

Tips & Tricks für Profis

Wir können Ihnen als weltweit erste Computerzeitschrift eine wirklich vollständige POKE-Liste für den C 64 anbieten! Interessant ist auch ein Trick, um den \$C000-Bereich für Basic-Programme zu nutzen.

Man möchte es nicht wahrhaben: Obwohl der C 64 nun schon drei Jahre alt ist, gibt es doch immer wieder ein paar unbekannte Tricks. Selbst die eingefleischten C 64-Profis geraten mitunter ins Staunen!

Vollständige POKE-Liste für den C 64!

Das in Listing 1 wiedergegebene Programm liefert dem Benutzer eine nach Wissen und Gewissen der Autoren wirklich vollständige Liste aller bekannten und unbekannt POKEs für den C 64. Es beantwortet damit die allerletzten Fragen der C 64-Fan-Gemeinde.

Programmtechnisch sei auf die aufwendige Seiten- und Zeilenformatierung hingewiesen, die ein angenehmes Druckbild auf dem (angeschlossenen) Drucker gewährleisten. Allerdings sollte dieser Drucker vor dem »RUN«-Befehl mit reichlich Papier versorgt werden. Abtipphinweise: Bitte gehen Sie bei der Eingabe dieses äußerst komplizierten Listings sorgfältig vor und verwenden Sie unbedingt den Checksummer. Nur dann kann ein fehlerfreier Programmauslauf gewährleistet werden!

(Dr. Peter Kittel, Hartmut Bützer/tr)

```

5 Z$="A L L E{3SPACE}POKES FUER DEN C 64,
  VON H. BUETZER UND P. KITTEL" <025>
6 Z$=Z$+"{16SPACE}" +CHR$(13) <186>
50 OPEN 1,4:ZZ=99:SZ=0 <212>
70 FOR I=0 TO 65535:FOR J=0 TO 255 <026>
80 A$="POKE"+MID$(STR$(I),2)+" "+MID$(STR$(
  J),2) <055>
81 IF LEN(A$)+LEN(Z$)<79 THEN Z$=Z$+" "+A$
  :GOTO 85 <090>
83 ZZ=ZZ+1:IF ZZ>60 THEN ZZ=0:SZ=SZ+1:PRIN
  T#1,CHR$(12);"SEITE";SZ:PRINT#1 <093>
84 PRINT#1,Z$:Z$=A$ <042>
85 NEXT:PRINT#1,Z$ <205>
90 CLOSE 1:END <024>

```

© 64'er

Listing 1. »Vollständige POKE-Liste«.

»Device not present Error«

Mit folgender Befehlsfolge können Sie überprüfen, ob Floppy oder Drucker am seriellen Bus angeschlossen und eingeschaltet sind:

```
OPEN 1f,ga,sa:CLOSE 1f:IF ST < 0 THEN PRINT" Fehler!":
STOP
```

Bedeutung der Variablen:

- lf: logische Filenummer, beliebig, sollte nicht anderweitig verwendet werden.
- ga: Geräteadresse des zu überprüfenden Geräts, üblicherweise 4 für den Drucker, 8 oder 9 für die Floppy.
- sa: Sekundäradresse, darf bei der Floppy nicht 15 sein, da sonst der CLOSE-Befehl auch alle anderen Floppy-Kanäle schließt; kann beim Drucker entfallen.
- ST ist die Computer-interne Statusvariable, in der das Betriebssystem Fehler bei der Ein- und Ausgabe festhält.

Funktionsweise: Der OPEN-Befehl aktualisiert, da kein Filenamen angegeben ist, nur die Computer-interne Tabelle der geöffneten Files. Daher kann der OPEN-Befehl auch niemals einen »DEVICE NOT PRESENT ERROR« nach sich ziehen. Entscheidend ist der CLOSE-Befehl: Da die in dieser Tabelle eingetragene Geräteadresse auf den seriellen Bus weist, wird auf jeden Fall auf den Bus zugegriffen und dabei festgestellt, daß das Gerät nicht angeschlossen beziehungsweise eingeschaltet ist. Dieser Zustand setzt Bit 7 der Statusvariablen ST. Der Basic-Interpreter verzichtet beim CLOSE-Befehl auf die Überprüfung dieser Systemvariablen, also tritt kein »device not present error« auf. Wird ST dann von Basic aus angesprochen, wird Bit 7 als Vorzeichenbit aufgefaßt, folglich enthält ST bei fehlendem/ausgeschaltetem Gerät einen negativen Wert und der IF-Befehl wird nach dem THEN fortgesetzt.

Übrigens: Kannten Sie schon diesen Betriebssystemfehler: Geben Sie bei ausgeschalteter Floppy ein:

```
10 OPEN 1, 8, 15 : INPUT #1, E, E$
RUN
```

Der Computer wartet, bis Sie die Floppy einschalten, und gibt dann erst »?device not present error« aus.

(Rainer Perske/tr)

Hardcopy für Star SG/10, 15

Die »Hardcopyroutine auf Tastendruck für Epson-Drucker« aus 64'er, Ausgabe 1/86, läßt sich für den STAR SG/10,15-Drucker modifizieren.

Programm »Hardcopy.obj.« mit 8,1 laden, »SYS 49152« eingeben und danach im Direktmodus folgende POKEs eingeben:
POKE 49441,75 : POKE 49442,27 : POKE 49443,13 : POKE 49444,49
: POKE 49445,27.

Soll noch statt der F1-Taste, die in vielen Programmen zur Auswahl in Menüs verwendet wird, eine andere Taste gewählt werden (ich verwende die »-«-Taste), so muß noch POKE 49168,53 eingegeben werden.

(Heden Wolfgang/tr)

Basic-Programm im \$C000-Bereich?

In der Tat kann es manchmal bei Verwaltung größerer Datenmengen beim C 64 sinnvoll sein, den für Basic-Programme normalerweise nicht zugänglichen Bereich \$C000 bis \$D000 (C 64) beziehungsweise \$A000 bis \$C000 (VC 20) mit dem Basic-Programm zu belegen und damit den Grundspeicher voll für Variablenfelder zur Verfügung zu haben. Für die Programmeingabe (beziehungsweise zum Laden) lautet die POKE-Kombination:

```
POKE44,192:POKE56,208:POKE49152,0:NEW
```

Die Programmzeile 0 eines solchen Programms muß dann lauten:

```
0 POKE56,PEEK(644):POKE45,3:POKE46,PEEK(642):CLR
```

Achtung: Nach »RUN« kann zwar noch mit »LIST« das Programm abgefragt werden, aber Änderungen im Programmlisting führen unweigerlich zum Systemabsturz (hängt mit der Verschieberoutine für Basic-Zeilen zusammen)! Für den VC 20 gilt für die Programmverlegung nach \$A000-\$C000 (sofern dieser Bereich mit einer RAM-Erweiterung belegt ist) für die Programmeingabe die Befehlsfolge POKE 44,160:POKE 56,192:POKE 40960,0:NEW. Die Zeile 0 gilt unverändert auch beim VC 20.

(Barnim König/tr)

Grafikzeichen in der ID

Durch folgenden kleinen Trick ist es möglich, auch Grafikzeichen in der ID einer Diskette zu verwenden: Man schiebt beim Formatieren vor der ID noch ein Anführungszeichen (CHR\$(34)) ein:

```
OPEN 1,8,15:"N:NAME"+CHR$(34)+",ID":CLOSE1
```

Der Kreativität sind keine Grenzen gesetzt.

(Wolf Schenk/tr)

Hypra-Platos und Görlitz-Interface

Laden Sie nun zuerst einmal das Programm »2. PRG« mit dem MSE. Ist das geschehen, ist mit dem Befehl »CTRL-N« auf den Eingabemodus zu schalten. Als Startadresse geben Sie »9240« ein, tippen die Zeile »9240« aus Ausgabe 11/85 ab, jedoch mit dem Unterschied, daß Sie die vierte Zahl (0A) durch »0D« ersetzen. Als Prüfsumme ist statt 40 A0 einzusetzen. Mit der Zeile »9278« ist genauso zu verfahren. Ersetzen Sie hier das siebte Byte (0A) ebenfalls durch »0D«. Die Prüfsumme ändert sich. Statt »26« muß hier »32« eingesetzt werden. Speichern Sie das Programm. Als Sekundäradresse ist nun im Druckertreiber die \$14 zu empfehlen. (Achim Hübner/tr)

Computer-Logbuch

Dieses Assemblerprogramm (Listing 2) dient dazu, die aktuelle Speicherstelle im Programmablauf in die linke obere Ecke des Bildschirms zu schreiben.

Mich hatte es gereizt, bei fremden Programmen in Maschinensprache die zu den verschiedenen Routinen gehörenden Adreßbereiche zu lokalisieren.

Ursprünglich wurde bei jedem zehnten Interrupt die Adresse aus den Speicherstellen 105 und 106 gelesen und in die oberste Zeile des Bildschirms geschrieben.

Durch den schnellen Wechsel der Anzeige war dies jedoch mühsam abzulesen. Aus diesem Grund habe ich jede zweite Ausgabe in die zweite Zeile umgeleitet. Nun ist die Ablesung kein Problem mehr. Es ist schon interessant, bei dieser Adressenfolge zuzuschauen. Besonders lehrreich ist es für Anfänger und Fortgeschrittene, zu sehen, welche Bereiche im Kernel angesprungen werden.

Wenn zum Beispiel der blinkende Cursor auf eine Eingabe wartet, so kann man in bunter Folge die Adressen zwischen \$E5CA bis \$E5D4 Revue passieren sehen.

Denkbar wäre noch, die Ablesung bei jedem Interrupt vorzunehmen, und die Adressen auf den Drucker zu geben. Dabei könnten alle Adressen im ROM-Bereich aussortiert werden, wenn man sich auf die Struktur eines noch unbekanntes Programms konzentrieren möchte. (Ralf Störmer/tr)

programm : outadr	c000	c097
c000 :	78 a9 0d a0 c0 8d 14 03	73
c008 :	8c 15 03 58 60 ce 96 c0	43
c010 :	ad 96 c0 c9 0f f0 11 ad	8a
c018 :	96 c0 c9 0a f0 25 ad 96	de
c020 :	c0 c9 01 f0 39 4c 31 ea	b4
c028 :	a0 03 a9 0e 99 00 d8 88	84
c030 :	10 f8 a0 00 bd 06 01 20	35
c038 :	66 c0 bd 05 01 20 66 c0	3b
c040 :	4c 31 ea a0 03 a9 0e 99	dc
c048 :	28 d8 88 10 f8 a0 00 bd	10
c050 :	06 01 20 7e c0 bd 05 01	bf
c058 :	20 7e c0 4c 31 ea a9 10	a2
c060 :	8d 96 c0 4c 31 ea 48 4a	12
c068 :	4a 4a 4a 20 71 c0 68 29	7f
c070 :	0f c9 0a 90 02 e9 39 69	20
c078 :	30 99 00 04 c8 60 48 4a	3b
c080 :	4a 4a 4a 20 89 c0 68 29	19
c088 :	0f c9 0a 90 02 e9 39 69	38
c090 :	30 99 28 04 c8 60 10 18	17

Listing 2. »Outadr« zeigt ständig den Prozessor-Programmzähler an.

Absturz-POKES

Durch »Verbiegen« diverser Vektoren lassen sich einige interessante Effekte erreichen:

POKE768,226:POKE769,252 — Reset nach Programmfehler oder Programmende (nicht im Direktmodus verwenden!)

POKE770,226:POKE771,252 — Reset nach Drücken der STOP-Taste (nicht im Direktmodus verwenden!)

POKE816,226:POKE817,252 — Reset nach Eingabe des LOAD-Befehls

Mit diesem Trick lassen sich noch andere Zeiger verändern. Wo diese Zeiger im Speicher des C64 liegen, läßt sich aus der Speicherbelegungstabelle des Handbuchs (Anhang Q, Seite 160) entnehmen.

Eine andere Möglichkeit ist, anstatt eines Resets einen Overflow-Error auszulösen. Dafür muß man die Werte 92 und 189 anstatt 226 und 252 in die entsprechenden Speicherstellen POKEn. Das läßt sich gut bei LOAD (816/817) und SAVE (818/819) machen. Vorsicht bei 768/769 und 770/771! Bei diesen Speicherstellen funktioniert der Trick nicht. Bei 770/771 wird zum Beispiel nach der Ready-Meldung immer wieder die Overflow-Routine angesprungen! (Stefan Pohl/tr)

Sortierroutine für zweidimensionale Felder

Array-Sort (Listing 3) wird absolut geladen (Sekundäradresse 1) und steht dann im Bereich \$C000 bis \$C31B. Es ist wegen einiger JMPs und JSRs nicht relocatibel. Aufgerufen wird die Sortierung mit SYS49152, Variable, wobei das \$-Zeichen nicht angegeben wird.

Beispiele: SYS49152,A, oder SYS49152,NM, aber nicht SYS49152,A\$. Bei zweidimensionalen Arrays muß vorher noch das Element der 2. Dimension, nach dem sortiert werden soll, in die Speicherstelle 780 gePOKEt werden.

Beispiel:

Eine kleine Adreßverwaltung besteht aus Y\$(X,0)=Vorname, Y\$(X,1)=Nachname, Y\$(X,2)=Straße und Y\$(X,3)=Ort.

So wird nun mit »POKE780,1:SYS48152,Y« das Array nach den Nachnamen sortiert.

Der POKE darf nicht entfallen, da sich in Zelle 780 zufällige Werte befinden!

Sollte das Array nicht dimensioniert oder das Element das in 780 gePOKEt wird, nicht vorhanden sein, so führt das zu der Fehlermeldung »Bad Subscript«. Im Programm werden einige Zeropage-Adressen und Teile des Kassettenpuffers verwendet, was jedoch für Basic nicht sehr von Belang ist.

(Franz Schinagl/tr)

SMON verbessert

Der Vergleichsbefehl des SMON läßt sich noch wesentlich optimieren: Dadurch bricht der SMON nicht bei dem ersten ungleichen Byte den Vergleich ab, sondern vergleicht solange, bis die Stop-Taste gedrückt wird.

Hierzu wird einfach aus dem »JMP \$C323« bei Adresse »\$CB0E« der Befehl »JSR \$C323« gemacht. Zusätzlich müssen nur noch zwei Befehle in das SMON-Disassembler-Listing eingefügt werden. Daraus folgt, daß der SMON ab \$CB11 um 6 Byte nach hinten verschoben werden muß und einige Speicherzellen ebenfalls geändert werden müssen.

Die zu ergänzenden 2 Befehle haben die Aufgabe, die Stop-Taste-Routine aufzurufen und den Vergleich weiterlaufen zu lassen.

Um dies möglichst einfach zu machen (Verschieben, Umrechnen und Einfügen von Befehlen), schildere ich nun die einfachste Methode.

1. SMON \$C000 laden und den Befehl NEW eingegeben.
2. Reassembler laden, nach \$9000 verschieben und »NEW« eingeben.
3. Listing 4 eingeben und den Reassembler mit »SYS 36864, \$C000, \$CFEA : RUN« starten
4. Das erhaltene Quelltextprogramm speichern (zirka 119 Blocks)
5. Hypra-Ass laden und starten
6. Das Quelltextprogramm mit /1" name" laden
7. Listing 5 eintippen.
8. Hypra-Ass starten.
9. Wenn der Assembler fertig ist (nach 1 Minute), den neuen SMON starten und mit »S"NAME" C000 D000« speichern.

```

programm : array-sort      c000 c31c
c000 : 85 b5 a9 00 8d 34 03 8d 6c
c008 : 35 03 20 73 00 8d 34 03 79
c010 : 20 73 00 c9 3a f0 0a c9 0a
c018 : 00 f0 06 8d 35 03 20 73 96
c020 : 00 20 4b c0 20 9b c0 a5 48
c028 : fb c9 01 d0 03 4c 17 c2 d7
c030 : 20 cd c0 a5 b5 c5 fc 10 b9
c038 : 5e c9 00 f0 03 20 9c c2 c2
c040 : 20 fd c0 a5 b5 f0 03 20 73
c048 : 9c c2 60 a5 2f 85 22 a5 05
c050 : 30 85 23 20 8b c0 a0 00 51
c058 : ad 34 03 d1 22 d0 15 c8 a9
c060 : ad 35 03 09 80 d1 22 d0 4b
c068 : 0b a5 22 8d 36 03 a5 23 d8
c070 : 8d 37 03 60 a0 02 b1 22 8b
c078 : 18 65 22 aa c8 b1 22 65 8e
c080 : 23 85 23 86 22 20 8b c0 d2
c088 : 4c 56 c0 a5 22 c5 31 d0 9b
c090 : 09 a5 23 c5 32 d0 03 4c 3c
c098 : 45 b2 60 ad 36 03 85 19 c8
c0a0 : ad 37 03 85 1a a0 04 b1 74
c0a8 : 19 85 fb c9 03 10 1b c8 6b
c0b0 : b1 19 99 f7 00 c0 08 30 da
c0b8 : f6 a5 fc a6 fd 85 fd 86 a6
c0c0 : fc a5 fe a6 ff 85 ff 86 5c
c0c8 : fe 60 4c 48 b2 a9 00 85 96
c0d0 : 02 a5 fe 18 65 fe 90 03 fe
c0d8 : 20 f6 c0 65 fe 90 03 20 11
c0e0 : f6 c0 8d 38 03 a5 ff 18 2e
c0e8 : 65 02 65 ff b0 0c 65 ff a9
c0f0 : b0 08 8d 39 03 60 e6 02 02
c0f8 : 18 60 4c 8a bb ad 36 03 ad
c100 : 18 69 09 85 22 ad 37 03 32
c108 : 69 00 85 23 a5 22 18 69 d6
c110 : 03 85 24 a5 23 69 00 85 1c
c118 : 25 20 a1 c1 a0 01 b1 22 0b
c120 : 99 56 00 b1 24 99 58 00 8b
c128 : c8 b1 22 99 56 00 b1 24 f9
c130 : 99 58 00 a0 00 b1 22 d1 c3
c138 : 24 30 02 b1 24 85 5b c9 9b
c140 : 00 f0 34 a0 00 b1 59 d1 70
c148 : 57 f0 08 b0 0b 20 bc c1 58
c150 : 4c 58 c1 c8 c4 5b d0 ed 98
c158 : a5 24 c5 5c d0 09 a5 25 42
c160 : c5 5d d0 03 4c 77 c1 a5 3b
c168 : 24 18 69 03 85 24 a5 25 ad
c170 : 69 00 85 25 4c 1c c1 a5 d7
c178 : 22 18 69 03 85 22 a5 23 a7
c180 : 69 00 85 23 a5 22 c5 5c ea
c188 : d0 07 a5 23 c5 5d d0 01 3c
c190 : 60 a5 22 18 69 03 85 24 56
c198 : a5 23 69 00 85 25 4c 1c 14
c1a0 : c1 a5 22 18 6d 38 03 85 6f
c1a8 : 5c a5 23 6d 39 03 85 5d ca
c1b0 : a5 5c 38 e9 03 b0 02 c6 1a
c1b8 : 5d 85 5c 60 a5 22 48 a5 d3
c1c0 : 23 48 a5 24 48 a5 25 48 cc
c1c8 : a6 fc 86 be 20 05 c2 a5 e6
c1d0 : 22 18 6d 38 03 85 22 a5 91
c1d8 : 23 6d 39 03 85 23 b0 22 d9
c1e0 : a5 24 18 6d 38 03 85 24 45
c1e8 : a5 25 6d 39 03 85 25 b0 f5
c1f0 : 11 c6 be d0 d7 68 85 25 4f
c1f8 : 68 85 24 68 85 23 68 85 57
c200 : 22 60 4c 7e b9 a0 00 b1 39
c208 : 22 aa b1 24 91 22 8a 91 e8
c210 : 24 c8 c0 03 30 f1 60 18 6d
c218 : ad 36 03 69 07 85 6e ad 80
c220 : 37 03 69 00 85 6f a0 00 8a
c228 : b1 6e f0 6f 85 22 c8 b1 2a
c230 : 6e 99 22 00 c0 02 d0 f6 41
c238 : a5 6e 85 71 a5 6f 85 72 75
c240 : 18 a5 71 69 03 85 71 90 f8
c248 : 02 e6 72 a0 00 b1 71 f0 a3
c250 : 3d 85 4d c5 22 90 02 a5 56
c258 : 22 85 55 c8 b1 71 99 4d 53
c260 : 00 c0 02 d0 f6 a0 00 b1 33
c268 : 23 d1 4e f0 04 b0 0b 90 39
c270 : cf c8 c4 55 d0 f1 c4 22 73
c278 : b0 c6 a0 02 b1 6e aa b1 90
c280 : 71 91 6e 99 22 00 8a 91 f8
c288 : 71 88 10 f0 30 b2 18 a5 a4
c290 : 6e 69 03 85 6e 90 8f e6 9c
c298 : 6f d0 8b 60 ad 36 03 18 27
c2a0 : 69 09 85 22 85 24 ad 37 d2
c2a8 : 03 69 00 85 23 85 25 20 44
c2b0 : 01 c3 a6 b5 a5 24 18 6d aa
c2b8 : 38 03 85 24 a5 25 6d 39 03
c2c0 : 03 85 25 ca d0 ee a0 00 2f
c2c8 : b1 22 aa b1 24 91 22 8a d8
c2d0 : 91 24 c8 c0 03 d0 f1 a5 87
c2d8 : 22 c5 5c d0 07 a5 23 c5 c4
c2e0 : 5d d0 01 60 a5 22 18 69 90
c2e8 : 03 85 22 a5 23 69 00 85 74
c2f0 : 23 a5 24 18 69 03 85 24 ff
c2f8 : a5 25 69 00 85 25 4c c6 ca
c300 : c2 a5 22 18 6d 38 03 85 d0
c308 : 5c a5 23 6d 39 03 85 5d 2a
c310 : a5 5c 38 e9 03 b0 02 c6 7a
c318 : 5d 85 5c 60 ff ff ff ff 5b
    
```

Listing 3. »Array-Sort«, eine schnelle Sortieroutine für String-Arrays.

- SMON nach \$9000 verschieben:
1. W C000 D000 9000
 2. V C000 D000 900 920B 9FD8
 3. M 902B 906B :In jedem zweiten Byte das erste »C« in eine »9« umwandeln
 4. FIC*\$9000,9FFA :Die Befehle so umändern wie in der 64'er, Ausgabe 12/85 erwähnt. (Achtung: ab \$9B00 stimmen die Adressen der Befehle mit den alten Adressen nicht mehr überein)
 5. M 9FDE 9FEA :Wie unter »3«.
- (Reiner Gebhardt/tr)

Diverse POKEs		
POKE	53265JL27	Bildschirm aus.einschalten
POKE	204,O	CursorBlinken einschalten
POKE	207,O : POKE 204J	CursorBlinken ausschalten
POKE	56325,5,255	Cursor schnell.Langsam
POKE	56334, PEEK (56334) OR 1	Interrupt ausschalten
POKE	56334, PEEK (56334) AND 254	Interrupt einschalten
POKE	788,52,49	RUN STOP Taste aus.einschalten
POKE	792, 193,71	RESTORE Taste aus.einschalten
POKE	651,255	Repeat um 4 Sekunden verzögern
POKE	646,O,15	Schriftfarbe setzen
POKE	199,O,J	Revers-Modus aus.ein
POKE	649,O,JO	Tastatur ab.einschalten
SYS	65499	TI\$="OOOOOO"
SYS	42115	END ohne READY.
SYS	44808	? SYNTAX ERROR
SYS	62255	Alle Files schließen
WAIT	653J,2,4	Warten bis SHIFTCommodore.CTRL gedrückt wird
WAIT	203,63	Warten auf Tastendruck
WAIT	203,64	Warten solange Taste gedrückt ist
PRINT	PEEK (152)	Anzahl der offenen Files
PRINT	PEEK (153)	Aktuelles Eingabegerät
PRINT	PEEK (154)	Aktuelles Ausgabegerät
PRINT	PEEK (184)	Aktuelle Filenummer
PRINT	PEEK (185)	Aktuelle Sekundäradresse
PRINT	PEEK (647)	Aktuelle Zeichenfarbe

(Alexander Niepel/tr)

```

100 ← T$COOB,$C213
110 ← T$CFD2,$CFE9
120 ← E32
READY.
    
```

Listing 4. Damit definieren Sie die Tabelle des SMON.

1350-	.BY \$3C,\$CA,\$5C,\$C5,\$16,\$CB,\$E2,\$C3
1380-	.BY \$F6,\$CB,\$42,\$CA,\$D2,\$C9,\$6D,\$C3
1390-	.BY \$09,\$CE,\$00,\$00,\$00,\$00,\$00,\$00
12970-LXXXX	JSR LC367
13020-LCBOB	JSR LC351
13030-	JSR LC323
13031-	JSR LC486
13032-	JSR LXXXX
14300-	LDA # \$3C
14560-	LDA # \$58
15220-	LDA # \$76
16000-	LDA # \$9B
16370-	LDA # \$9B
17960-	LDX # \$F7
18400-TLCPD8	.BY \$78,\$CE,\$02,\$CD,\$02,\$CD,\$4E,\$CE
18410-	.BY \$CB,\$CF,\$8B,\$CF

Listing 5. Diese Zeilen müssen Sie zum Quelltext des SMON eintragen.