

Das Laden von Programmen dauert, selbst wenn man Hypra-Load verwendet, doch immer noch eine geraume Zeit. Viel praktischer sind die Ihnen sicher schon bekannten EPROMs. Der Anschluß von EPROMs stellt beim C 64 kein allzu großes Problem dar. Am Expansion-Port befinden sich alle für diesen Zweck wichtigen Adreß-, Daten- und Steuerleitungen. Analog dazu verfügt auch ein EPROM über Adreß-, Daten- und einige Steueranschlüsse. Letztere unterteilen sich in Pins für die Stromversorgung (0 V, +5 V), Pins zur Programmierung und Pins zur letztendlich gewünschten Verbindung mit dem Bus-System des C 64. Da man nach einer Programmierung des EPROMs die Anschlüsse für die Programmiersteuerung (PGM) nicht mehr benötigt, diese aber im Betrieb einen definierten Zustand haben müssen, werden beide auf logisch High (+5V) gelegt. Der Anschluß »Chip Enable« (CE = Chipauswahl) teilt dem EPROM mit, daß es angesprochen ist, und »Output Enable« (OE = Ausgabe möglich) fordert das EPROM auf, die Daten an den Datenbus zu legen. Danach stehen die Daten genauso zur Verfügung, als ob der Computer sie aus dem RAM lesen würde.

den. \overline{ROML} wählt den Speicherbereich ab \$8000, \overline{ROMH} den Bereich ab \$A000:
 $\$8000: \overline{ROML} \quad \overline{OE}$
 $\$A000: \overline{ROMH} \quad \overline{CE}$

Jetzt sollte man dem Computer noch mitteilen, daß er auf die EPROMs und nicht, wie gewohnt, auf sein internes RAM zugreift, denn die Leitungen \overline{ROML} und \overline{ROMH} können nicht direkt beeinflußt werden. Erst die Verwendung von zwei weiteren Leitungen bringt das gewünschte Ergebnis. Diese Leitungen heißen \overline{EXROM} und \overline{GAME} . Sobald eine dieser Leitungen auf Masse gelegt wird, teilt sie der PLA (Programmable Logic Array = Adreßmanager) mit, daß die Daten im Bereich \$8000, beziehungsweise \$A000 vom Expansion-Port geholt werden sollen.

Damit sind die wesentlichen Funktionen der 8-KByte-Platine beschrieben. Sie ist so konzipiert, daß sie ein EPROM in den Bereich von \$8000 bis \$9FFF einblendet. Mit den vier Lötbrücken auf der Oberseite der Platine kann allerdings auch eine andere Speicheraufteilung eingestellt werden. Bei Bedarf trennen sie die beiden geschlossenen Lötbrücken auf und verbinden die beiden geöffneten. Damit wird der Speicherbereich von \$A000 bis

EPROMs

\$BFFF für die EPROMs selektiert. Dies ist jedoch eine sehr ausgefallene Konfiguration, die Sie in der Praxis wahrscheinlich selten brauchen werden.

Mehrere Betriebsarten

Bei der 16-KByte-Platine wird zwischen mehreren Betriebsarten unterschieden. Im einfachsten Fall wird das rechte EPROM A (vom Expansion-Port gesehen) in den Bereich ab \$8000 und das linke in Bereich ab \$A000 eingeblendet. Aus den einleitenden Betrachtungen wissen wir, daß für den Bereich ab \$8000 \overline{EXROM} auf »Low« und für den Bereich »\$A000 \overline{GAME} auf »low« gelegt (mit der Masse Leitung verbunden) werden müssen. Dies geschieht mit den Schaltern S4 (\overline{EXROM}) beziehungsweise S3 (\overline{GAME}). Weiterhin erfolgt über den Schalter S5 eine Verbindung von \overline{ROML} mit CE/OE des rechten EPROMs und über den Schalter S1 ein Kontakt zwischen \overline{ROMH} und CE/OE des linken EPROMs.

Als nächstes soll die Schaltung so verändert werden, daß beide EPROMs im Bereich ab \$8000 ste-

Zwei Platinen zur Wahl

Zum Aufbau der Schaltung ist es am besten, eine Platine anzufertigen, die in den Expansion-Port paßt. Dabei haben wir uns für zwei verschiedene Ausführungen, eine mit 8 KByte für kleinere Programme und eine mit 16 KByte Speicherplatz entschieden. Es können wahlweise EPROMs vom Typ 2732 (4KByte) oder 2764 (8 KByte), verwendet werden. Zusätzlich bietet die 16 KByte-Platine mehrere Betriebsmodi, die im einzelnen noch beschrieben werden. Doch zurück zum Anschluß der EPROMs an den C 64. Da ein EPROM maximal 8 KByte Speicherplatz hat und damit sicherlich nicht den ganzen Adreßraum des 6510-Prozessors ausfüllt, muß der C 64 den Speicherbereich für das EPROM festlegen. Für diesen Zweck sind zwei Leitungen vorgesehen: \overline{ROML} und \overline{ROMH} . Beide sind »aktiv-low«, das heißt sie wirken erst, wenn sie vom Zustand »1« in den Zustand »0« übergehen, beziehungsweise wenn sie ein Potential von 0 Volt gegenüber Masse haben. Diese Leitungen werden mit den Anschlüssen CE und OE (beide ebenfalls aktiv-low) des EPROMs verbun-

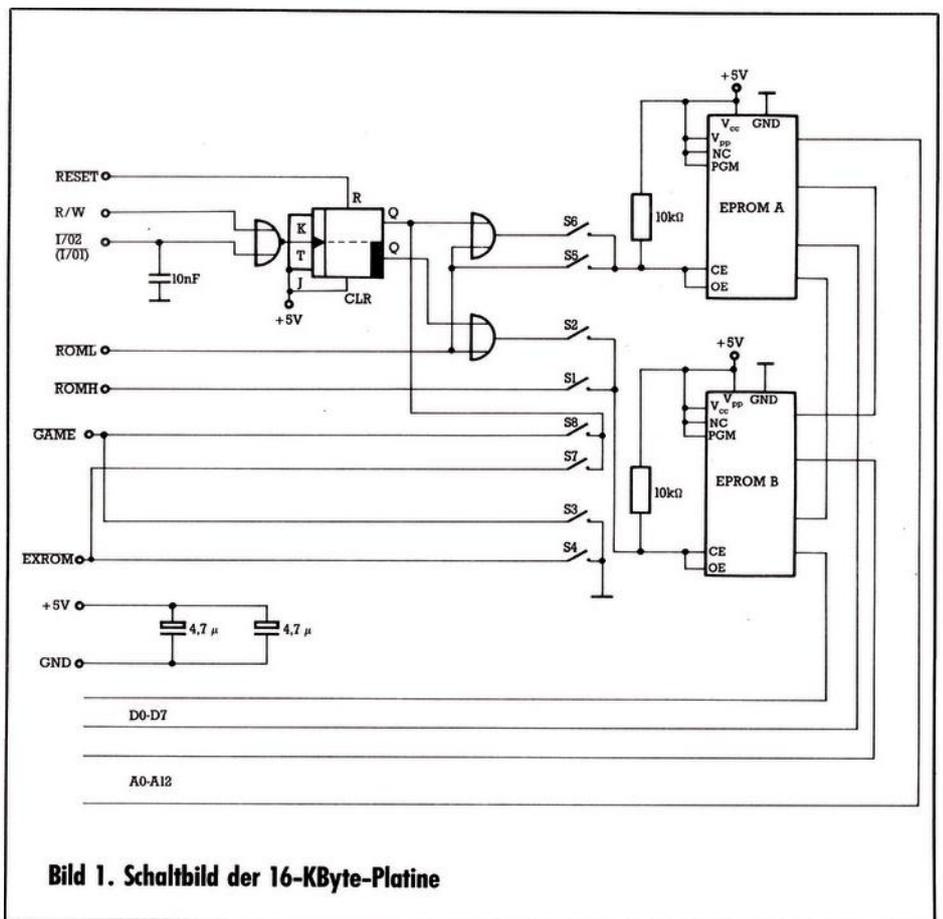


Bild 1. Schaltbild der 16-KByte-Platine

im Expansion-Port

Es gibt kaum ein Speichermedium, von dem Daten so schnell geladen werden können wie von EPROMs. Bauen Sie sich selbst eine Erweiterungsplatine mit wahlweise 8 oder 16 KByte Speicher!

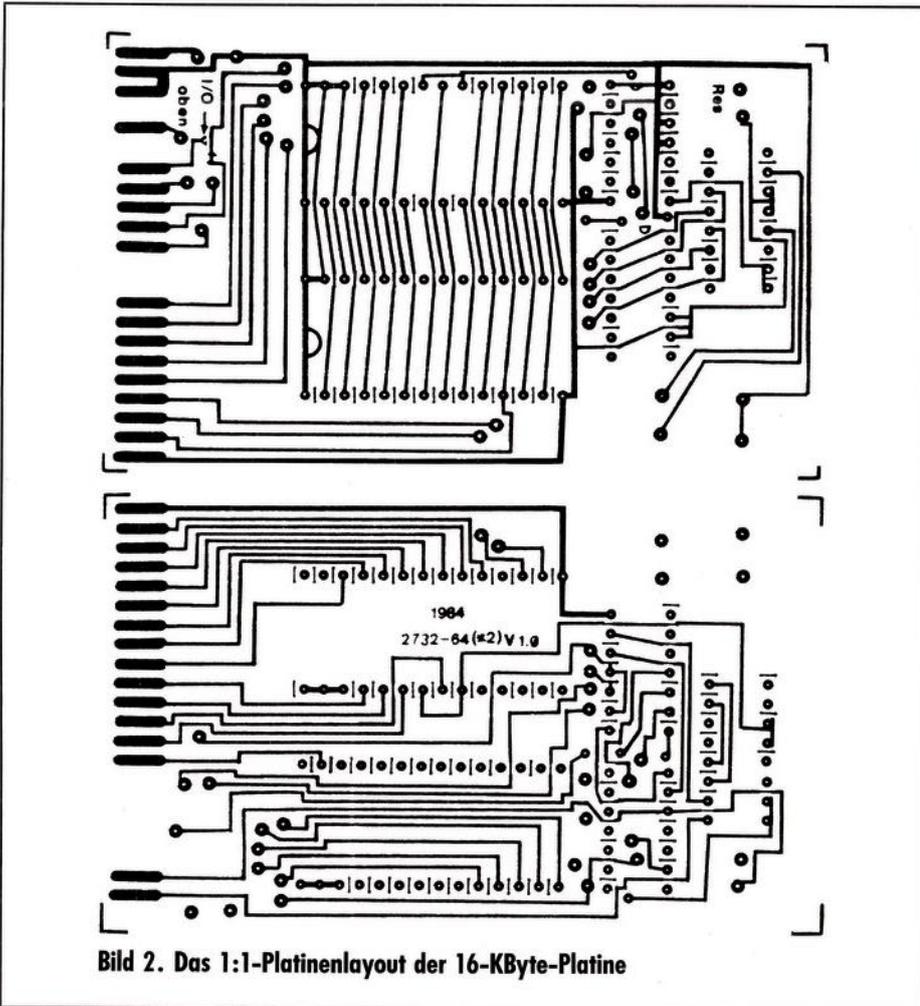


Bild 2. Das 1:1-Platinenlayout der 16-KByte-Platine

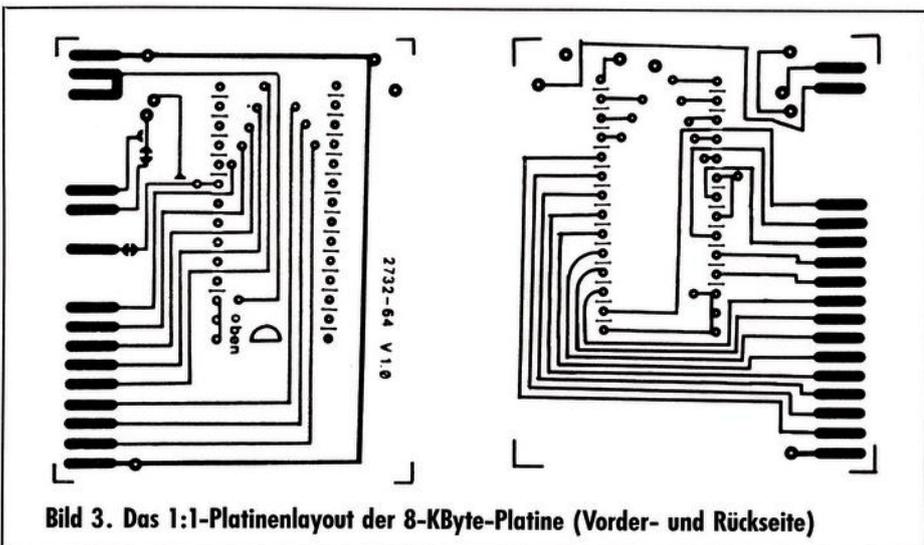


Bild 3. Das 1:1-Platinenlayout der 8-KByte-Platine (Vorder- und Rückseite)

hen, besser gesagt, daß sich immer nur eines in Betrieb befindet und zwischen beiden EPROMs umgeschaltet werden kann. Dazu benötigt man eine kleine Logikschaltung mit einem JK-Flip-Flop und zwei OR-Gattern (Bild 1). Beim JK-Flip-Flop werden die Eingänge J und K auf +5 V gelegt und bei jedem Wechsel von »High« nach »Low« an T (Takt) wird das Flip-Flop umgeschaltet. Der Ausgang Q wechselt dann von »0« nach »1«, »Q« wechselt von »1« nach »0«. Man braucht noch eine Leitung, die vom Computer aus kontrolliert, beziehungsweise geschaltet werden kann und die mit dem Takt-Eingang des Flip-Flop verbunden wird. Zu diesem Zweck gibt es zwei I/O-Leitungen im C 64, nämlich I/O1 und I/O2, denen zwei Register im Computer entsprechen. Diese Register (I/O1: \$DE00; I/O2: \$DF00) kann man zum Beispiel mit POKE 57088,X oder STA \$DE00 ansprechen. Dabei wechselt ihr Zustand kurzzeitig von »high« nach »low« und es entsteht gerade das Schaltverhalten, das man für das Flip-Flop benötigt. Die Zusammenhänge beim Schalten gehen am besten aus einer Logiktable hervor. Es darf dabei immer nur ein EPROM an den Pins CE/OE ein Signal »low« bekommen.

Einschalten (Hardware-Reset) beziehungsweise Zustand 1:

$$Q = L, \bar{Q} = H$$

A1	B1	Q1	A2	B2	Q2
Q=L	H=ROML	H	Q=H	H=ROML	H
Q=L	L=ROML	L	Q=H	L=ROML	H

Q1=L: EPROM A angewählt

Q2=H: EPROM B nicht angewählt

Sobald der Takt-Eingang für kurze Zeit von »High« auf »Low« geht, erhöht man den anderen Zustand des Flip-Flops.

Umschalten beziehungsweise Zustand 2: Q = H, Q̄ = L

A1	B1	Q1	A2	B2	Q2
Q=H	H=ROML	H	Q=L	H=ROML	H
Q=H	L=ROML	H	Q=L	L=ROML	L

Q1=H: EPROM A nicht angewählt

Q2=L: EPROM B angewählt

Die letzte Verwendungsmöglichkeit der Platine erlaubt das Ein- und Ausschalten von wahlweise einem 8-KByte-, oder 16-KByte-Bereich. Mit dem Schalter X5 wird zunächst ROML mit dem rechten EPROM verbunden (\$8000). Für ein 16-KByte-Modul wird zusätzlich der Schalter S1 geschlossen und damit das linke EPROM im Bereich \$A000 selektiert. Doch wie bereits beschrieben, greift der Computer immer noch nicht auf die EPROMs zu. Darum müssen die Leitung EXROM (zusätz-

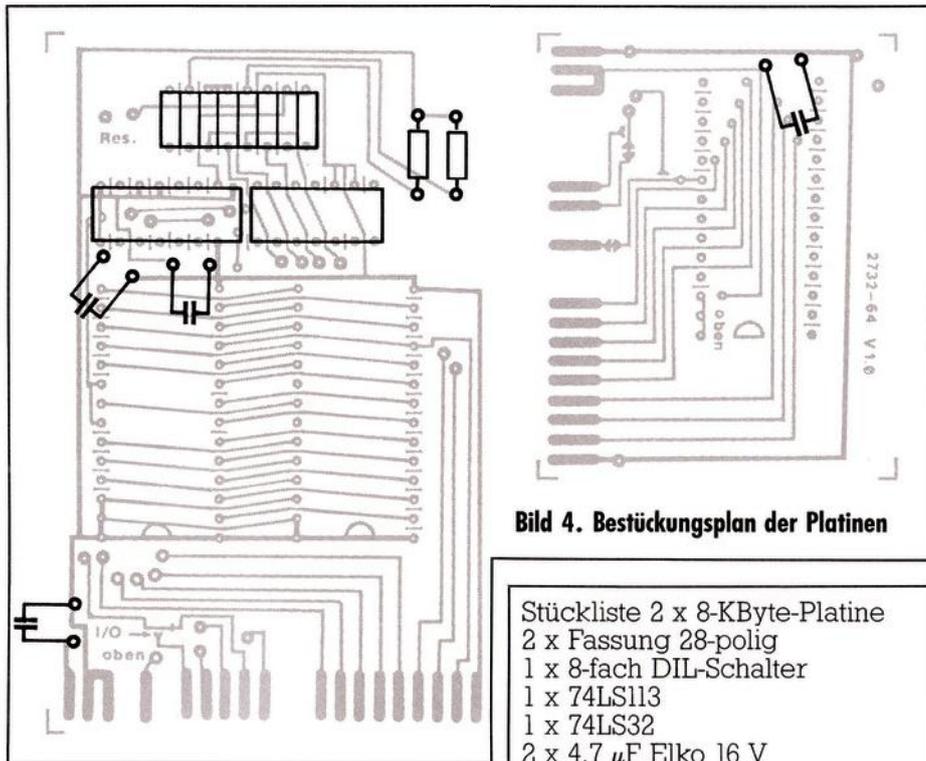


Bild 4. Bestückungsplan der Platinen

Stückliste 2 x 8-KByte-Platine
2 x Fassung 28-polig
1 x 8-fach DIL-Schalter
1 x 74LS113
1 x 74LS32
2 x 4,7 μ F Elko 16 V
1 x 10 nF Keramik Kondensator
2 x 10 kOhm Widerstand $\frac{1}{4}$ V/A
Stückliste 1 x 8-KByte-Platine
1 x Fassung 28-polig
1 x 4,7 μ Elko 16 V
Stückliste

lich GAME bei 16 KByte) mit Masse verbunden werden. Setzt man in diese Verbindung nun das schon verwendete Flip-Flop, so können die Leitungen, je nach Zustand, von I/O₂ auf Masse gelegt werden oder

nicht. Das Ein- und Ausschalten erfolgt jetzt mit dem schon erwähnten Anspringen von I/O₂ (\$DF00). Zu erwähnen ist noch, daß das Flip-Flop nach einem Hardware-Reset in den Grundzustand ($Q=L$, $\bar{Q}=H$) geht, da sein Reset-Eingang \bar{R} mit dem Anschluß RESET des Expansion-Port verbunden ist. Dann ist im zweiten Fall das rechte EPROM selektiert und im letzten Fall das Modul eingeschaltet. Bei einem Software-Reset (SYS 64738) bleibt der Zustand des Flip-Flops, und damit der Platine, erhalten.

Bei der Eigenherstellung ist zu beachten, daß sich auf den Platinen einige Durchkontaktierungen (Bild 2 und 3) befinden. Außerdem sollten Sie darauf achten, ob die Bauteile mit der Platinenoberseite oder Unterseite verlötet werden müssen. Bestücken Sie die Platinen zuerst mit den Kondensatoren, und danach mit den Fassungen und Bauteilen (Bild 4). Achten Sie bei den Kondensatoren auf die richtige Polung. Bei der Verwendung von EPROMs des Typs 2732, die vier Pins weniger als ein 2764 haben, werden diese nicht bündig mit der Kerbe in die Fassung gesetzt. Die zwei Pins auf beiden Seiten der Kerbe bleiben frei.

(Markus Braun/aw)

Der Hexer — Zusatz tastatur für den MSE

Als Programmierfan stehen Sie auch häufig vor dem Problem, seitenweise Zahlen eingeben zu müssen, zum Beispiel MSE-Listings. Meist erfolgen die Angaben im Hexadezimalsystem, das die Ziffern 0 bis F umfaßt. Deshalb zeigen wir Ihnen, wie man sich einfach eine Hex-Tastatur selbst bauen kann.

Der Bau dieser Hex-Tastatur dürfte dem geübten Bastler keine Schwierigkeiten bereiten. Dem handwerklich weniger Erfahrenen, können wir in Kürze eine Bezugsadresse für die entsprechende, fertig aufgebaute Tastatur geben.

Das wichtigste Kriterium der Tastatur war die Kompatibilität zu allen existierenden Programmen. Es mußte also eine Hardwarelösung geschaffen werden, die keinen Speicherplatz benötigt.

Der Verbindungsstecker zwischen Tastatur und C 64-Platine besitzt 19 Kontakte und eine sogenannte Codierungslücke. Diese Lücke in der Stiftleiste verhindert ein falsches Aufstecken. Die 19 Kontakte

Wer schon seitenweise Hex-Code eingetippt hat, weiß wie nervtötend das auf der Schreibmaschinentastatur des C 64 sein kann. Wir stellen Ihnen deshalb eine komfortable Selbstbau-Hex-Tastatur vor.

lassen sich in drei Gruppen aufteilen (Tabelle 1).

1. Die Kontakte 1 und 3 sind der RESTORE-Taste zugeordnet. Der zweite Kontaktstift fehlt. Die entsprechende Bohrung im Stecker ist verschlossen. Diese Codierung verhindert ein falsches Aufstecken des Tastatursteckers.

2. Die Stifte 5 bis 12 sind an die acht Reihen (Rows) der Tastaturmatrix (Bild 1) angeschlossen.

3. Die Stifte 13 bis 20 geben den logischen Zustand der acht Matrixspalten (Columns) wieder.

Jede Taste verbindet eine Spalte der Matrix mit einer Reihe der Matrix. Bei acht Reihen und acht Spalten gibt es folglich $8 \times 8 = 64$ Möglichkeiten der Verschaltung, die der C 64 vollständig ausnutzt. Bild 1 zeigt,

welche Matrixpunkte die einzelnen Tasten einnehmen.

Alle Tastenkontakte sind im Ruhezustand geöffnet und werden durch Tastendruck geschlossen. Will man eine Taste von außen ersetzen, braucht nur die entsprechende Matrixspalte und Reihe über einen Taster miteinander verbunden zu werden.

Der Anschluß der Zusatz tastatur erfolgt über eine Adapterleiste (Bild 2).

Nach dem Öffnen des C 64 (Garantieverlust!) wird der Tastaturstecker abgezogen und zusammen mit dem Zwischenstück wieder aufgesteckt. Im einzelnen besteht der Adaptersockel aus einem kleinen Stück Lochrasterplatine mit Kupferbahnen im Abstand von 2,54 mm.