grafikzeichen, die über Shift zu erreichen sind.

— entsprechendes gilt für sämtliche Commodore-Grafikzeichen, die über die Commodore-Taste zu erreichen sind. Hier wird jedoch das jeweilige Klartextzeichen nicht unterstrichen, sondern überstrichen.

Erläuterungen zu den Cursorsteuerzeichen

Cursorsteuerzeichen werden, wie schon oben erwähnt, umdefiniert. Sie sehen hier eine Liste der möglichen Ausdrücke, die für ein Cursorsteuerzeichen im Listing auftauchen können. Gleichzeitig ersehen Sie aus der Tabelle, welche Taste beziehungsweise Tastenkombination zu drücken ist, damit dieses Steuerzeichen richtig in Ihr Programm übernommen wird. Beachten Sie, daß Sie die Steuer-codes nur dann als reverses Zeichen sehen können, wenn der Rechner im »Quote-Modus« arbeitet, das heißt er sich im Gänsefüßchenmodus befindet.

Wenn Sie lesen ! drücken Sie

CTRL steht für Control-Taste, so bedeutet[CTRL-A], daß Sie die Control-Taste und die Taste »A« drücken müssen. Im folgenden steht:

[down]	! Taste neben rechtem Shift, Cursor unten
[up]	! Shift-Taste & Taste neben rechtem Shift, Cursor hoch
[clear]	! Shift-Taste & 2. Taste ganz rechts oben
[inst]	! Shift-Taste & Taste ganz rechts oben
[home]	! 2. Taste von ganz rechts oben
[del]	! Taste ganz rechts oben
[right]	! Taste ganz rechts unten
[left]	! Shift-Taste & Taste unten rechts
[space]	! Leertaste
[f1]	! grauer Tastenblock rechts
[f3]	! grauer Tastenblock rechts
[f5]	! grauer Tastenblock rechts
[f7]	! grauer Tastenblock rechts
[f2]	! grauer Tastenblock rechts & Shift
[f4]	! grauer Tastenblock rechts & Shift
[f6]	! grauer Tastenblock rechts & Shift
[f8]	! grauer Tastenblock rechts & Shift
[return]	! Shift-Taste & Return
[black]	! Control-Taste & 1
[white]	! Control-Taste & 2
[red]	! Control-Taste & 3
[cyan]	! Control-Taste & 4
[purple]	! Control-Taste & 5
[green]	! Control-Taste & 6
[blue]	! Control-Taste & 7
[yellow]	! Control-Taste & 8
[rvson]	! Control-Taste & 9
[rvoff]	! Control-Taste & 0
[orange]	! Commodore-Taste & 1
[brown]	! Commodore-Taste & 2
[lig.red]	! Commodore-Taste & 3
[grey 1]	! Commodore-Taste & 4
[grey 2]	! Commodore-Taste & 5
[lig.green]	! Commodore-Taste & 6
[lig.blue]	! Commodore-Taste & 7
[grey 3]	! Commodore-Taste & 8

Wenn Sie sich erst einmal an die in Klartext geschriebenen Steuerzeichen gewöhnt haben, werden Sie den Vorteil dieser Schreibweise erkennen. Der zu dem jeweiligen Steuerzeichen gehörende Klartext ist so verfaßt, daß Sie leicht die Taste beziehungsweise die Tastenkombination finden, die Sie drücken müssen.

Checksummer VC 20

Der Checksummer VC 20 ist im Prinzip genauso aufgebaut wie der Checksummer 64. Da beim VC 20 jedoch nicht die Möglichkeit besteht, das ROM softwaremässig zu modifizieren, mußte ein anderer Weg als beim Commodore 64 gewählt werden, um die Checksumme zu generieren.

In ihrer Funktionsweise unterscheiden sich der Checksummer VC 20 und der Checksummer 64 nicht. Es gelten folgende Sonderregelungen bei der Benutzung des Checksummer VC 20:

— da der Basic-Bereich nicht belegt werden soll, ist das Programm im Kassettenpuffer abgelegt.

— angeschaltet wird der Checksummer VC 20 mit »SYS 955«

— Abschaltung des Checksummer VC 20 wird mit »SYS 58459« vollzogen

ACHTUNG: Nehmen Sie keine Kassetten-Operationen vor, wenn der Checksummer VC 20 eingeschaltet ist. Da das Betriebssystem den Kassettenpuffer mit Daten belegt, kann der Checksummer VC 20 überschrieben werden, was zur Folge hat, daß sich der Rechner bei aktiviertem Checksummer VC 20 »aufhängt«. Wollen Sie deshalb ein Programm auf (von) Kasette abspeichern (laden), so müssen Sie erst den Checksummer VC 20 abschalten (SYS 58459).

Daraufhin kann der Kassettenpuffer mit Daten überschrieben werden, ohne daß der Rechner »aussteigt«.

Als Sicherung wird bei der Initialisierung geprüft, ob das zuletzt angesprochene Peripherie-Gerät der Kassettenrecorder war. Ist das der Fall, so werden die Betriebssystemroutinen LOAD und SAVE für die Benutzung gesperrt. Der Rechner meldet bei Aufruf einer dieser beiden Routinen READY, ohne weitere Aktionen durchzuführen. Diese Sicherung kann man nach der Tipparbeit aufheben, wenn man den Checksummer VC 20 mit SYS 58459 abschaltet. Dadurch wird der Kassettenpuffer für andere Daten freigemacht. Weiterhin wird dann durch gleichzeitiges Drücken der Tasten »Run-Stop & Restore« erreicht, daß die Betriebssystemroutinen LOAD und SAVE wieder eingerichtet werden.

— Bei Benutzung einer Diskettenstation brauchen Sie nicht darauf zu achten, daß bei LOAD beziehungsweise SAVE der Checksummer VC 20 überschrieben wird, da der Kassettenpuffer für die Diskettenstation normalerweise nicht genutzt wird. Deshalb können Sie die beiden Routinen weiterhin normal nutzen, sofern der Rechner bei der Initialisierung des Checksummer VC 20 feststellt, daß das zuletzt angesprochene Peripherie-Gerät nicht der Kassettenrecorder war.

— bedingt durch den anderen Aufbau des Checksummer VC 20 wird anders als beim Checksummer 64 nach der LOAD-Routine keine Checksumme ausgegeben.

— wird eine Zeile gelöscht, also eine Zahl zwischen 0 und 63999 eingegeben, und danach Return gedrückt, so wird eine Checksumme ausgegeben, die aber keine Bedeutung hat.

Viel Spaß beim Eintippen von Programmen mit dem neuen Checksummer!

(F. Lonczewski / gk)

```

10 REM ***** <175>
20 REM * <247>
30 REM * CHECKSUMMER 64 * <162>
40 REM * * <011>
50 REM * 64'ER * <061>
60 REM * * <031>
70 REM * COMMODORE 64 * <056>
80 REM * * <051>
90 REM ***** <255>
100 PRINT "[CLEAR,SPACE13,RVSON]CHECKSUMMER
[SPACE]64[RVOFF]" <025>
110 PRINT <007>
120 PRINT "[DOWN2,SPACE4]ICH[SPACE]ARBEITE!
[SPACE]BITTE[SPACE]ETWAS[SPACE]GEDULD." <071>
130 FOR I=40960 TO 49151:POKE I,PEEK(I):NEXT
<007>
140 FOR I=57344 TO 65535:POKE I,PEEK(I):NEXT
<025>
150 POKE 1,53:POKE 42289,96:POKE 42290,228 <001>
160 FOR I=58464 TO 58554:GOSUB 220:POKE I,A
<184>
170 PS=PS+A+1:NEXT I <109>
180 IF PS<>11187 THEN PRINT"PRUEFSUMMENFEHLER
[SPACE]!":END <180>
190 PRINT "[DOWN4,SPACE9]CHECKSUMMER[SPACE]
AKTIVIERT." <247>
200 PRINT "[DOWN2]AUSSCHALTEN[SPACE]
:[SPACE]POKE1,55" <050>
210 PRINT "[DOWN]ANSCHALTEN[SPACE2]
:[SPACE]POKE1,53":NEW <171>
220 READ A$:IF LEN(A$)<>2 THEN PRINT"TIPPFEHLER
[SPACE]IN[SPACE]ZEILE"PEEK(63)+PEEK(64)*256
:END <201>
230 A1=ASC(A$):A2=ASC(RIGHT$(A$,1)) <216>
240 IF A1<48 OR A1>57 THEN IF A1<65 OR A1>70 TH
EN 310 <130>
250 IF A2<48 OR A2>57 THEN IF A2<65 OR A2>70 TH
EN 310 <144>
260 IF A1>64 THEN A1=A1-55:GOTO 280 <204>
270 IF A1<58 THEN A1=A1-48 <128>
280 IF A2>64 THEN A2=A2-55:GOTO 300 <220>
290 IF A2<58 THEN A2=A2-48 <151>
300 A=A1*16+A2:RETURN <138>
310 PRINT"UNGUELTIGER[SPACE]HEXCODE[SPACE]IN
[SPACE]ZEILE"PEEK(63)+PEEK(64)*256:END <021>
320 DATA A0,02,A9,00,85,02,B1,5F <098>
330 DATA F0,0F,C9,20,D0,03,C8,D0 <146>
340 DATA F5,18,65,02,85,02,4C,6E <126>
350 DATA E4,C0,04,30,F1,C6,D6,A5 <169>
360 DATA D6,48,A2,03,A9,20,9D,01 <150>
370 DATA 04,BD,B7,E4,20,D2,FF,CA <238>
380 DATA 10,F2,A6,02,A9,00,20,CD <169>
390 DATA BD,A9,3E,20,D2,FF,68,85 <245>
400 DATA D6,20,6C,E5,A9,8D,20,D2 <229>
410 DATA FF,4C,80,A4,5C,48,20,C9 <244>
420 DATA FF,AA,68,90,01,8A,60,09 <234>
430 DATA 3C,12,13 <199>

```

```

10 REM***** <057>
20 REM* * <247>
30 REM* CHECKSUMMER * <056>
40 REM* VERSION VC20 * <044>
50 REM* * <021>
60 REM***** <107>
70 PRINT "[CLEAR,SPACE2,RVSON]CHECKSUMMER[SPACE2]
VC-20[RVOFF]" <185>
80 PRINT <233>
90 PRINT "[DOWN]EINEN[SPACE]MOMENT,[SPACE]
BITTE..." <181>
100 FOR I=827 TO 993:GOSUB 180:POKE I,A <177>
110 PS=PS+A+1:NEXT I <049>
120 IF PS<>20612 THEN PRINT"[DOWN]
PRUEFSUMMENFEHLER[SPACE]!":END <130>
130 SYS 955:PRINT"CHECKSUMMER[SPACE]AKTIVIERT."
<242>
140 PRINT"AN[SPACE]:SYS955" <212>
150 PRINT "[DOWN]AUS:SYS58459,[SPACE]BEI[SPACE]
CAS-[SPACE4]SETTE[SPACE]ZUSAETZLICH[SPACE5]
RUN/STOP[SPACE]&[SPACE]RESTORE" <068>
160 PRINT "[DOWN]BEI[SPACE]AKTIVIERTEM[SPACE]
CHECK-SUMMER[SPACE]KEIN"; <105>
170 PRINT "[SPACE]CASSETTEN-BETRIEB[SPACE](LOAD,
[SPACE]SAVE)[SPACE2]ERLAUBT!":NEW <051>
180 READ A$:IF LEN(A$)<>2 THEN PRINT"TIPPFEHLER
[SPACE]IN[SPACE]ZEILE"PEEK(63)+PEEK(64)*256
:END <161>

```

```

190 A1=ASC(A$):A2=ASC(RIGHT$(A$,1)) <176>
200 IF A1<48 OR A1>57 THEN IF A1<65 OR A1>70 TH <177>
    EN 270 <095>
210 IF A2<48 OR A2>57 THEN IF A2<65 OR A2>70 TH <178>
    EN 270 <109>
220 IF A1>64 THEN A1=A1-55:GOTO 240 <159>
230 IF A1<58 THEN A1=A1+48 <087>
240 IF A2>64 THEN A2=A2-55:GOTO 260 <184>
250 IF A2<58 THEN A2=A2+48 <110>
260 A=A1*16+A2:RETURN <098>
270 PRINT"UNGUELTIGER[SPACE]HEXCODE[SPACE]IN <237>
    [SPACE]ZEILE"PEEK(63)+PEEK(64)*256:END <061>
280 DATA 20,5F,03,86,7A,84,7B,20 <130>
290 DATA 73,00,AA,F0,F3,A2,FF,86 <097>
300 DATA 3A,90,0A,A2,00,86,FF,20 <127>
310 DATA 79,C5,4C,E1,C7,A2,01,86 <199>
320 DATA FF,4C,9C,C4,A6,FF,E0,01

```

```

330 DATA F0,03,4C,60,C5,A0,02,A9 <125>
340 DATA 00,85,FE,B1,5F,F0,0F,C9 <186>
350 DATA 20,D0,03,C8,D0,F5,18,65 <141>
360 DATA FE,85,FE,4C,76,03,C0,04 <193>
370 DATA 30,F1,C6,D6,A5,D6,48,A2 <198>
380 DATA 03,A9,20,9D,01,04,BD,B7 <180>
390 DATA 03,20,D2,FF,CA,10,F2,A6 <217>
400 DATA FE,A9,00,20,CD,DD,A9,3E <016>
410 DATA 20,D2,FF,68,85,D6,20,87 <220>
420 DATA E5,A9,8D,20,D2,FF,A2,00 <003>
430 DATA 86,FF,F0,AE,09,3C,12,13 <002>
440 DATA A9,3B,8D,02,03,A9,03,8D <249>
450 DATA 03,03,A5,BA,C9,01,D0,10 <235>
460 DATA A9,74,8D,30,03,8D,32,03 <239>
470 DATA A9,C4,8D,31,03,8D,33,03 <007>
480 DATA AD,8B,02,8D,90,03,60 <113>

```

M-P-S: Multi-Programm-System

Mehr als 30 Programme gleichzeitig im Speicher

Wer hat nicht schon den Wunsch verspürt, mehr als ein Programm gleichzeitig im Speicher zu haben. M-P-S erlaubt sogar bis zu 32 verschiedene Basic-Programme. Dafür stehen 42 KByte(!) freier Basic-Speicher zur Verfügung. Jedes Programm kann unabhängig von einem anderen aufgerufen

und gestartet werden. Zu jeder Zeit behalten Sie den Überblick über die im Computer stehenden Programme. Auch Directories lassen sich permanent im Speicher halten. Eine tolle Sache.

M-P-S bietet die Möglichkeit, mehr als nur ein Basic-Programm im Speicher zu halten. Vorgesehen ist eine Verwaltung von maximal 32 Basic-Programmen, wobei die Anzahl der tatsächlich speicherbaren Programme aber auch von deren Umfang abhängt. Jedes Programm erhält eine Kennnummer, die zusammen mit den Programmnamen, Anfangs- und Endadresse in einem Inhaltsverzeichnis abgelegt wird. Durch die Eingabe der Kennnummer kann dann jedes einzelne Basic-Programm aufgerufen werden. Für die Basic-Programme stehen etwa 42-KByte zur Verfügung. Dies wurde dadurch erreicht, daß Tabellen und Programmspeicher teilweise in den RAM-Bereich unter dem ROM gelegt wurde.

Um die Länge von M-P-S in Grenzen zu halten, wurde die Länge der Befehlswörter auf drei Buchstaben begrenzt.

Die Befehle:

1. OFF
2. STO (Store)
3. REC (Recall)
4. DSP (Display)
5. SPA (Space)
6. DEL (Delete)

1. OFF

OFF bewirkt ein Abschalten von M-P-S. Möchte man anschließend M-P-S wieder aktivieren, so gibt es zwei Möglichkeiten: