

Latvijas Universitāte  
Datorikas fakultāte

# Atmiņa. Fizikālās realizācijas varianti.

Kurss "ievads digitālajā projektēšanā"  
Lekcija 02.12.2011

Autors: Rinalds Ruskuls

# *Lekcijas saturs*

- Atmiņu īpašības
- Atmiņas, tās iespējamie iedalījumi
- Atmiņas tipu uzbūves veidi

# ***Atmiņu īpašības Nr.1.***

- Datoram ir divu veidu atmiņas:
  - RAM – **R**andom **A**ccess **M**emory:
    - Static RAM – pārsvarā lieto, reģistru failiem, Kešatmiņām
    - Dynamic RAM – operatīvā atmiņa
  - ROM – **R**ead **O**nly **M**emory, lieto nemainīgu programmu ielādei

# Atmiņu īpašības Nr.2.

- Atmiņas, kuras pēc sprieguma noņemšanas pazaudē atmiņas saturu (Volatile)



- Atmiņas, kurās pēc sprieguma noņemšanas, paliek tekošā atmiņas informācija (Non-Volatile)



# Atmiņu iedalījums

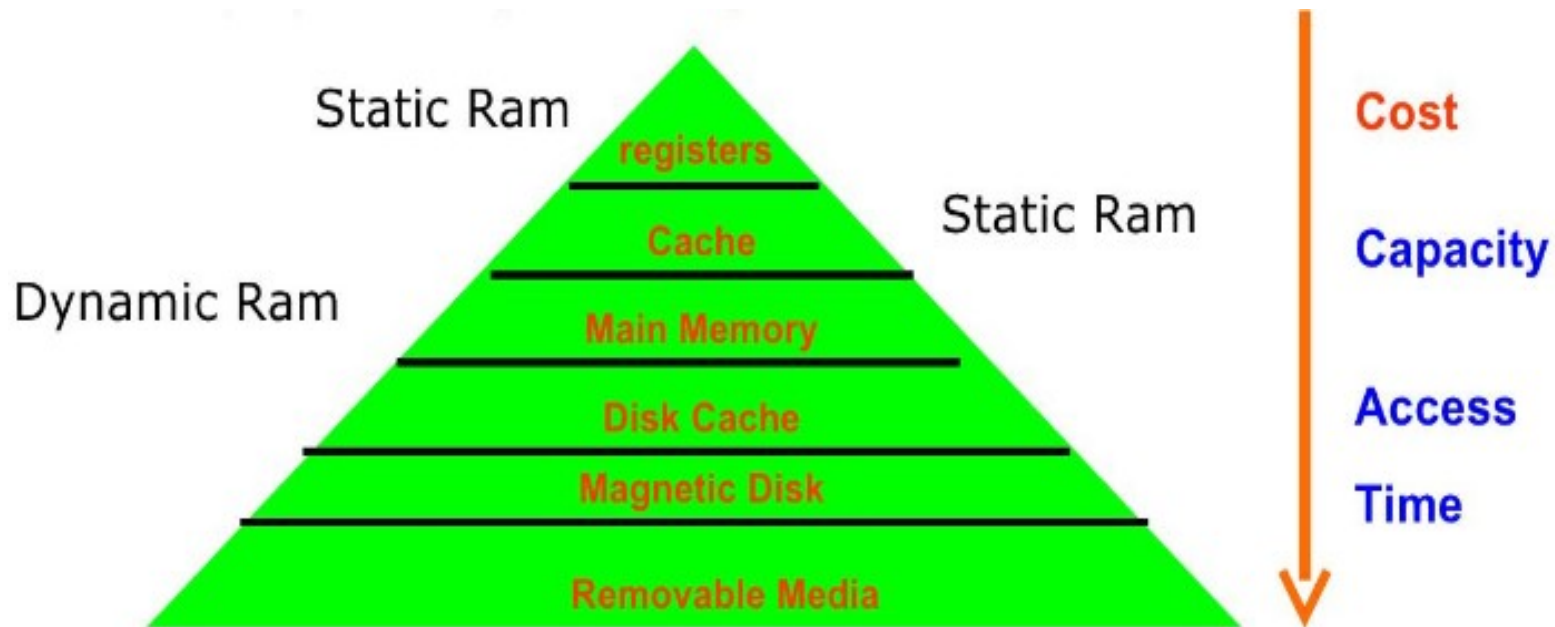
Lasāmās – Rakstāmās atmiņas RAM (saturs <b>nepaliek</b> pēc barošanas sprieguma noņemšanas)		Lasāmās-Rakstāmās Atmiņas (saturs <b>paliek</b> pēc barošanas sprieguma noņemšanas)	Lasāmatmiņas ROM
Brīvpiekļuves	Ne-Brīvpiekļuves	EPROM	Ar masku programmējamās  M-PROM
SRAM	FIFO	EEPROM (E <sup>2</sup> PROM)	
DRAM	LIFO	FLASH	
PSRAM			

Atmiņu salīdzināšanas kritēriji:

- Atmiņas blīvums (bitu skaits uz laukuma vienību)
- Piekļuves laiks (rakstīšanas, lasīšanas laiks)
- Jaudas patēriņš

# Atmiņas hierarhija

- Atmiņas izvēli raksturo šādi kritēriji – cik daudz, cik ātri, cik tas izmaksā?
- Jāatrod līdzsvars starp pieejamām iespējām un vēlmēm
- Dažādas tehnoloģijas ir izveidotas, lai implementētu atmiņas

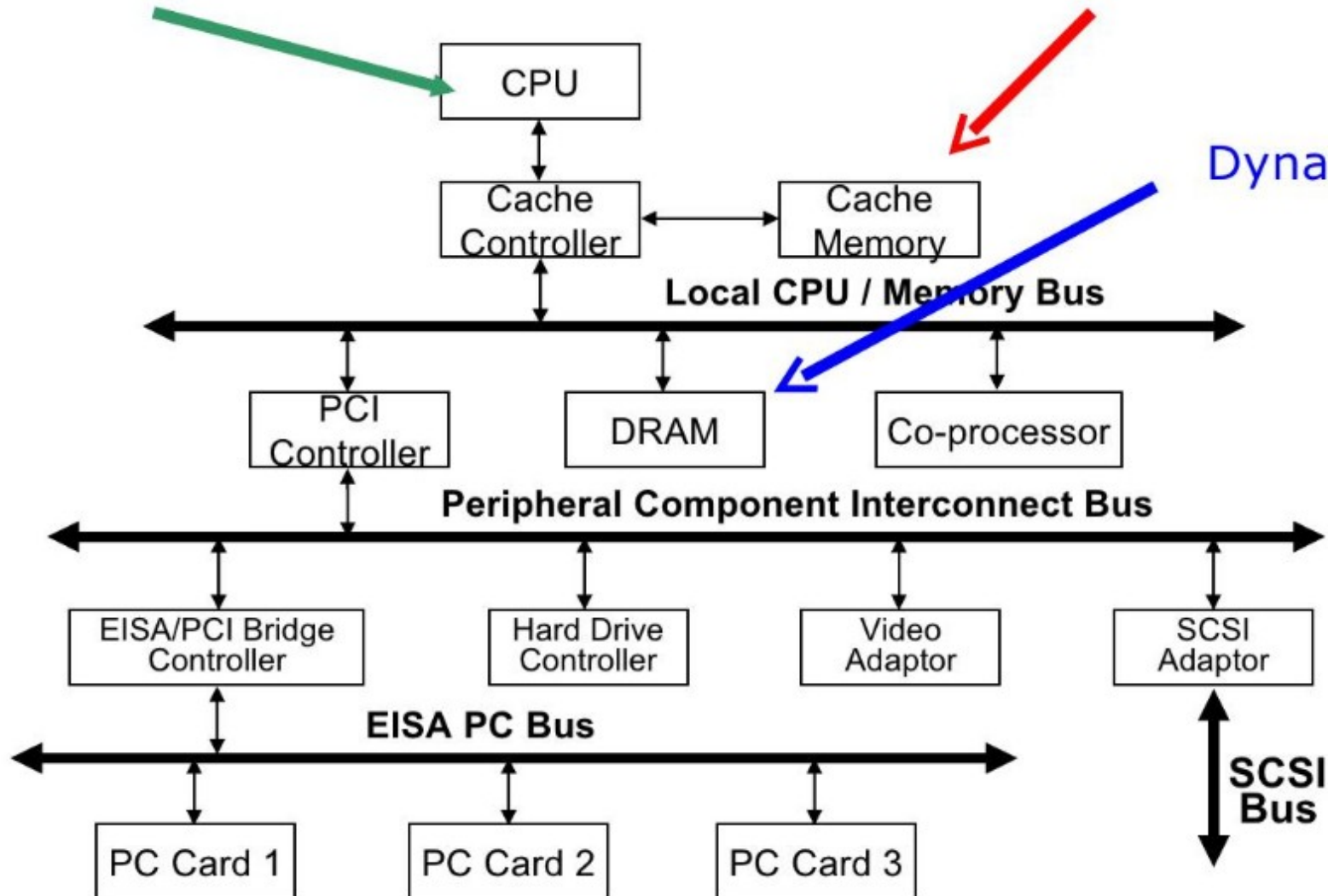


# Memory

**Registers**

**Static RAM**

**Dynamic RAM**



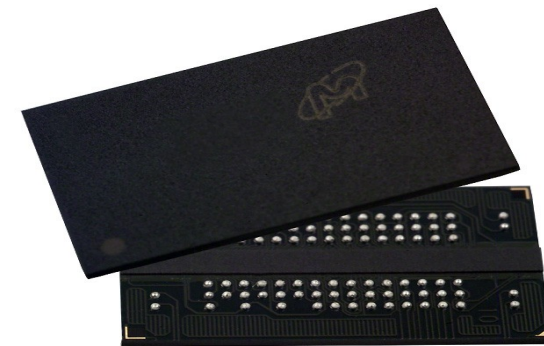
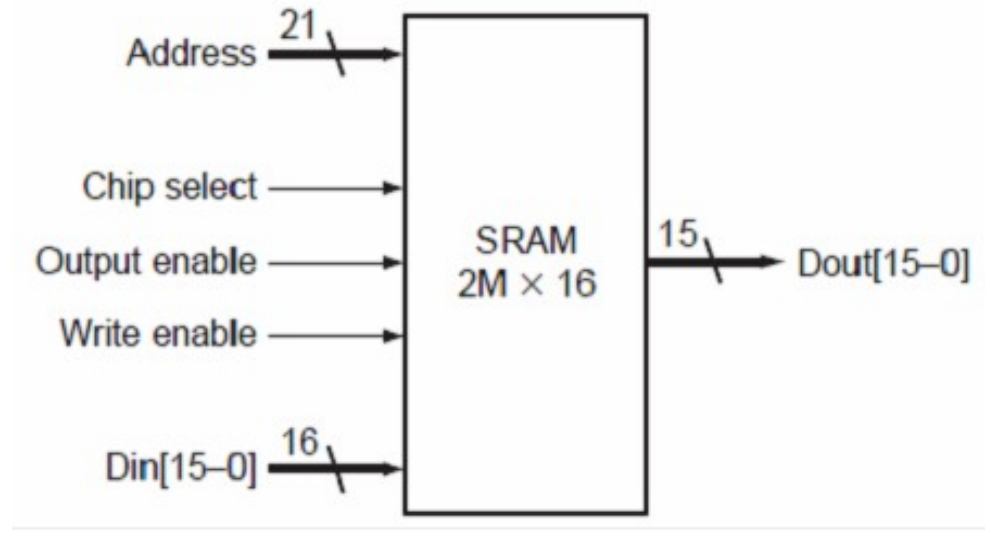
# *Random Access Memory*

- Jebkuram atmiņas datu apgabalam piekļūst vienādā laikā
- RAM tipi:
  - Static RAM (SRAM) – ātrs, dārgs
  - Dynamic RAM (DRAM) – lēns, lēts
- RAM piekļūst:
  - Izmantojot adreses un datu līnijas
  - Kontrolējot piekļuves signālus (R/W, chip select)



# SRAM – Static Random Access Memory

- Dati tiek glabāti statiski, piemēram, D-trigeros
- Atmiņas mikroshēmu raksturo
  - Adrešu skaits
  - Bitu skaits vienai adresei (Din)



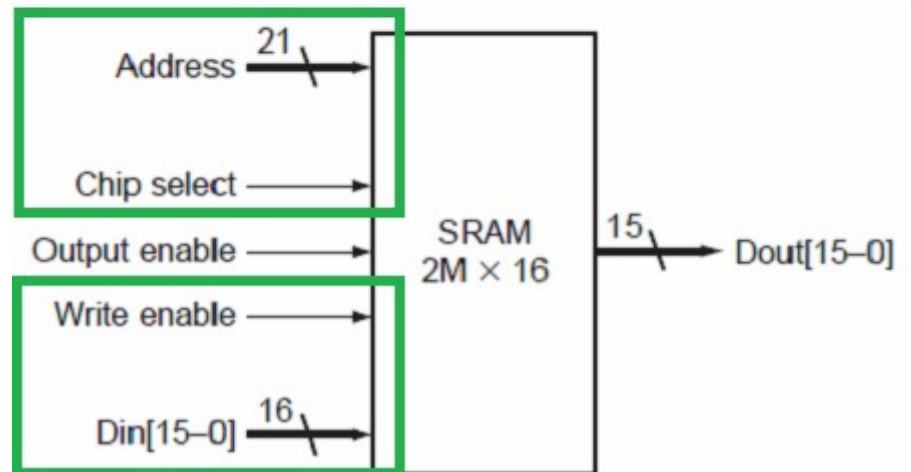
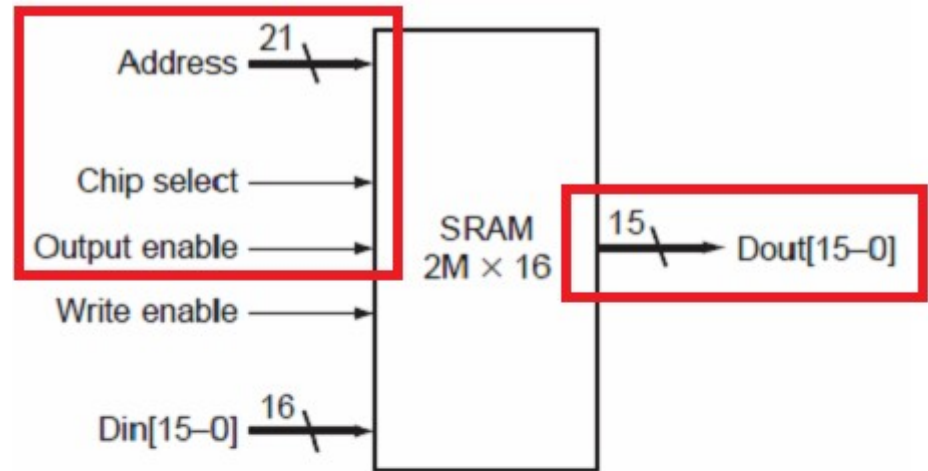
# SRAM vadība

- Lasīšanas režīms:

- Address
- Chip Select
- Output Enable

- Rakstīšanas režīms:

- Address
- Chip Select
- Write Enable



# SRAM uzbūve - vienkāršota

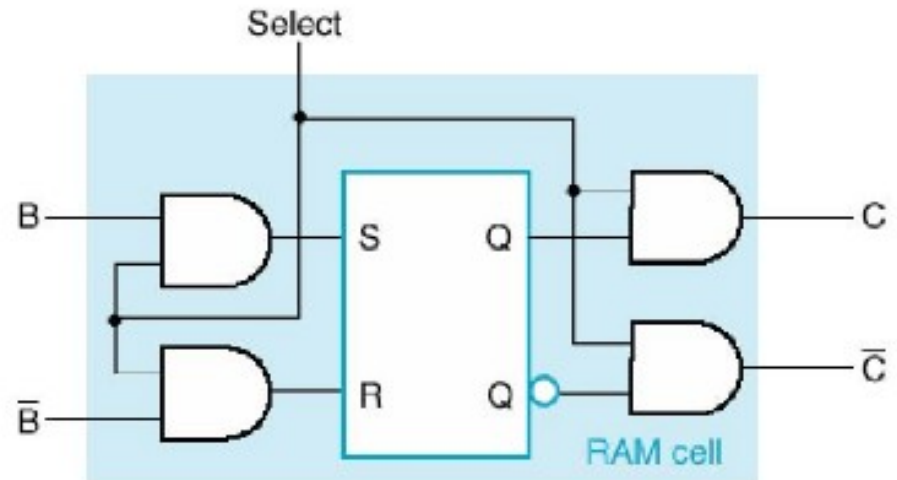
- Viena bita atmiņas izveidei izmantosim:
  - RS trigeru
  - kontroles izvadus

Kad kontroles izvads Select ir '0'

Atmiņas elements saglabā iepriekšējo vērtību, bet nepadod uz izeju

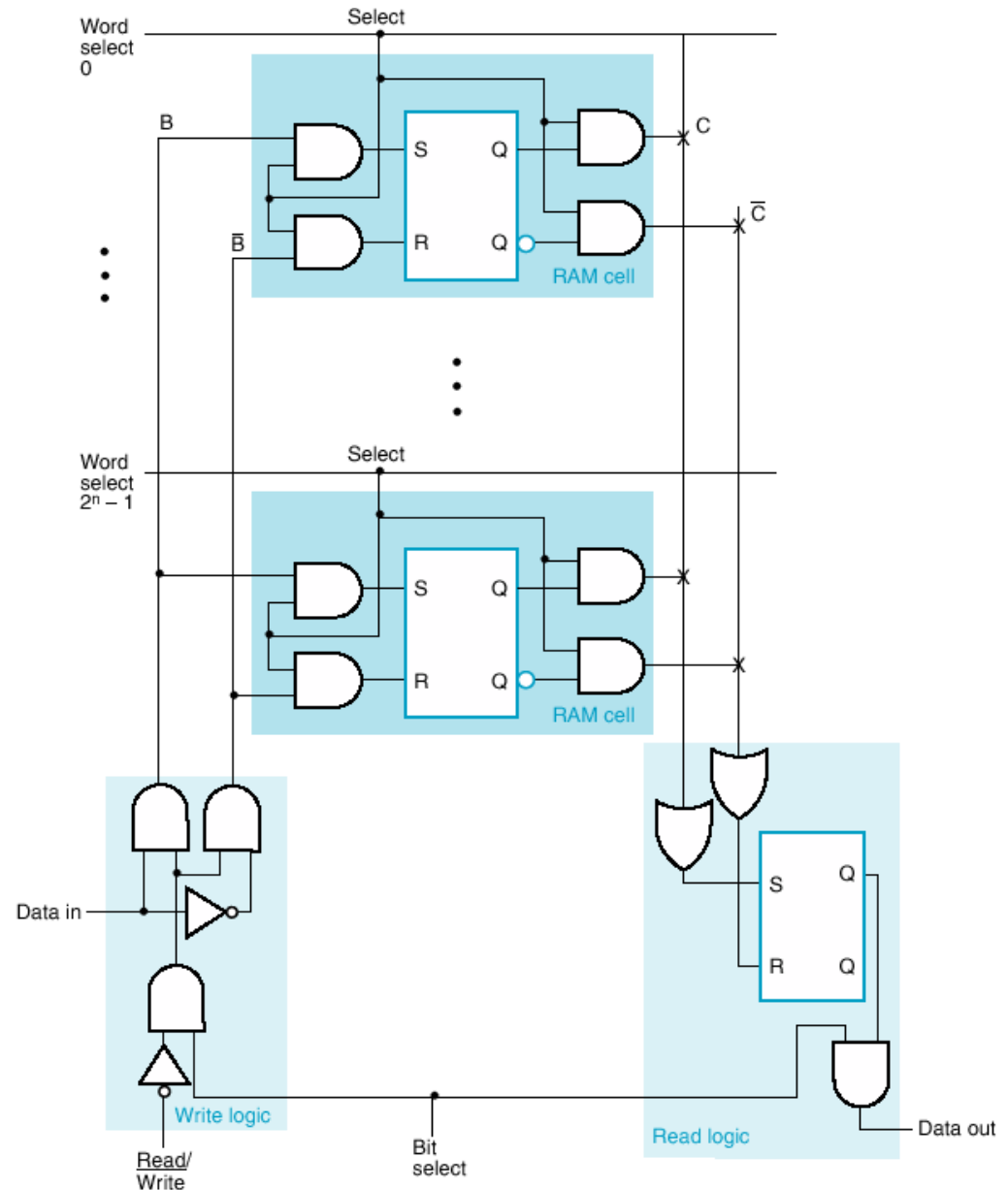
Kad kontroles izvads Select ir '1'

Atmiņas elements veic operāciju, kādu definē ieejas signāli



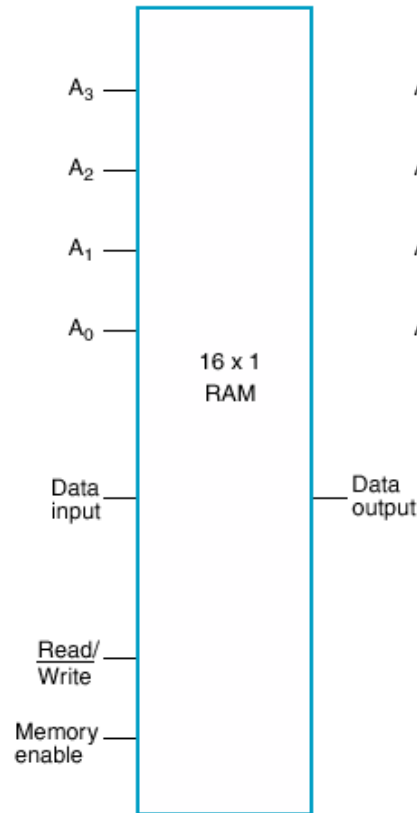
# Viendimensionāla atmiņa

- Ar **Word select** ir iespējams adresēt vienu atmiņas bitu
- **Read/Write** nosaka operācijas tipu
- Data in – ierakstāmie dati
- Data out – izejas dati

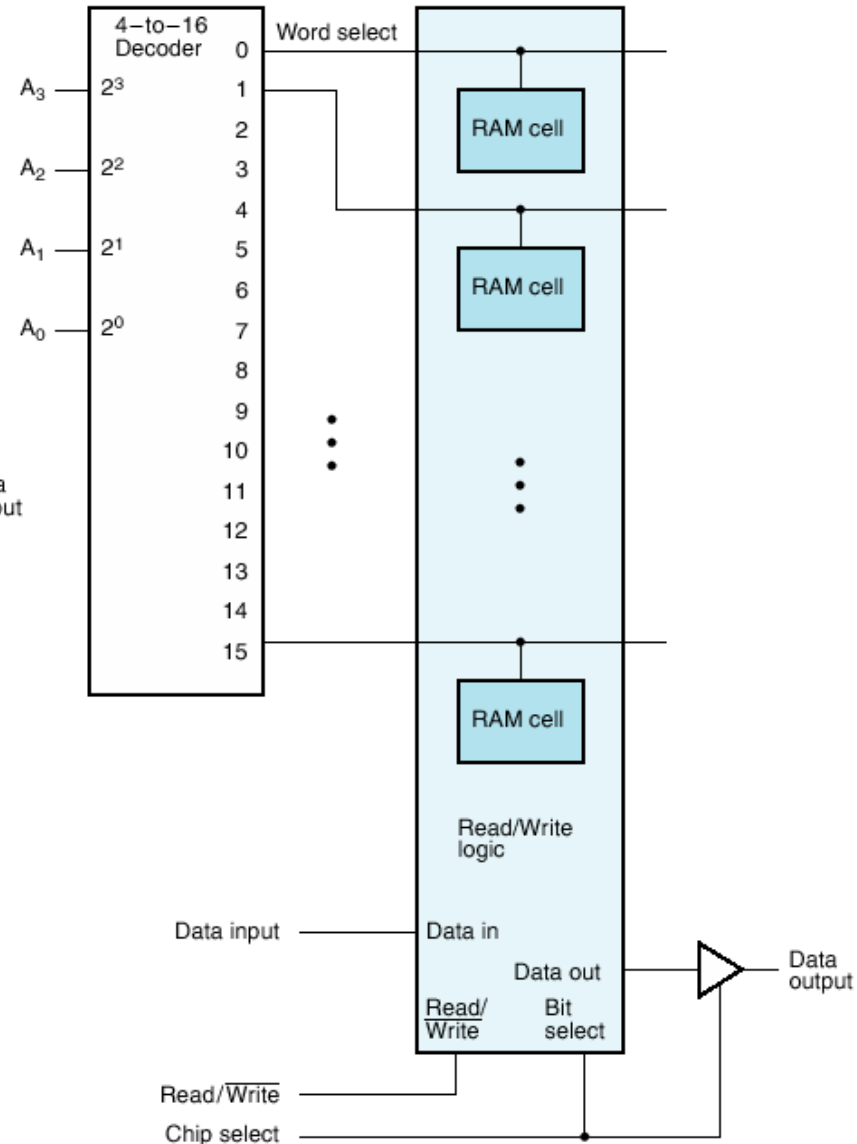


# 16x1 RAM uzbūve

- 4 bitu kopne ir nepieciešama lai piekļūtu 16 atmiņas vērtībām
- Nepieciešams izveidot dekoderi, kas spēj adresēt 16 bitu atmiņu (4 → 16)



(a) Symbol



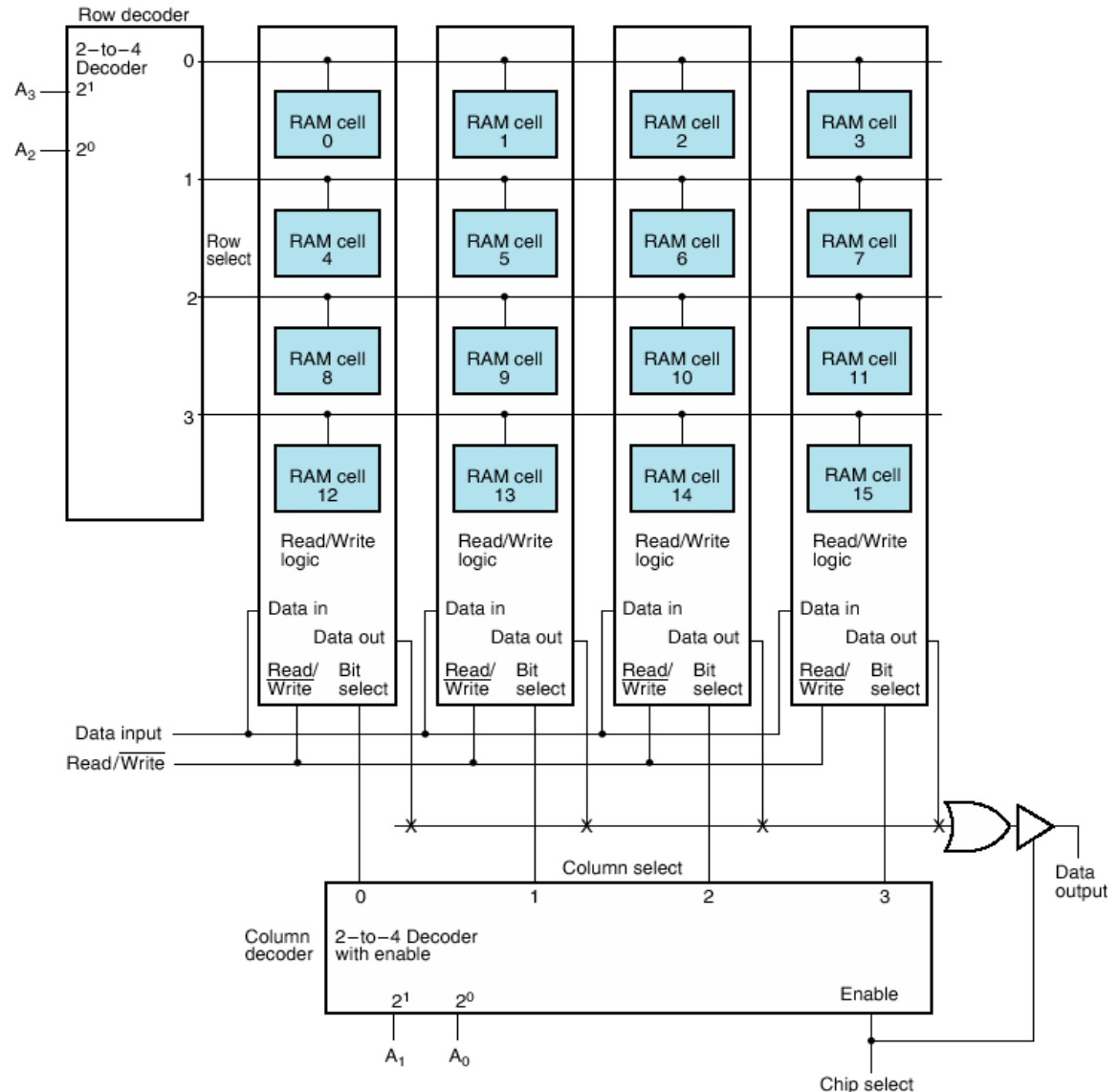
# *RAM uzbūve*

- Praksē SRAM atmiņas sastāv no tūkstošiem vārdu
  - Pieaugot atmiņas daudzumam pieaug arī dekodera izmērs

Vai iepriekšējā pieeja būs pielietojama šādam risinājumam?

# 16x1 SRAM -> 4x4 SRAM

- Viena dekodera vietā izmantot divus (row, column)
- Adrese tiek sadalīta
- Lietotājam nekā nemainās



Verilog

# 16x1 SRAM -> 4x4 SRAM

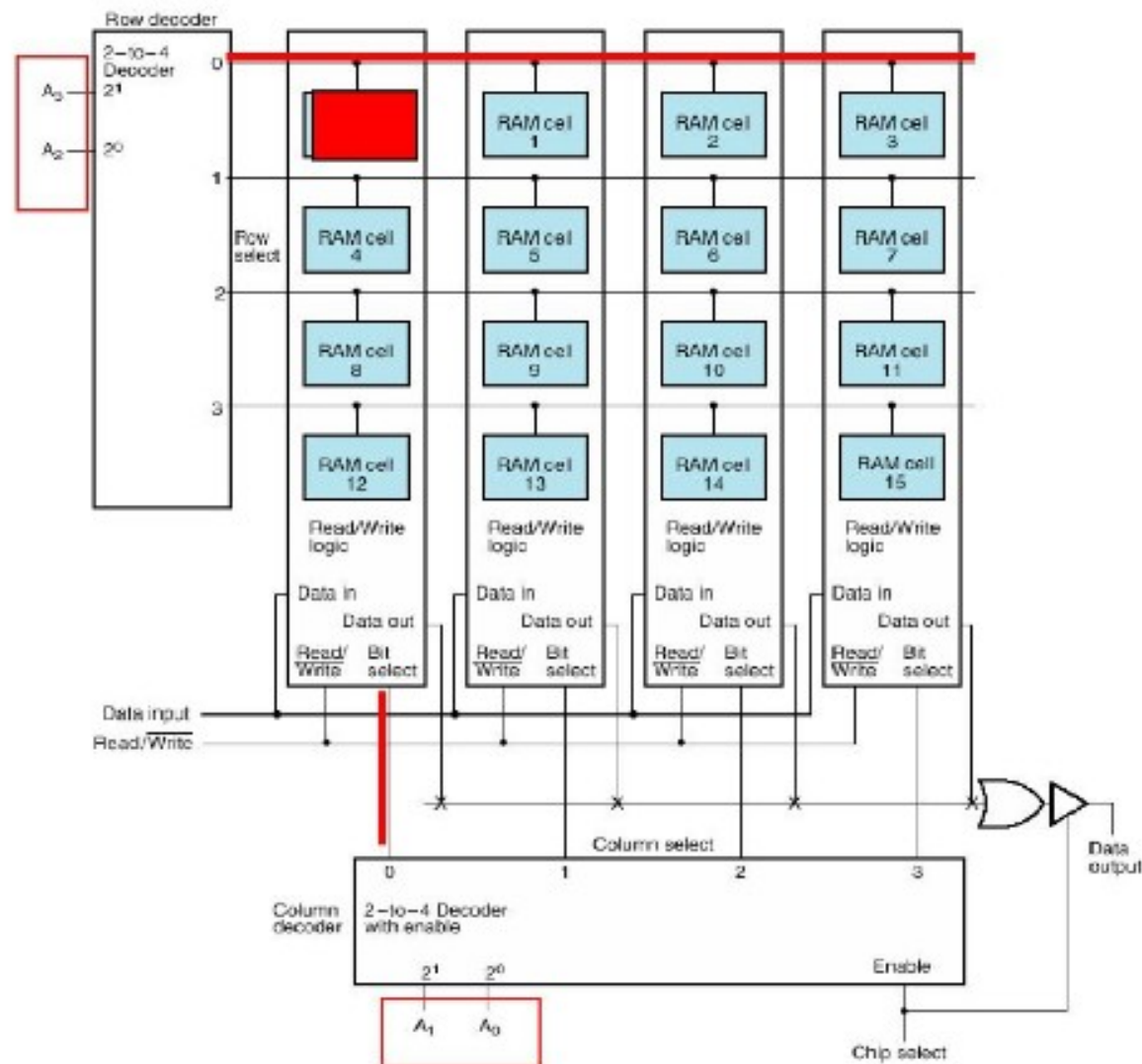
- Izvēloties

$A_0 A_1 A_2 A_3$  adreses

vērtību "0000", tiks

izvēlēts nultais

atmiņas bits

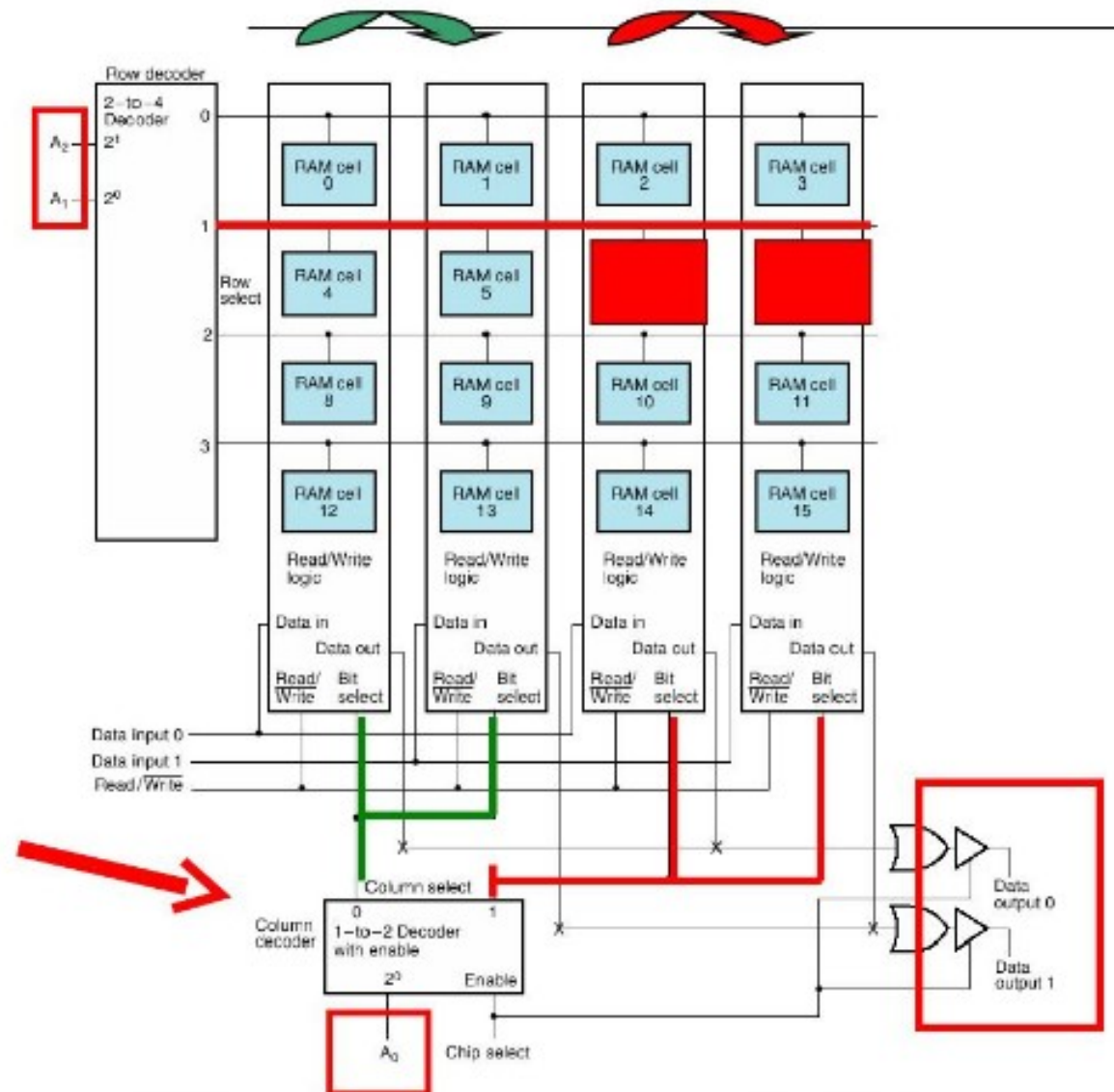


Verilog



# SRAM no 16x1 uz 8x2

- Nelielas izmaiņas loģikā
- Vienā piegājienu var nolasīt divus bitus



Verilog

# ***Komplicētāka SRAM uzbūve***

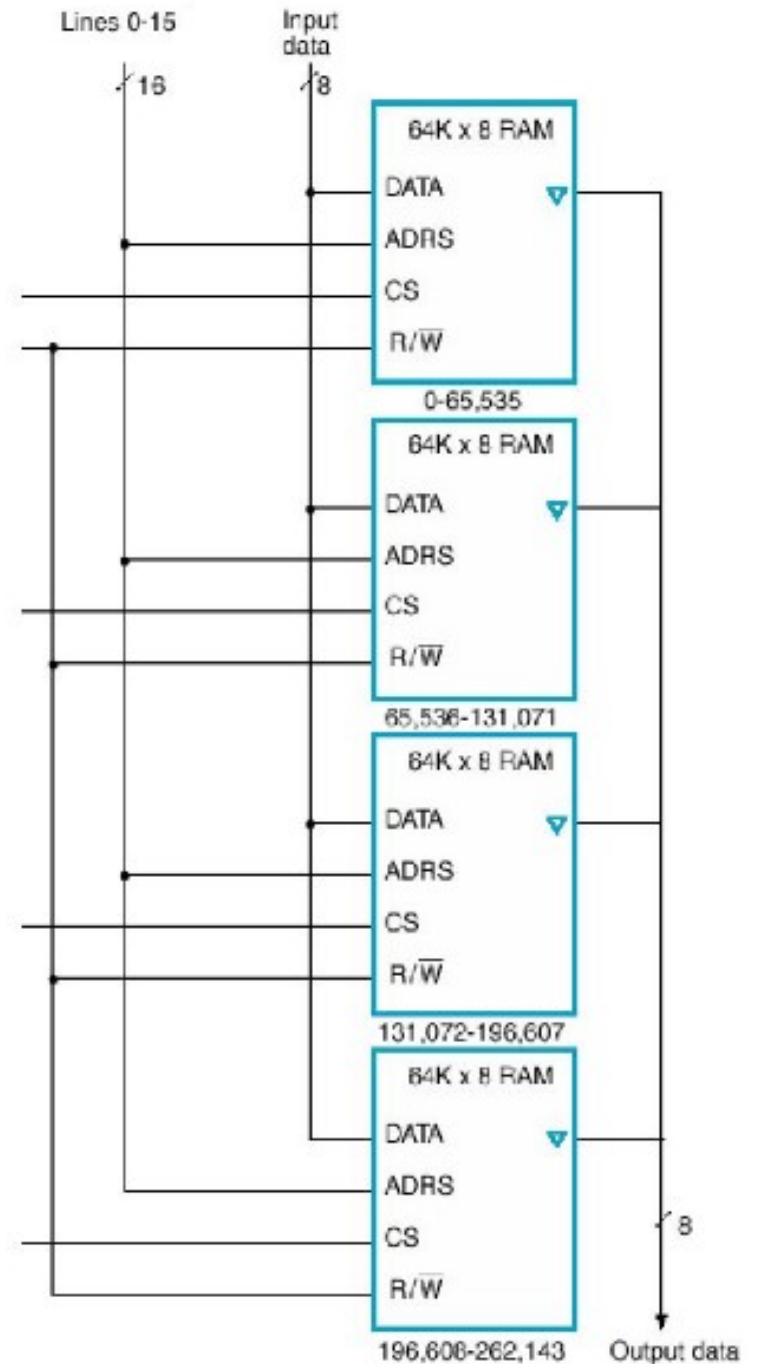
- Nepieciešams izveidot  $32k \times 8 = 256k$ 
  - Adreses līnijas platums – 15 biti
- Rindas dekoderim būtu jāspēj atkodēt 15 bitus uz 32'768 izejas pozīcijām
  - Vai tas ir reāli?

# ***Komplicētāka SRAM uzbūve***

- Projektējamai atmiņai ir jāsaturs  $32k \times 8 = 256k$  bitu
- Jāizlīdzina rindas un kolonas dekodējamo pozīciju skaits  
(Jāizvelk kvadrātsakne no  $256k = 512$ )
  - Rindu dekoderim ir nepieciešams 9bitu dekoders uz 512 pozīcijām
  - Kolonu dekoderim  $512/8 = 64$ , kas kopsummā dod 6 bitu dekoders uz 64 pozīcijām
- Nepieciešamā loģika dekoderu implementācijai  $\sim 608$ , kas no iepriekšējās pieejas atšķiras par 50 reizēm!

# Vēl lielāks SRAM!?

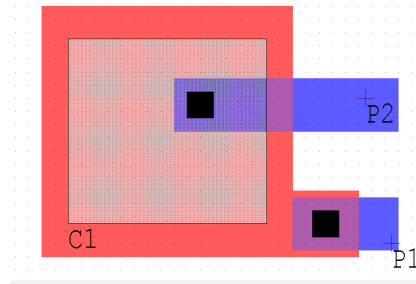
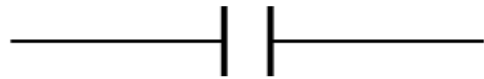
- Izmantot gatavus SRAM blokus
  - 64k x 8
- Saslēgt paralēli adreses, datu, r/w signālus
- Atstāt katram SRAM modulim savu *chip select*



# ***SRAM veiktspēja***

# Dynamic RAM

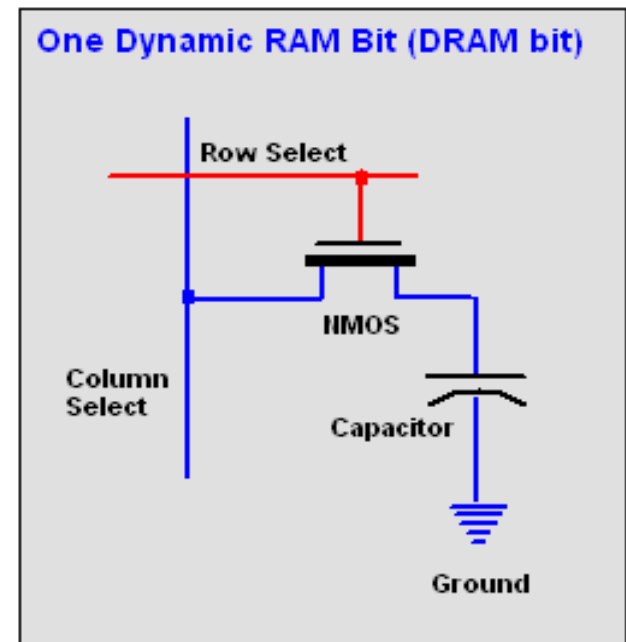
- Viena bita glabāšanai izmanto kondensatoru



Kondensators spēj saglabāt lādiņu

Tranzistors ir kā slēdzis, kas spēj:

- Uzlādēt / izlādēt
- atvienoties no kondensatora
- lai dati nepazustu periodiski jāveic kondensatora pārlādēšana



# *Kas tad tas?*



Pareizā atbilde – šeit pirmo reizi izmantota atmiņa, kas ir bāzēta uz kondensatora lādiņa

# ***DRAM īpašības***

- Destruktīvā lasīšana
  - Nolasot datus, dati pazūd
  - Jāatjauno pēc nolasīšanas
- Dati periodiski jāatjauno
  - Kondensatoriem ir noplūdes strāvas, kas laika gaitā izlādē kondensatoru
- Liela atmiņas ietilpība uz laukuma vienību



# ***DRAM rakstīšana un lasīšana***

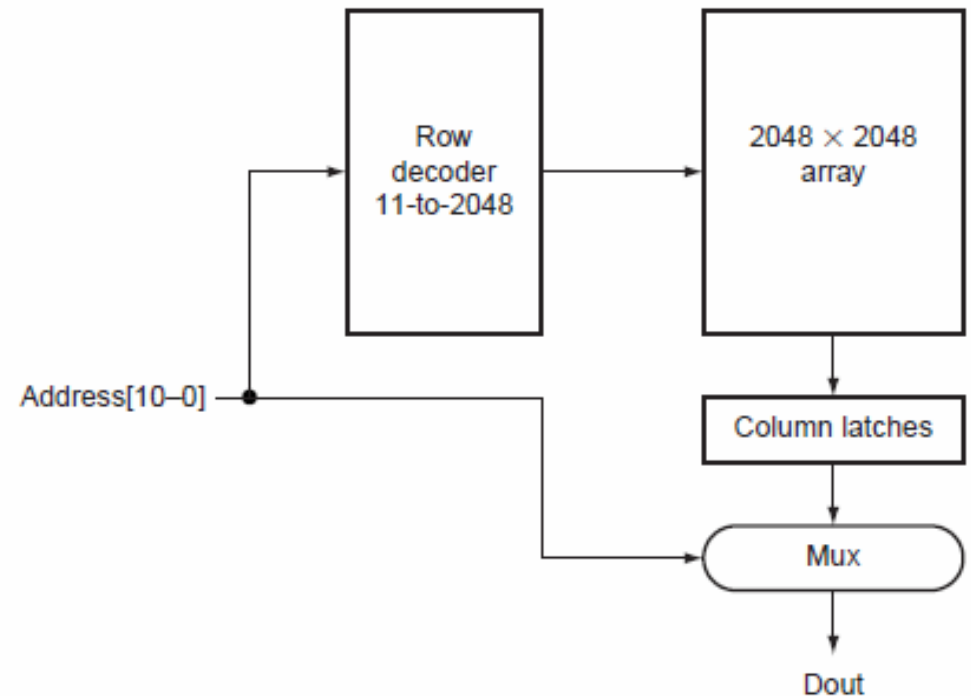
- Rakstīšana:
  - Uz bit line padod datu signālu (0 vai 1)
  - Uz word line padodot 1, kondensatoru vai nu uzlāde (1),vai izlāde (0)
- Lasīšana
  - Uz bit line padod pusi no 1 atbilstoša sprieguma
  - Uz word line padodot 1, uz bit line kondensatora del rodas sprieguma leciens 1 vai 0 virziena, kas tiek detektets, izmantojot jutigu pastiprinataju

# ***DRAM un 2-līmeņu dekodēšana***

- Divas secīgas fāzes
  - Rindas izvēle
    - Aktivizējam atbilstošo ***word line***
    - Visas kolonnas savu rezultātu saglabā trigeros
  - Kolonnas izvēle
    - Izvēlamies datus no kolonnu trigeriem
    - Regenerēšanas gadījumā rakstam trigeru saturu atpakaļ attiecīgajās kolonnās
- Lai ietaupītu uz vadiem, rindu un kolonnu adresācijai izmanto tos pašus vadus un dažus papildsignālus
  - Row Access Strobe (RAS)
  - Column Access Strobe (CAS)

# 4Mx1 DRAM atmiņas implementācija

- 11 biti, lai izvēlētos 1 no 2048 rindām (RAS)
- Rindas saturu saglabā 2048 viena bita trigeros
- 11 biti, lai izveletos 1 no 2048 trigeriem (CAS)



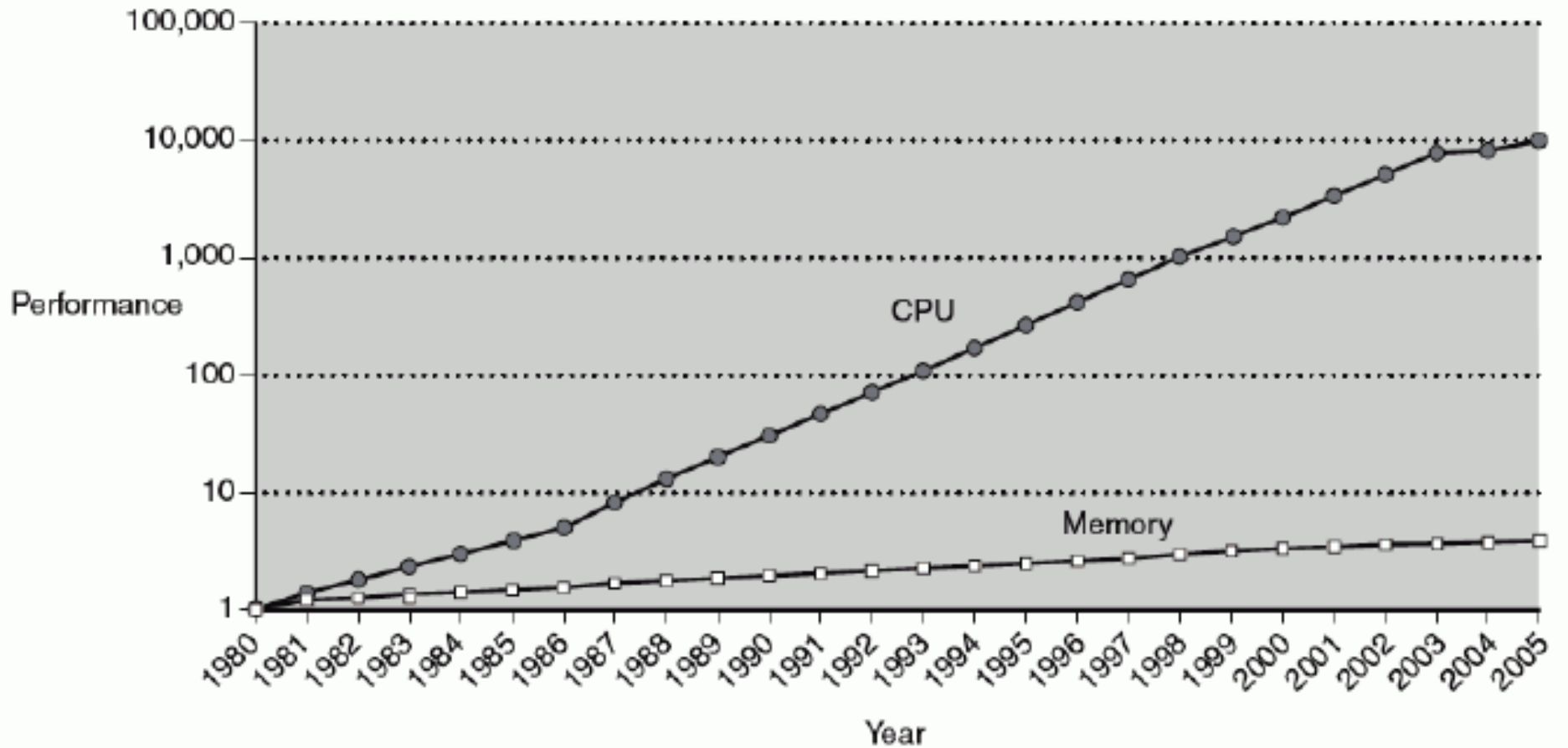
# ***DRAM attīstība laika gaitā***

- Ir bijuši vairāki uzlabojumi šai tehnoloģijai
  - Izveidota DRAM atmiņa ar sinhro-impulsu (SDRAM)
  - Datu lasīšana/rakstīšana notiek uz abām sinhro-impulsa frontēm (DDR SDRAM)
  - Datu lasīšana/rakstīšana tiek dalīta vēl sīkāk, sinhro-impulsa fāzēs – 90, 180, 270, 360
  -

# ***DRAM attīstība laika gaitā II***

- Procesora veikspēja pieaug par 60% gadā jeb dubultojas 1,5 gados (Moore's Law)
- DRAM veikspēja pieaug par 9% gadā jeb dubultojas 10 gados
- Līdz ar to starpība starp procesora un DRAM veiktspeju pieaug par 50% gadā (Joy's Law)

# DRAM attīstība laika gaitā II



***Paldies par uzmanību!  
Jautājumi?***