



UNIUNEA EUROPEANĂ



GUVERNUL ROMÂNIEI



Instrumente Structurale  
2007-2013



# Platformă de e-learning și curriculum e-content pentru învățământul superior tehnic

## Proiectarea cu Microprocesoare

### **17. Proiectarea memoriei EEPROM**

## MEMORIA EPROM

Pe modulul de bază al structurii ce se va proiecta s-a prevăzut posibilitatea cuplării unei memorii permanente EPROM de capacitate între 12K octeți și 64K octeți. De fapt s-au prevăzut 6 socluri de 28 de pini care permit cuplarea unui număr de 6 capsule de memorie EPROM. S-a ales această organizare pentru scop didactic.

În funcție de tipul de memorie EPROM folosit se realizează capacitatea memoriei astfel:

Tip memorie	Capacitate	Spațiu de adresare
2716 (2K x 8)	12K	0FD000H - 0FFFFFFH
2732 (4K x 8)	24K	0FA000H - 0FFFFFFH
2764 (8k x 8)	48K	0F4000H - 0FFFFFFH
27128 (16K x 8)	64K	0F0000H - 0FFFFFFH

Pentru a introduce diferite tipuri de EPROM - uri în soclurile prevăzute pe modulul de bază este necesar să se prevadă un mecanism hardware de comutare a adresei astfel încât să se asigure liniile de adrese corespunzătoare capsulei de memorie.

Având în vedere spațiul de adresare în care este plasată memoria EPROM rezultă urmatorul tabel de utilizare a liniilor de adresă:

Tip memorie	A19	A18	A17	A16	A15	A14	A13	A12	A11 - A1
2716	1	1	1	1	x	x	selecție capsulă	Adresă pt. capsulă	
2732	1	1	1	1	x	selecție capsulă	Adresă pt. capsulă		
2764	1	1	1	1	selecție capsulă	Adresă pt. capsulă			
27128	1	1	1	1	sel. capsulă	Adresă pt. capsulă			

Se observă că în cazul utilizării memoriei 2716 sau 2732, biții de adresă ADR15:14 respectiv ADR15 nu sunt utilizați, ceea ce conduce la faptul că informația din memoria EPROM se regăsește în mai multe spații de adresare.

Astfel memoria EPROM realizată cu capsule 2716 poate să fie adresată în spațiul:

0FD000H - 0FFFFFFH sau

0F1000H - 0F3FFFH

0F5000H - 0F7FFFH

0F9000H - 0FBFFFH

iar când este realizată cu capsule 2732 poate să fie adresată în spațiile:

0FA000H - 0FFFFFFH sau

0F2000H - 0F7FFFH

De asemenea, se observă că în cazul utilizării memoriei 27128, la decodificatorul de adresare participă numai linia ADR15 ceea ce conduce la posibilitatea de adresare numai a 4 circuite din cele 6.

Pentru implementarea în memoria EPROM a unui set de programe de depanare selectabile prin intermediul unor comutatoare, trebuie să se prevadă o posibilitate hardware de interschimbare a unor zone de memorie (din punct de vedere al adresabilității). Este necesar ca fiecare program de depanare (de lungime maximă 256 octeți) aflat fizic în zona de memorie 0FD000H - 0FDFFFH, indiferent de tipul de memorie utilizat, în momentul selectării pentru execuție să fie văzut de către microprocesor în spațiul 0FFF00H - 0FFFFFFH. Acest lucru este necesar pentru a putea fi lansat direct în execuție în momentul pornirii sistemului.

În cele ce urmează se prezintă modul de implementare a memoriei EPROM pentru structura proiectată.

### SCHEMA BLOC A MEMORIEI EPROM

Având în vedere cele precizate anterior rezultă următoarea schemă bloc pentru memoria EPROM a structurii propuse.

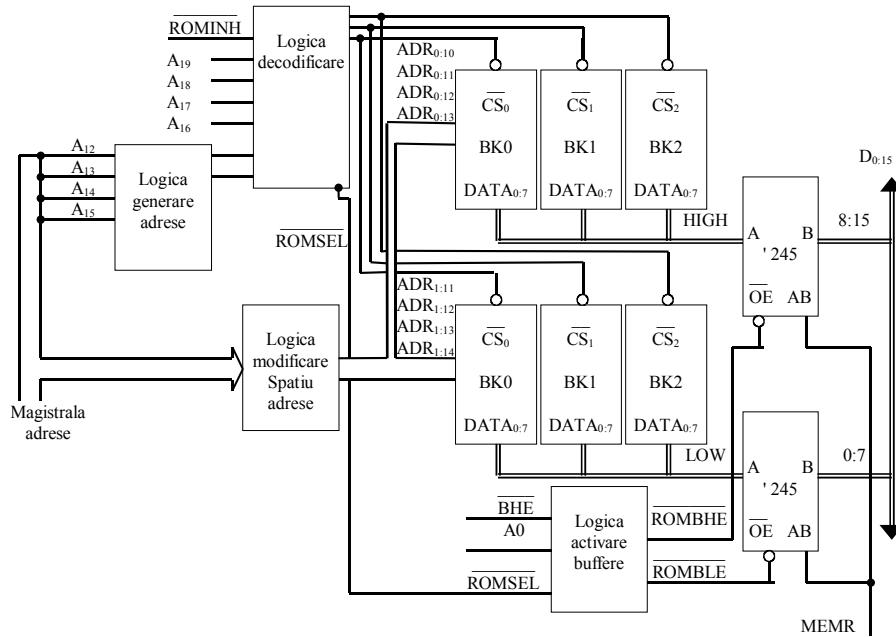


Figura 4.16. Schema bloc a memoriei EPROM

### LOGICA DE GENERARE ADRESE

Are scopul de a asigura, pentru decodificatorul spațiului de adrese, liniile de adrese corespunzătoare tipului de memorie utilizat. Constă dintr-o schemă de multiplexare ca în figura 4.17.

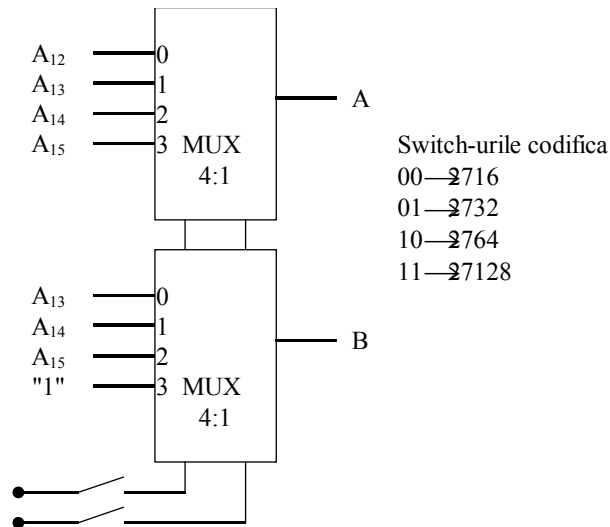


Figura 4.17. Logica de generare adrese

În funcție de *strap*-urile S36, S37 se asigură următoarele adrese pentru decodificarea spațiului de adrese.

S36	S37	Tip memorie	Intrare B	Intrare A
da	da	2716	A <sub>13</sub>	A <sub>12</sub>
da	nu	2732	A <sub>14</sub>	A <sub>13</sub>
nu	da	2764	A <sub>15</sub>	A <sub>14</sub>
nu	nu	27128	1	A <sub>15</sub>

## LOGICA DE DECODIFICARE

Asigură selecția celor trei subspații de memorie prin generarea semnalelor de selecție:

$$\overline{CS}_j = DCD_{j+1}(A_{i+1}, A_i) + \overline{ROMSEL}$$

$$\overline{ROMSEL} = \overline{ROMINH} \cdot A_{19} \cdot A_{18} \cdot A_{17} \cdot A_{16}$$

unde  $j=0, 1, 2$  pentru memoriile de tipul: 2716, 2732, 2764

$j=1, 2$  pentru memoriile de tipul 27128

respectiv

$i = 12$  pentru memoriile de tipul 2716

$i = 13$  pentru memoriile de tipul 2732

$i = 14$  pentru memoriile de tipul 2764

$i = 15$  pentru memoriile de tipul 27128

$\overline{CS}_i$  - reprezintă semnalele de selectare ale circuitelor de memorie corespunzătoare

$\overline{ROMINH}$  - este un semnal de pe magistrala de extensie a sistemului care poate inhiba memoria EPROM.

Dacă  $\overline{ROMINH} = 0$  memoria EPROM de pe modulul de bază este inhibată, nu poate fi adresată de către microprocesor,  $\overline{ROMINH} = 1$  memoria EPROM de pe modulul de bază este accesibilă microprocesorului.

$\overline{ROMSEL}$  - este un semnal care este activ în momentul în care memoria EPROM nu este inhibată și liniile de adresă specifică o locație din spațiul alocat memoriei EPROM.

Pentru memoria de tip 27128 se consideră  $A_{16} = 1$  ca intrare în decodificator, deoarece se utilizează numai 64K. În acest caz numai CS1 și CS2 pot să fie active. Deci programul din memoria permanentă trebuie să fie plasat numai în 4 din cele 6 circuite de memorie.

## LOGICA DE GENERARE *BUFFER*-E ACCES PE MAGISTRALĂ

Rolul acestei scheme logice este de a asigura activarea pe magistrala de date a informațiilor citite din memoria EPROM.

Citirea se poate face pe 16 biți sau 8 biți în funcție de ciclul mașină curent, pe baza valorilor semnalelor  $\overline{BHE}$  și  $A_0$ .

$$\overline{ROMHBE} = \overline{ROMSEL} + \overline{HIGHBE} \quad - \text{ activare } buffer \text{ date } D_{8:15}$$

$$\overline{ROMLBE} = \overline{ROMSEL} + A_0 \quad - \text{ activare } buffer \text{ date } D_{0:7}$$

$$\overline{HIGHBE} = \overline{BHE} \cdot A_0$$

*Buffer*-ele de date sunt orientate dinspre memorie spre magistrala de date de către semnalul MEMR.

Schema detaliată de implementare a memoriei EPROM este prezentată în figura 4.18.

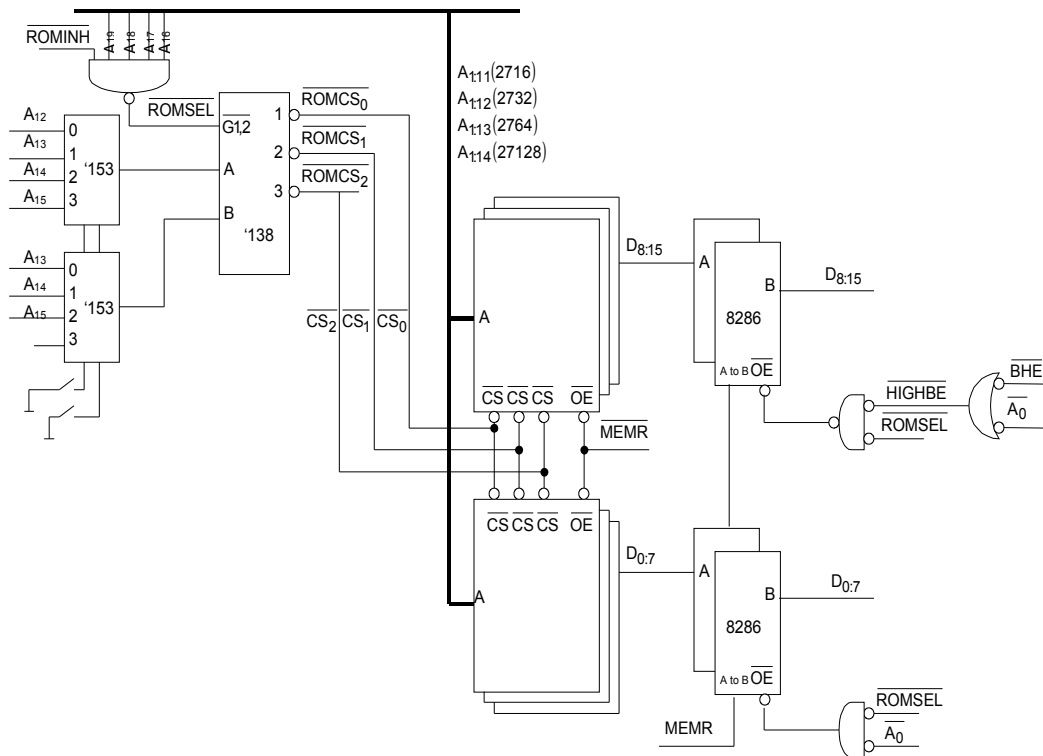


Figura 4.18. Schema detaliată a memoriei EPROM

## LOGICA DE MODIFICARE A SPAȚIULUI DE ADRESARE.

Așa cum s-a prezentat fiecare dintre programele de depanare trebuie să fie lansate în execuție la punerea sub tensiune a sistemului după ce în prealabil a fost selectat cu ajutorul unor comutatoare.

Pentru aceasta fiecare program de depanare, în momentul selecției sale trebuie să fie văzut de către microprocesor în ultimii 256 de octeți ai spațiului de adresare de 1Mo, adică în zona 0FFF00H - 0FFFFFFH.

Modificarea spațiului de adresare trebuie făcută ca în figura 4.19.

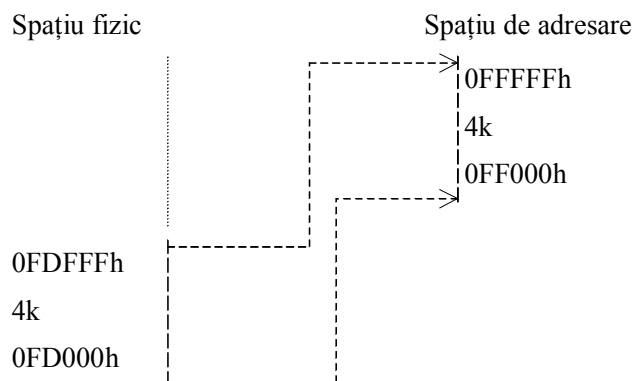


Figura 4.19. Modificarea spațiului de adresare

Comutarea acestor două zone de memorie se realizează prin modificarea bitului de adresă  $A_{13}$  utilizat de logica de decodificare astfel:

$$ADR_{13} = A_{13} \oplus DSW_1 7$$

După ce s-au comutat aceste două zone de memorie, de 4K octeți este necesar ca subzone de 256 octeți să fie translatare în spațiul 0FFF00H – 0FFFFFFH.

Această translatare se face cu ajutorul comutatoarelor de selecție a programelor de test: DSW1 modificând liniile de adrese care ajung la circuitele de memorie în modul următor:

$$ADR_i = A_{i+1} \text{ pentru } 0 \leq i \leq 6, i=13;$$

$$ADR_i = A_{i+1} \oplus DSW_1 7 \text{ pentru } 7 \leq i \leq 10$$

$$ADR_{12} = A_{13} \oplus DSW_1 7$$

unde  $A_i$  - sunt liniile de adrese de pe magistrala sistemului,  $A_i$  - sunt liniile de adrese ale circuitelor de memorie,

DSW<sub>i</sub> – comutator de pe placa de bază (“on” reprezintă 1 logic, “off” reprezintă 0 logic).

Modul în care se translatează programele de depanare ca să ajungă în ultima zonă de 256 de octeți este prezentat în figura 4.20.

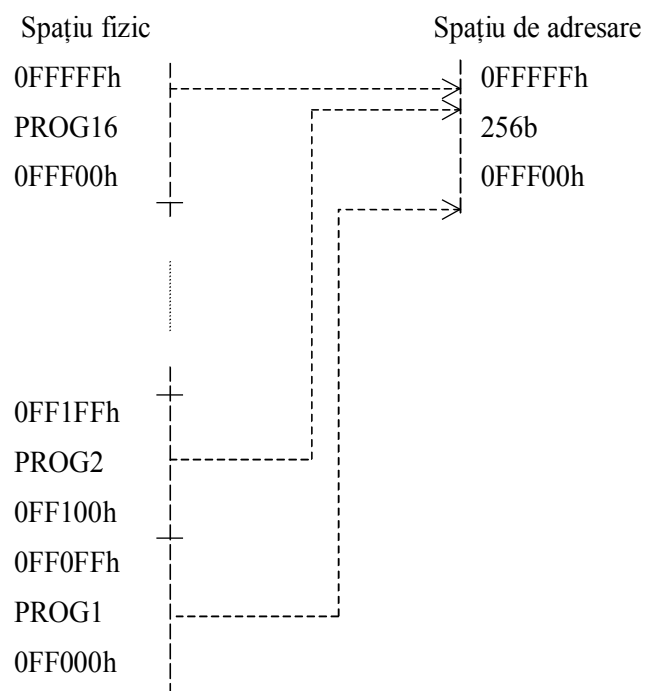


Figura 4.20. Modul de traducere a programelor de depanare în memoria sistemului