



# Proiectarea cu Micro-Procesoare

**Lector: Mihai Negru**

An 3 – Calculatoare și Tehnologia Informației

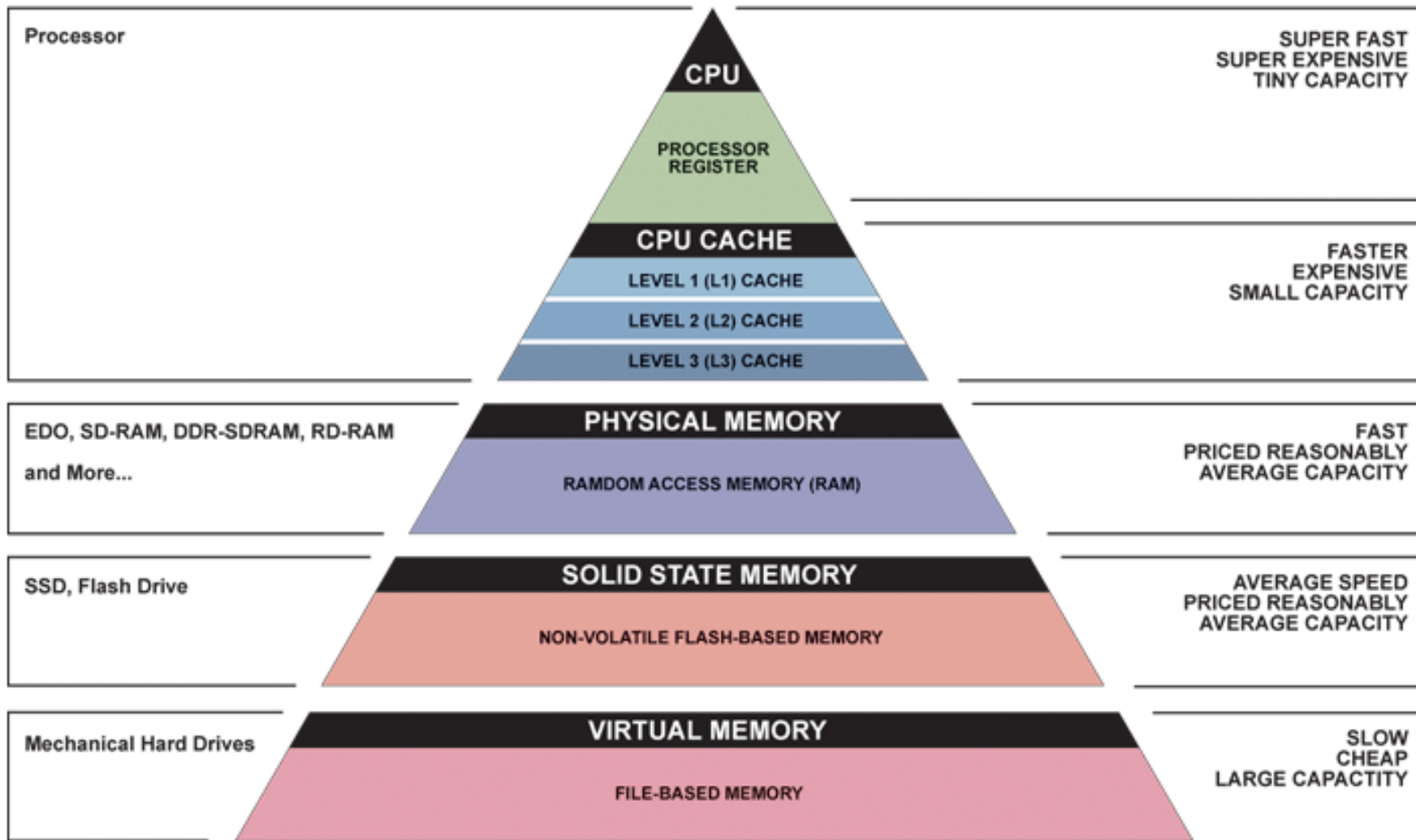
Seria B

**Curs 11: Interfațarea Memoriei**

<http://users.utcluj.ro/~negrum/>



# Ierarhia Tipică a Memoriei



▲ Simplified Computer Memory Hierarchy  
Illustration: Ryan J. Leng



# Clasificarea Memoriilor



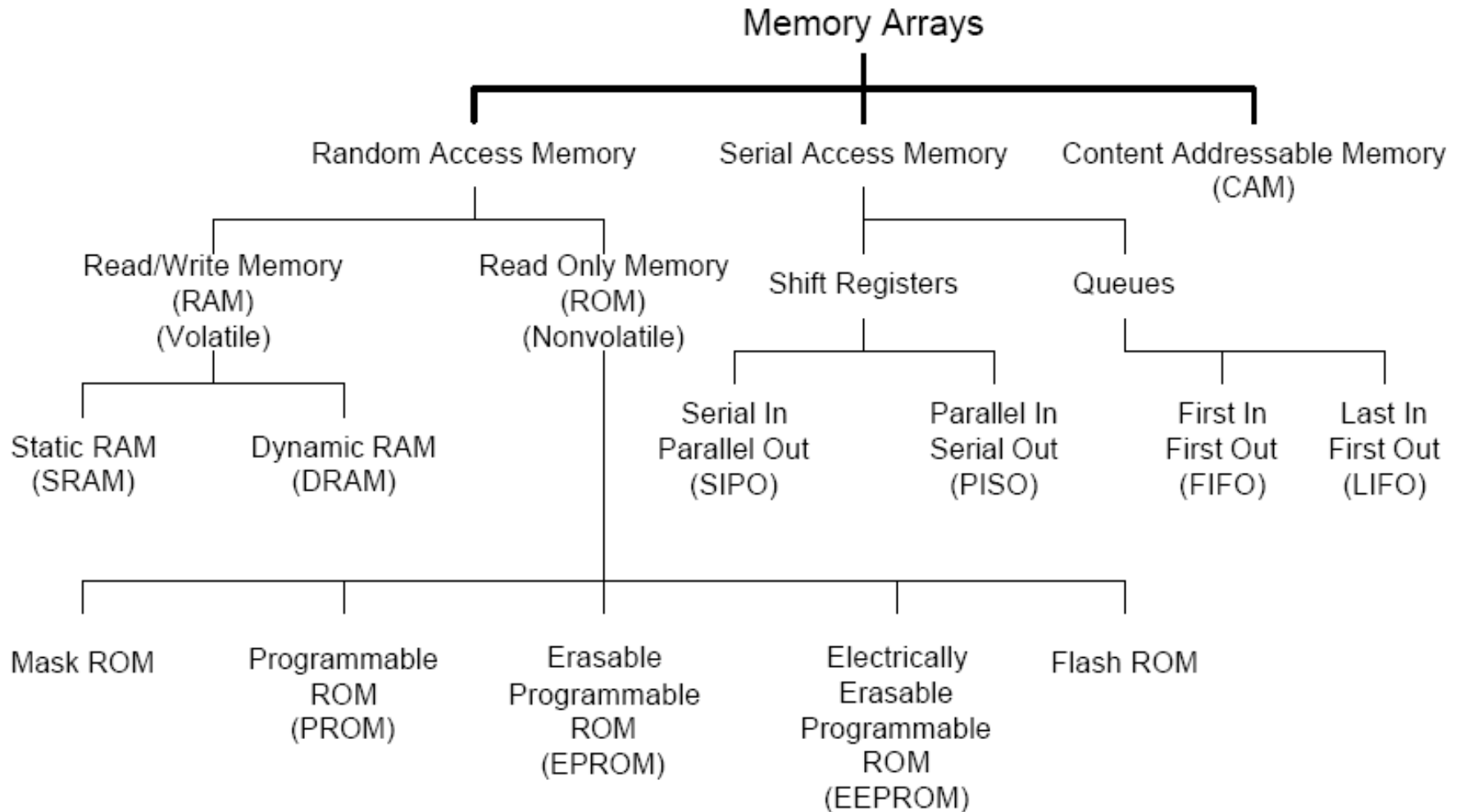
Read Write (RWM)		NVRWM	ROM
Random Access	Non-Random Access	EPROM	Mask-prog. <b>ROM</b>
<b>SRAM</b> (cache, regiștri) <b>DRAM</b> (memoria principală)	FIFO, LIFO Registru de deplasare	EEPROM FLASH	Electrically-prog. PROM

- Memorie statică / Memorie dinamică
- Memorie volatilă / Memorie nonvolatilă (NV)
- Read only (ROM) / Random Access (RAM)

Modul de acces – aleator, serial, adresabil prin continut

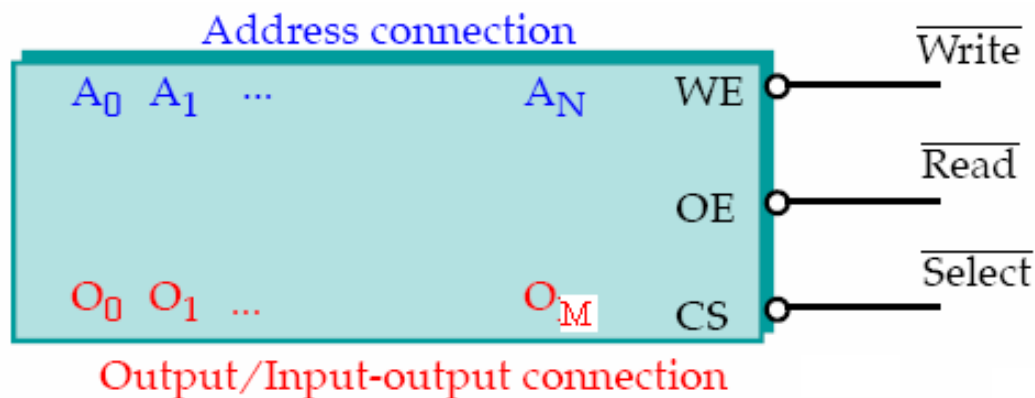


# Clasificarea Memoriilor





# Configurația generală a pinilor



- **Pini de adresă**

- $A_0 \dots A_N \Rightarrow$  numărul locațiilor de memorie =  $2^{N+1}$

- **Pini de date**

- $M \Rightarrow$  dimensiunea locațiilor de memorie
- Bidirecționali, ieșire 3-state (#OE)

- **Pini de control**

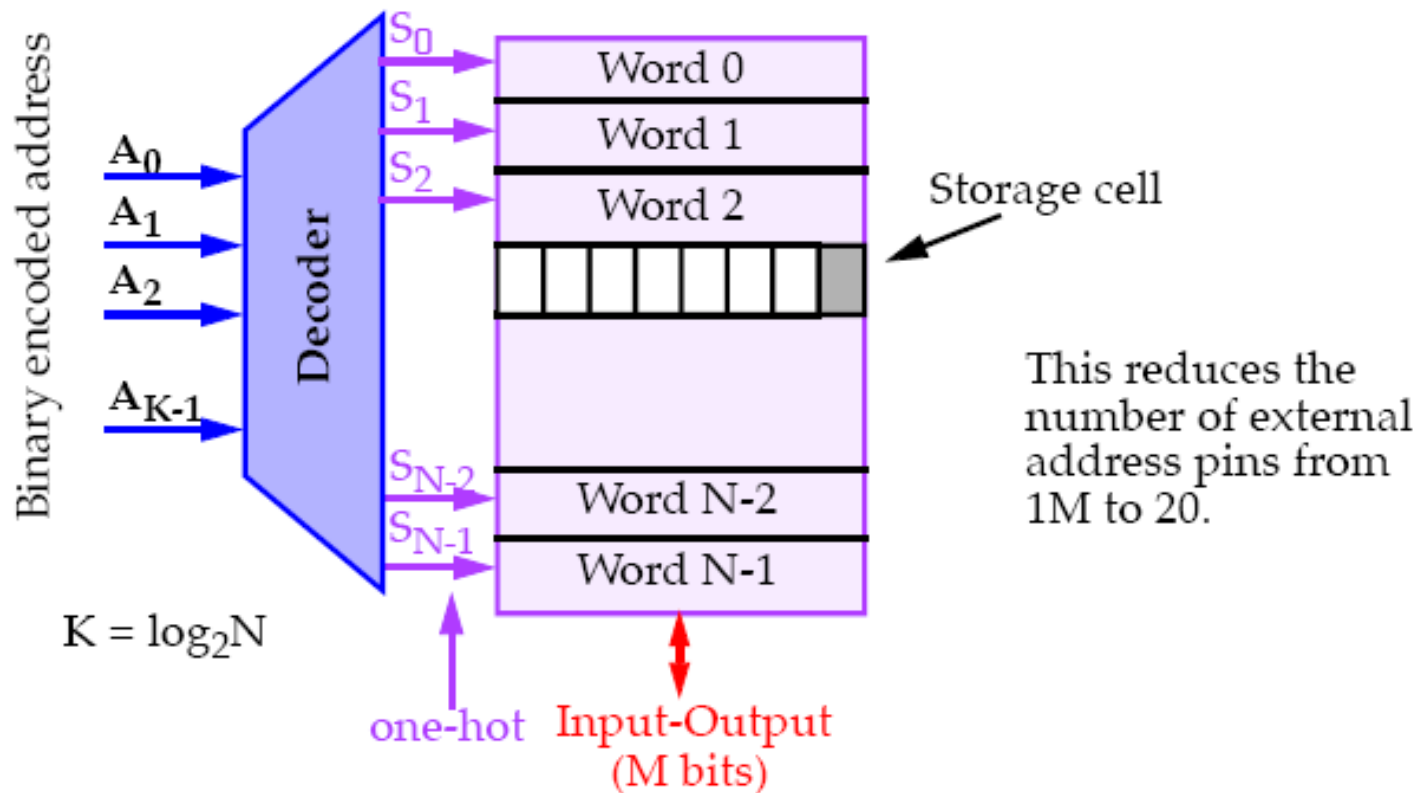
- Chip select/enable  $\Rightarrow$  activează dispozitivul
- Read (#OE) / Write (#WE)  $\Rightarrow$  selecția operației (citire / scriere)



# Arhitectura memoriei

## Memory Architecture

Add a decoder to solve the package problem:

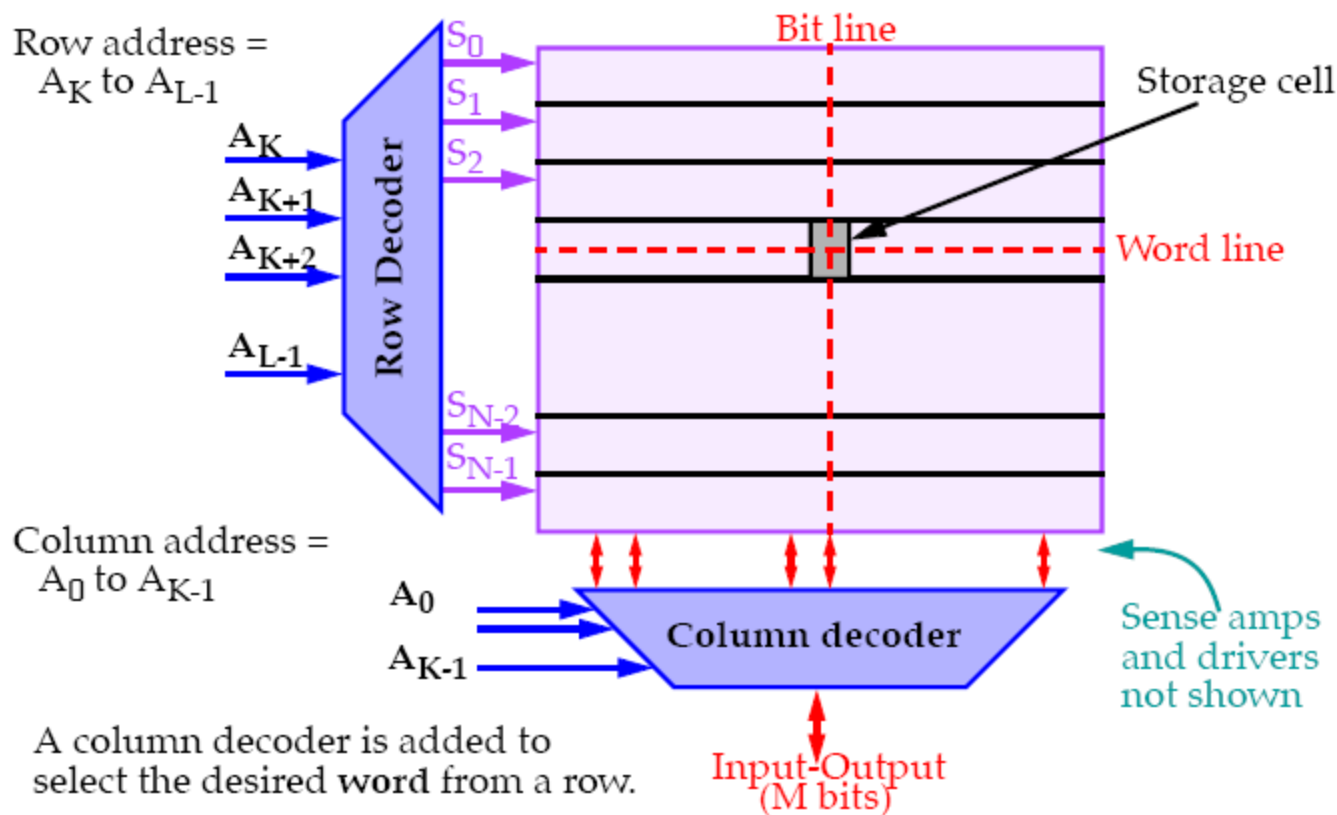


Nu se rezolvă problema raportului dintre lățimea și lungimea matricii

Această organizare este foarte lentă, deoarece firele verticale sunt foarte lungi



# Arhitectura memoriei



Dimensiunile verticală și orizontală sunt de obicei similare

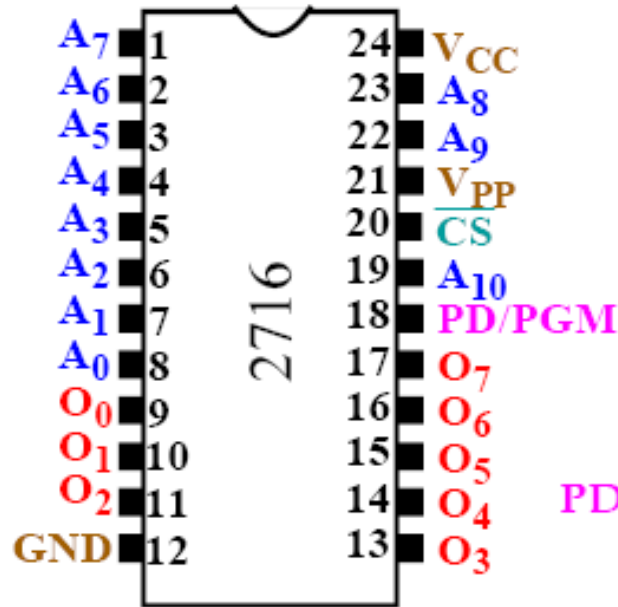
Mai multe cuvinte sunt stocate pe aceeași linie



# ROM / EPROM



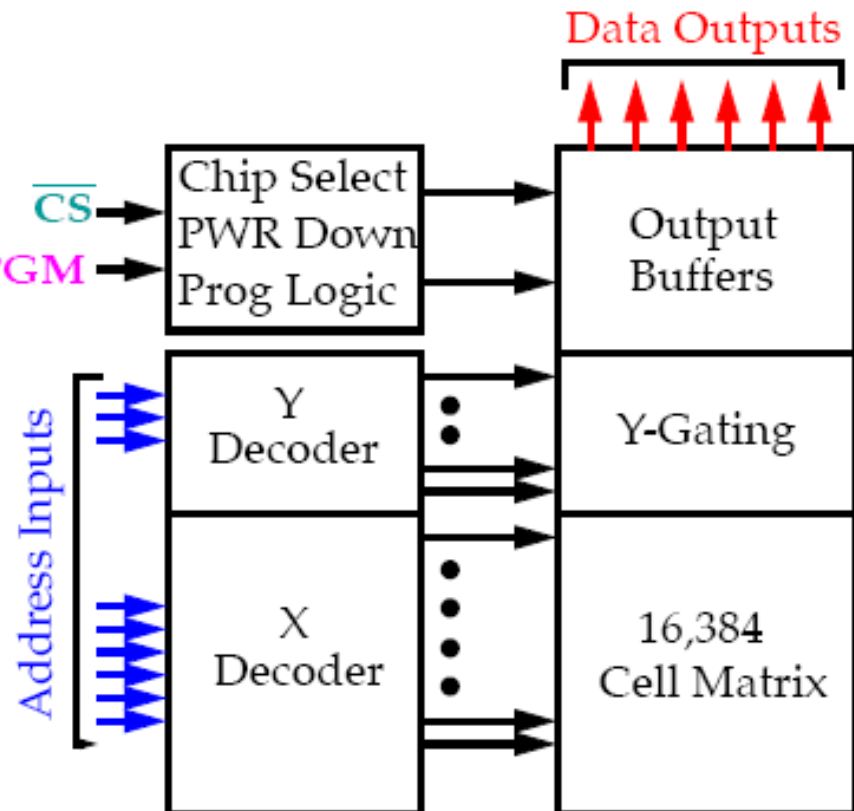
Intel 2716 EPROM (2K X 8):



2K x 8 EPROM

Pin(s)	Function
$A_0-A_{10}$	Address
PD/PGM	Power down/Program
$\overline{CS}$	Chip Select
$O_0-O_7$	Outputs

$V_{PP}$  is used to program the device by applying 25V and pulsing PGM while holding  $\overline{CS}$  high.



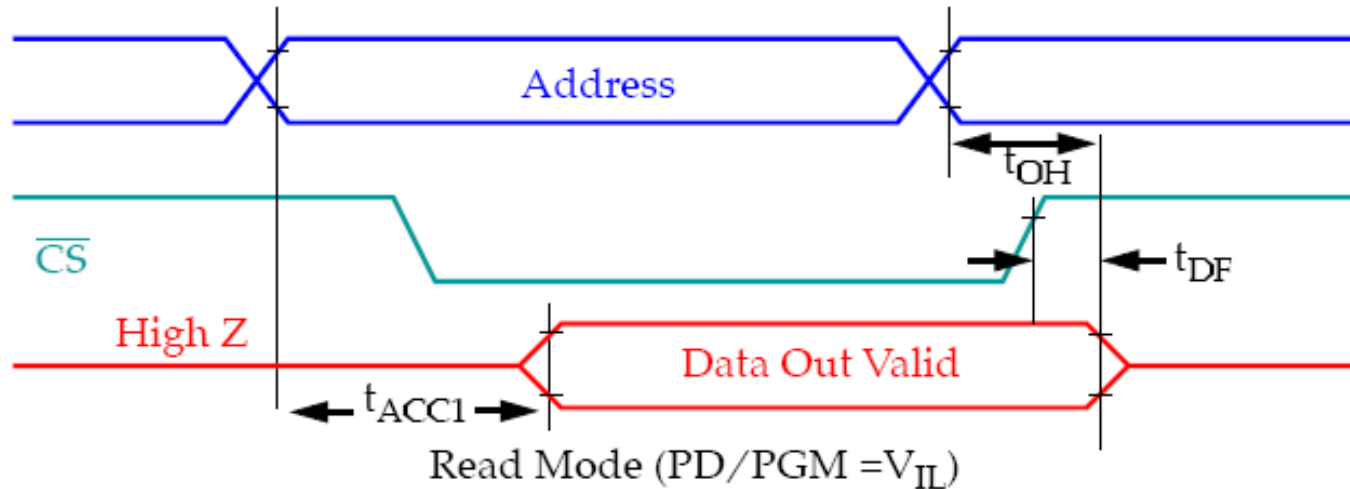




# ROM / EPROM



2716 Timing diagram:



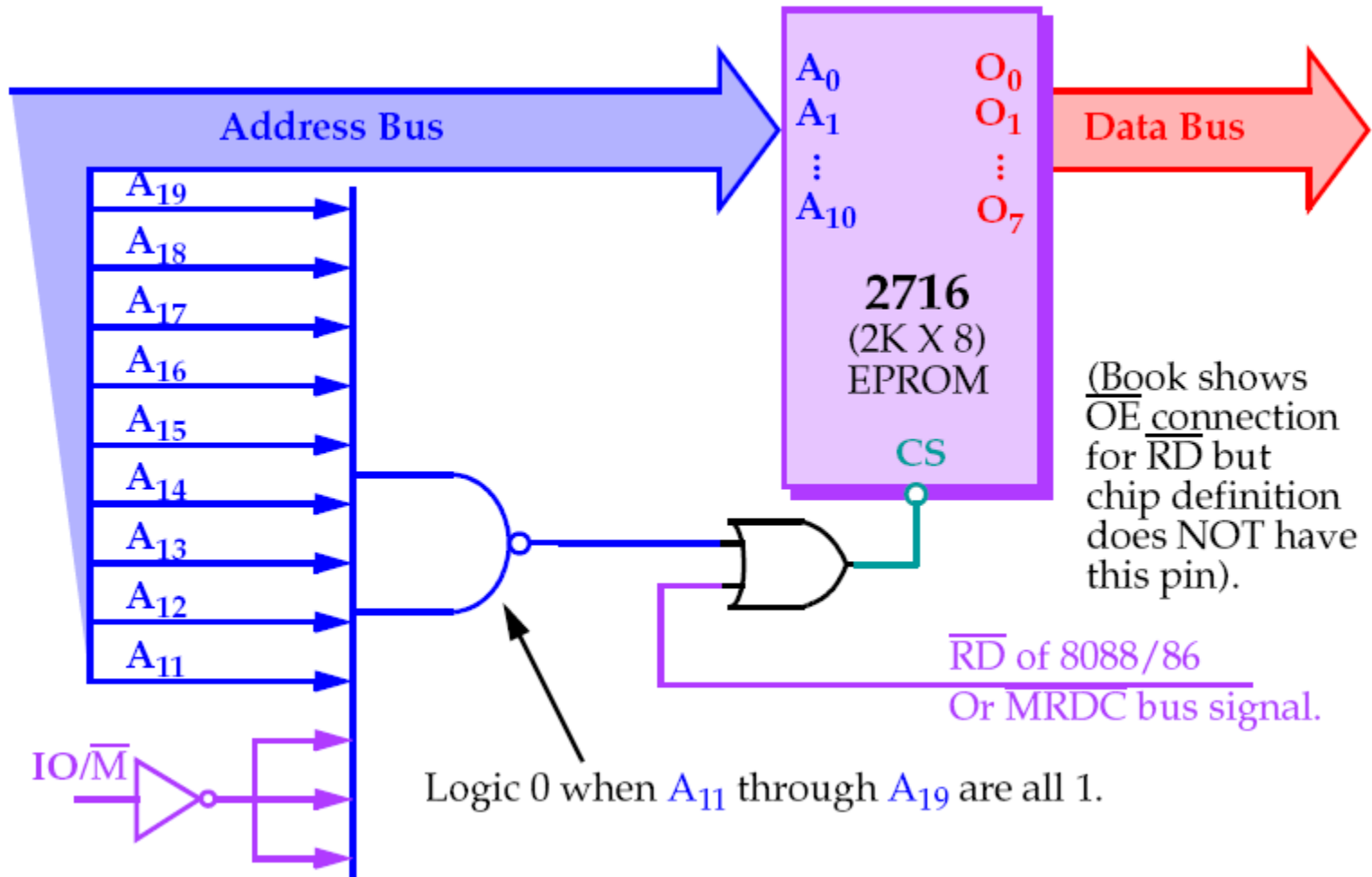
Sample of the data sheet for the 2716 A.C. Characteristics.

Symbol	Parameter	Limits			Unit	Test Condition
		Min	Typ.	Max		
$t_{ACC1}$	Addr. to Output Delay		250	450	ns	PD/PGM = $\overline{CS} = V_{IL}$
$t_{OH}$	Addr. to Output Hold	0			ns	PD/PGM = $CS = V_{IL}$
$t_{DF}$	Chip Deselect to Output Float	0		100	ns	PD/PGM = $V_{IL}$
...	...	...	...	...	...	...

This EPROM requires a wait state for use with the 8086 (460ns constraint).



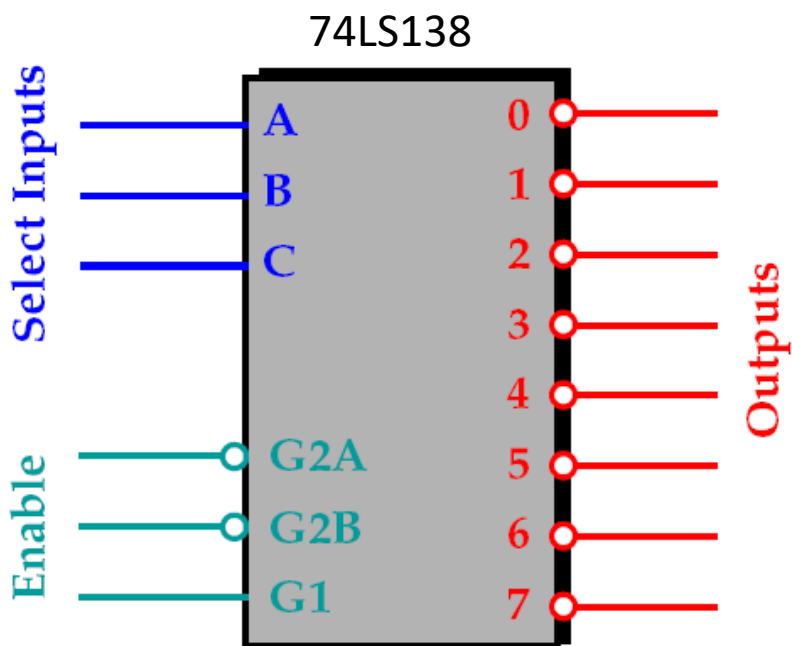
# ROM / EPROM



Domeniul de adrese: FF800h - FFFFFh



# Decodificarea adreselor



Inputs						Output							
Enable			Select			0	1	2	3	4	5	6	7
G2A	G2B	G1	C	B	A	0	1	2	3	4	5	6	7
1	X	X	X	X	X	1	1	1	1	1	1	1	1
X	1	X	X	X	X	1	1	1	1	1	1	1	1
X	X	0	X	X	X	1	1	1	1	1	1	1	1
0	0	1	0	0	0	0	1	1	1	1	1	1	1
0	0	1	0	0	1	1	0	1	1	1	1	1	1
0	0	1	0	1	0	1	1	0	1	1	1	1	1
0	0	1	0	1	1	1	1	1	0	1	1	1	1
0	0	1	1	0	0	1	1	1	1	0	1	1	1
0	0	1	1	0	1	1	1	1	1	1	0	1	1
0	0	1	1	1	0	1	1	1	1	1	1	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	0

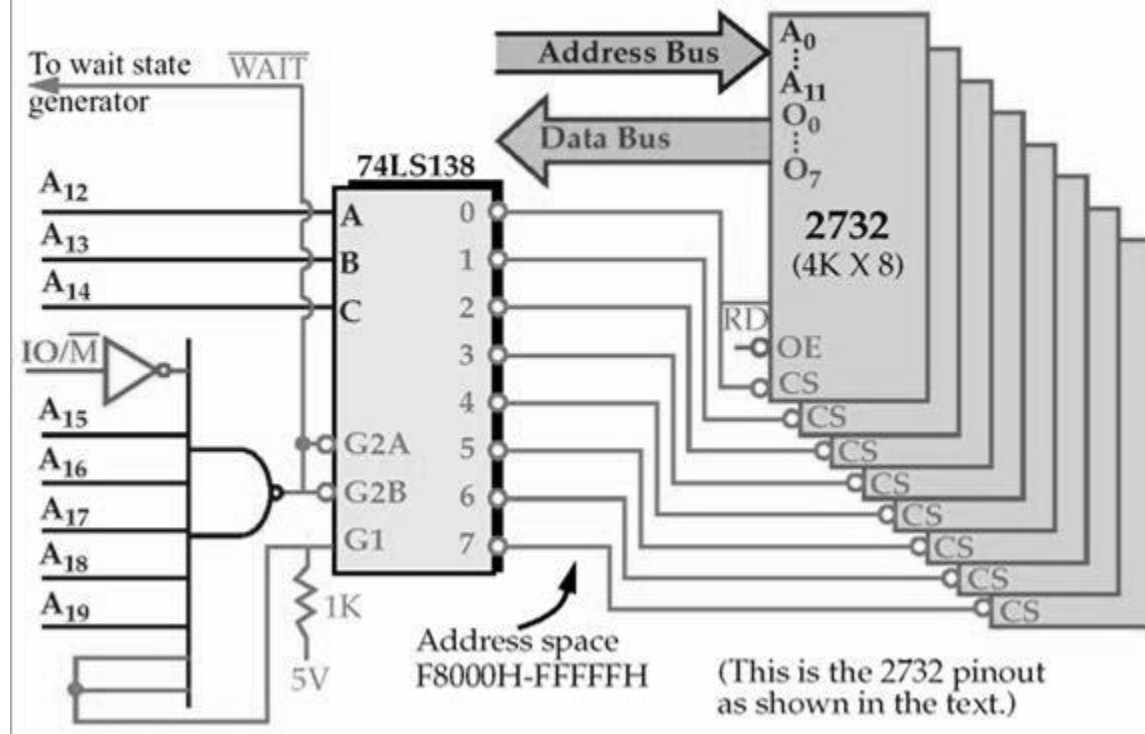
Note that all *three* Enables (G2A, G2B, and G1) must be active, e.g. low, low and high, respectively.



# ROM / EPROM



8088 and 80188 (8-bit) EPROM Memory Interface



EPROM	Density (bits)	Capacity (bytes)
2716	16K	2K × 8
2732	32K	4K × 8
27C64	64K	8K × 8
27C128	128K	16K × 8
27C256	256K	32K × 8
27C512	512K	64K × 8
27C010	1M	128K × 8
27C020	2M	256K × 8
27C040	4M	512K × 8

8088 – mod minim

EPROM  $8 \times 2732 = 8 \times 4\text{kB} = 32 \text{ KB}$

Organizarea adreselor:

0: F8000- F8FFF;

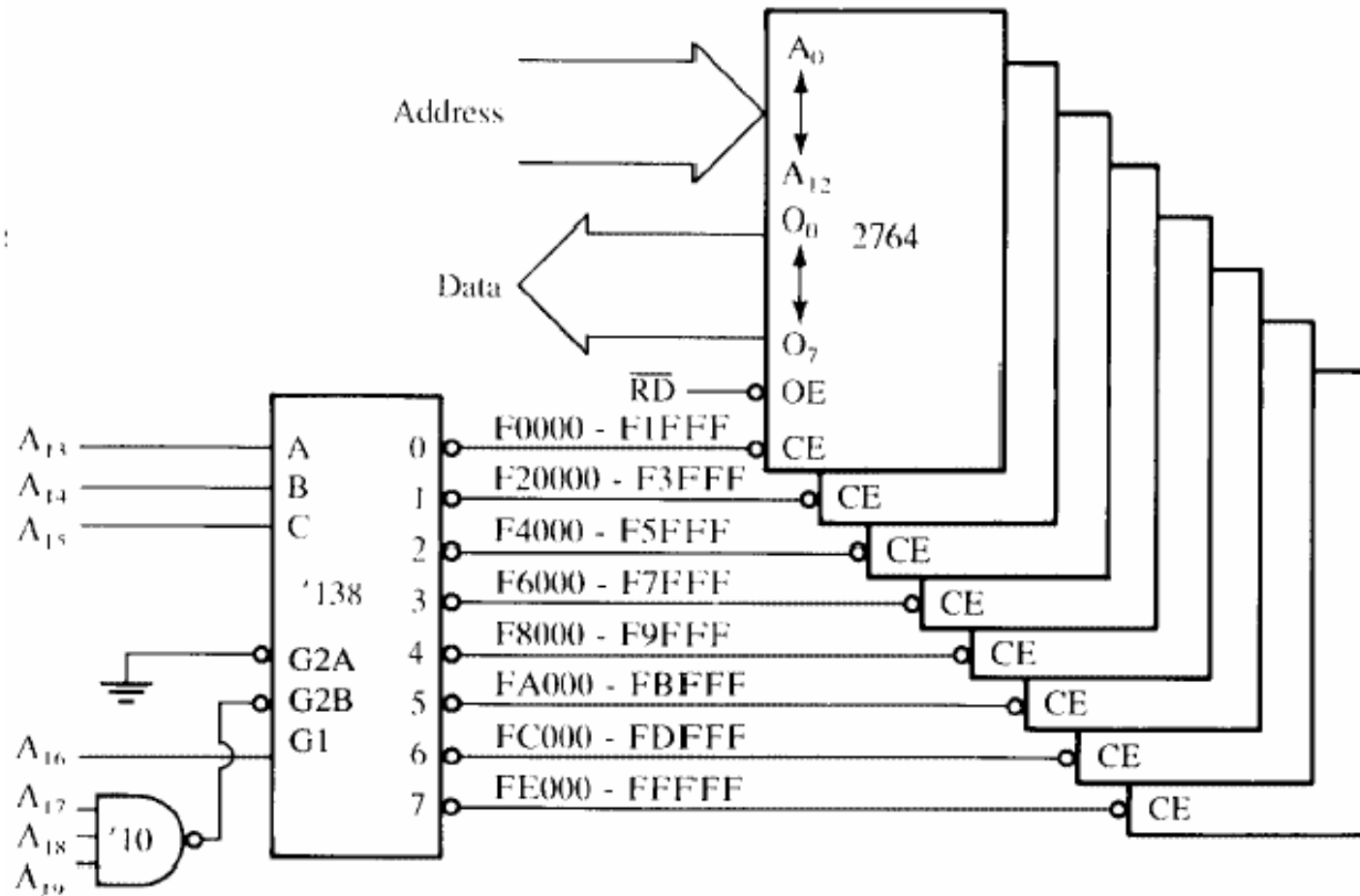
1: F9000- F9FFF;

...

7: FF000-FFFFF



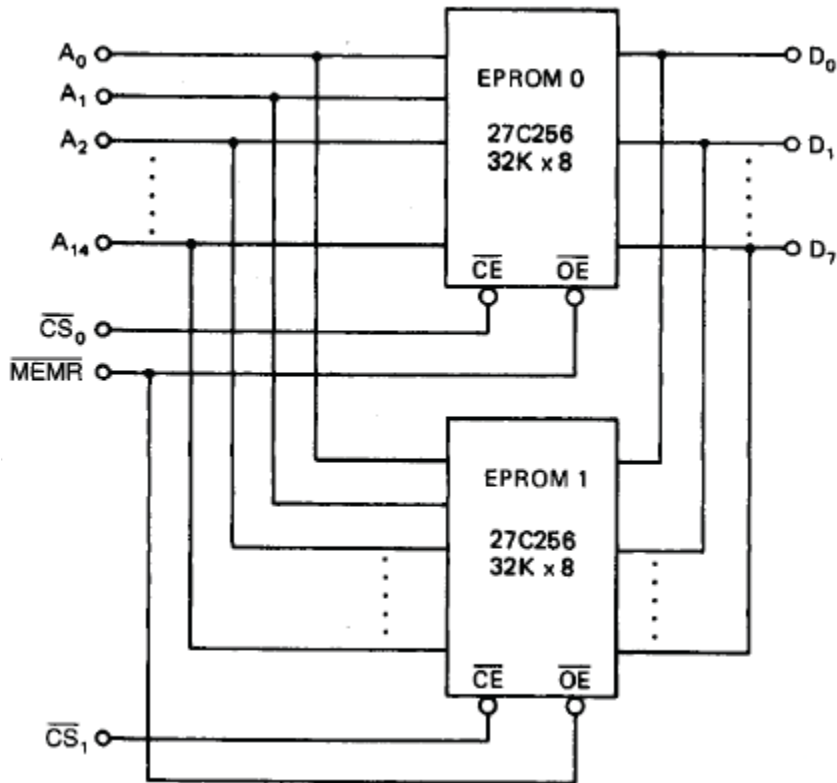
# ROM / EPROM



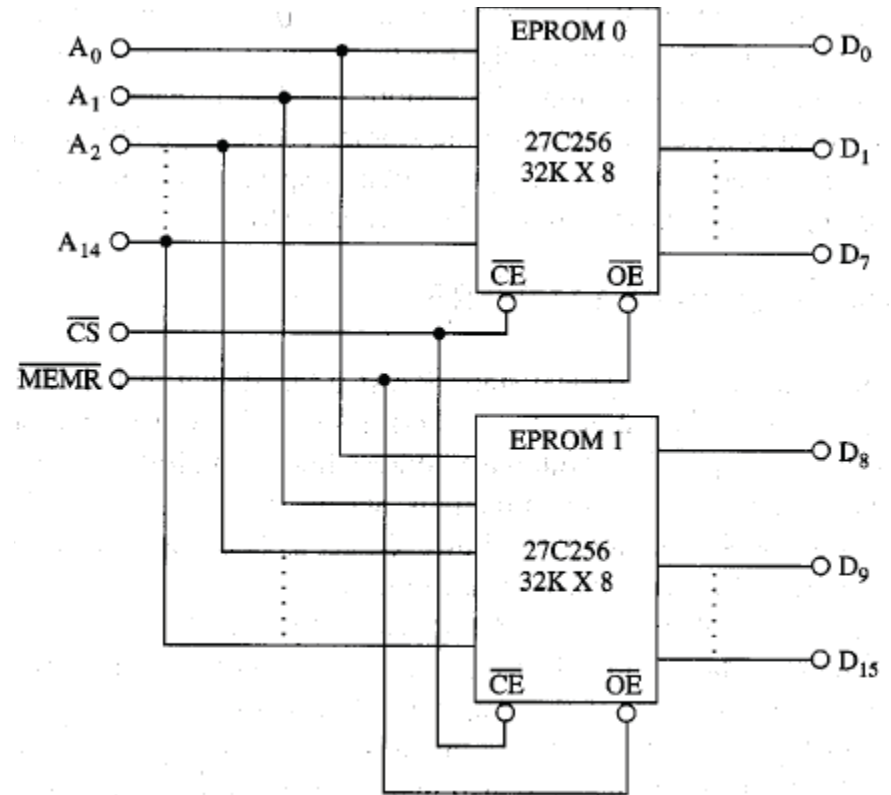
- 8088:  $8 \times 2764 = 8 \times 8 \text{ KB} = 64 \text{ KB}$
- Spațiul de adrese: F0000 – FFFFF



# Conectarea memoriilor



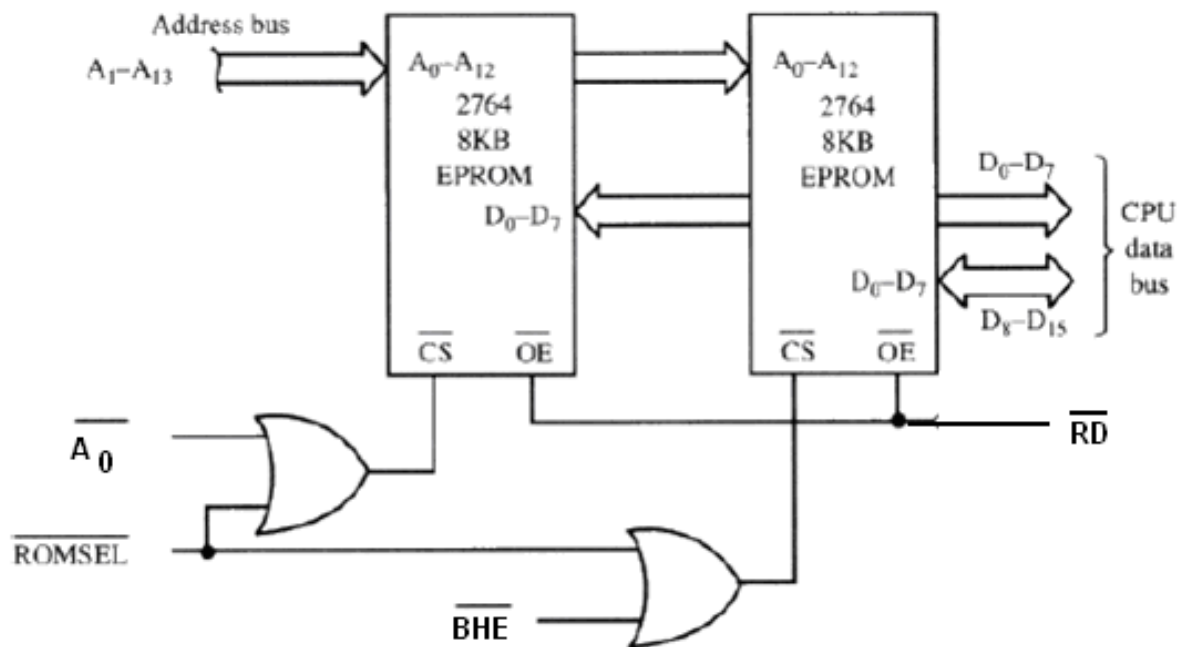
**Conectarea paralelă:** extinde adâncimea memoriei



**Conectarea serială:** extinde lațimea cuvântului



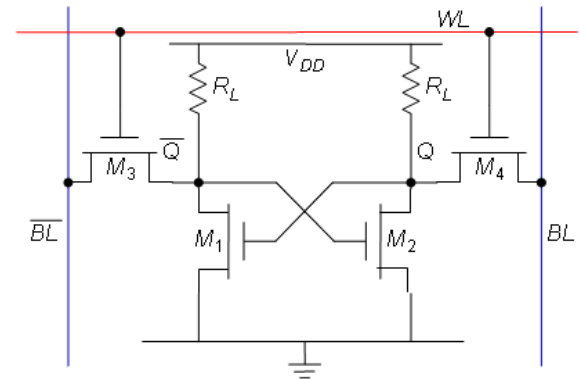
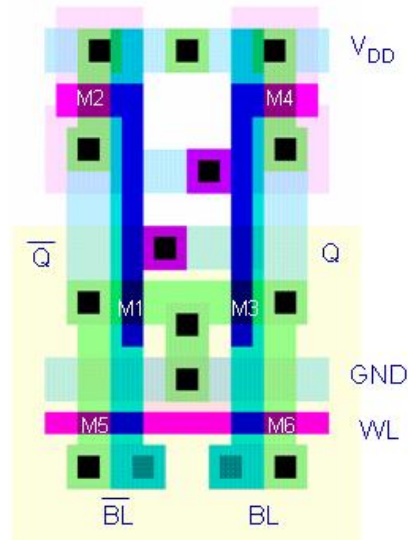
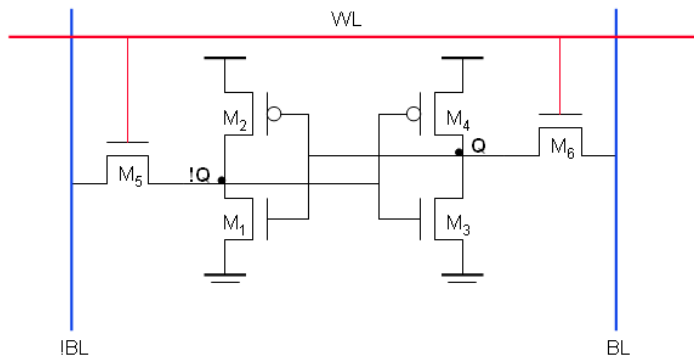
# Conexiunea la o magistrală de 16 biți (8086)



#BHE	A0	Explicatie
0	0	Acces pe 16 biți (aliniat)
0	1	Byte superior, de la adresă impară
1	0	Byte inferior, de la adresă pară
1	1	Combinatie nepermisă



# Static RAM (SRAM)



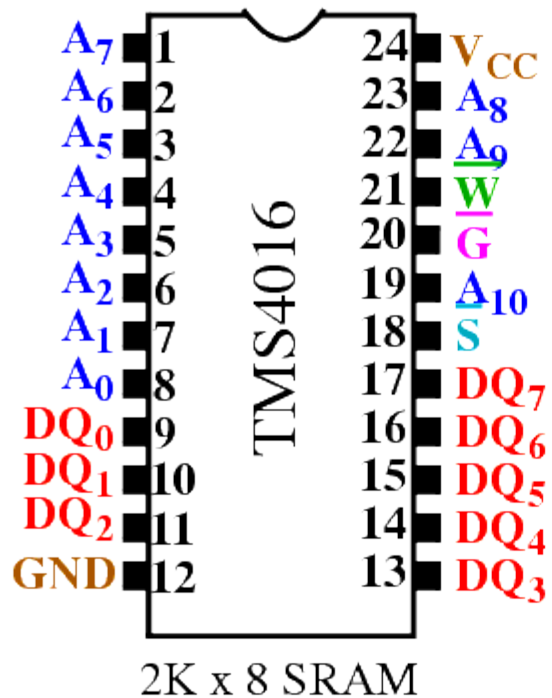
**Table 12-2** Comparison of CMOS SRAM cells used in 1-Mbit memory (from [Takada91])

	Complementary CMOS	Resistive Load	TFT Cell
Number of transistors	6	4	4 (+2 TFT)
Cell size	58.2 $\mu\text{m}^2$ (0.7- $\mu\text{m}$ rule)	40.8 $\mu\text{m}^2$ (0.7- $\mu\text{m}$ rule)	41.1 $\mu\text{m}^2$ (0.8- $\mu\text{m}$ rule)
Standby current (per cell)	$10^{-15}$ A	$10^{-12}$ A	$10^{-13}$ A





# Static RAM (SRAM)



Pin(s)	Function
$A_0-A_{10}$	Address
$DQ_0-DQ_7$	Data In/Data Out
$\overline{S}$ (CS)	Chip Select
$\overline{G}$ (OE)	Read Enable
$\overline{W}$ (WE)	Write Enable

[1]

Pini similari cu cei ai EPROM-ului, cu excepția semnalului de scriere

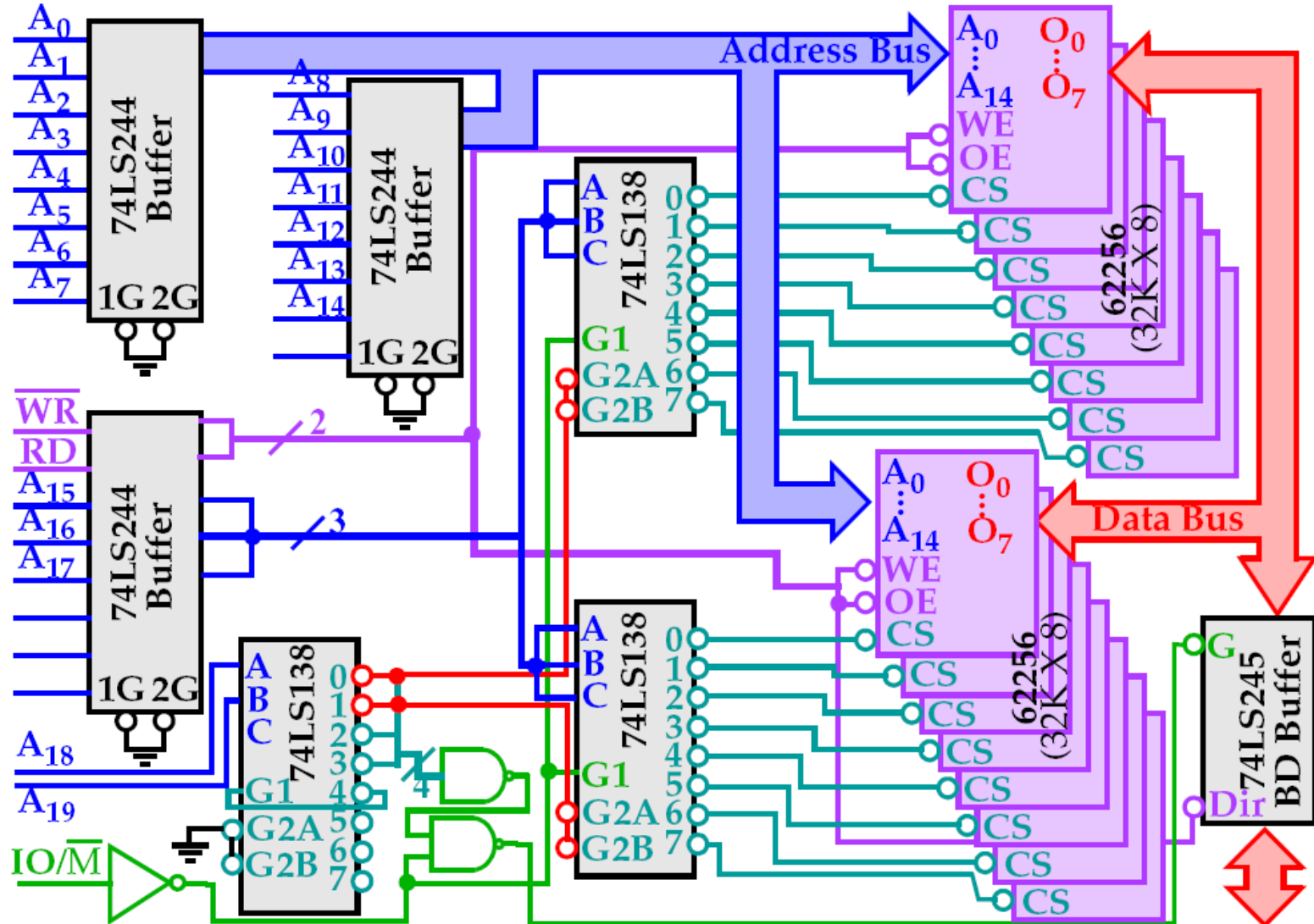
Timp de acces mai rapid

Folosit pentru memorii Cache



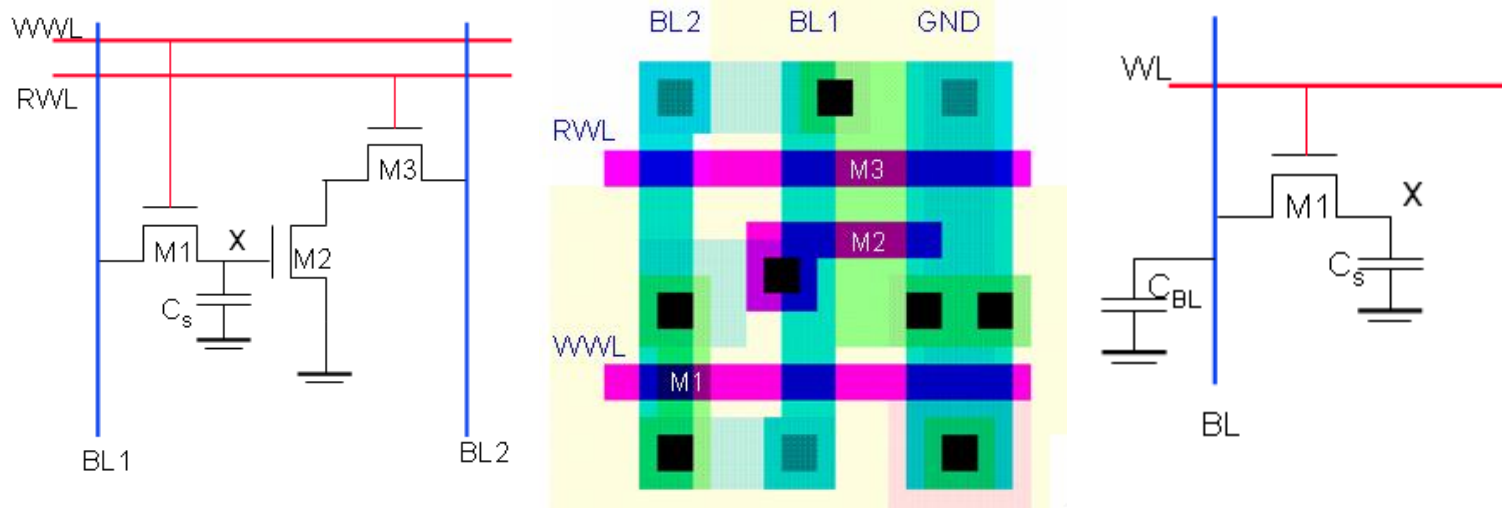
# Static RAM (SRAM)

## 8088 and 80188 (8-bit) RAM Memory Interface





# Dynamic RAM (DRAM)



1

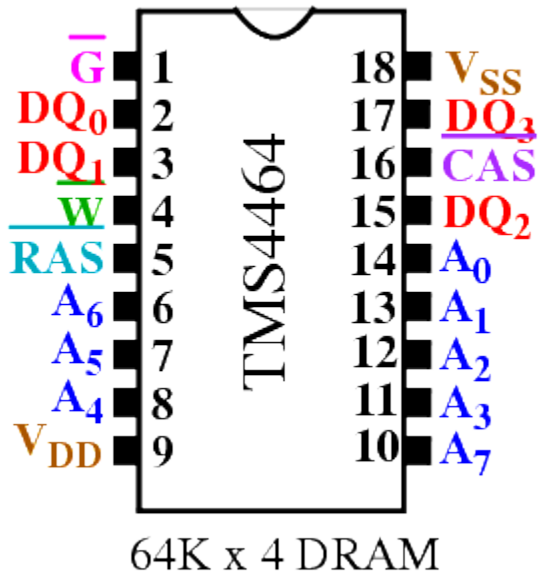
Dimensiune:  $\frac{1}{2}$  din dimensiunea celulei SRAM  $\Rightarrow$  capacitate sporită  $\Rightarrow$  pinii de adresă sunt multiplexați

Refresh: 1 ... 4 ms  $\Rightarrow$  circuit special  $\Rightarrow$  cicluri de citire, scriere, refresh

Tipuri: SDR, DDR, Rambus



# Dynamic RAM (DRAM)



Pin(s)	Function
$A_0-A_7$	Address
$DQ_0-DQ_3$	Data In/Data Out
$\overline{RAS}$	Row Address Strobe
$\overline{CAS}$	Column Address Strobe
$\overline{G}$	Output Enable
$\overline{W}$	Write Enable

[1]

64 K locații adresabile – necesită 16 linii de adresă, dar are numai 8

Adresa randului ( $A_8:A_{15}$ ) este plasată pe pinii de adresă, și memorată în latch-uri interne, folosind RAS (row address strobe)

Adresa coloanei ( $A_0:A_7$ ) este memorată ulterior, folosind CAS (column address strobe)

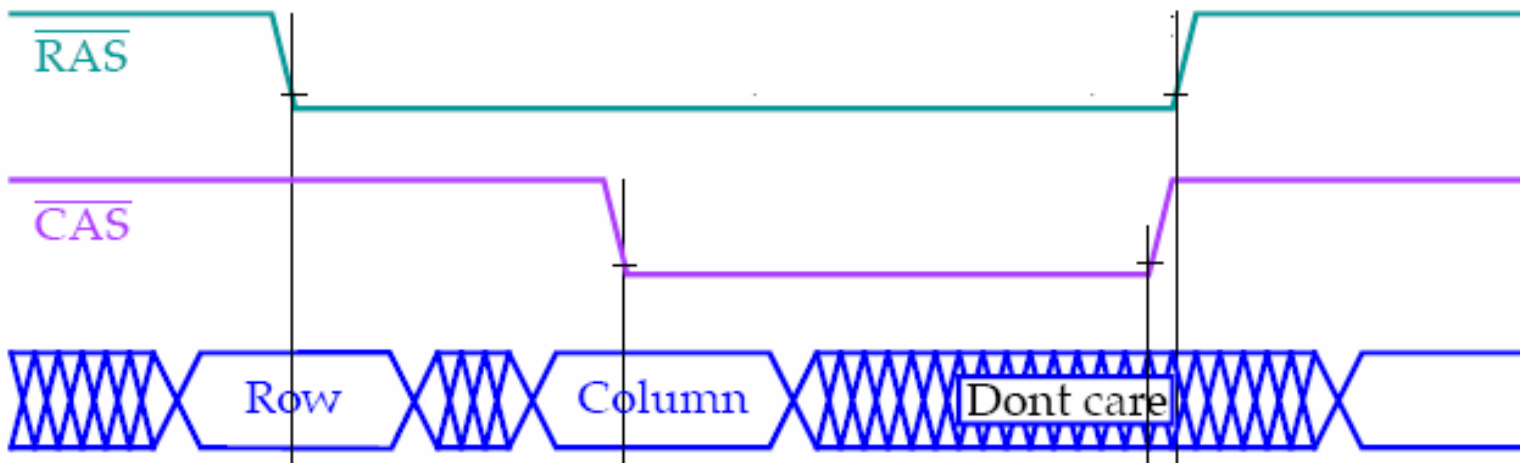


# Dynamic RAM (DRAM)



## DRAMs

TI TMS4464 DRAM (64K X 4) Timing Diagram:



CAS also performs the function of the chip select input.

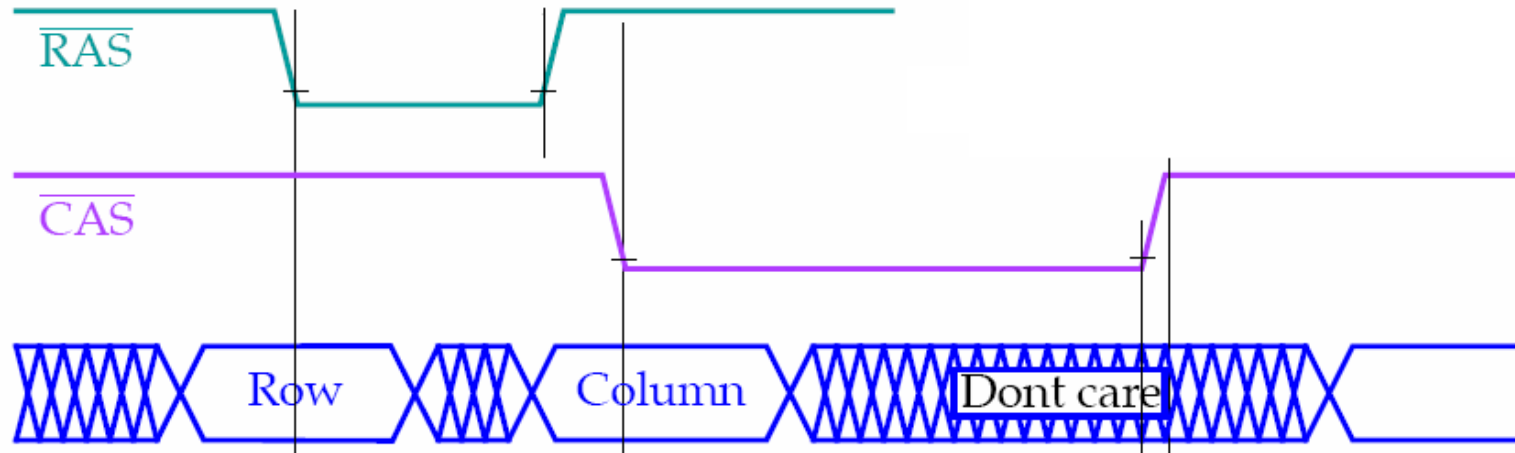
#RAS & #CAS trebuie transmise de un controller DRAM

Controllerul DRAM trebuie sa multiplexeze (în timp) liniile de adresa, ca:

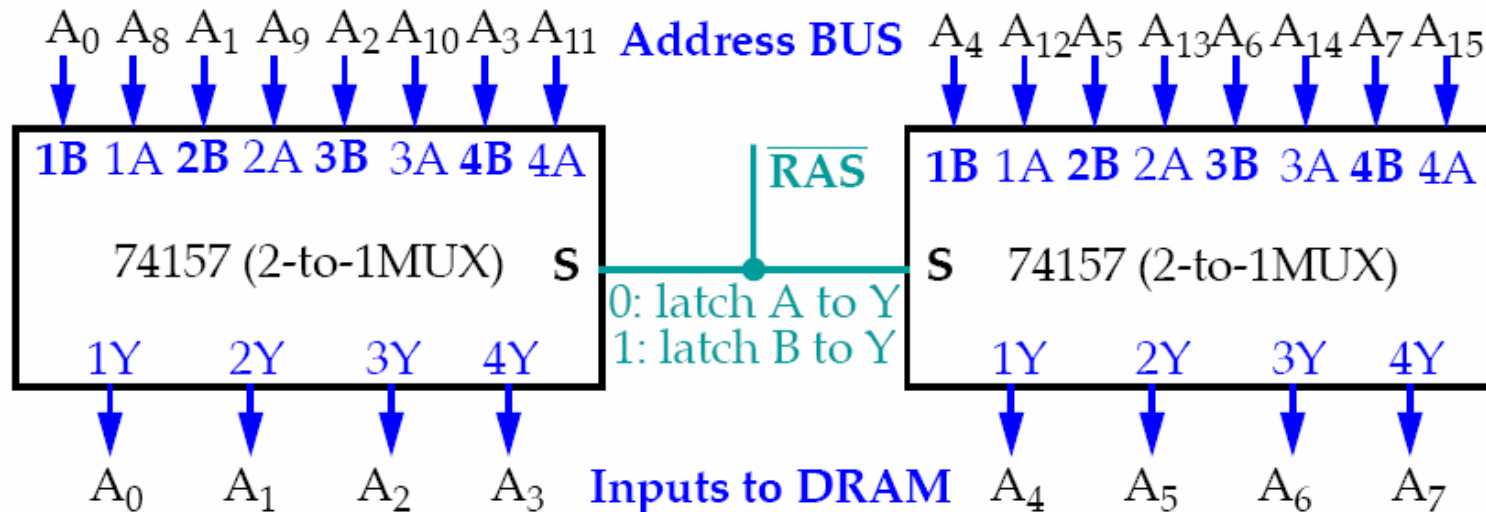
- Adresa randului (ex. A8-15)
- Adresa coloanei (ex. A0-7)



# Dynamic RAM (DRAM)



CAS also performs the function of the chip select input.

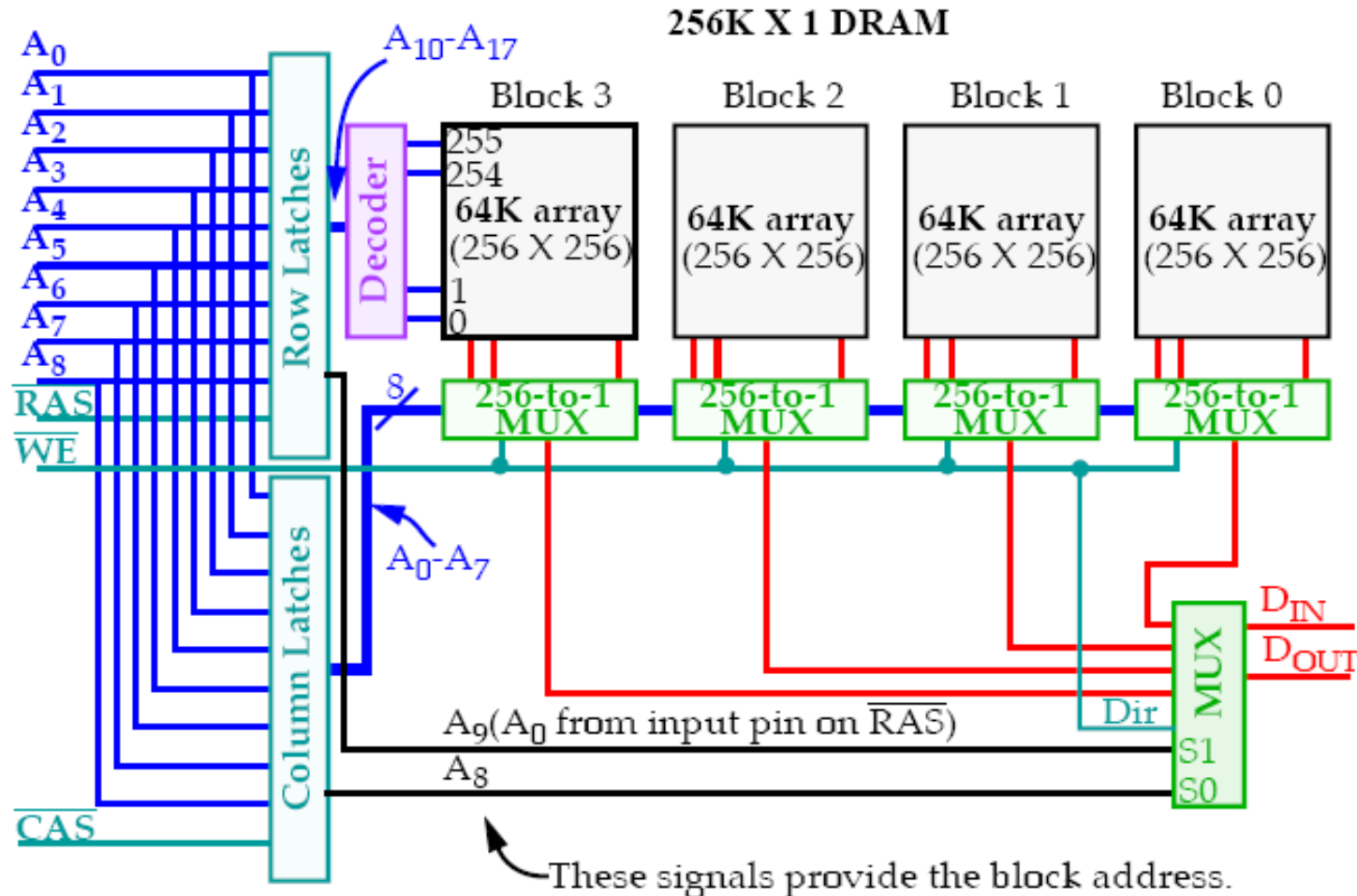




# Dynamic RAM (DRAM)

## 256K X 1 DRAM – structura internă

Dynamic RAM





- **Ciclu special de refresh**

- Se petrece transparent cand sunt folosite alte componente – *transparent refresh sau cycle stealing*.
- Un ciclu ce folosește doar RAS încarcă o adresă a rândului în DRAM
- Condensatorii rândului selectat sunt reîncărcați prin citirea internă a biților și scrierea lor inapoi.

**Exemplu:**

- **256K X 1 DRAM** (256 randuri x 256 coloane x 4 blocuri)
  - $\Rightarrow$  refresh este necesar la fiecare  $15.6\text{ ms}$  ( $4\text{ ms} / 256$ ).
  - Pentru 8086, o citire sau o scriere se petrece la fiecare  $800\text{ns}$  ( $4 \times 200 = 4 \times T_{clk}$ ).
- $\Rightarrow$  **19** citiri/scrieri per refresh ( $15.6\mu\text{s} / 0.8\mu\text{s} = 19.5$ )
- $\Rightarrow$  Ciclurile de citire /scriere iau  $\sim$  **5%** din timpul de refresh





# Dynamic RAM (DRAM): Controller DRAM

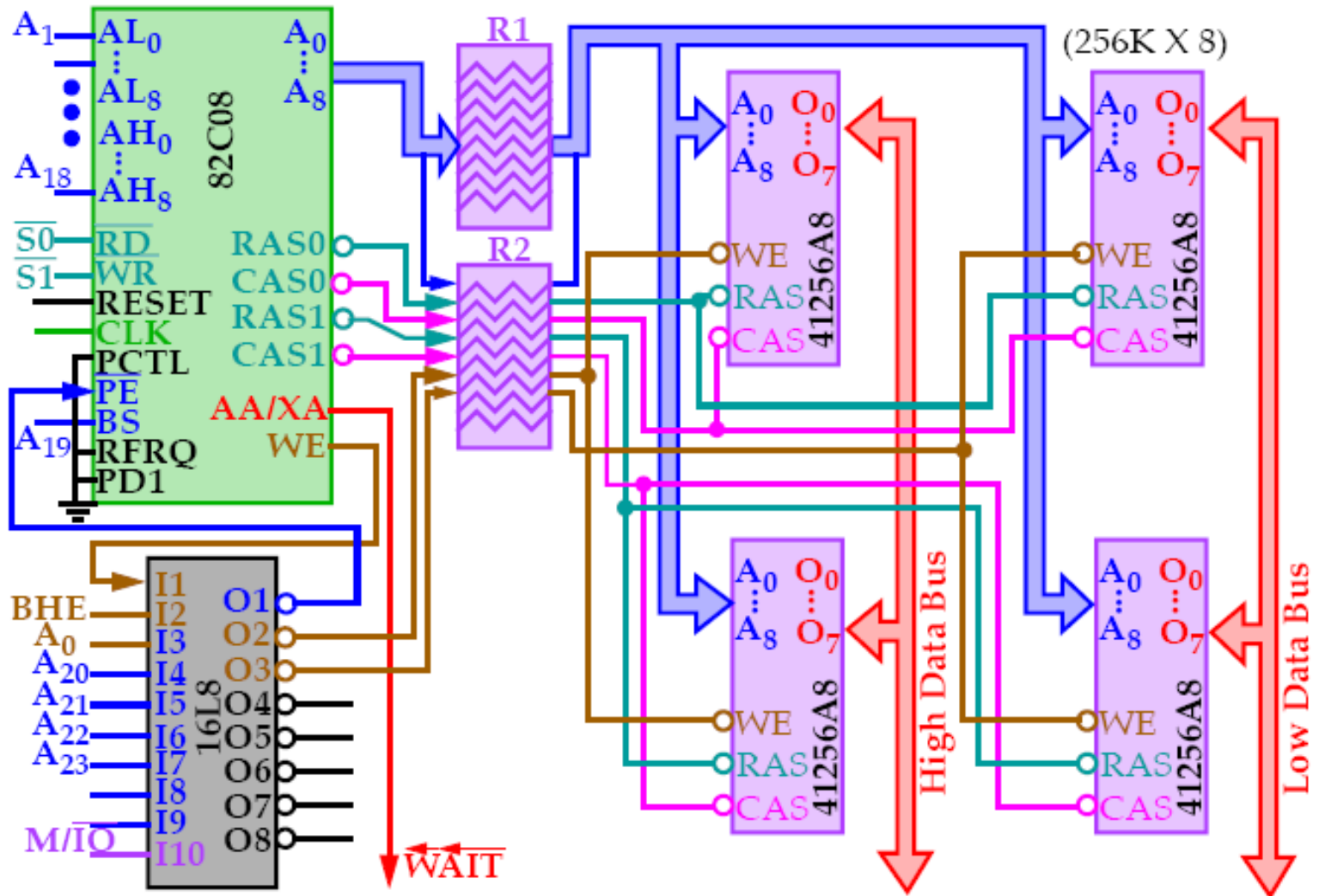


- Multiplexarea adreselor și generarea semnalelor de control pentru DRAM.
- **Intel 82C08**, poate controla doua bancuri de memorie DRAM 256 K X 16 , pentru un total de 1 MB.
- Bitii A1 - A18 (18 bits) sunt conectati la intrarile (AL – coloana) și (AH – linie) a lui 82C08.
- În funcție de adresă, se activează RAS0/CAS0 sau RAS1/CAS1.
- WE , BHE si A0 sunt folosiți pentru determinarea scrierii, și unde anume se va scrie



# RAM – Dynamic (DRAM)

## DRAM Controllers

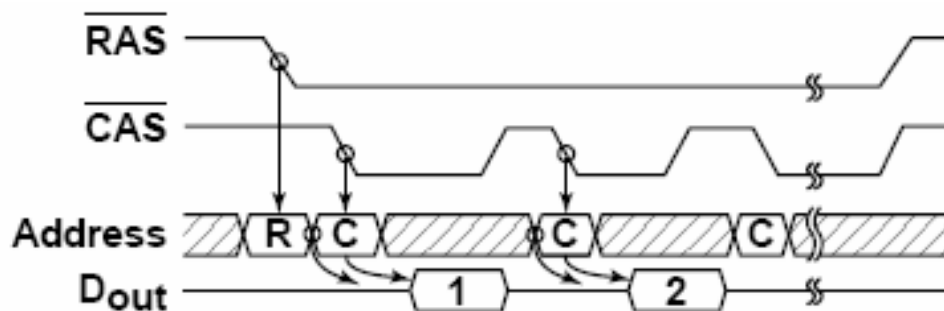




# DRAM – Fast Page Mode (FPM)



- Permite citirea mai multor coloane pentru același rând
  - 1 RAS, multiple CAS
- Variații:
  - Static column – nu necesită pulsarea CAS pentru schimbarea coloanei, ci CAS este menținut activ, adresa coloanei se schimbă și ieșirea o urmează cu o anumită întârziere
  - Nibble mode – adresele consecutive primei coloane sunt generate de un contor intern, la fiecare puls CAS, nemaifiind preluate de pe liniile de adresa.

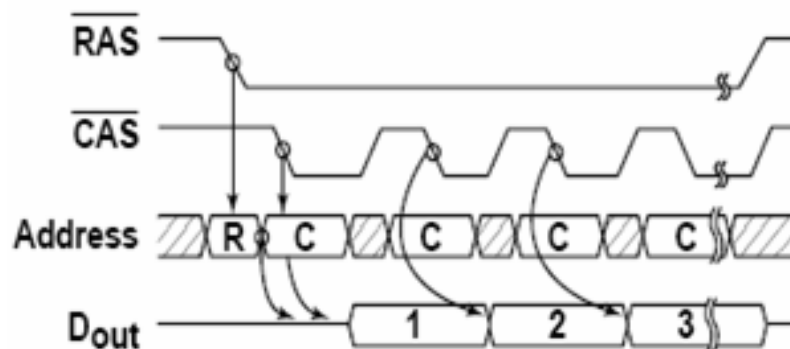




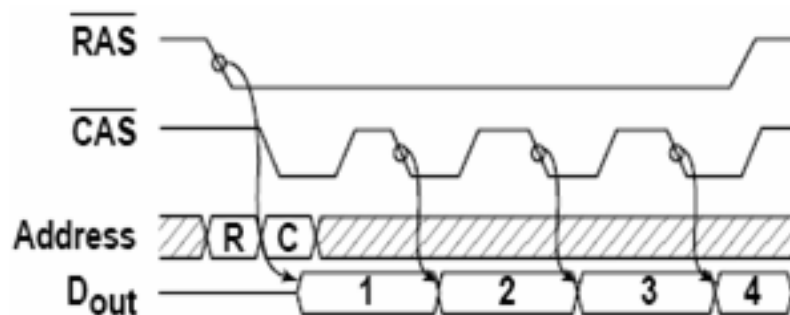
# DRAM – Extended Data Out (EDO)



- Un ciclu de acces poate menține datele din ciclul anterior active



- Burst EDO (BEDO) – adresele coloanei sunt generate intern după primul CAS

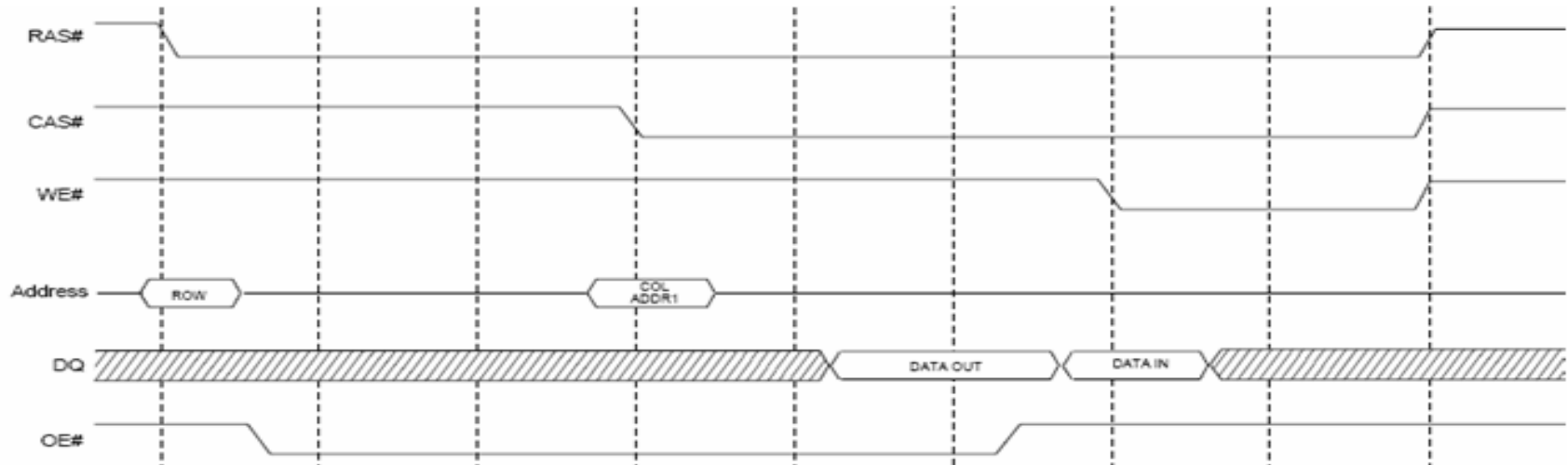




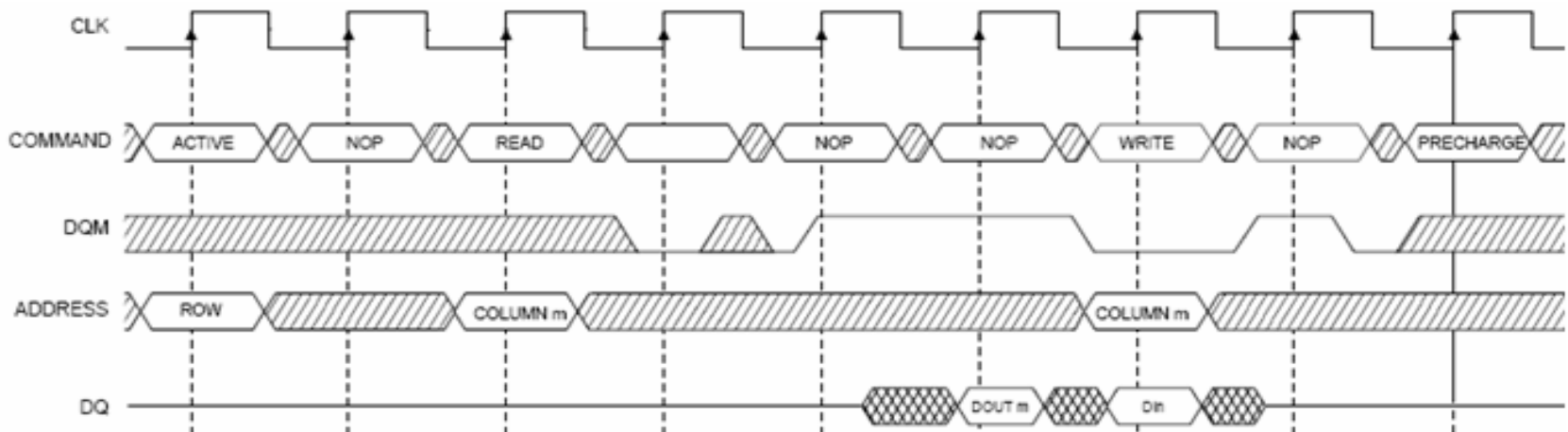
# DRAM asincron / sincron



- Asincron

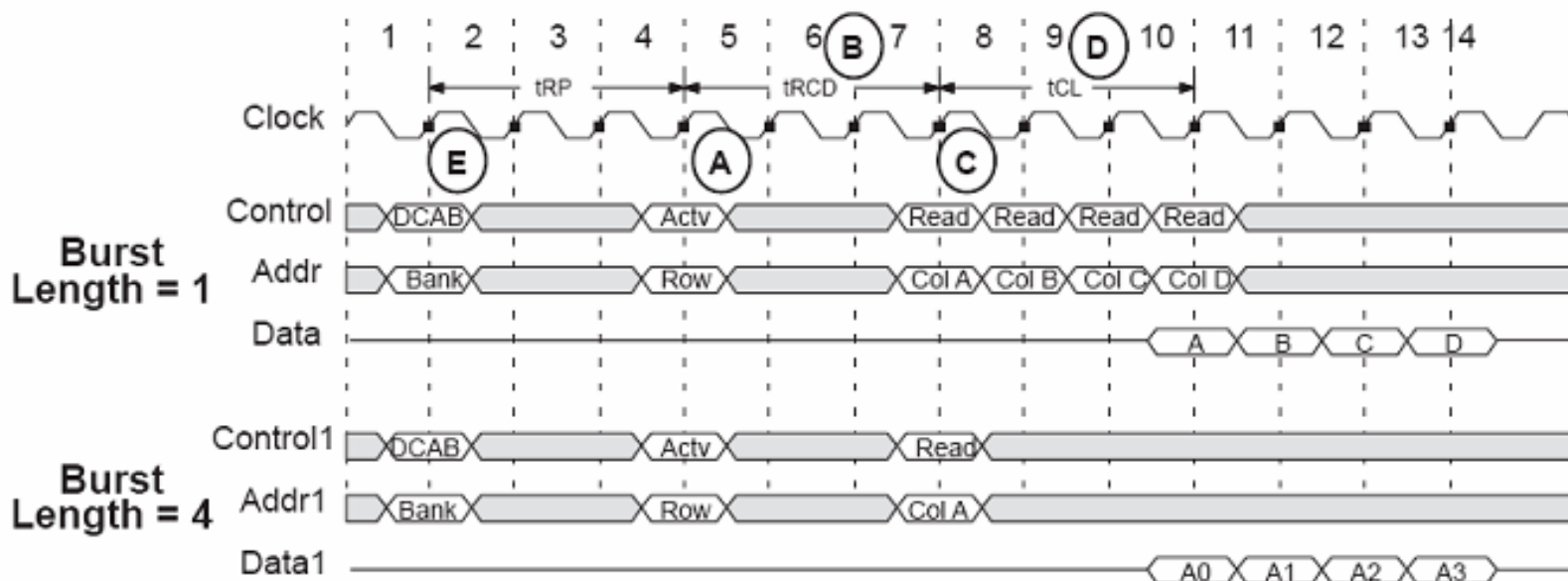


- Sincron





# SDRAM – Synchronous DRAM



## Semnale SDRAM:

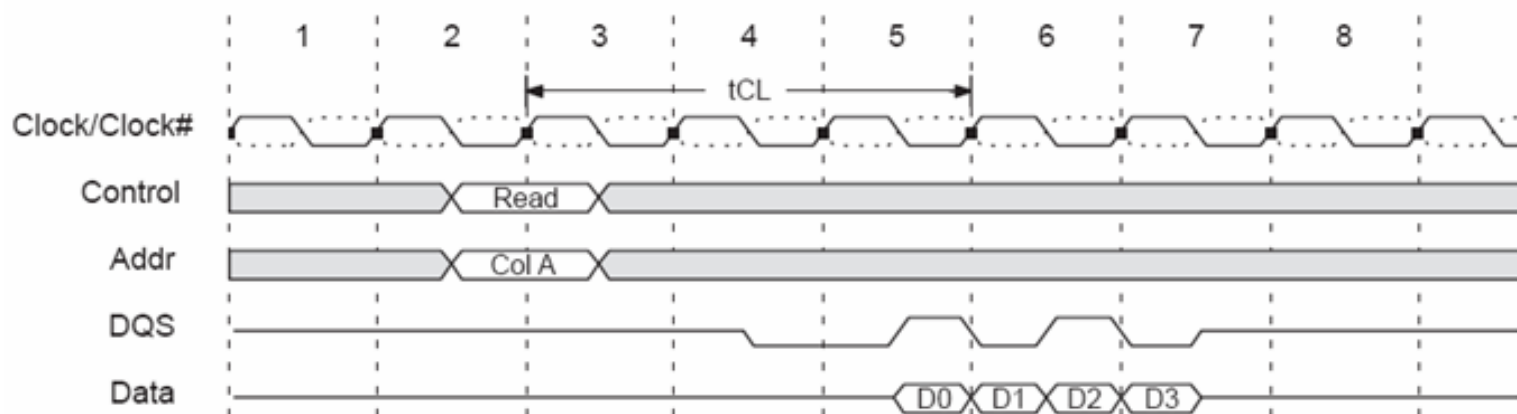
- **CKE Clock Enable.** Daca acest semnal este zero, memoria nu executa nici o operatie.
- **/CS Chip Select.** Când acest semnal este 1, memoria ignoră toate intrările.
- **DQM Data Mask.** Masca pentru octeții din cuvantul de date (1 linie de mască pentru fiecare octet).
- **/RAS Row Address Strobe.** Formează, împreună cu /CAS și /WE, comenzi pentru memorie.
- **/CAS Column Address Strobe.**
- **/WE Write enable.**



# DDRAM – Double Data Rate SDRAM



- Transferul de date se efectuează pe ambele fronturi ale semnalului de ceas





## Referințe



- [https://www.csee.umbc.edu/courses/undergraduate/CMPE310/Fall07/cpatel2/slides/html\\_versions/chap10\\_lect04\\_memory.html](https://www.csee.umbc.edu/courses/undergraduate/CMPE310/Fall07/cpatel2/slides/html_versions/chap10_lect04_memory.html)
- [https://www.csee.umbc.edu/courses/undergraduate/CMPE310/Fall07/cpatel2/slides/html\\_versions/chap10\\_lect06\\_memory3.html](https://www.csee.umbc.edu/courses/undergraduate/CMPE310/Fall07/cpatel2/slides/html_versions/chap10_lect06_memory3.html)