



1802 EPROM- programmeerprogramma

H. B. Stuurman

Het EPROM-programmeerapparaat dat door de heer Van de Peijl is beschreven in RB december 1980 leek mij een nuttige uitbreiding van mijn Cosmac microcomputer-systeem. Het programmeerapparaat is door mij gebouwd als een zelfstandige eenheid, dus met voeding. De voeding is in een TEKOKastje gemonteerd, het programmeerprintje is er met enkele strookjes dubbelzijdig kleefband op vastgezet.

Wat mij enigszins verbaasde is dat

het printje niet voorzien is van bevestigingsgaten op de hoeken en de curieuze printsteek die sommige componenten hebben. Dit doet echter geen afbreuk aan de schakeling, die voortreffelijk werkt.

Aan de koperzijde heb ik verder onder de EPROM-voet 2 extra ont-koppelcondensatoren van 22 nF aangebracht voor Vcc en Vpp. Met een stukje 9-aderig bandkabel wordt het programmeerapparaat met de computer verbonden.

Korte beschrijving

Het EPROM-programmeerapparaat wordt gestuurd door een speciaal daarvoor geschreven programma. Dit programma is uitsluitend bestemd voor EPROM's van het type 2716 (5 V). Getracht is het programma – en daarmee de EPROM – zo goed mogelijk te beveiligen, zonder echter in extremiteiten te vervallen. Zo wordt bijv. de adresruimte getest: is deze negatief of groter dan $7FF_H$ dan wordt niet ►



EPROM

verder gegaan, maar de fout wordt gemeld, waarna de correcte adressen kunnen worden ingetikt. Ditzelfde gebeurt ook bij foutieve inbreng, dat wil zeggen een letter in plaats van een cijfer.

Van het geheugenblok, waarvan men de inhoud betrekking wil laten hebben op de EPROM in het programmeerapparaat, corresponderen altijd de adressen A0 t/m A10. Het eerste adres in de EPROM is daardoor altijd X000_H of X800_H.

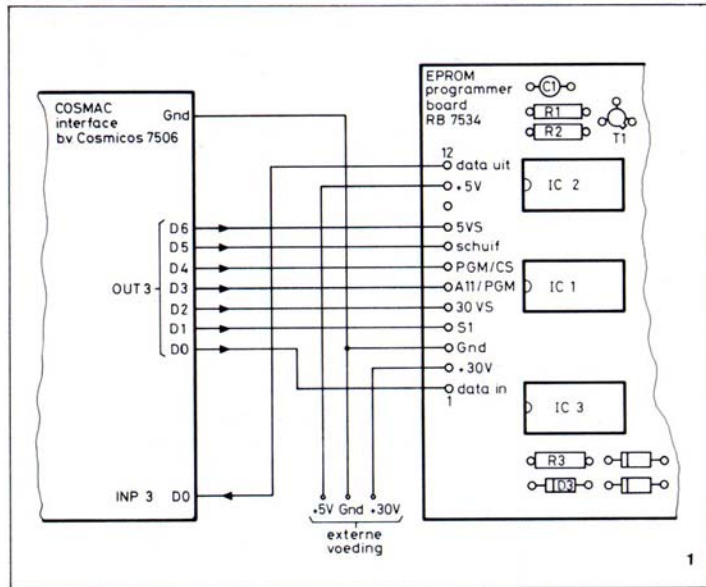
In het eerste geval is het bestreken gebied X000_H t/m X7FF_H; in het tweede geval X800_H t/m XFFF_H. X kan een willekeurig 4K blok zijn (dat wil zeggen een hexa-decimaal cijfer 0 t/m F).

Nadat het programma is gestart meldt het zich met een introductie en wordt u verzocht de EPROM in het programmeerapparaat te plaatsen. Vervolgens wordt gevraagd om het begin- en eindadres van het relevant zijnde geheugenblok. Het eindadres is inclusief. Als slechts één plaats wordt geprogrammeerd, zijn begin- en eindadres gelijk.

Als bij het intikken van een adres een fout wordt gemaakt, kunt u doorgaan tot de laatste 4 cijfers juist zijn alvorens op carriage return te drukken. Vervolgens wordt onderzocht of de adressen in een juiste relatie tot elkaar liggen. Is dat het geval dan kan worden gekozen uit 3 mogelijkheden.

1. READ
2. VERIFY
3. PROGRAM

Bij Read wordt de inhoud van de EPROM gekopieerd in het aangegeven geheugenblok. Vanzelfsprekend dient dit wel RAM te zijn.



Bij Verify wordt de inhoud van de EPROM vergeleken met die van het geheugenblok. Fouten worden gemeld door middel van het adres, de foute byte en de correcte byte. Met Program wordt het geheugenblok in de EPROM weggeschreven. Afhankelijk van de lengte kan dit even duren. Program wordt altijd gevolgd door Verify. Program en Verify kunnen betrekking hebben op RAM of ROM.

Het is zeer eenvoudig programma's in een EPROM te multipliceren. Met Read leest men het programma in RAM; vervolgens kunnen een onbeperkt aantal EPROM's worden geprogrammeerd. Het enige dat men moet hebben is tijd. De computer doet het werk.

Het is eveneens eenvoudig te verifiëren of een EPROM goed is gewist. In dat geval moet op ieder adres FF_H staan. Door de RAM-buffer te vullen met FF_H en vervolgens de EPROM hiermee te verifiëren komt een slecht gewiste EPROM snel aan het licht.

De RAM-buffer kan met FF_H worden gevuld door Read zonder EPROM in het programmeerapparaat. (Open ingangen van TTL IC's zijn „1“.)

Tot slot een waarschuwing. Plaats de EPROM pas in het apparaat als u daartoe wordt verzocht. Het programma heeft dan met zekerheid

het apparaat spanningsloos gemaakt.

Aansluiting op de computer

Om het programmeerapparaat op de computer aan te sluiten moet men beschikken over een 8 bits uitgangspoort, een ingangspoort en een Gnd-aansluiting.

Zeer geschikt is bijv. het Cosmos

Tabel 1

OUT 3	D0	- data in
	D1	- S1
	D2	- 30 VS
	D3	- A11/PGM
	D4	- PGM/CS
	D5	- schuif
	D6	- 5 VS
INP 3	D7	- niet verbinden
	D0	- data uit
	D1	
	D2	
	D3	
	D4	- niet verbinden
	D5	
D6		
D7		
Gnd	- massa	

EPROM



interface kaartje dat beschreven is in RB september 1980. Alle vereiste signaalaansluitingen zijn daarop aanwezig.

In verband met de bandkabel is de indeling van de databitjes anders gekozen dan door de heer Van de Peijl (zie tabel 1). Van de uitgangspoort worden D0 t/m D6 gebruikt en van de ingangspoort alleen D0. Verder is de Gnd van het programmeerapparaat verbonden met Gnd op de interfacekaart. Door de gekozen indeling van de bitjes liggen tenminste D1 t/m D6 van de uitgangspoort naast elkaar (afb. 1). De interfacekaart dient te zijn geprogrammeerd voor N-lijn 3 (OUT 3 = 63; INP 3 = 6B). Dit is overigens gemakkelijk te wijzigen.

Programma

Het EPROM-programmeerprogramma kan draaien op elke 1802 microcomputer die is uitgerust met een ASCII-keyboard en videomonitor of printer. Aan geheugenruimte moet minstens 4K aanwezig zijn voor het Programma en de RAM-buffer.

Het EPROM-programma loopt van 0000_H t/m 02FF_H.

Het is page relocatable, dat wil zeggen het kan op iedere gewenste pagina beginnen. Het bevat een eigen stackje en een eigen I/O op 300 baud met Q en EF4. Het programma dient in Register 0 te starten. Tijdens „Mark” geldt dat EF4 = 0, dat wil zeggen pen 21 van de 1802 is laag. (Dit komt overeen met EF4 = 1 in de instructieset.)

Programma 1

```
1802 EPROM PROGRAMMER FOR 2716 SINGLE 5V
AUTHOR: H. P. STUURMAN
```

moet zijn, n 0A. Het 02B4.

handen is, wijze wor-

R3 en ke- ata-uitwis- l. oor Type en voor Type

geprogrammeerd voor N-lijn 3 (OUT 3 = 63; INP 3 = 6B). Dit is overigens gemakkelijk te wijzigen.

Programma

Het EPROM-programmeerprogramma kan draaien op elke 1802 microcomputer die is uitgerust met een ASCII-keyboard en videomonitor of printer. Aan geheugenruimte moet minstens 4K aanwezig zijn voor het Programma en de RAM-buffer.

Het EPROM-programma loopt van 0000_H t/m 02FF_H.

Het is page relocatable, dat wil zeggen het kan op iedere gewenste pagina beginnen. Het bevat een eigen stackje en een eigen I/O op 300 baud met Q en EF4. Het programma dient in Register 0 te starten. Tijdens „Mark” geldt dat EF4 = 0

Afb. 1 Zo wordt de programmeerprint met de Cosmac interface verbonden.

Programma 1 Volledige listing van een 1802 programma om de EPROM-schakeling aan te drijven.

Testrun

```
1802 EPROM PROGRAMMER
FOR 2716 .SINGLE 5V
VER 0

PUT EPROM IN SOCKET!

START? 0000
FINISH? 07FF
READ?
VERIFY?
PROGRAM?
ENTER R, V OR P! P
WAIT FOR VERIFY!
ERROR AT #06B3 FF MUST BE 45

END!
```

Voor inverted EF4 (ELF II) dienen de volgende adressen te worden gewijzigd.

Adres Byte

```
02D8 37 (B4) → 3F (BN4)
02DC 37 (B4) → 3F (BN4)
02E6 3F (BN4) → 37 (B4)
```

```
0160 17A6 FF00 81FB 023A 8F33 79E8 D512 304B;
0170 08D7 089B F676 7676 D703 8BD7 09E4 4326;
0180 F808 B3FB 19A3 2393 3A5n
```

```
1802 EPROM PROGRAMMER
FOR 2716 .SINGLE 5V
VER 0

PUT EPROM IN SOCKET!

START? 0000
FINISH? 07FF
READ?
VERIFY?
PROGRAM?
ENTER R, V OR P! P
WAIT FOR VERIFY!
ERROR AT #06B3 FF MUST BE 45

END!
```

Voor inverted EF4 (ELF II) dienen de volgende adressen te worden gewijzigd.

Adres Byte

Na carriage return (0D) volgen 2 nul karakters.

Als dit na Line feed (0A) moet zijn, verander dan M 02B0 in 0A. Het aantal nul's staat op M 02B4.

Indien een eigen I/O voorhanden is, kan deze op de volgende wijze worden gekoppeld.

Adres

```
006B F8 (LDI)
006C high byte „Type”
006F low byte „Type”
0087 F8 (LDI)
0088 high byte „Read”
008B low byte „Read”
```

Type en Read starten in R3 en keren terug met D5. De data-uitwisseling geschiedt in R(F)1.

Beschikbare registers voor Type en Read: RC, RE, R(F)0 en voor Type bovendien R(D)0.

Dit is in overeenstemming met Utility 4 (UT 4) van RCA. Het programma bevat **geen** SCRT-routines!

Als dit na Line feed (0A) verander dan M 02B0, aantal nul's staat op M 02B4. Indien een eigen I/O voorhanden is, kan deze op de volgende wijze worden gekoppeld.

Adres

```
006B F8 (LDI)
006C high byte „Type”
006F low byte „Type”
0087 F8 (LDI)
0088 high byte „Read”
008B low byte „Read”
```

Type en Read starten in R3 en keren terug met D5. De data-uitwisseling geschiedt in R(F)1. Beschikbare registers voor Type en Read: RC, RE, R(F)0 en bovendien R(D)0.

Dit is in overeenstemming met Utility 4 (UT 4) van RCA. Het programma bevat geen SCRT-routines!