

## 5. MEMORII SEMICONDUCTOARE

### 5.0. Introducere

Sunt circuite integrate care au ca scop stocarea informatiei sub forma binara. Memoriile reprezinta categoria de circuite integrate cu cea mai mare raspândire si vânzare în lume în prezent. Ele se gasesc sub forma distincta sau intra în componenta altor circuite integrate cum ar fi microprocesoare, microcontrolere, FPGA, circuite de telecomunicatii, etc.

### 5.1. Clasificarea memoriilor

*Traditional* se deosebesc doua categorii largi de memorii:

- memorii care pot fi doar citite, denumite ROM (*Read Only Memory*);
- memorii care pot fi si scrise si citite, denumite RAM (*Random Access Memory*).

În functionare normala, memoriile ROM se pot doar citi. Înscrisura informatiei într-o astfel de memorie se realizeaza într-o etapa anterioara utilizarii ei printr-o procedura numita **programare** (*programming* sau mai corect *burning*).

Memoriile RAM ar trebui denumita corect RWM (*Read Write Memory*) deoarece se pot citi si scrie în orice moment si la orice adresa, ele fiind memorii la care accesul poate fi aleator.

Din punct de vedere al *tehnologiei de fabricatie* exista memorii cu tranzistoare bipolare, MOS, CMOS respectiv BiCMOS.

Memoriile *volatile* își pierd continutul la deconectarea alimentarii, pe când memoriile *nevolatile* își mentin informatia stocata si fara prezenta tensiunii de alimentare. În general memoriile ROM sunt nevolatile, iar cele RAM sunt volatile.

În functie de *organizarea magistralei de date*, deosebim memorii *paralel* si *seriale*. În prezent se fabrica memorii seriale EEPROM si SRAM.

În functie de *modul de utilizare*:

- memorii de program.
- memorii de date.

În general memoriile ROM sunt folosite pentru stocarea programelor iar memoriile RAM a datelor.

Clasificarea în memorii RAM si ROM mai are în prezent doar o semnificatie academica, deoarece memoriile EEPROM si Flash pot fi citite si scrise în functionare normala la orice adresa (comportament de RAM), iar memoriile SRAM nevolatile (NV SRAM) stocheaza informatia peste 10 ani fara alimentare externa (comportament de ROM).

Câteva caracteristici importante ale principalelor tipuri de memorii produse în prezent sunt prezentate comparativ în tabelul 5.1. Reprogramabilitatea si posibilitatea de programare în montaj sunt specifice doar unor tipuri de memorii ROM, de aceea câmpurile aferente lor au fost hasurate în cazul memoriilor RAM.

O memorie ideala se caracterizeaza prin capacitate ridicata de stocare (si implicit densitate mare de integrare), timp de acces redus, programabilitate în circuit, informatie nevolatila, consum de curent scazut si cost redus. Categoriile de memorii din tabelul 5.1 au doua sau mai multe avantaje din cele de mai sus – singura care le îndeplineste pe *toate* fiind însa memoria Flash.

Comparatie între caracteristicile diverselor tipuri de memorii

Tipul memoriei	Capacitate	Volatila	Celula elementara cu 1 tranzistor	Reprogramabila	Programare în montaj
ROM	Mare	NU	DA	NU	NU
PROM	Mica	NU	DA	NU	NU
EPROM	Medie	NU	DA	DA	NU
OTP	Medie	NU	DA	NU	NU
EEPROM	Medie	NU	NU	DA	DA
Flash	Mare	NU	DA	DA	DA
NV RAM	Medie	NU	NU		
SRAM	Mare	DA	NU		
DRAM	Foarte mare	DA	DA		

## 5.2. Unitati de masura a informatiei

Bitul este cea mai mica unitate de masura a informatiei. Un *bit* este prin definitie masura cantitatii de informatie necesara eliminarii nedeterminarii la aruncarea cu banul. În algebra booleana un bit poate memora un 0 sau un 1. Multiplii bitului (notat cu b) sunt kilobitul, megabitul, gibabitul si terabitul:

$$1 \text{ kb} = 2^{10} \text{ b} = 1024 \text{ b}$$

$$1 \text{ Mb} = 2^{20} \text{ b} = 1024 \text{ kb} = 1.048.576 \text{ b}$$

$$1 \text{ Gb} = 2^{30} \text{ b} = 1024 \text{ Mb} = 1.073.741.824 \text{ b}$$

$$1 \text{ Tb} = 2^{40} \text{ b} = 1024 \text{ Gb} = 1.099.511.627.776 \text{ b}$$

În echipamentele numerice informatia este organizata *fizic* si *logic*. Prin *organizare fizica* se înțelege modul de stocare propriu memoriei semiconductoare, iar prin *organizare logica* – modul de organizare a informatiei în echipament. Cele doua marimi nu sunt neaparat egale. Exista memorii integrate cu organizare multipla de obicei pe 8 sau 16 biti (memoria din figura 5.1 are o capacitate de 64 de biti si se poate organiza în 64 de cuvinte de 1 bit, 16 cuvinte de 4 biti sau 8 cuvinte de 8 biti).

Marimea logica care caracterizeaza latimea unei adrese de memorie adresata unic se numeste *cuvânt*. Cuvântul are o latime de unul sau mai multi octeti (1, 2, 4 sau 8). Mai multe locatii de memorie de la adrese succesive formeaza o *pagina* de memorie. O pagina de memorie poate avea 2 kB sau mai mult.

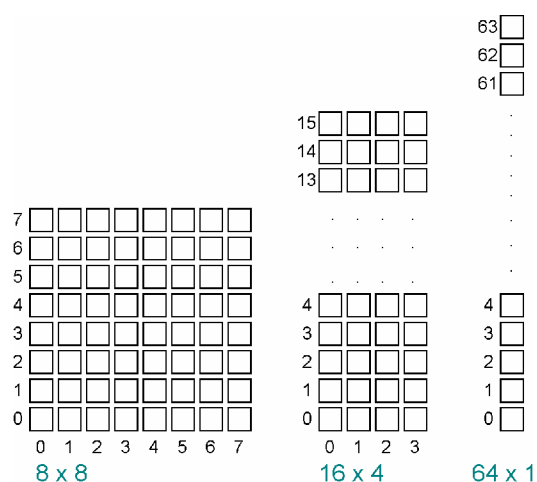


Figura 5.1. Posibilitati de organizare a unei memorii de 64 de biti.

În circuitele de memorie, informatia este organizata fizic pe 1 bit, 4 biti, 8 biti, 16 biti sau 32 de biti. O locatie de memorie de 8 biti se mai numeste octet (*byte*), iar una de 4 biti o jumatate de octet sau *nibble*. Octetul (notat o sau B) are aceeasi multipli ca si bitul – kB, MB, GB, TB.

### 5.3. Parametrii memoriei

*Timpul de acces* este cel mai important parametru dinamic al unei memorii si reprezinta durata dintre momentul adresarii memoriei pâna la momentul în care la iesirea de date devine disponibila informatia accesata. Timpul de acces este cuprins între câteva ns la cele mai rapide memorii SRAM la câteva sute de ns la cele mai lente memorii EPROM.

*Capacitatea memoriei* (exprimata de obicei în multipli de biti sau de octeti) este  $C = m \cdot k$  unde:

m = numarul de cuvinte distincte ce pot fi adresate;

k = numarul de biti ce corespund fiecarui cuvânt memorat.

### 5.4. Functionarea unei memorii

**Selectia** unui cuvânt din memorie se poate face *liniar*, utilizând un singur decodificator sau prin *coincidenta*, utilizând doua decodificatoare – unul pentru linii si altul pentru coloane (figura 5.2). Selectia prin coincidenta este singura utilizata la memoriile de capacitate mare. De exemplu în cazul unei memorii având o capacitate de 1 Mcuvânt, sunt necesare 20 de linii de adresa, iar utilizarea unui singur DCD ar presupune ca acesta sa aiba 1.048.576 linii de iesire, pe când utilizarea a doua DCD, fiecare cu 10 intrari, reduce numarul de linii de iesire pentru fiecare DCD la 1024.

În figura 5.2 matricea de memorare este presupusa bidirectionala, la selectia prin coincidenta cuvântul având o latime de 1 bit. În cazul des întâlnit în practica de cuvinte pe 4, 8 sau 16 biti modelul matricii de memorare bidirectional nu mai satisface si se utilizeaza un model tridimensional (figura 5.21).

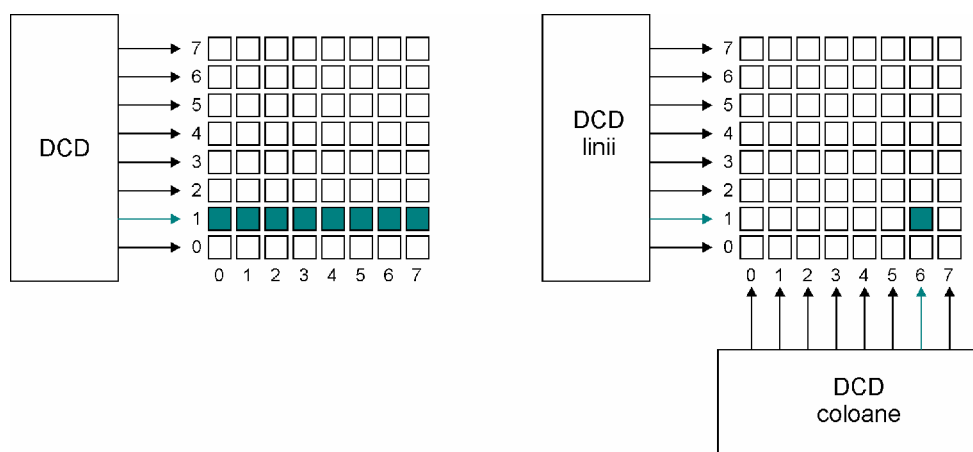


Figura 5.2. Selectia liniara si prin coincidenta.

### 5.5. Memorii ROM

Exista mai multe variante constructive pentru memoriile ROM (figura 5.3).

Memoriile ROM cu *programare prin masca* – se refera la memorii programate la producator, caz în care informatia se scrie la locul fabricatiei si nu poate fi modificata ulterior de utilizator. Continutul memoriei este conform cu dorintele utilizatorului. Aceste memorii sunt ieftine, dar se produc doar la comenzi ridicate (peste 10.000 de bucati).

Memoriile PROM (*Programmable ROM*) pot fi programate la utilizator, dar o singura data (informatia scrisa nu mai poate fi rescrisa).

Memoriile EPROM (*Erasable PROM*) sunt reprogramabile electric de catre utilizator. Circuitele EPROM se sterg prin expunere la radiatii ultraviolete, permitând apoi o noua înscriere.

Memoriile OTP (*One Time Programmable*) sunt o varianta a circuitelor EPROM încapsulate în plastic fara fereastra de cuarț. Acestea se pot programa o singura data, fiind însa mai ieftine decât omoloagele EPROM.

Memoriile EEPROM (*Electrically Erasable PROM*) – asigura utilizatorului posibilitatea programarii si stergerii electrice în circuitul final, fara a fi nevoie de expunerea structurii la radiatii ultraviolete.

Memoriile FLASH sunt o varianta de EEPROM la care procesul de stergere este foarte scurt (câteva microsecunde) – fata de zeci de ms la EEPROM, respectiv zeci de minute la expunere la radiatii ultraviolete la EPROM).

Memoriile ROM programabile prin masca si cele PROM s-au produs si cu tranzistoare bipolare; aceste tipuri nu se mai utilizeaza în proiectele moderne. Memoriile EPROM, OTPROM, EEPROM sunt fabricate în tehnologie NMOS si mai ales CMOS.

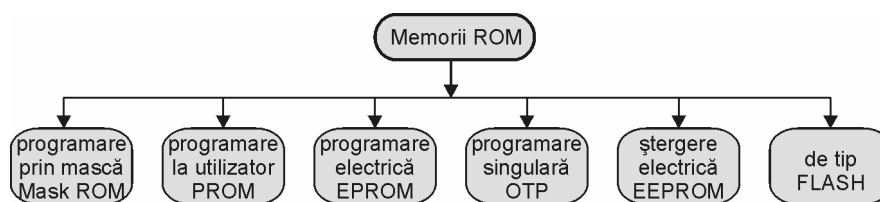


Figura 5.3. Clasificarea memoriilor ROM.

### 5.5.1. Schema bloc a unei memorii ROM

Se pot imagina mai multe arhitecturi didactice pentru a putea explica si înțelege functionarea unei memorii ROM. Una dintre acestea (figura 5.4) se obtine interconectând un decodicator cu un codificator. În figura 5.4, semnificatia liniilor electrice este:

- $A_0 - A_{n-1}$  sunt liniile de adresa prin intermediul carora se specifica cuvântul ce va fi citit;
- $W_0, \dots, W_{m-1}$  sunt  $m = 2^n$  linii de cuvânt, ele selecteaza intern cuvântul adresat.
- $D_0, \dots, D_{k-1}$  sunt  $k$  linii de date,  $k$  fiind latimea cuvântului memorat.

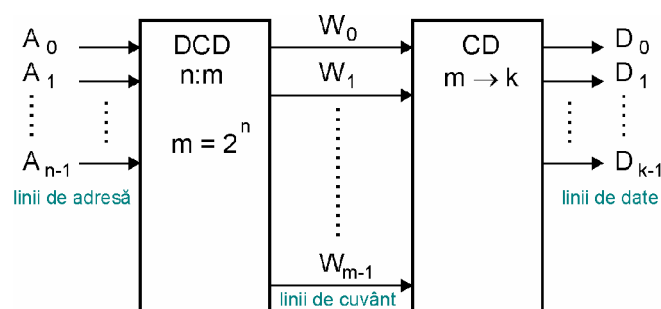


Figura 5.4. O arhitectura posibila pentru memoria ROM.

Aplicând o adresa la liniile  $A_0 - A_{n-1}$ , la iesirea DCD se activeaza o singura linie  $W_i$ . La iesirea CD se obtine un cod pe  $k$  biti ce reprezinta chiar informatia înmagazinata în locatia ce corespunde liniei de cuvânt  $W_i$  active, respectiv adresei selectate la intrarea DCD si implicit a memoriei ROM.

În aceasta schema sediul informatiei memorate este CD, DCD necontinând informatia ce urmeaza sa fie citita. Cuvântul binar obtinut la iesirea CD depinde de structura codicatorului, înscrierea sau programarea unui cuvânt într-o anumita locatie se face prin modificarea structurii schemei CD ce corespunde liniei de cuvânt respective.

### 5.5.2. Memorii ROM programabile prin masca

Un asemenea circuit este produs parcurgându-se aproape toate etapele normale de fabricatie cu exceptia unei ultime etape în care folosind o masca specifica cerintelor utilizatorului se adapteaza memoria pe baza unui tabel furnizat de utilizator. Prin mascare, grilele tranzistoarelor MOS se conecteaza selectiv la liniile de cuvânt (figura 5.5). O grila conectata înseamna ca la activarea liniei de cuvânt ce este conectata la acel tranzistor, acesta va conduce, la iesire pe linia de bit fiind pe 1 logic. O grila neconectata presupune un 0 logic la iesire.

Memoria din figura 5.5 contine 8 cuvinte a 4 biti, având o capacitate de 32 de biti. În vederea programarii utilizatorul completeaza un tabel similar cu tabelul 5.2. Daca la adresa 1 se mascheaza baza celui de-al doilea tranzistor, la activarea liniei W1 ( $W_1 = 1$ ), la iesirea de date se va gasi combinatia dorita, adica  $Q_0Q_1Q_2Q_3 = 1011$ .

Tabelul 5.2

Continutul memorie PROM (exemplu)

Adresa	Date iesire $Q_0 - Q_3$
000	1010
<b>001</b>	<b>1011</b>
.	.
.	.
111	0100

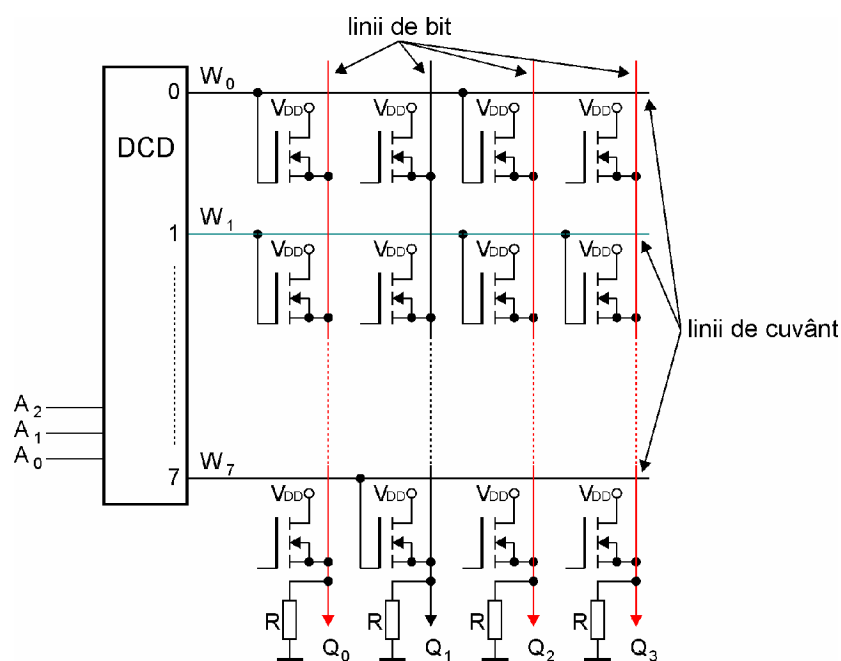


Figura 5.5. Schema simplificata a unei memorii ROM programate prin masca.

### 5.5.3. Memorii PROM

Memoriile PROM s-au realizat cu tranzistoare bipolare si ulterior cu tranzistoare MOS. Varianta cu tranzistoare bipolare oferea capacitati extrem de mici (maxim sute de biti), puteri disipate ridicate (sute de mW) si de aceea ele nu se mai folosesc în prezent.

Varianta cu tranzistoare MOS ofera capacitati de memorare mult mai mari si putere disipata semnificativ mai redusa decât în cazul celei bipolare.

Utilizatorul folosește un dispozitiv special numit programator prin intermediul căruia poate înscrie informația dorită într-o astfel de memorie. După programare, doar în anumite cazuri se mai pot face modificări respectiv programa circuitul.

În stare neprogramată, fuzibilele sunt intacte, ieșirea fiind pe 1 logic. Programarea se bazează pe întreruperea unei conexiuni existente în memoriile neprogramate. Întreruperea se face prin vaporizarea unor trasee prin impulsuri de curent generate în mod corespunzător de echipamentul de programare. Ieșirea unui bit programat este pe 0 logic.

Memoria PROM din figura 5.6 are o capacitate de 8 cuvinte a 4 biți, adică de 32 de biți. Pentru a memora la adresa 1 ( $A_2A_1A_0 = 001$ ) cuvântul 1011, în timpul programării se va „arde” doar cel de-al doilea fuzibil corespunzător liniei  $W_1$ . După programare, la selecția liniei de cuvânt  $W_1$  la ieșire se va regăsi informația dorită, adică 1011.

Trebuie remarcat că un bit neprogramat poate fi ulterior trecut în 0 la o nouă programare, dar un bit programat nu va putea fi readus în 1 logic, cu alte cuvinte la adresa 1 cuvântul 1011 va putea fi reprogramat în 0011, dar nu în 0111.

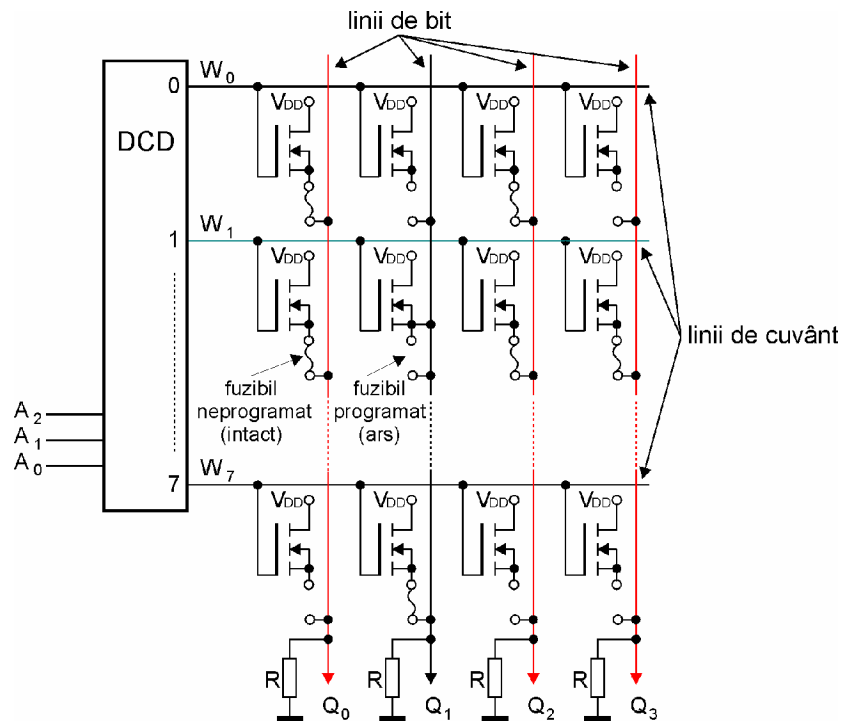


Figura 5.6. Schema simplificată a unei memorii PROM cu tranzistoare MOS.

### 5.5.4. Memoria EPROM

Schema simplificată a unei memorii EPROM la care partea de programare a fost omisă este prezentată în figura 5.7. Cele 4 tranzistoare MOS din partea superioară a schemei formează sarcinile active pentru tranzistoarele utilizate în nodurile matricii de memorare. S-a ales această variantă în locul unor rezistențe de sarcină deoarece un tranzistor MOS ocupă un spațiu mai redus în aria de siliciu, iar consumul de putere este mai mic.

Dacă tranzistoarele matricii de memorare ar fi tranzistoare MOS obișnuite, la activarea liniei de cuvânt  $W_i$ , toate ieșirile ar fi puse la masă (0000). Pentru a trece una din liniile de bit pe 1 ar fi necesar ca tranzistorul din nodul ce corespunde liniei  $W_i$  să nu conducă atunci când  $W_i = 1$ , iar programarea ar trebui realizată fără întreruperea legăturii fizice a grilei la linia de cuvânt. Pentru aceasta este necesară folosirea unui nou tip de tranzistor MOS, și anume tranzistorul MOS cu grila flotantă, dezvoltare tehnologică care a permis realizarea memoriei EPROM.

Caracteristica  $i_D - U_{gs}$  a unui astfel de tranzistor MOS (figura 5.8) depinde de încărcarea cu sarcini negative a grilei flotante. Tranzistoarele din nodurile corespunzătoare unei linii de bit care trebuie să

fie pe 1 trebuie sa aiba poarta flotanta încarcata cu sarcina negativa q-. Programarea este facuta prin încarcarea grilei flotante cu ajutorul unui impuls de programare (10V-15V, tipic 12 V sau uneori 12,5 V) între drena si sursa tranzistorului, cu durata de câteva zeci de ms, dupa selectarea liniei de cuvânt  $W_i = U_H$  (figura 5.9).

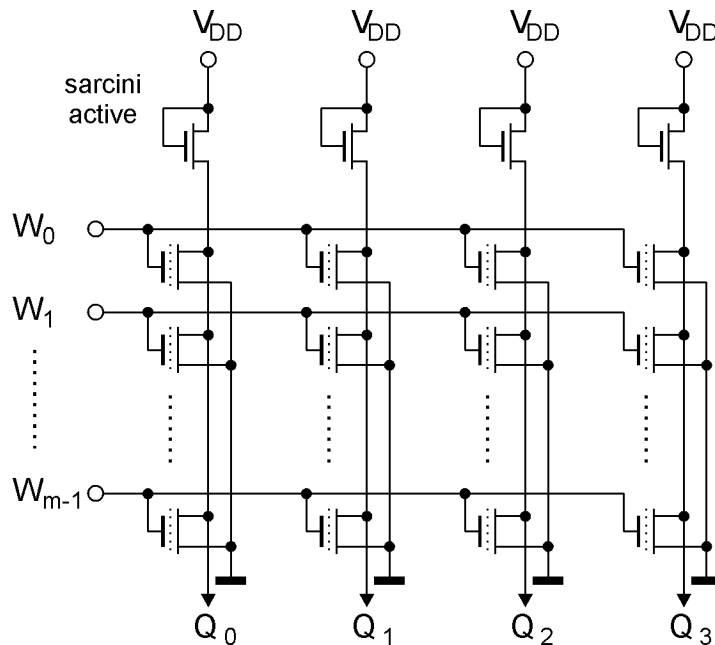


Figura 5.7. Schema simplificata a unei memorii EPROM.

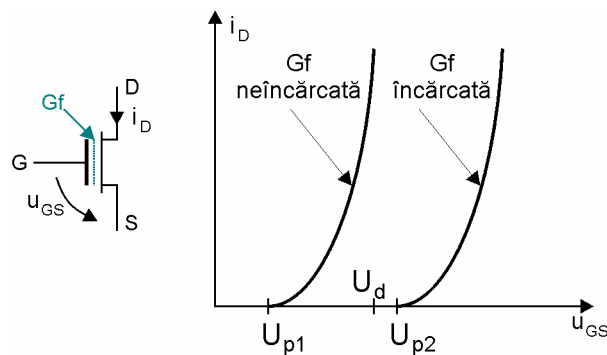


Figura 5.8. Tranzistorul MOS cu grila flotanta si caracteristica electrica.

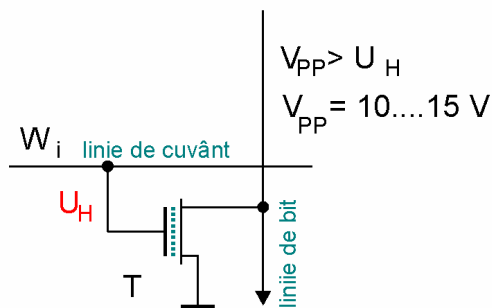


Figura 5.9. Programarea unui 1 logic în tranzistorul MOS cu grila flotanta.

Tranzistorul MOS cu grila flotanta din figura 5.9 are  $U_{p2} > U_d$ . La activarea liniei de cuvânt  $W_i$  ( $W_i = 1$ ), T nu va conduce, iar linia de bit corespunzatoare va fi în 1 logic datorita sarcinii active formate din alt tranzistor MOS, asa cum se poate observa în figura 5.7.

*Stergerea* informatiilor se face iradiind matricea CD cu radiatii UV (ultraviolete) un interval de timp de ordinul zecilor de minute. Deoarece aceasta stergere nu se poate efectua selectiv, dupa expunere memoria EPROM va fi integral stearsa, toti bitii fiind pe 1 logic.

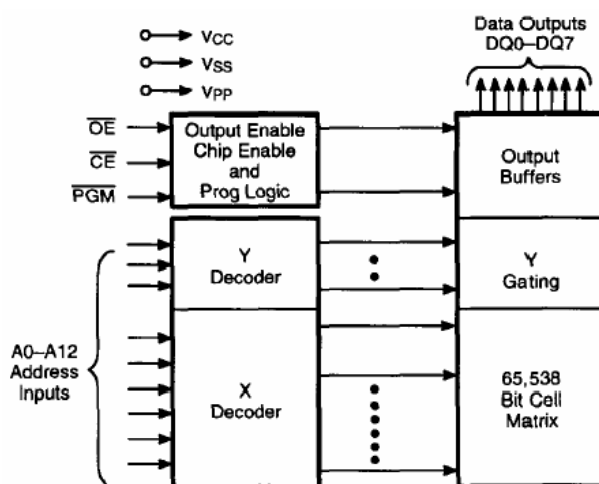
Radiatia ultravioleta determina stergerea memoriei EPROM daca are o lungime de unda mai mica de 4000 Å, valoarea recomandata de catalog fiind de 2537 Å. Distanța dintre lampa UV si cip trebuie sa fie de circa 2,5 cm, iar puterea radiatiei 12000 mW/cm<sup>2</sup>; în aceste conditii stergerea completa a circuitului survine dupa 15-20 minute. Studii de specialitate au demonstrat ca expunerea *continua* a unei memorii EPROM la lumina fluorescanta din camera (care are si o componenta cu lungimea de unda între 3000 si 4000 Å) poate determina stergerea informatiei în circa 3 ani acelasi circuit se sterge în circa 1 saptamâna daca este expus la lumina solara directa.

Numarul garantat de programari si de stergeri este mai mare de 100, dar defectele sunt frecvente chiar dupa câteva cicluri stergere – programare; durata de mentinere a informatiei memorate este minim zece ani.

Pentru circuitele EPROM moderne, având capacitati de peste 64 kbiti, o programare octet cu octet la zeci de ms pentru un octet ar fi inacceptabil de lunga (la un EPROM 27512 de 512 kbiti, programarea ar dura 20 ms x 65536 octeti = 22 minute). Pentru reducerea timpului de programare au fost imaginati algoritmi de *programare rapida*, la care pe durata programarii se face  $V_{DD} = 6\text{ V}$  si  $V_{PP} = 12,5\text{ V}$ , impulsul de programare reducându-se la 1 ms sau în unele cazuri chiar de 100 µs! Prin aceasta chiar la memoriile EPROM de capacitati relative ridicate (peste 1 Mbit), programarea dureaza maxim câteva minute.

### Memoria EPROM 2764A

2764A este o memorie NMOS de 64 kbiti organizata ca 8 k cuvinte de 8 biti, oferita de o serie de producatori. Dispunerea pinilor este prezentata în figura 5.26, iar aspectul capsulei în figura 5.13.



**Figura 5.10.** Arhitectura simplificata a memoriei EPROM 2764A (conform foii de catalog ST).

Tabelul 5.3

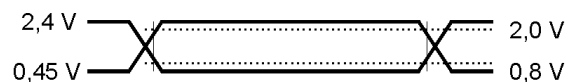
*Modurile de operare pentru memoria EPROM 2764A*

Mod	nCS	nOE	nPGM	A <sub>9</sub>	V <sub>PP</sub>	Q <sub>0</sub> – Q <sub>7</sub>
Citire ( <i>Read</i> )	0	0	1	X	V <sub>DD</sub>	Data Out
Inhibare iesiri ( <i>Output Disable</i> )	0	1	1	X	V <sub>DD</sub>	HiZ
Asteptare ( <i>Standby</i> )	1	X	X	X	V <sub>DD</sub>	HiZ
Programare ( <i>Program</i> )	0	1	impuls	X	V <sub>PP</sub>	Data In
Verificare ( <i>Verify</i> )	0	0	1	X	V <sub>PP</sub>	Data Out
Inhibarea programarii ( <i>Program Inhibit</i> )	1	X	X	X	V <sub>PP</sub>	HiZ
Identificare inteligenta ( <i>Electronic Signature</i> )	0	0	1	12 V	V <sub>DD</sub>	Octet identificare

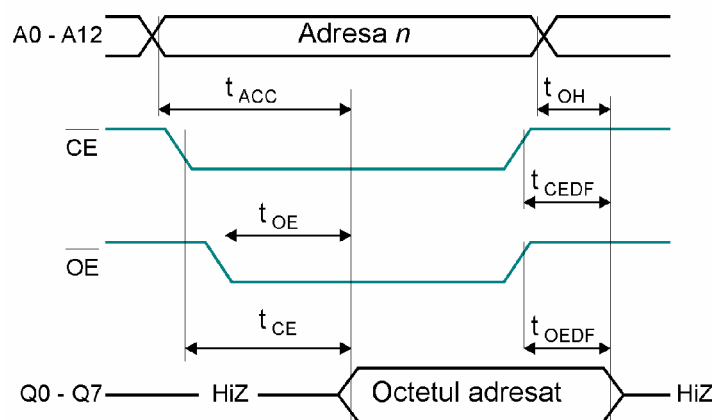


Memoriile EPROM prezinta doua terminale de control asemanatoare: nCS si nOE. Pentru nCS = 1, circuitul este trecut în modul asteptare, în care iesirile sunt în starea de înalta impedanta, iar consumul de curent este substantial redus. nOE controleaza doar bufferele de iesire, care sunt inhibitate pentru nOE = 1 si validate pentru nOE = 0. Motivatia existentei a *doua* terminale de control este mai subtila: într-un sistem cu microprocesor, controlul cu doua terminale evita conflictele pe magistrala daca se utilizeaza mai multe asemenea memorii si un decodoficator pentru selectia lor. Microprocesorul va emite semnalele de adresa, din care o parte (cele mai semnificative) sunt folosite de DCD si semnalul nRD, care va fi conectat la intrarile nOE ale tuturor memoriilor.

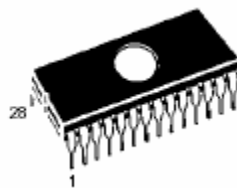
Fabricantul specifica în foaia de catalog punctele de test pentru 0 si 1 logic. În cazul memoriei 2764A aceste puncte de test sunt 0,8 V si 2 V (niveluri TTL), figura 5.11. Aceste puncte de test sunt valabile si pentru memorii EPROM mai noi, de exemplu pentru memoria de 1 Mbit 27C010, realizata în tehnologie CMOS.



**Figura 5.11.** Punctele de test pentru starea 0 si 1 în cazul memoriei EPROM 2764A.



**Figura 5.12.** Diagramele de semnal pentru citirea memoriei EPROM 2764A.



**Figura 5.13.** Aspectul unei memorii EPROM cu 28 de pini. Se remarca fereastra circulara de cuar.

Datorita numarului mare de variante de memorii EPROM si a specificatiilor uneori usor diferite pentru acelasi tip de circuit la diferiti producatori, memoriile EPROM moderne dispun de un mecanism de identificare inteligenta pe baza unei semnaturi electronice (*Electronic Signature*). Pe baza semnaturii electronice se identifica tipul circuitului (codificat pe un octet) si producatorul (codificat tot pe un octet). Aceste informatii sunt citite automat de echipamentul de programare care va selecta algoritmul de programare potrivit circuitului utilizat fara interventia operatorului uman, eliminându-se astfel o posibilitate de eroare.

Pentru a activa modul de identificare inteligenta, adresa A9 trebuie adusa la o tensiune de 12 V, toate celelalte linii de adresa se mentin la 0 logic; pentru A0 = 0 logic la liniile de date se va putea citi codul producatorului, (de exemplu pentru SGS-THOMSON se va citi 20H), iar pentru A0 = 1 se va putea citi codul circuitului, în acest caz 08H.

Caracteristici principale 2764A:

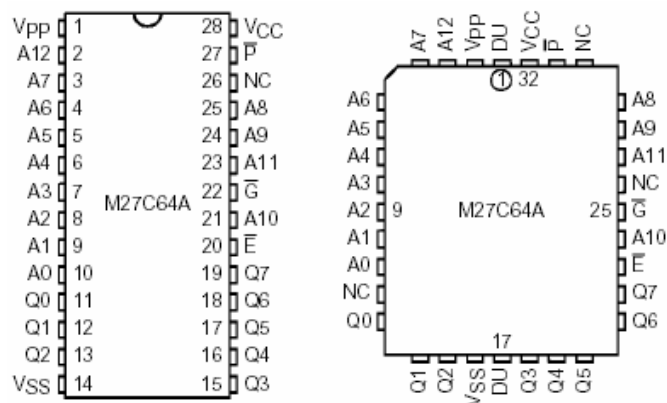
- timp de acces: 180 ns;
- curent de alimentare în repaus: 35 mA;
- curent de alimentare în funcționare normală: 75 mA
- tensiune de programare  $V_{PP} = 12$  V;
- algoritm de programare rapid (durata totală de programare este sub 1 minut);
- identificare inteligentă prin semnatura electronică;

Varianta CMOS pentru același circuit, notată 27C64 este compatibilă pin la pin cu predecesoarea sa și oferă următoarele caracteristici îmbunătățite:

- timp de acces: 150 ns
- curent de alimentare în repaus: 100  $\mu$ A
- curent de alimentare în funcționare normală: 30 mA
- tensiune de programare  $V_{PP} = 12,5$  V;
- algoritm de programare rapid (durata totală de programare este sub 1 minut);
- identificare inteligentă prin semnatura electronică;

### 5.5.5. Memoria OTP (*One Time Programmable*)

Este tot o memorie EPROM programată electric la producător sau la utilizator fără fereastră de cuarț pentru ștergere, având un preț mai mic cu circa 40% decât la varianta EPROM. Utilizatorul nu o mai poate reprograma ulterior. Este rentabilă în producția de masă, fiind asemănătoare cu memoria ROM programată prin mască. Memoriile OTP sunt de obicei disponibile în capsule de plastic ieftine, cum ar fi PLCC (*Plastic Leaded Chip Carrier*). Memoria 27C64A de exemplu este oferită în varianta OTP într-o capsulă PLCC 32 (cu 32 de terminale față de 28 la varianta DIP, cu mențiunea că cele 4 terminale suplimentare nu se folosesc).



**Figura 5.14.** Dispunerea terminalelor la 27(C)64A, (DIP28) și OTP 27C64A, (PLCC32).  
NC = neconectat, DU = nu se folosește

### 5.5.6. Memoria EEPROM

Elimină dificultățile de extragere din soclu și expunere la lumina ultravioletă. Atât programarea cât și ștergerea se execută electric. Prin perfecționarea tehnologiei și micșorarea grosimii stratului izolator al grilei flotante există posibilitatea programării și ștergerii electrice cu tensiuni mici aplicate între drenă și poartă. Polaritatea caderii de tensiune drenă-poartă este inversată la ștergere față de programare. Sunt mai scumpe decât memoriile Flash; se pot rescrie în timpul funcționării, numărul de ștergeri și reprogramări fiind cu un cel puțin un ordin de mărime mai mare decât la memoriile Flash.

## Memoria EEPROM 2864B

2864B este o memorie EEPROM CMOS de 64 kbiti organizata ca 8 k cuvinte de 8 biti, oferita de mai multi producatori. Dispunerea pinilor este redata în figura 5.26, iar modurile de operare în tabelul 5.4.

Tabelul 5.4

Modurile de operare pentru memoria EEPROM 2864B

Mod	nCE	nOE	nWE	I/O <sub>0</sub> – I/O <sub>7</sub>
Citare ( <i>Read</i> )	0	0	1	Data Out
Sciere ( <i>Write</i> )	0	1	0	Data In
Inhibare scriere ( <i>Write Inhibition</i> )	X	X	1	-
	X	0	X	-
Asteptare ( <i>Standby</i> )	1	X	X	HiZ

Caracteristici principale pentru memoria EEPROM 2864B:

- timp de acces – 150 ns;
- curent de alimentare în repaus: 1  $\mu$ A;
- curent de alimentare în functionare normala: 30 mA
- numar minim de stergeri si de reprogramari:  $10^5$
- durata de mentinere a informatiei memorate: minim 10 ani;

### 5.5.7. Memoria Flash

Memoriile Flash permit atât citirea cât și înscrierea informatiei în timpul functionarii normale; sunt memorii de densitate mare, nevolatile, folosite în cele mai diverse aplicatii – de la aparatele de fotografiat digitale la înlocuirea de hard-diskuri.

Celula de memorare a unui bit consta dintr-un singur tranzistor MOS cu grila flotanta. Stocarea propriu-zisa este realizata prin prezenta sau absenta sarcinilor în grila flotanta. O sarcina relativ ridicata acumulata în grila flotanta (figura 5.15) este echivalenta cu 0 logic, iar o sarcina redusa sau absenta cu 1 logic.

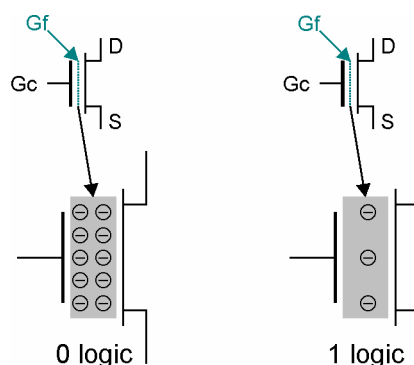


Figura 5.15. Celula elementara a memoriei Flash.

#### Programarea

În stare neprogramata, toate celulele memoriei sunt în 1 logic (sarcini reduse sau nule în grila flotanta). Daca în procesul de programare se doreste memorarea unui 0 logic, grila de control se aduce la un potential pozitiv  $+V_{prog}$ , prin aceasta fiind atrasi electroni spre grila flotanta, ea încarcându-se negativ. O data programata, sarcina grilei flotanta se mentine un timp îndelungat – minim 10 ani.

În cazul în care se doreste stocarea unui 1 logic, în timpul programarii celula respectiva este lasata nemodificata.

### Citirea

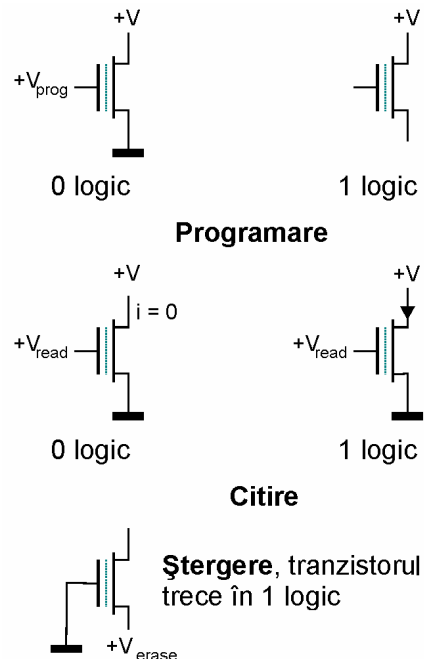
Pe durata operatiei de citire, la grila de control se aplica o tensiune pozitiva  $+V_{\text{read}}$ . Cantitatea de sarcina stocata în grila flotanta va determina în acest caz daca sub actiunea acestei tensiuni, tranzistorul MOS va conduce sau nu: daca se memoreaza un zero, datorita numarului mare de electroni de pe grila flotanta, tranzistorul ramâne blocat. În contrast, daca grila flotanta contine putini electroni, sub influenta tensiunii  $+V_{\text{read}}$  tranzistorul MOS va conduce (figura 5.16).

### Stergerea

Operatia de stergere se rezuma la înlaturarea electronilor din grila flotanta – aducând grila de control la potentialul masei si sursa la o tensiune pozitiva ( $+V_{\text{erase}}$ ), electronii vor fi atrasi spre sursa si prin aceasta grila flotanta nu va mai contine sarcini negative sau numarul acestora va fi foarte redus.

O memorie Flash este întotdeauna stearsa înainte de a fi programata. La memoriile Flash moderne stergerea se realizeaza pe blocuri sau paginat, dar important este ca un octet singular din cadrul unui bloc nu poate fi scris (programat) decât dupa ce este sters împreuna cu întregul bloc din care face parte.

Memoriile Flash ofera suplimentar posibilitatea stergerii globale rapide (*bulk erase*).



**Figura 5.16.** Functionarea simplificata a celulei elementare din memoria Flash.

Schema simplificata pentru operatia de citire a unei memorii Flash este ilustrata în figura 5.17. Aria de memorie contine  $m$  cuvinte a  $n$  biti fiecare, adica un total de  $m \times n$  tranzistoare MOS cu grila flotanta pentru acelasi numar de biti de informatie. La fiecare linie de bit un tranzistor MOS formeaza sarcina activa (aceasta abordare este mai eficienta decât cea în care ar fi dispuse rezistente de sarcina, *de ce?*). Pentru citire se activeaza o linie de cuvânt, fiecare din cele  $n$  comparatoare furnizând la iesire 1 sau 0 logic, dupa cum tranzistoarele MOS de memorare conduc sau nu (1 logic daca tranzistorul conduce, 0 logic în caz contrar).

Caracteristici generale:

- timp de acces: 45 – 150 ns;
- numar de stergeri si de reprogramari este de ordinul  $10^4 - 10^5$ ;
- durata de mentinere a informatiei memorate este de minim 10 ani;
- sunt cele mai ieftine memorii nevolatile;
- se pot rescrie in timpul functionarii.

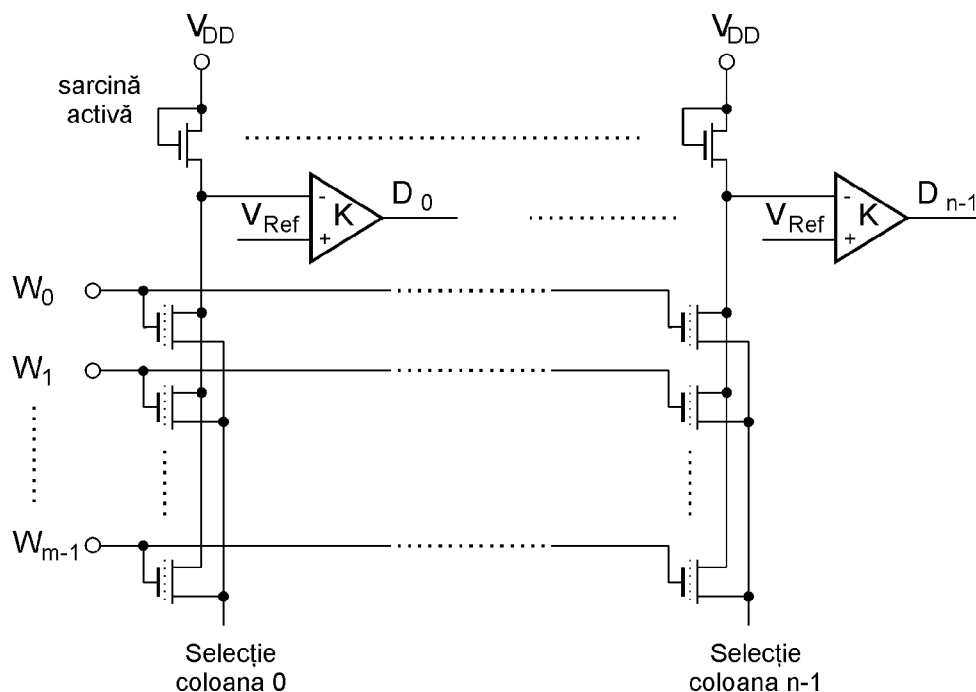


Figura 5.17. Structura simplificata a unei memorii Flash (sectiunea de programare este omisa).

### Memoria Flash 29F010

29F010 este o memorie Flash CMOS de 1 Mbit organizata ca 128 k cuvinte de 8 biti, oferita de mai multi producatori. Dispunerea pinilor este prezentata în figura 5.28, iar modurile de operare în tabelul 5.5.

Tabelul 5.5

Modurile de operare pentru memoria Flash 29F010

Mod	nCE	nOE	nWE	A0	A1	A9	I/O0-I/O7
Citire (Read)	0	0	1	A0	A1	A9	Data Out
Scriere (Write)	0	1	0	A0	A1	A9	Data In
Asteptare (Standby)	1	X	X	X	X	X	HiZ
Inhibare iesire (Output Disable)	0	1	1	X	X	X	HiZ
Identificare – Codul circuitului	0	0	1	1	0	12 V	ID
Id. inteligenta – Codul producatorului	0	0	1	0	0	12 V	ID
Protectie sector (Verify Sector Protect)	0	0	1	0	1	12 V	ID

Caracteristici principale 29F010:

- timp de acces – 45 ns;
- curent de alimentare în repaus: 25  $\mu$  A;
- curent de alimentare în functionare normala: 30 mA la citire, 50 mA la programare sau stergere;
- este împartit în 8 sectoare de 16 kocteti;
- permite stergerea întregului circuit sau a oricarui sector;
- stergerea si verificarea unui sector dureaza sub 1 secunda;
- permite identificarea inteligenta în montaj, activata pentru A9 = 12 V, similar cu memoriile EPROM;
- identificarea inteligenta se poate realiza în circuit si în absenta tensiunii ridicate de 12 V printr-o serie de comenzi;
- numar minim de stergeri si de reprogramari:  $10^5$ ;
- durata de mentinere a informatiei memorate: minim 10 ani;

## 5.6. MEMORII RAM

Sunt circuite integrate cu integrare pe scara foarte larga (VLSI) care permit în timpul functionarii atât citirea cât si înscrierea în locatia adresata. O denumire mai potrivita pentru aceste memorii este de memorii RWM (*Read-Write Memory*).

Clasificare RAM:

- RAM statice (SRAM), realizate cu bistabile în tehnologie bipolară sau unipolară;
- RAM dinamice (DRAM), numai în tehnologie unipolară NMOS sau CMOS.

Dupa modul de selectie a locatiei:

- selectie liniara, în cazul memoriilor de capacitate redusa, organizate pe cuvinte de mai multi biti;
- selectie prin coincidenta, în cazul memoriilor de capacitate relativ ridicata, organizate pe cuvinte de un bit;
- selectia tridimensională, în cazul memoriilor de capacitate foarte mare, organizate pe cuvinte de mai multi biti.

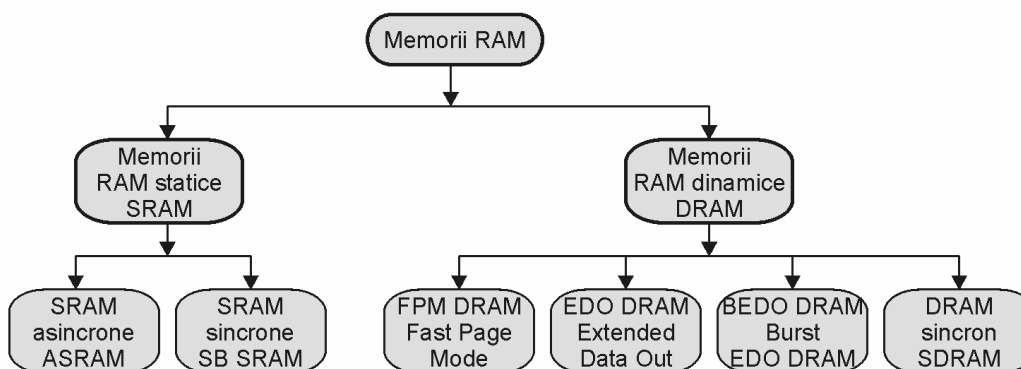


Figura 5.18. Principalele tipuri de memorii RAM disponibile în prezent.

### 5.6.1. Selectia cuvântului adresat

Selectia liniara se realizeaza similar cu selectia omonima prezentata la memoriile ROM. Memoria din figura 5.19 este organizata pe  $m = 2^n$  octeti. Pentru combinatia de adrese  $A_{n-1} \dots A_0 = 0 \dots 0$ ,  $W_0 = 1$ , selectându-se primul octet, care se va regasi la iesire (10111011).

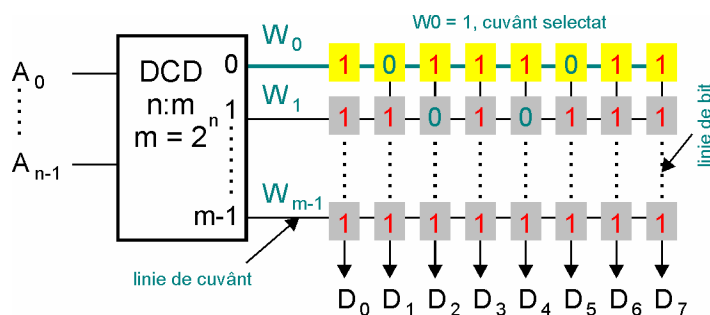
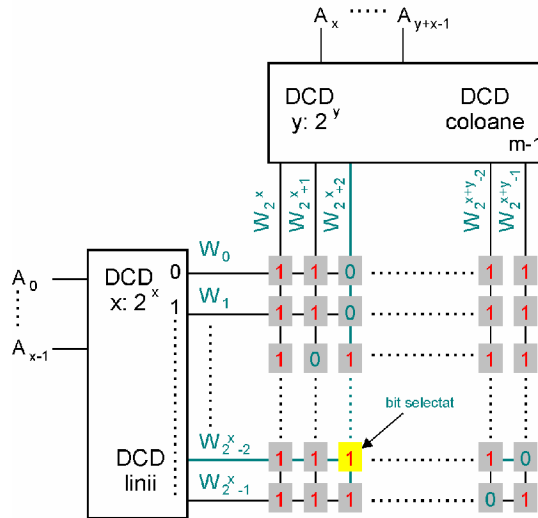


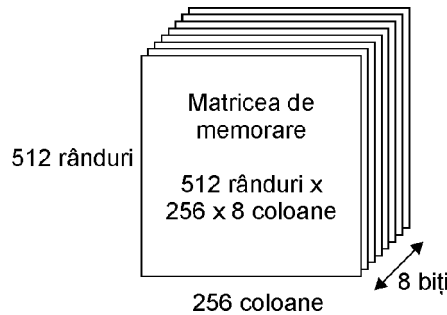
Figura 5.19. Selectia liniara într-o memorie RAM.

Selectia prin coincidenta utilizeaza doua decodificatoare de adresa, unul pe linii si celalalt pe coloane. Nodul selectat din memoria din figura 5.20 corespunde adresei de linii  $A_0 \dots A_{x-1} = 0 \ 1 \ \dots 1$  si de coloane  $A_x \dots A_{x+y-1} = 0 \ 0 \ 1 \ 0 \ \dots 0$ . Pentru o folosire eficienta a ariei de siliciu ocupate se cauta ca matricea de memorare sa fie patrata sau cât mai aproape de un patrat în cazul unui numar impar de adrese, dimensiunile celor doua DCD fiind egale sau apropiate.



**Figura 5.20.** Selectia prin coincidenta.

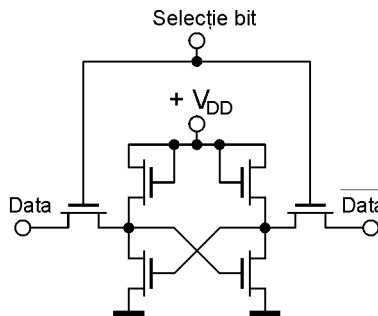
La memoriile RAM de capacitate mare (de peste 1 Mbit) decodificarea bidirectionala nu mai este suficienta, deoarece numarul de linii ale fiecaruia dintre cele doua DCD va depasi 1000. Din acest motiv informatia stocata este dispusa într-o matrice tridimensionala. De exemplu la memoria 628128A de 128 x 8 kbiti, matricea de memorare se prezinta ca în figura 5.21. În acest caz decodificarea se face bidimensional, existând 512 rânduri a 256 de coloane, dar în punctul de selectie, cuvântul nu are lungimea de 1 bit, ci de 8 biti.



**Figura 5.21.** Matricea de memorare la memoria SRAM 628128A, de 1 Mbit.

### 5.6.2. Memoria SRAM

Aceste memorii sunt realizate cu bistabile, celula elementara fiind cea din figura 5.22.



**Figura 5.22.** Bistabilul de memorare al celulei elementare pentru memoria SRAM.

Terminale memoriei sunt:

A0... An-1	- linii de adresa,	/CS	- selectie capsula,
Do0...Do(k-1)	- date iesire,	/OE	- validare iesire,
Di0...Di(k-1)	- date intrare,	/nWE	- validare scriere.

Celula completa de memorie pentru 1 bit cu latch D (figura 5.23).

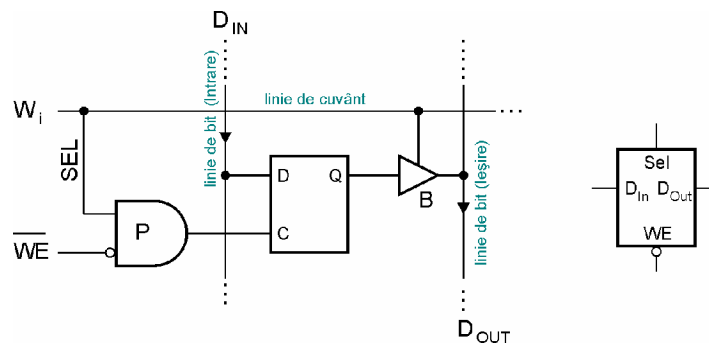
*Descrierea functionarii:*

Pentru a putea face o operatie de citire sau de scriere este necesar ca celulele sa fie selectata cu ajutorul liniei de cuvânt  $W_i$  activa pe 1. Linia de cuvânt furnizeaza deci semnalul SEL ce actioneaza asupra portii P si a bufferului B. O data selectat, bufferul B se afla în stare normala de functionare. Cealalta intrare a portii SI se foloseste pentru stabilirea operatiei ce se va efectua: citire ( $nWE = 1$ ) sau scriere ( $nWE = 0$ ).

La citire ( $nWE = 1$ ) trebuie selectata celula ( $SEL=1$ ); în acest caz intrarea C a bistabilului este 0 ( $C=0$ ) si oricare ar fi datele pe linia de intrare ele nu se înmagazineaza în D-latch în schimb B fiind în stare normala, obtinem la iesirea Do pe Q memorat in bistabil.

La scriere ( $nWE = 0$ ) trebuie selectata celula ( $SEL=1$ ); în acest caz ceea ce exista pe linia LBIN se înscrie in D-latch. Cu alte cuvinte ceea ce am pe liniile de intrare  $D_{IN}$  se va transfera în bistabil (adica la iesirea Q a acestuia).

Daca celula nu este selectata ( $SEL = 0$ ),  $C = 0$ , iar B se afla în starea de impedanta ridicata (HiZ).



**Figura 5.23.** Model functional pentru celula de memorare pentru memoria SRAM si simbolul utilizat.

Consideram în cele ce urmeaza o memorie ipotetica de 1024 cuvinte de câte 4 biti. Aceasta memorie va contine 4192 de celule, selectate liniar prin intermediul unui DCD 10:1024 (figura 5.24).

Tabelul 5.6

*Functionarea memoriei SRAM*

Operatie	$\overline{CS}$	$\overline{OE}$	$\overline{WE}$
Citire ( <i>Read</i> )	0	0	1
Scriere ( <i>Write</i> )	0	1	0
Memorare ( <i>Hold</i> )	1	X	X

*Descrierea functionarii*



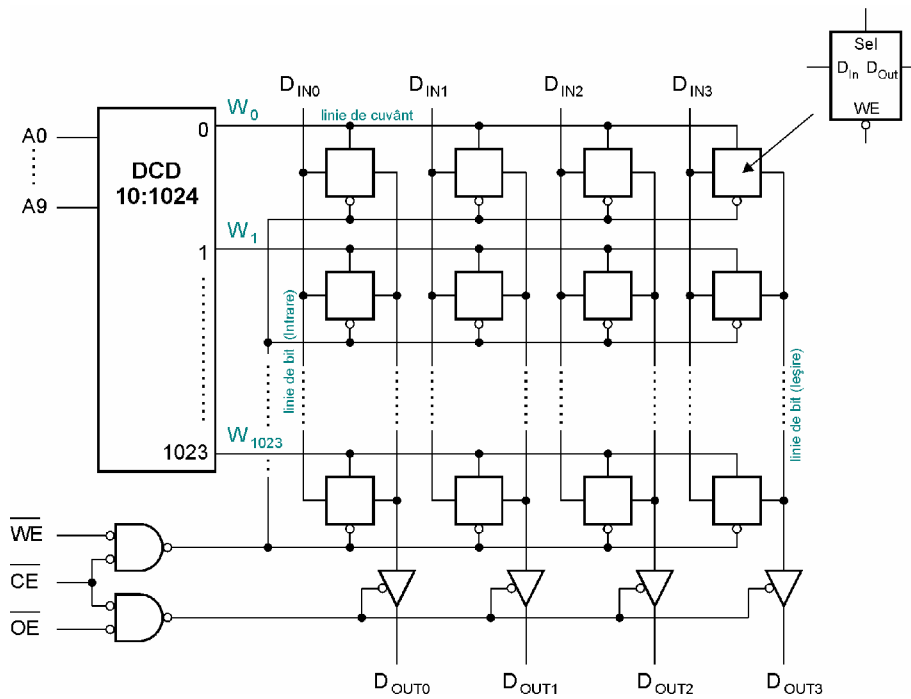


Figura 5.24. Structura simplificata a unei memorii SRAM de 4 x 2 biti.

Reducerea numarului de pini ai capsulei circuitului integrat se poate realiza folosind pini unici pentru intrare si iesire, caracteristica obtinuta adoptând urmatoarea structura I/O (figura 5.25):

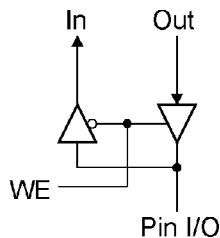


Figura 5.25. Reducerea numarului de pini ai capsulei SRAM.

Dispunerea terminalelor la memoriile SRAM respecta aranjamentul initiat de memoriile EPROM si continuat de memoriile EEPROM, în figura 5.26 fiind prezentata dispunerea terminalelor pentru memoriile de 64 kbiti.

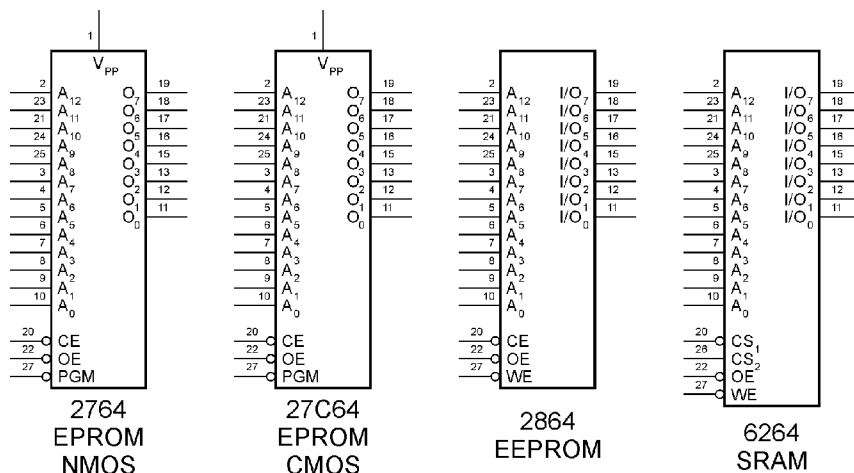


Figura 5.26. Dispunerea terminalelor la memoriile 8 k x 8 biti.

## Memoria SRAM HM6264A de 64 kbiti (Hitachi)

Memoria SRAM 6264 nu are pin de tensiune de programare (pinul 1, VPP de la EPROM), acesta rămânând neconectat, iar selectia circuitului se face cu 2 semnale (nCS1 si CS2), aspect folosit pentru extinderea capacitatii (vezi si aplicatiile de la §5.8). De cele mai multe ori producatorul de memorii specifica timpul de acces ca un prefix la denumirea circuitului, de exemplu Hitachi ofera circuitele HM6264A-10 (100 ns), HM6264A-12 (120 ns) si HM6264A-15 (150 ns).

Caracteristici principalele pentru memoria HM6264-10:

- timp de acces – 100 ns;
- putere disipata în repaus: 100  $\mu$ W;
- putere disipata în functionare normala: 15 mW / MHz;
- 8 intrari si iesiri trei-stari comune.

Diagramele de citire (figura 5.27) pentru memoria 6264 sunt asemanatoare cu cele de la memoria EPROM. Toate valorile urmatoare se vor referi la memoria HM6264A-10, la care timpul de acces ( $t_{ACC}$ ) este de 100 ns. Pentru citire, trebuie specificata o adresa pe liniile A0-A12 pe durata ciclului de citire, a carui durata minima  $t_{RC}$  este de 100 ns. Deoarece memoria SRAM este *statica*, nu este specificata si o valoare maxima –  $t_{RC}$  poate fi oricât de lung.

Tabelul 5.7

Principalii parametri dinamici ai ciclului de citire pentru memoria SRAM HM6264A-10

Parametru	Symbol	Min	Max	UM	
Durata ciclului de citire ( <i>Read cycle time</i> )	$t_{RC}$	100	—	ns	
Timpul de acces ( <i>Address access time</i> )	$t_{ACC}$	—	100	ns	
Timpul de acces selectie circuit - iesire ( <i>Chip Selection to Output</i> )	CS1	$t_{CS1}$	—	100	ns
	CS2	$t_{CS2}$	—	100	ns
Timp de acces validare iesire ( <i>Output enable to output valid</i> )	$t_{OE}$	—	50	ns	
Timp inhibare circuit ( <i>Chip deselection to output in HiZ</i> )	CS1	$t_{CS1Z}$	0	35	ns
	CS2	$t_{CS2Z}$	0	35	ns
Timp inhibare iesiri ( <i>Output disable to output in HiZ</i> )	$t_{O EZ}$	0	35	ns	
Timpul de mentinere de la modificarea adresei ( <i>Output hold from address change</i> )	$t_{OH}$	10	—	ns	

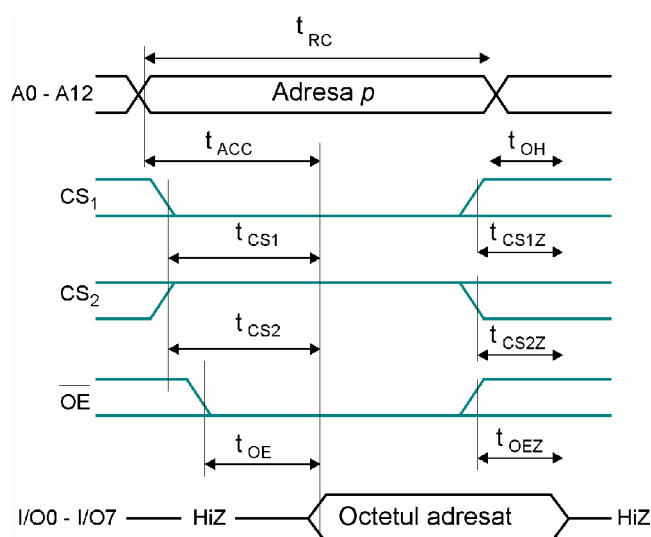


Figura 5.27. Diagrame de semnal pentru operatia de citire, memoria SRAM 64 kb, HM6264A.

Scrierea informatiei în memorie se realizeaza conform diagramelor din figura 5.28.

Tabelul 5.8

Principalii parametri dinamici ai ciclului de scriere pentru memoria SRAM HM6264A-10

Parametru	Simbol	Min	Max	UM
Durata ciclului de scriere (Write cycle time)	$t_{WC}$	100	—	ns
Timpul selectie circuit – sfârșitul nWE (Chip selection to end of write)	$t_{CW}$	80	—	ns
Timpul adrese valide – sfârșitul nWE (Address valid to end of write)	$t_{AW}$	80	—	ns
Latimea impulsului de scriere (Write pulse width)	$t_{WP}$	60	—	ns
Timpul de recuperare nWE (Write recovery time)	$t_{WR}$	0	—	ns
Timp suprapunere Dout cu nWE (Data to write time overlap)	$t_{DW}$	40	—	ns
Timpul de mentinere pentru DiN (Data hold from write time)	$t_{DH}$	0	—	ns
Timp inhibare iesiri (Output enable to output in HiZ)	$t_{OHZ}$	0	35	ns

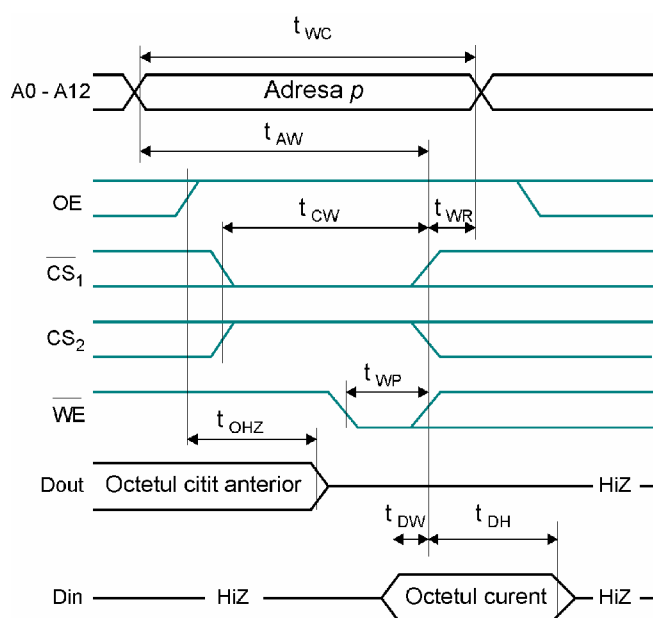


Figura 5.28. Diagrame de semnal pentru operatia de scriere, memoria SRAM 64 kb, HM6264A.

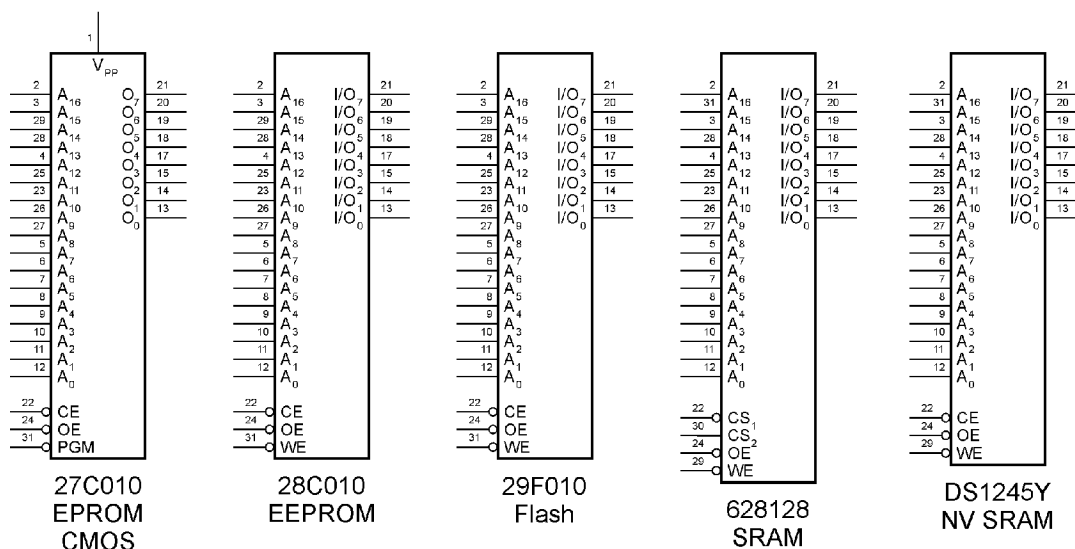


Figura 5.29. Dispunerea terminalelor la memoriile 128 k x 8 biti.



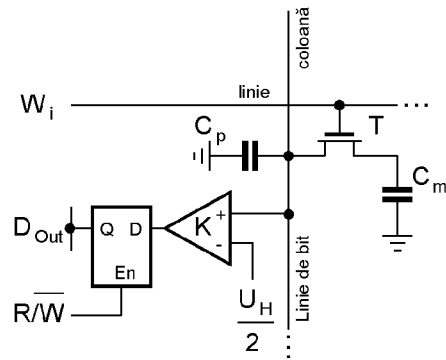


Figura 5.32. Arhitectura bufferului de ieșire pentru o linie de bit în cazul unei memorii DRAM.

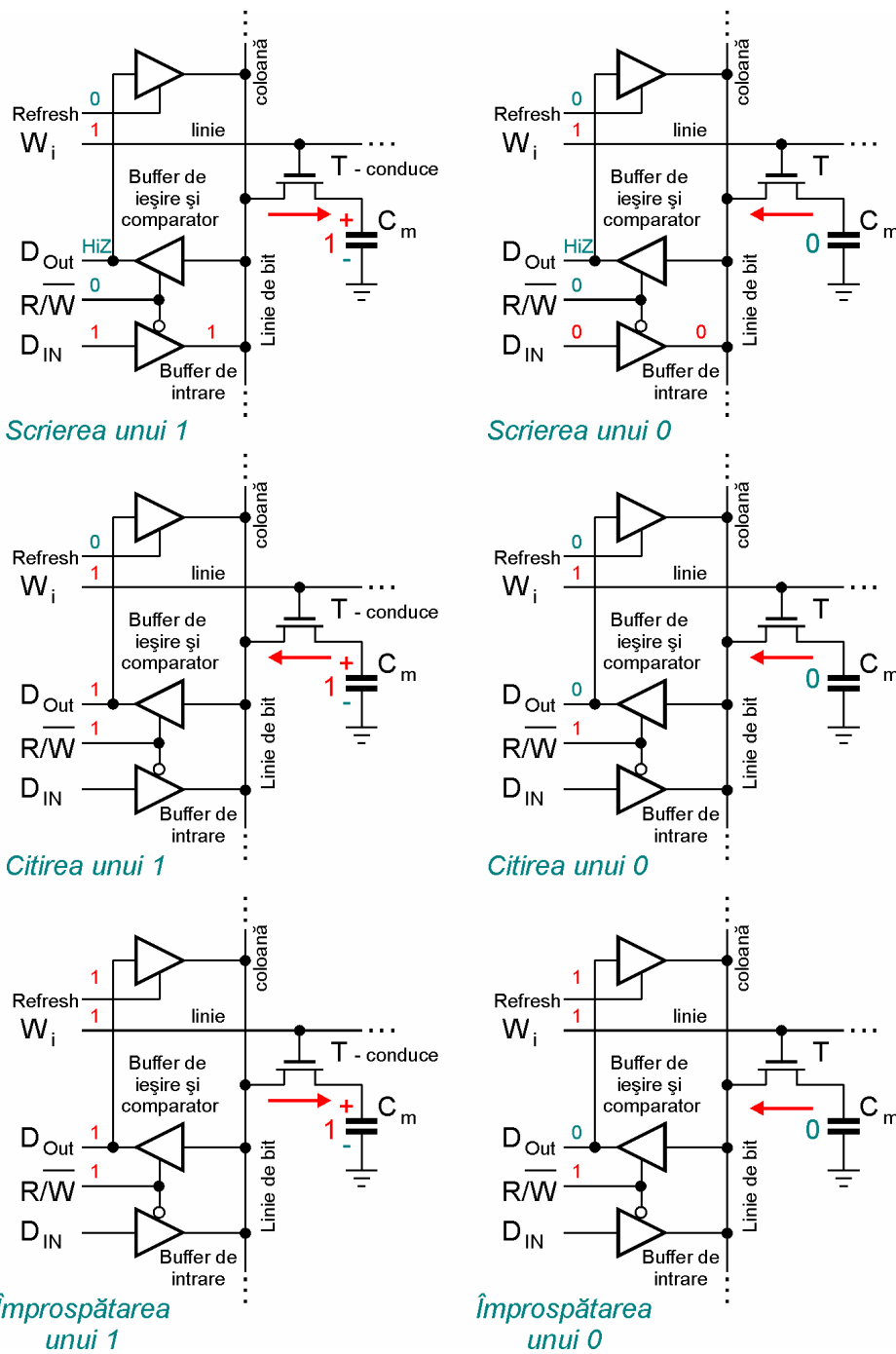


Figura 5.33. Scrierea, citirea și împrospătarea memoriei DRAM.

## Descrierea functionarii

### Scrierea

Înscrierea unui 1 în celula se face punând linia de bit LB la  $U_H$  prin activarea bufferului B. În acest timp se aplica un impuls de înălțime tot  $U_H$  liniei de cuvânt  $W_i = 1$  care determina intrarea în conducție a tranzistorului T și încărcarea lui  $C_m$  la  $U_H$ .

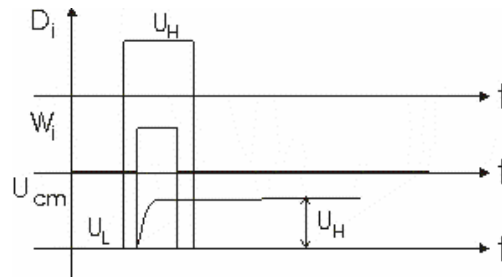


Figura 5.34. Înscrierea unui 1 în celula de memorare.

Pentru înscrierea unui 0 în celula, linia LB rămâne pe  $U_L$  (aproapiat de 0). Liniei de cuvânt  $i$  se da un impuls de înălțime  $U_H$ , care și în acest caz pune în conducție tranzistorul T descarcând condensatorul  $C_m$ .

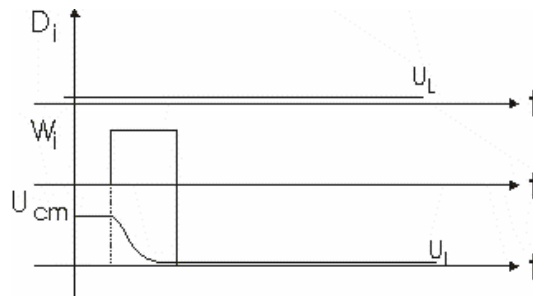


Figura 5.35. Înscrierea unui 0 în celula de memorare.

### Citirea

Se trece  $D_i$  pe  $U_H/2$  cu  $B \rightarrow N$ ,  $C_p$  se încarcă la  $U_H/2$ , se pune  $B \rightarrow Z$  izolând  $(LB)_i$  se aplica un impuls  $U_H$  pe  $LC$ . și se pune în conducție T.

1. Dacă  $C_m \rightarrow U_H$  se va citi un 1, atunci  $C_m$  se descarcă parțial peste  $C_p$  iar tensiunea pe  $LB$  va fi:

$$u_{Cp} = \frac{U_H}{2} + \Delta u$$

Iesirea comparatorului K trece pe 1 care este încărcat în bistabilul D-latch și transmis la Dout.

2.  $C_m \rightarrow U_L$ ,  $C_p$  se descarcă peste  $C_m$ , iar tensiunea:  $u_{Cp} = \frac{U_H}{2} + \Delta u'$

Comparatorului K va avea iesirea pe 0 care este memorat de bistabilul D-latch.

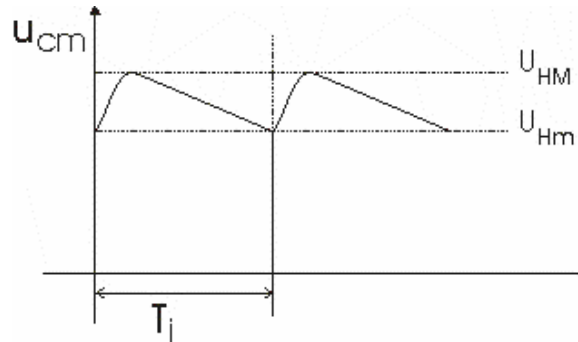
Dupa fiecare operatie de citire, care altereaza nivelul de tensiune de la bornele  $C_m$  se efectueaza o operatie de reanscriere a bitului citit folosind iesirea bistabilului D-latch (reîmprospatarea informatiei alterate prin citire).

Nu numai prin citire are loc o alterare a informatiei înscrise în  $C_m$  ci și datorita descărcării exponențiale în timp a acesteia datorita curenților de pierdere. Din acest motiv, chiar dacă nu se efectueaza citiri, este necesara reîmprospatarea sarcinii acumulate în  $C_m$  la fiecare 2-4ms (depinde de tipul memoriei).

Timpul mediu necesar unei împrospatari este în medie de  $0.3 \mu s$  și dacă împrospatarea s-ar face bit după bit în intervalul de  $4 ms$  ar putea fi împrospatate:

$$\frac{10^6 \cdot 4000}{10^6 \cdot 0.3} = 13.333$$

celule, neexistând timp pentru alte operații utile.



**Figura 5.36.** Evoluția tensiunii pe capacitatea de memorare.

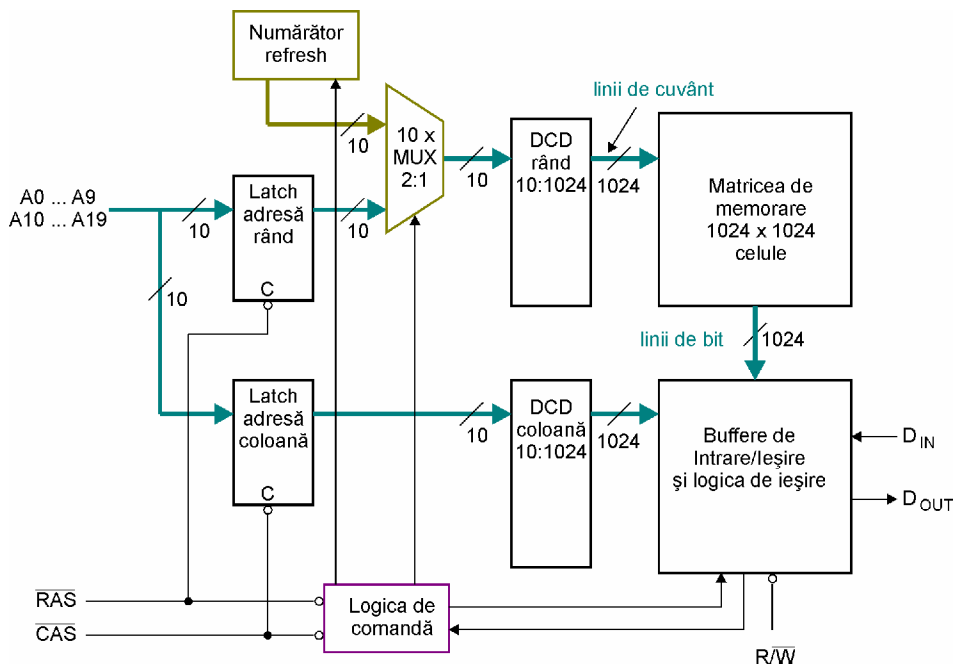
Pentru o memorie de  $64kb$  ( $65536$  biti) satisfacerea timpului disponibil de împrospatare impune adaptarea unei structuri patrate a matricii de celule de memorie, corespunzătoare la  $256/256$  biti adică o structură organizată pe  $256$  linii de cuvânt, fiecărui cuvânt corespunzându-i  $k = 256$  biti.

În acest caz datorită împrospatării simultane a tuturor celulelor selectate de o linie de cuvânt timpul necesar pentru împrospatarea întregii memorii va fi de:

$$256 \times 0.3 \cdot 10^{-6} = 76.8 \mu s, \text{ care constituie: } \frac{76.8}{4000} \cdot 100 = 1.92\%$$

din timpul disponibil pentru reamprospatare. Deci  $98\%$  din intervalul de timp dintre două împrospatări succesive poate fi folosit pentru efectuarea unor operații utile de citire și înscrisere a informației.

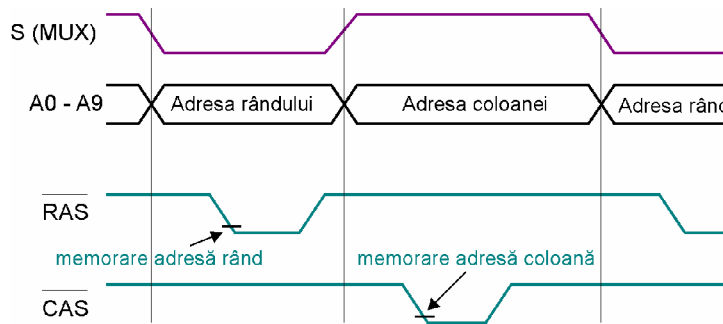
Pentru exemplificare, în figura 5.37 este prezentată schema bloc a unei memorii DRAM de  $1024k$  biti organizată pe cuvinte de câte  $1$  bit, a căror selecție se realizează prin coincidență.



**Figura 5.37.** Arhitectura unei memorii DRAM moderne.

Terminalele circuitului sunt: A0...A9, linii de adresa, /RAS – semnal de tip strobe pentru memorarea adreselor de linie, (*Row Address Strobe*); /CAS – semnal de tip strobe pentru memorarea adreselor de coloana (*Column Address Strobe*); R/nW – comanda citirii (= 1), respectiv a scrierii (= 0) *Read / Write Enable*; D<sub>IN</sub>, linia de date de intrare, D<sub>OUT</sub> – linia de date de iesire.

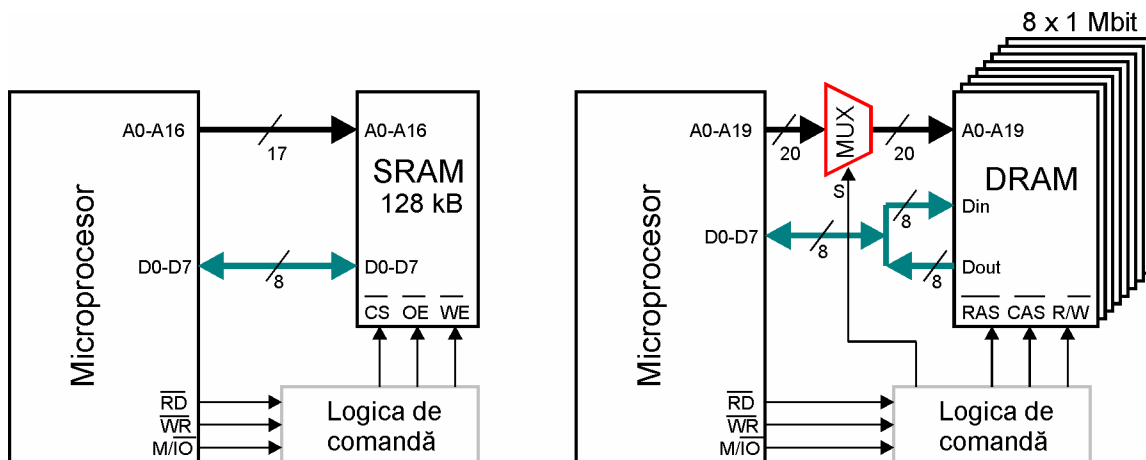
Pentru a accesa  $2^{20} = 1.048.567$  locatii de memorie sunt necesare 20 de linii de memorie (A0 .. A19). Începând cu primele circuite DRAM, în scopul reducerii numarului de pini ai capsulei, s-a utilizat o tehnica de *multiplexare* a adreselor, circuitul având doar 10 pini de intrare pentru adrese si doua semnale specifice de comanda – nRAS pentru comanda adreselor corespunzatoare decodificatorului de linii, respectiv nCAS pentru comanda adreselor corespunzatoare decodificatorului de coloane. Deoarece un circuit tipic de memorie DRAM nu are nici o intrare nCS, economia este de 10 terminale, rezultând o capsula cu 18 terminale la o capacitate de stocare de 1 Mbit, ceea ce este remarcabil.



**Figura 5.38.** Multiplexarea adreselor în cazul unei memorii DRAM (S este semnalul de selecție asociat MUX).

Întâi se aplica prima jumatate a cuvântului de adresa A0...A9, apoi a doua jumatate A10...A19. Circuitul de comanda contine doua lach-uri D de memorare a celor doua jumatati ale cuvântului de adresa. Prin intermediul liniilor A0...A9 se comanda decodificatorul care are rolul de a selecta o singura linie de cuvânt. Încarcarea A0...A9 în registrul din circuitul de comanda se face prin activarea liniei /RAS. Cu liniile A10...A19 se comanda blocul MUX/DMUX prin care se selecteaza o singura linie de bit (la citire prin MUX, iar la înscriere prin DMUX). Încarcarea adreselor A10...A19 în registrul corespunzator din circuitul de comanda se face prin activare liniei /CAS (*Column Address Strobe*). Linia de control stabileste care dintre cele doua circuite (MUX sau DMUX) este activ. Evident la citire este activ MUX-ul, iar la scriere DMUX-ul.

Mai exista la iesirea magistrelor liniilor de bit un registru RLB cu 256 bistabile D-latch. Schimbul de informatie între matrice si registru este bidirectional. In cazul operatiilor de citire este activata si linia ce valideaza in stare normala bufferul B pentru Dout.



**Figura 5.39.** Interfatarea memoriilor RAM la un sistem cu microprocesor (scheme de principiu)



## Împrospatarea

Pe frontal scazator al semnalului /RAS se înregistrează adresa liniei de cuvânt în registrul din circuitul de comanda. Se selectează o linie de cuvânt ce corespunde acestei adrese și se înscrie conținutul fiecărei celule ce corespunde lui LC selectat în RLB. Pe frontul crescător al semnalului /RAS se reînscris informația din registru regenerată în celulele apartinătoare liniei de cuvânt selectată. În continuare se trece la următoarea linie de cuvânt în vederea împrospătării tuturor celulelor.

Generarea adreselor pentru împrospatarea memoriei se face folosind un numărator pe 8 biți ce funcționează continuu. Se constată că nu intervine adresa ce stabilește linia de bit în care se face înscrierea sau citirea.

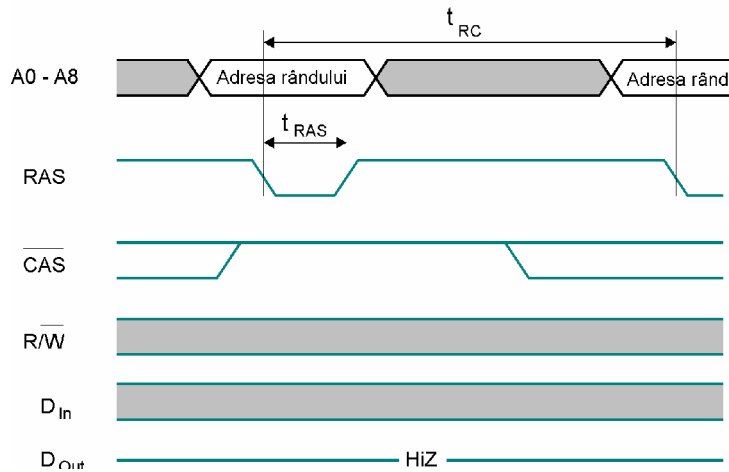


Figura 5.39. Diagramele de semnal pentru împrospatare (variantea simplificata).

## Citirea din memorie

Se aplică A0...A9 și se activează linia /RAS pentru încărcarea adresei liniei de cuvânt în latch-ul din circuitul de comandă, fapt ce duce la selecția unei linii de cuvânt și se scrie conținutul celulelor de pe linia de cuvânt în registrul RLB. Apoi într-o altă secvență se aplică la intrare A10...A19. Semnalul /WE trebuie să fie pe 1 (se execută o operație de citire). După aplicarea celei de-a doua jumătăți a adresei se aplică semnalul /CAS activ pe "0". Pe frontul scazator al semnalului /CAS se memorează A10...A19 în latch-ul de 10 biți din circuitul de comandă. Se aplică acești 10 biți MUX-ului, se selectează una dintre cele 1024 de ieșiri ale registrului RLB și se transmite la bufferul B înspre ieșire D<sub>Out</sub>. Simultan bufferul B este trecut în stare normală.

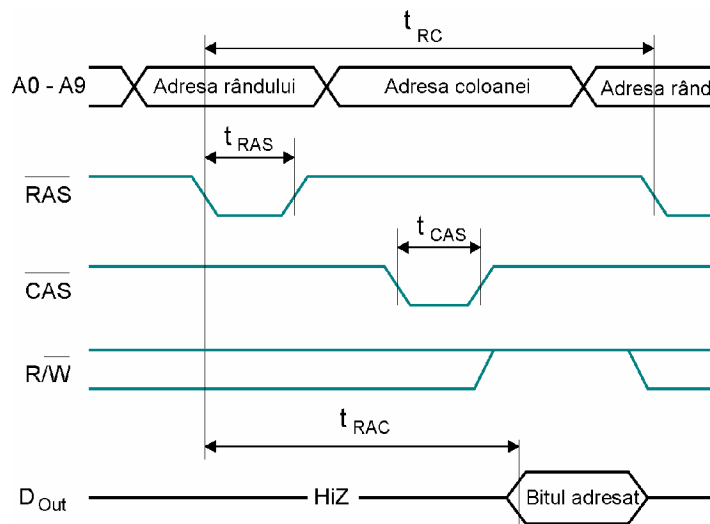


Figura 5.40. Diagramele de semnal pentru citire (variantea simplificata).

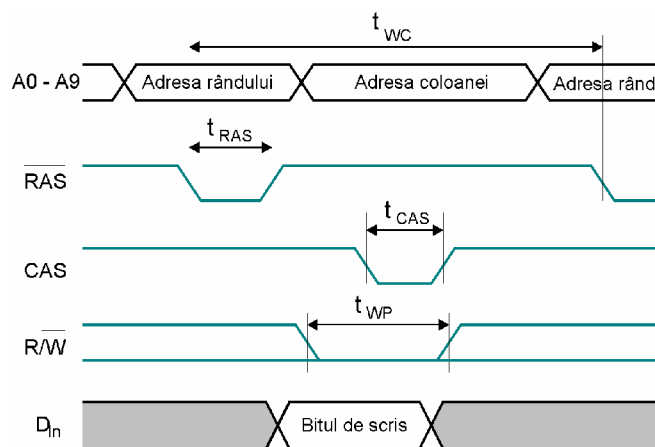


Figura 5.41. Diagramele de semnal pentru scriere (varianta simplificata).

## 5.7. EXTINDEREA CAPACITATII DE MEMORARE

Extinderea capacitatii de memorare este un caz des întâlnit în practica, prin care se dorește realizarea unei memorii având o capacitate mai mare decât cea a unui singur circuit integrat, fie ca număr de biți ai cuvântului memorat (extensia în latime), fie ca număr de cuvinte memorate (extensia în adâncime), fie în ambele direcții (extensia mixta).

În toate aceste situații, etapele de proiectare sunt:

- determinarea numărului de circuite de memorie necesare
- determinarea schemei electrice a electronicii suplimentare, dacă ea există.

### 5.7.1. Extensia numărului de biți ai cuvântului adresat

Acesta este cel mai simplu caz – deoarece numărul de cuvinte memorate este suficient pentru aplicația concretă. În exemplul următor, presupunem ca se dorește modernizarea memorie operative a unui sistem de achiziție video prin trecerea de la imagini alb-negru în 256 de nuanțe de gri (codificate pe 8 biți) la imagini color codificate pe 24 de biți (16.777.216 culori). Imaginea are o rezoluție de 320 x 240 de puncte (76.800 puncte), se achiziționează și prelucrează o singură imagine și de aceea o memorie de 128 kbit x 8 era suficientă pentru cazul alb-negru.

În acest exemplu determinarea numărului de circuite necesare este banal, fiind evident necesare 3 asemenea circuite (figura 5.41). Memoria inițială folosea un singur circuit, cea modernizată utilizează trei asemenea circuite. Conectarea se face legând în paralel linie cu linie toate intrările celor trei memorii (adresele A0-A15, liniile de secție SEL și linia de comandă a scrierii nWR). Liniile de ieșire sunt *separate* și formează noua magistrală de 24 de biți.

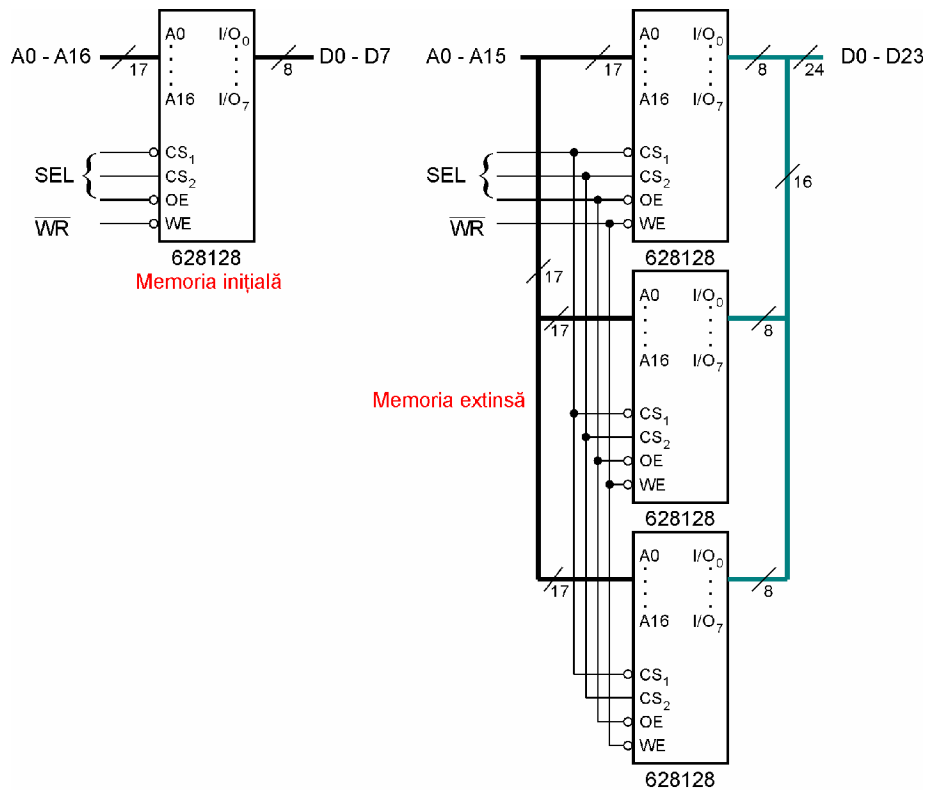


Figura 5.41. Extinderea latimii cuvântului de memorare de la 8 la 24 de biti.

### 5.8.2. Extensia numărului de cuvinte adresate

În acest caz presupunem ca la sistemul de achiziție prezentat anterior se dorește mărirea rezoluției de la 320 x 240 de pixeli la 640 x 400 pixeli (256.000 de puncte), iar reprezentarea pe 8 biti cu niveluri de gri este suficientă. Numărul de circuite de memorie necesare este:

$$N = \frac{\text{Capacitatea totală}}{\text{Capacitatea unui circuit}} = \frac{256.000 \text{ pixeli}}{131.072 \text{ biti}} = 2 \text{ cipuri}$$

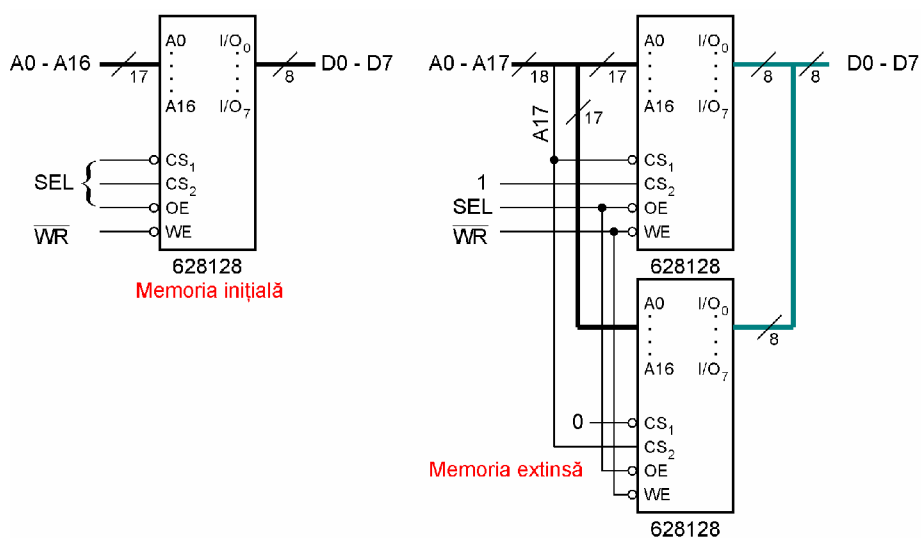


Figura 5.42. Extinderea capacității de memorare de la 128 la 256 kbiti.

În acest caz numărul de adrese necesare sistemului de achiziție este 17, intervenind suplimentar și A16, care va fi utilizată pentru selectarea alternativă a celor două circuite de memorare. Dacă memoria ar fi avut un singur pin nCS ar fi fost necesar un inversor între A16 și intrarea celui de-al doilea cip de memorie, A16 fiind legată direct la intrarea nCS a primului circuit. Deoarece sunt disponibile două intrări de selecție circuit, unul activ jos și altul sus, decodificarea este mult mai simplă (figura 5.42). Pretul plătit pentru această simplitate este reducerea numărului de linii de selecție de la 3 în cazul inițial la 1 la memoria extinsă.

Există situații în care se dorește o extindere substanțială, de exemplu atunci când se dorește realizarea unei memorii Flash de 1 Moctet utilizând 8 circuite 29F010 de 128 octeți. În acest caz prezenta unui decodificator 3:8 este indispensabilă. Pentru a dresa 1 Moctet sunt necesare 20 de linii de adresă, 17 mai puțin semnificative fiind legate în paralel la cele 8 memorii Flash notate #0...#7, cel 3 mai semnificative A17-A19 fiind intrări în DCD 3:8 74LS138. În acest caz memoria extinsă are aceleași linii de control ca memoria inițială (nSEL, nOE, nWR, nRD) nemaifiind necesare alte modificări.

Pentru schema electrică extinsă trebuie făcute câteva comentarii:

- a. Legarea liniei nRD la nOE este singura soluție posibilă. Analizând diagramele de timp pentru memoria Flash rezultă că mai întâi uP emite o adresă pe liniile A0 – A19 care este decodificată. Datorită hazardului combinational propriu DCD, dacă ar exista doar nCE, este posibil ca pentru intervale de timp scurte, de câțiva  $t_p$ , două ieșiri sau mai multe ieșiri ale DCD să fie simultan pe 0, validând accidental 2 sau multe memorii. Este puțin probabil ca memoriile să se distrugă datorită unui singur conflict pe magistrală (*bus contention*) de câteva zeci de ns, dar dacă acest proces se repetă, el va conduce la încălzirea etajelor finale din memoriile Flash și în final la distrugerea lor. Prezenta celei de-a doua linii de selecție și legarea ei la nRD permite strobarea ieșirii memoriilor la momente de timp bine definite și după stabilizarea ieșirii DCD și memoriilor, ceea ce elimină complet problema menționată.
- b. În vederea reducerii în continuare a consumului, se poate înlocui circuitul 74LS138 cu varianta 74HCT138.
- c. Linia nSEL poate fi legată la masă în cazul memoriei inițiale, utilizarea ei reduce doar semnificativ consumul, aspect esențial în cazul echipamentelor portabile, alimentate la baterii.
- d. Linia nSEL poate fi utilizată similar în cazul memoriei extinse.

### 5.8.3. Extensia mixtă

Cea mai complexă extindere este cea mixtă. Pentru ilustrare vom considera două implementări de memorie de 256 cuvinte pe 12 biți, utilizând circuite SRAM de 1 Mb (organizare 128 cuvinte de 8 biți, respectiv de 256 k cuvinte de 4 biți).

În ambele cazuri sunt necesare 3 circuite de memorie, iar al doilea caz este chiar simplu. Pentru primul caz se vor utiliza alternativ 128 cuvinte de 8 biți din primul și 128 cuvinte de 4 biți din al doilea circuit, respectiv 128 cuvinte de 8 biți din al treilea circuit împreună cu celelalte 128 cuvinte de 4 biți din al doilea circuit. Cu alte cuvinte, al doilea circuit de memorie este divizat logic în două jumătăți, asociate celorlalte două circuite (figura 5.44). Pentru a selecta care jumătate este utilizată, la ieșirea memoriei #2 se folosește un multiplexor cvadruplu 2:1, de tip 74LS157.

La schema din figura 5.44 trebuie făcute câteva observații:

- a. Inversorul de pe linia de adresă A17 poate lipsi dacă se renunță la linia globală de selecție nCE.
- b. Multiplexorul 74LS157 nu asigură transferul bidirecțional al informației, el trebuie înlocuit cu circuitele trei stări comandate de nRD și nWR pentru a putea asigura atât citirea cât și scrierea memoriei SRAM rezultate.

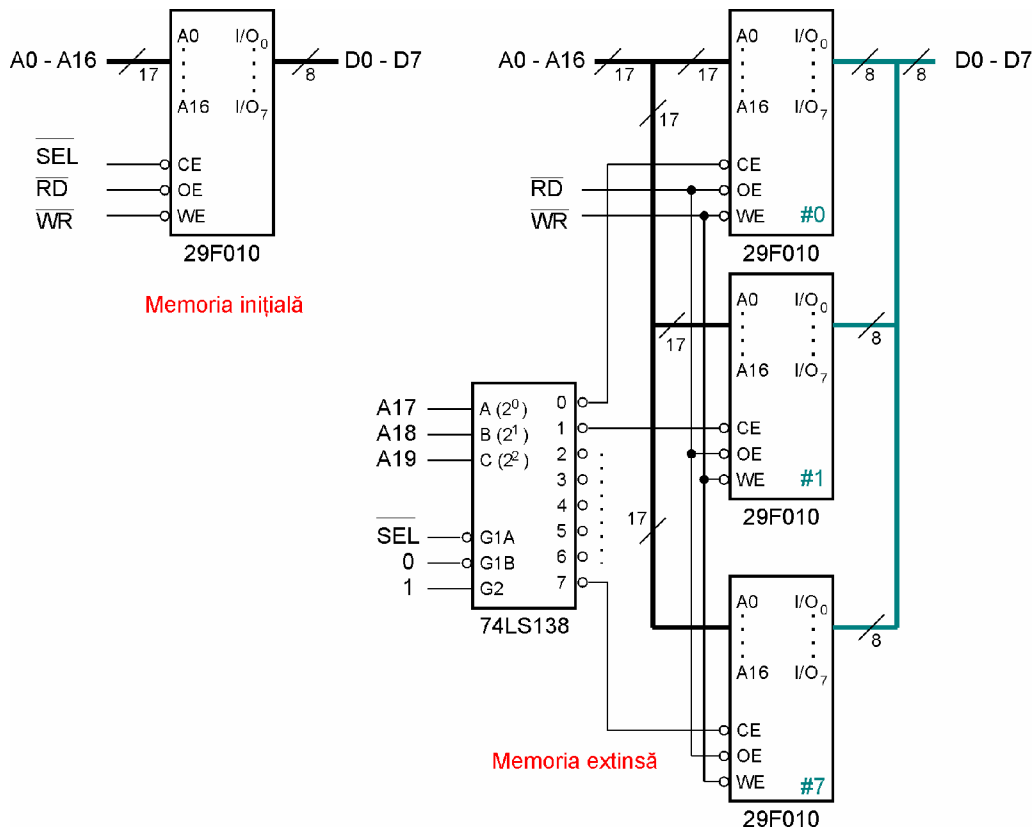


Figura 5.43. Extinderea capacitii de memorare de la 128 la 256 kbiti.

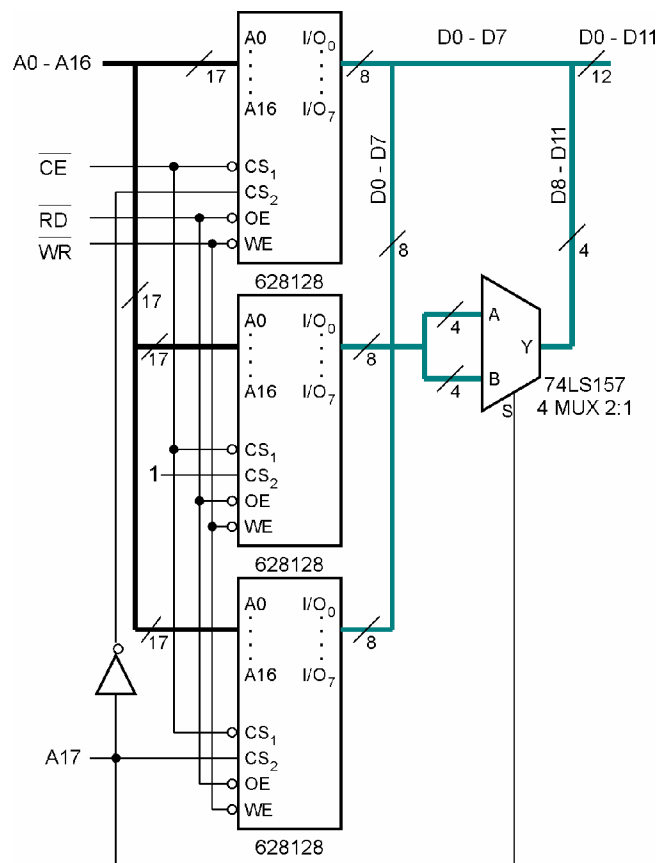


Figura 5.44. Extinderea mixta a capacitii de memorare de la 128 k x 8 la 256 k x 12 biti.

## 5.8. MEMORII SPECIALE

### 5.8.1. Memoria LIFO

### 5.8.2. Memoria FIFO

### 5.8.3. Memorii EEPROM seriale

### 5.8.4. Memorii SRAM pe magistrala I2C

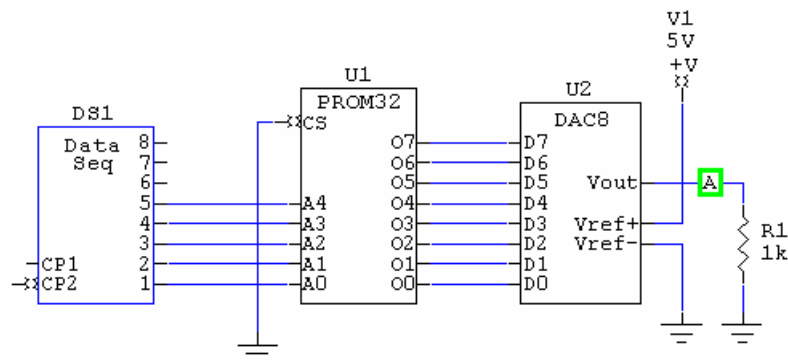
## 5.9. APLICATII ALE MEMORIILOR

### 5.9.1. Memorii auxiliare

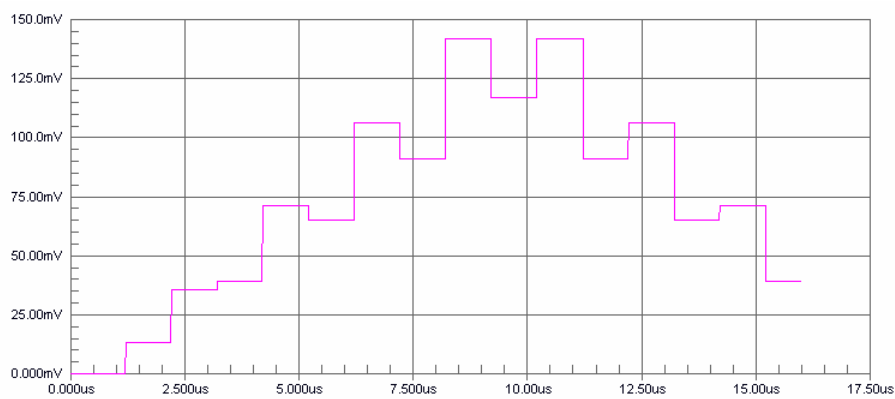
- Flash reader PC

### 5.9.2. Generatoare de functii (generatoare de semnal programabile)

*Descrierea functionarii*



**Figura 5.4x.** Generator numeric de functii – schema de principiu.



**Figura 5.4x.** Generator numeric de functii – diagrame de semnal.

### 5.9.3. Implementarea functiilor logice

### 5.9.4. Automate programabile

### 5.9.5. Firmware

Memoriile ROM sunt cel mai adesea utilizate în micro sisteme pentru stocarea programelor și a constantelor. Aceasta informație se numește *firmware* și este cea de-a treia componentă a unui sistem alături de *hardware* și *software*. Unele sisteme de calcul de dimensiuni reduse stochează în ROM interpretoare Basic (mai ales în cazul calculatoarelor de buzunar) sau sisteme de operare și aplicații (WinCE, Word, Excel și altele în cazul dispozitivelor *hand-held* sau PDA).

Sistemele cu microcontrolere moderne stochează codul firmware direct în memoria ROM a controlerului. Dacă această memorie este EEPROM sau Flash, reprogramarea ei este extrem de simplă și nu necesită extragerea circuitului din montaj (ce s-ar întâmpla dacă la o revizie proprietarul unui autoturism scump ar vedea că i se desface autoturismul pentru a extrage un circuit?).

Tendința prezenta este de a folosi memoria Flash internă a microcontrolerelor ca memorie de program, iar cea EEPROM ca memorie de date nevolatilă. De obicei memoria EEPROM este de dimensiune mai mică decât cea Flash și uneori este necesară utilizarea unui EEPROM extern, de cele mai multe ori serial (*de ce?*)

### 5.9.6. Memoria bootstrap

Pentru majoritatea calculatoarelor sistemul de operare și aplicațiile se încarcă în memoria RAM. Această abordare permite alegerea sistemului de operare potrivit și a aplicațiilor dorite. Pentru încărcarea efectivă, calculatorul rulează un mic program aflat în memoria ROM, numit *bootstrap* (ceea ce în traducere liberă înseamnă a te trage în sus de propriile șireturi de la cizme).

Este interesant de subliniat aici că această comportare o au și unele FPGA (cele produse de Xilinx sunt cel mai cunoscut exemplu), care la alimentare se *configurează* conform unui cod exterior stocat într-o memorie ROM.

### 5.9.7. Tabele de date

### 5.9.8. Conversoare de cod

Reprezintă un caz particular de implementare de funcții.

This document was created with Win2PDF available at <http://www.daneprairie.com>.  
The unregistered version of Win2PDF is for evaluation or non-commercial use only.