
AC1 – ROM – BANK

Allgemeine Beschreibung:

In der letzten Variante der EPROM – Bank hatte ich einen IC eingesetzt, den es zwar gibt, aber nicht bei den Händlern im Angebot ist. Aus diesem Grund habe ich auf die erste Variante zurückgegriffen und die Schaltung noch verbessert. Dadurch werden die 2k- und 4k-Eproms auch exakt selektiert.

Schaltung:

Diese Schaltung wurde so entworfen, dass die Leiterplatte zweimal aufgebaut werden kann. Somit stehen dann folgende Bänke zur Verfügung:

(bei einer Lp)	(bei zwei Lp)
4 x 8k-Byte	8 x 8k Byte
2 x 4k-Byte	4 x 4k-Byte
2 (1) x 2k-Byte	4 (3) x 2k-Byte

1x 2k-Byte wird für das Programm der ROM-Bank selbst belegt, da im Monitor kein Platz dafür ist. Dort wird nur eine Umladeroutine untergebracht, wofür der Platz noch reicht.

Auf der ersten Lp **muss** J1/2 gebrückt sein. Somit werden die Bänke F0 bis F7 angesprochen. Auf der zweiten Lp wird der Jumper bei J2/3 gesteckt. Somit werden die Bänke F8 bis FF angesprochen. Es kann auch eine Lötbrücke gesetzt werden.

Zur Einblendung der EPROM's wird der Bereich # A000 bis # BFFF verwendet. Da dort keine Programme laufen. Beim Einblenden können auch in diesen Bereich Programme geladen werden, da der Zugriff auf den EPROM nur im Lesezyklus entsteht. Beim Schreiben erfolgt der Zugriff in den RAM-Bereich.

Programm:

Im Monitor V.8 werden ab #0F22 folgende Bytes eingetragen (**ROT**).

```
#0F20 E1 C9 00 09 75 0D 3E F6 D3 0E 11 00 19 21 00 A0
#0F30 01 00 07 ED B0 3E 00 D3 0E C3 00 19
```

Diese Ruoutine wird mit „u“ angesprochen. Daraufhin wird das Umladeprogramm vom EPROM auf **Platz F6** in den Bereich #1900 umgeladen und angesprungen. Dieser Bereich wurde gewählt, da die Programme meist ab Adresse #2000, #4000 oder ab #E000 geladen werden.

Das nachfolgend beschriebene Programm wird auf einen 2716 gebrannt und **muss** auf dem **EPROM - Platz F6** gesteckt werden. Wo man die anderen EPROM's steckt ist jeden selbst überlassen. Wichtig ist dann nur der richtige Eintrag im Programm.

AC1 – ROM – BANK

Der erste Teil des Programms bleibt unverändert. Ab Adresse #1C00 werden dann die jeweiligen Einträge für die einzelnen Bänke vorgenommen. Es ist zu jeder Zeit möglich Bänke nach zu bestücken und den EPROM mit den Einträgen nach zu brennen, da auf den leeren Bänken die Bytes auf #FF stehen.

Ab Adresse # 1C00 sind folgende Einträge zu finden:

```
1C00 FF 41 FF FF FF FF FF FF FF FF FF FF FF FF FF
1C10 FF FF FF FF FF FF FF FF FF FF FF FF FF FF FF
1C20 FF 42 FF FF .....
```

Auf Steckplatz F3 wird z.B. EDAS*4 gesteckt – dann wird folgendes eingetragen:

```
1C00 00 41 20 45 44 41 53 2A 34 20 20 20 20 20 20
1C10 F3 40 00 20 00 A0 00 FF FF FF FF FF FF FF FF
1C20 FF 42 FF FF .....
```

00 = Freischaltung der Bank
 41 20 = Nummer und Namen des Programms
 (leere Stellen bis Zeilenende (#0F) mit #20 auffüllen, denn diese werden auf dem Bildschirm ausgegeben)
 F3 40 00 = F3 – Banknummer
 40 00 Ladeadresse des Programms
 20 00 Länge des Programms

A0 00 = Startadresse – wo das Programm steht – bei einem Programm immer #A000

Der Name des Programms kann max. 13 Zeichen lang sein.

Natürlich habe ich auch bedacht, das vielleicht schon alle 8k-Plätze belegt sind und man aber noch ein Programm von 8k Länge hat, oder ein Programm länger ist als 8k-Byte, wie zum Beispiel das 12k-Basic, was aber sogar über 12k-Byte ist.

In dem Fall muss man ja 2 EPROM's nehmen. Bleiben wir bei dem Beispiel des 12k-Basic. Da muss das Programm auf 2 x 2764 gebrannt werden. Nehmen wir an diese EPROM's werden auf die Plätze F1 und F2 gesteckt. Dann sieht der Eintrag wie folgt aus. (Ich belasse den Eintag von EDAS*4 im Programm ab #1C00, da wir diesen ja schon gesteckt hatten)

```
1C00 00 41 20 45 44 41 53 2A 34 20 20 20 20 20 20
1C10 F3 40 00 20 00 A0 00 FF FF FF FF FF FF FF FF
1C20 00 42 20 31 32 2D 4B 20 2D 20 42 41 53 49 43 20
1C30 F1 40 00 20 00 A0 00 2B F2 60 00 14 00 A0 00 FF
1C40 FF 43 FF FF .....
```

2B (+) = dadurch erkennt das Programm, das noch ein zweiter Teil geladen werden muss.

AC1 – ROM – BANK

Danach stehen die Banknummer, die Ladeadresse und die Startadresse für den zweiten Teil und die Länge. Somit wird das komplette Programm automatisch geladen.

Es ist auch möglich, dass auf einen EPROM zwei Programme gespeichert sind. Dann sieht der Eintrag als Beispiel wie folgt aus:

```
1C00 00 41 20 45 44 41 53 2A 34 20 20 20 20 20 20
1C10 F3 40 00 10 00 A0 00 FF FF FF FF FF FF FF FF
1C20 00 42 20 54 65 73 74 20 20 20 20 20 20 20 20
1C30 F3 50 00 08 00 B0 00 FF FF FF FF FF FF FF FF
1C40 FF 43 FF FF
```

00 = Bank freischalten
 42 20 = Nummer und Name des Programms
 F3 = Banknummer – man beachte, das es die gleiche ist wie auf Adresse #1C10
 50 00 = Ladeadresse
 08 00 = Länge des Programms – hier 2kB
 B0 00 = Startadresse, wo das Programm steht – hier #B000, da ab #A000 bis #AFFF bereits das andere Programm steht.

Wenn ein Programm auf dem EPROM ist, ist die Ladeadresse immer #A000!!

Ich habe mit Absicht keinen automatischen Ansprung der Programme vorgesehen, da viele Programme auch einen Warmstartadresse habe, dadurch wäre es nicht ratsam, wenn man z.B. schon die Assemblerliste geladen hat und dann das EDAS vom EPROM liest, dass dieses dann ab #4000 anspringt. Dann wären die Daten verloren.

Ab Adresse #1C00 sind bereits die Plätze für die Einträge 41 – 53 vorgesehen. Somit dürfte es keine Probleme bei den Eintragungen geben.

Mit den Eintragungen ab Adresse #1C00 fortlaufend beginnen. Also nicht bei 41 den ersten Eintag und den nächsten bei 46, denn dann wird der Eintag auf 46 nicht erkannt, da die dazwischen liegenden Plätze nicht freigeschaltet sind.

Nach dem Eintragen der Bänke ab Adresse #1C00 einen EPROM brennen und diesen dann auf die Bank **F6** stecken und dann kann's losgehen.

Ich bin der Meinung, dass zwei dieser Leiterplatten für die vorhanden Programme reichen dürften.

Es hat sich gezeigt, dass das Umladen viel Zeit spart (ca. 2 sec. für 8k-Byte) und das lästige Laden vom Band entfällt. Des Weiteren kann man schnell zwischen den wichtigsten Programmen wechseln.

Für Hinweise zur Verbesserung bin ich dankbar.

Andreas
DL9UNF

22.04.2010