Tips & Tricks zum C16 und Plus/4

Wir zeigen Ihnen, wie Sie im Grafikmodus des C 16 bis zu 4 KByte Basic-Speicher zur Verfügung haben. Außerdem bringen wir einige interessante PEEKs und POKEs und ein Grafikprogramm zum Experimentieren.

Achtung: An alle versierten C 16- und Plus/4-Programmierer! Wir suchen dringend einen »MSE« und einen »Checksummer« für obengenannte Computer. Die Programme sollten mit denselben Prüfsummern arbeiten, wie die C 64-Versionen. Es wird also am günstigsten sein, wenn die bereits vorhandenen Programme nur noch auf diese Computer mit den geänderten Adressen angepaßt werden. Der »Checksummer C 16« muß natürlich auch den erweiterten Befehlssatz des Basic V3.5 verarbeiten können.

Passend zu diesem Checksummer benötigen wir ein Druckprogramm, das die Steuerzeichen des Computers (zum Beispiel Cursor-Bewegungen innerhalb von PRINT-Anweisungen) in Klartext übersetzt (siehe Checksummer C 64). Wenn Sie einen entsprechenden Lösungsvorschlag entwickelt haben, schicken Sie ihn an die Redaktion, zu Händen Herrn Röder oder setzen Sie sich telefonisch mit uns in Verbindung. (tr)

PEEKs und POKEs

Und hier wieder einmal eine Liste mit interessanten Speicheradressen für den C 16 und Plus/4:

POKE 0,0	Datasettenmotor aus
POKE 0,15	Datasettenmotor ein
PRINT PEEK(1)	Abfrage der Datasette: 200 = Aus, 216 = Störung, 192 = eine der Tasten PLAY/REW/FWD gedrückt, 208 = PLAY und REC gedrückt
SYS 32768	Reset
POKE 240,255	Wartet, bis eine Taste gedrückt wird
POKE 1351,128	Groß/Klein-Umschaltung wird blockiert
PRINT PEEK(1347)	Sondertastenabfrage <SHIFT>/<CTRL> und <CBM>
POKE 65286, PEEK(65286) AND239	Bildschirm abschalten
POKE 65286, PEEK(65286)OR16	Bildschirm einschalten
POKE 814,23	LOAD wird nicht mehr ausgeführt

(Oliver Eickhoff/tr)

Die kleine Schlange

Das Zeichenprogramm, das folgt, zeichnet eine Schlangenlinie. Es ist kein großartiges Programm, aber für C 16-Fans und Anfänger das richtige.

```
10 COLOR 4, 5
20 GRAPHIC 1, 1
30 FOR X = TO 3 1 9
40 Y = INT (100 + 99 * SIN (X/20))
50 DRAW 1 TO X,Y
60 NEXT X
70 GRAPHIC 0
80 END
```

Man kann das Programm auch mit Punkten zeichnen, dann muß man Zeile 50 umändern. Sie lautet dann `50 DRAW 1, X, Y.`

(Sven Burgdorf/tr)

Bis zu 4 KByte Basic-Speicher im Grafikmodus des C 16

Das größte Problem, mit dem sich die Benutzer des C 16 herumschlagen müssen, ist der kleine Arbeitsspeicher der Grundversion. Er beträgt im Normalmodus bekanntlich nur 12277 Byte; im Grafikmodus gar nur noch 2045 Byte.

Wer käufliche oder veröffentlichte Programme, die mit der hochauflösenden Grafik arbeiten, um eigene Routinen, Melodien oder Texte erweitern will, wird enttäuscht, weil diese Programme den Speicher meist bis aufs letzte Byte ausnutzen. Auch der Selbstentwickler wird nicht sehr erfreut sein, wenn der Speicher das Programm gerade nicht mehr aufnehmen kann. Diesen Problemen kann jedoch in den meisten Fällen, ohne einen Pfennig für Speichererweiterungen auszugeben, abgeholfen werden: denn so unglaublich es klingt, mit ein paar Tricks stehen Ihnen im Grafikmodus 4053 Byte zur Verfügung.

Um die Tricks auch einigermaßen zu verstehen, sollte man sich zunächst über den Aufbau des RAM klar werden. Der Speicher des C 16 gliedert sich in sieben Bereiche

Bereich 1:	**Adressen 0 bis 2047 umfassen die Zeropage und dient gewissermaßen als Notizblock für das Betriebssystem.**
Bereich 2:	**Adressen 2048 bis 3071 beinhalten die Farbinformationen für Hintergrund und Vordergrund jeder einzelnen der 25 x 40 Bildschirmpositionen im Textmodus.**
Bereich 3:	**Adressen 3072 bis 4095 umfassen den Speicherbereich für die auf dem Bildschirm im Textmodus dargestellten Zeichen.**
Bereich 4:	**Adressen 4096 bis 6143 umfassen den Bereich des Basic-Speichers, der auch im Grafikmodus verfügbar ist.**
Bereich 5:	**Adressen 6144 bis 16383 dienen im Textmodus als Basic-Speicher, im Grafikmodus ist dort der Bildschirminhalt abgelegt.**
Bereich 6:	**Adressen 16384 bis 64767 sind als RAM beim C 16 nicht bestückt.**
Bereich 7:	**Adressen 64768 bis 65535 werden vom Computer ähnlich gehandhabt wie der Bereich 1.**

Nun muß man sich folgendes verdeutlichen:
— Im Textmodus wird das Basic-Programm in den Bereichen 4 und 5 abgelegt, die Bereiche 2 und 3 dienen zur Speicherung des Bildschirminhalts.
— Im Grafikmodus ist das Basic-Programm im Bereich 4 enthalten, der Bereich 5 speichert den Bildschirminhalt.

Eine naheliegende Frage lautet nun, was bei eingeschaltetem Grafikmodus in den Bereichen 2 und 3, also auf dem Textbildschirm, vor sich geht. Die Antwort lautet: Nichts! Und man kann diese Bereiche für Basic-Programme nutzen. Vorzugsweise natürlich bei Grafikprogrammen, mit Einschränkungen aber auch im Textmodus.

Bei allen drei im folgenden dargestellten Möglichkeiten wird der Zeiger für den Basic-Anfang verschoben, und zwar entweder unmittelbar hinter den Bildschirmspeicher oder aber in die Bereiche 2 beziehungsweise 3. Die Verschiebung der Zeiger muß jeweils nach einem Reset, also vor dem Eingeben oder Laden von Programmen erfolgen. Natürlich können dann die Programme, die keine Speichererweiterung benötigen, ganz normal geladen werden. Programme, die den erweiterten Speicher benötigen, sind ohne die Verschiebung der Zeiger nicht lauffähig, da sie bei Erreichen des »GRAPHIC«-Befehls einen »out of memory error« erzeugen. Sie können jedoch (ohne die Zeiger zu verschieben) geladen, gelistet und bearbeitet werden.

Wie bereits erwähnt, gibt es drei Möglichkeiten, den Basic-Anfang zu verschieben.

1. Man verschiebt den Basic-Anfang unmittelbar hinter das letzte Zeichen des Bildschirmspeichers. Ausgenutzt wird da-

bei die Tatsache, daß der Bildschirm aus 25 x 40 = 1000 Zeichen besteht, der C 16 dem Bereich 3 aber 1 KByte, also 1024 Byte zur Verfügung stellt. Die überschüssigen Bytes können Sie mit folgendem Befehl an den Arbeitsspeicher anhängen:

```
POKE 44,15:POKE 43,233:POKE4072,0:NEW
```

Die beiden ersten POKEs setzen den Zeiger für den Basic-Anfang auf das erste freie Byte. Der dritte POKE-Befehl setzt an den neuen Basic-Anfang eine notwendige Null. Durch »NEW« wird der Computer veranlaßt, weitere interne Adressen an den neuen Basic-Anfang auszurichten. Sie haben nun 22 Byte gewonnen, der Textbildschirm steht voll zur Verfügung und Sie werden im weiteren Verlauf der Arbeit nichts von der Veränderung bemerken.

2. Wesentlich mehr, nämlich bis zu 822 Byte, stehen Ihnen mit der zweiten Möglichkeit zur Verfügung. Dabei wird der Basic-Anfang in den Textbildschirm gesetzt. Das Basic-Programm, das sich dann im Bildschirmbereich befindet, äußert sich durch einen Zeichensalat in der unteren Hälfte des Bildschirms. Als besonders vorteilhaft erweist sich hier einmal mehr die Möglichkeit des C 16, Windows zu definieren. Damit kann nämlich der Bildschirmbereich, in dem sich das Basic-Programm befindet, vor dem Überschreiben geschützt werden. Es steht Ihnen also bei dieser Möglichkeit ein Fenster in der oberen Bildschirmhälfte zur Verfügung, dessen Größe Sie je nach Bedarf selbst festlegen können. In diesem Fenster können Sie Ein- und Ausgaben durchführen, Programme schreiben, listen und testen wie gewohnt. Ein Nachteil ist wohl, daß Programme, die den Textbildschirm nutzen, eventuell so geändert werden müssen, daß sie mit dem definierten Fenster auskommen.

Wollen Sie diese Möglichkeit nutzen, so gehen Sie folgendermaßen vor:

— Überlegen Sie, wieviel Speicherplatz Sie brauchen, das heißt wie groß Ihr Fenster sein darf.
— Bringen Sie den Cursor auf das letzte Zeichen der letzten nutzbaren Zeile.
— Drücken Sie nacheinander die Tasten <ESC> und <B>.

nutzbare Zeichen	H	L	N	gewonnener Speicher
5	12	201	3272	822
10	13	145	3472	622
15	14	89	3672	422
20	15	33	3872	222

— Entnehmen Sie die entsprechenden Werte aus der untenstehenden Tabelle.
— Setzen Sie die Werte für H,L und N aus der Tabelle in den folgenden Befehl ein:

```
POKE 44,H : POKE 43, L : POKE N,0 : NEW
```

3. Die dritte Möglichkeit hat zwar leider mehr Nachteile, die aber durch den Gewinn von 2 KByte Speicher leicht wettgemacht werden. Da bei Ausnutzung dieser Speichergröße nur eine einzige Textzeile im Textmodus zur Verfügung steht, muß man folgendermaßen vorgehen: Die Programme werden im Normalmodus geschrieben, dabei darf der Grafikmodus auch vorher nicht eingeschaltet worden sein, da auch »GRAPHIC 0« den Grafikspeicher nicht für Basic freimacht, sondern nur in den Textmodus umschaltet. Ein Testen von Programmen ist also nur sehr eingeschränkt, eventuell nur durch Weglassen der »GRAPHIC«-Befehle möglich.

Nun muß in das fertige Programm eine REM-Zeile so eingebaut werden, daß die Codes der Eingaben (LOAD, RUN etc.) die in der ersten Bildschirmzeile durchgeführt werden müssen, im Programm hinter eben diese REM-Anweisung zu stehen kommen. Auf diese Weise kann trotz Betrieb der ersten Bildschirmzeile bei erweitertem Basic-Speicher dieser »am Stück« gehandhabt werden. Um die richtige Zeilennummer für diese REM-Zeile zu finden, gehen Sie folgendermaßen vor:

— Beginnen Sie mit dem Wert 5
— Zu Beginn jeder Zeile zählen Sie 5 dazu.
— In den Zeilen zählt jedes Zeichen 1, dabei gelten Basic-Befehlswörter als ein Zeichen.
— Am Ende von Zeilen dürfen keine unnötigen Leerstellen stehen.
— Die Zeilennummer und die darauffolgende Leerstelle werden nicht gezählt.
— Vor der Zeile, mit der Sie den Wert 984 erreichen, fügen Sie folgende Zeile ein: <Zeilennummer> REM {80 beliebige Zeichen} Beispiel: 980 REM EEEEEEEEEEEE... und so weiter.
— Nun speichern Sie das Programm.
— Cursor auf das letzte Zeichen der ersten Zeile setzen.
— <ESC> und dann <B> drücken.
— POKE 44,8 : POKE 43,41 : POKE 2088,0 : NEW eingeben.
— Programm laden und starten.

Anmerkung: Die Möglichkeit 1 funktioniert in allen Grafikmodi, bei den Möglichkeiten 2 und 3 dürfen nur die Grafikmodi mit ungeteiltem Bildschirm, also GRAPHIC 1 und GRAPHIC 3 benutzt werden. Ich meine, daß sich mit den dargestellten Tricks trotz der erwähnten Nachteile sehr viel anfangen läßt.

(Erwin Huber/tr)

Die (un)gleichen Brüder C 16 und Plus/4

Jeder C 16- oder Plus/4-Besitzer weiß es: Die Programme für den jeweils anderen Computer laufen auch auf dem eigenen. Zumindest die meisten. Oder doch nicht? Wir haben die Computer für Sie ausführlich verglichen.

In letzter Zeit ist viel Verwirrung um die Kompatibilität der Computer C 16/116 und Plus/4 entstanden (siehe 64'er, 9/86, Seite 67). Eindeutig geklärt war bisher nur, daß C 16 und C 116 absolut hard- und softwarekompatibel sind, da sie sich lediglich durch Gehäuse, Tastatur und eine geringfügig modifizierte Platine unterscheiden.

Weitgehend ungeklärt war dagegen ihr »verwandtschaftliches Verhältnis« zum Plus/4. Überraschend:

Beim C 16/116 handelt es sich lediglich um einen »abgemagerten« Plus/4: Die Hardware des C 16/116 ist mit der des Plus/4 — bis auf den statt 64 KByte nur 16 KByte großen Schreib/-Lesespeicher (RAM), den fehlenden User-Port mit seiner RS232-Schnittstelle und die nicht eingebaute Textverarbeitung- und Dateiverwaltungs-Software — praktisch identisch.

Gemeinsam ist den drei Computern der Mikroprozessor, der Quartz, der Tastatur-Port, der Videocontroller, das programmierbare Logik-Array sowie — die ROMs. — Ja, Sie haben richtig gelesen! Das Basic- und auch das Betriebssystem (»Kernel«)-ROM ist bei allen drei Computern absolut gleich.