ΗΥ220
Εργαστήριο Ψηφιακών Κυκλωμάτων

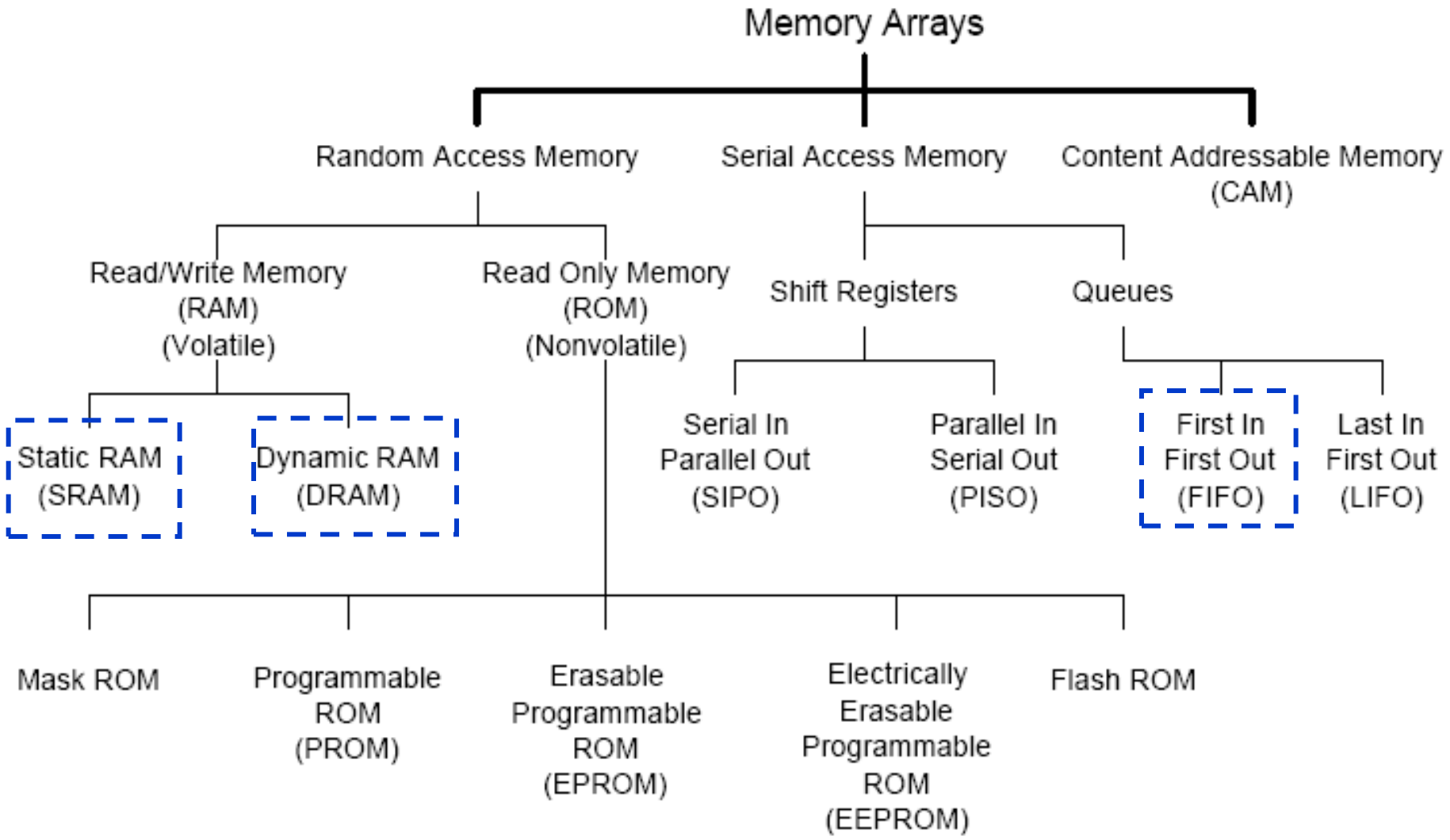
Χειμερινό Εξάμηνο
2015-2016

Στατικές Μνήμες - SRAM

Περίληψη

- Μνήμη είναι μια συλλογή από κελιά αποθήκευσης μαζί με κατάλληλα κυκλώματα για είσοδο και έξοδο από και προς την μνήμη.
  - Μπορούμε να γράφουμε και να διαβάζουμε κελιά
- Η μνήμη RAM (Random Access Memory) οργανώνει τα δεδομένα σε λέξεις
- Τα δεδομένα προσπελαυνονται μέσω μιας ακολουθίας από σήματα
  - Κυματομορφές χρονισμού (timing waveforms)
- Οι αποκωδικοποιητές είναι από τα πιο σημαντικά κομμάτια των μνημών
  - Επιλέγουν συγκεκριμένα δεδομένα
- Οι στατικές μνήμες χάνουν τα δεδομένα τους όταν τις αποσυνδέσουμε από το ρεύμα.

Τύποι Memory Arrays



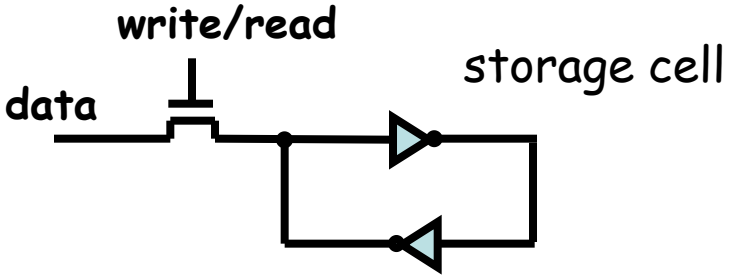
Τύποι RAMs

- Static Random Access Memory (SRAM)
  - Operates like a collection of latches
  - Once value is written, it is guaranteed to remain in the memory as long as power is applied
  - Generally expensive
  - Used inside processors (like the Pentium)
- Dynamic Random Access Memory (DRAM)
  - Generally, simpler internal design than SRAM
  - Requires data to be rewritten (refreshed), otherwise data is lost
  - Often hold larger amount of data than SRAM
  - Longer access times than SRAM
  - Used as main memory in computer systems

SRAM vs. DRAM

Static RAM (SRAM)

Δεδομένα αποθηκεύονται σε Latch.

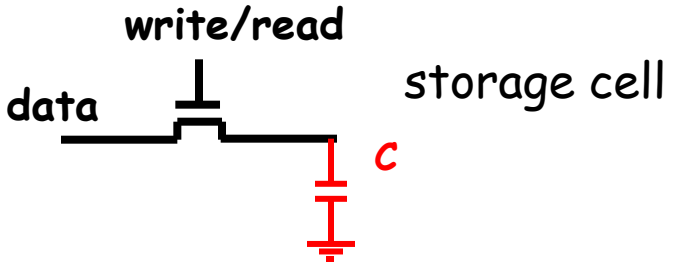


+Ταχύτερη προσπέλαση μνήμης.
+Δεν χρειάζεται refreshing.
+Καλή συμπεριφόρα στον θόρυβο.

-Μεγαλύτερο μέγεθος/bit από DRAM.

Dynamic RAM (DRAM)

Δεδομένα αποθηκεύονται σε φορτίο dynamic node.



+Μικρό μέγεθος/bit μνήμης.

-Χρειάζεται refreshing λόγω leakage. Πιο αργή από SRAM.
-Προβλήματα με θόρυβο (noise).

Τα βασικά σήματα των RAMs

- Γραμμές εισόδου και εξόδου δεδομένων (input and output lines, DIN - DOUT)
- Η μνήμη περιέχει 2^k λέξεις (memory words)
  - K γραμμές διεύθυνσης (address lines - ADDR) επιλέγουν 1 λέξη από τις 2^k
- Θέτουμε το σήμα Read (*asserted)* για να μεταφέρουμε δεδομένα στην έξοδο.(OE - output enable)
- Θέτουμε το σήμα Write (*asserted)* για να αποθηκέυσουμε τα δεδομένα της εισόδου. (WE - write enable)
- Chip Enable/ Chip Select (CE-CS) σαν γενικός διακόπτης λειτουργίας

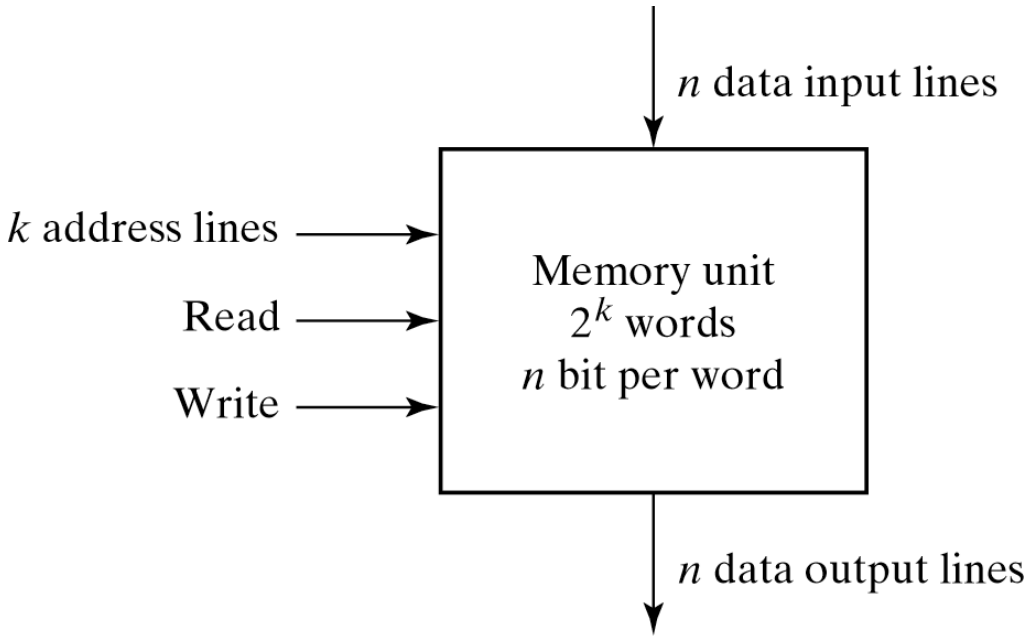
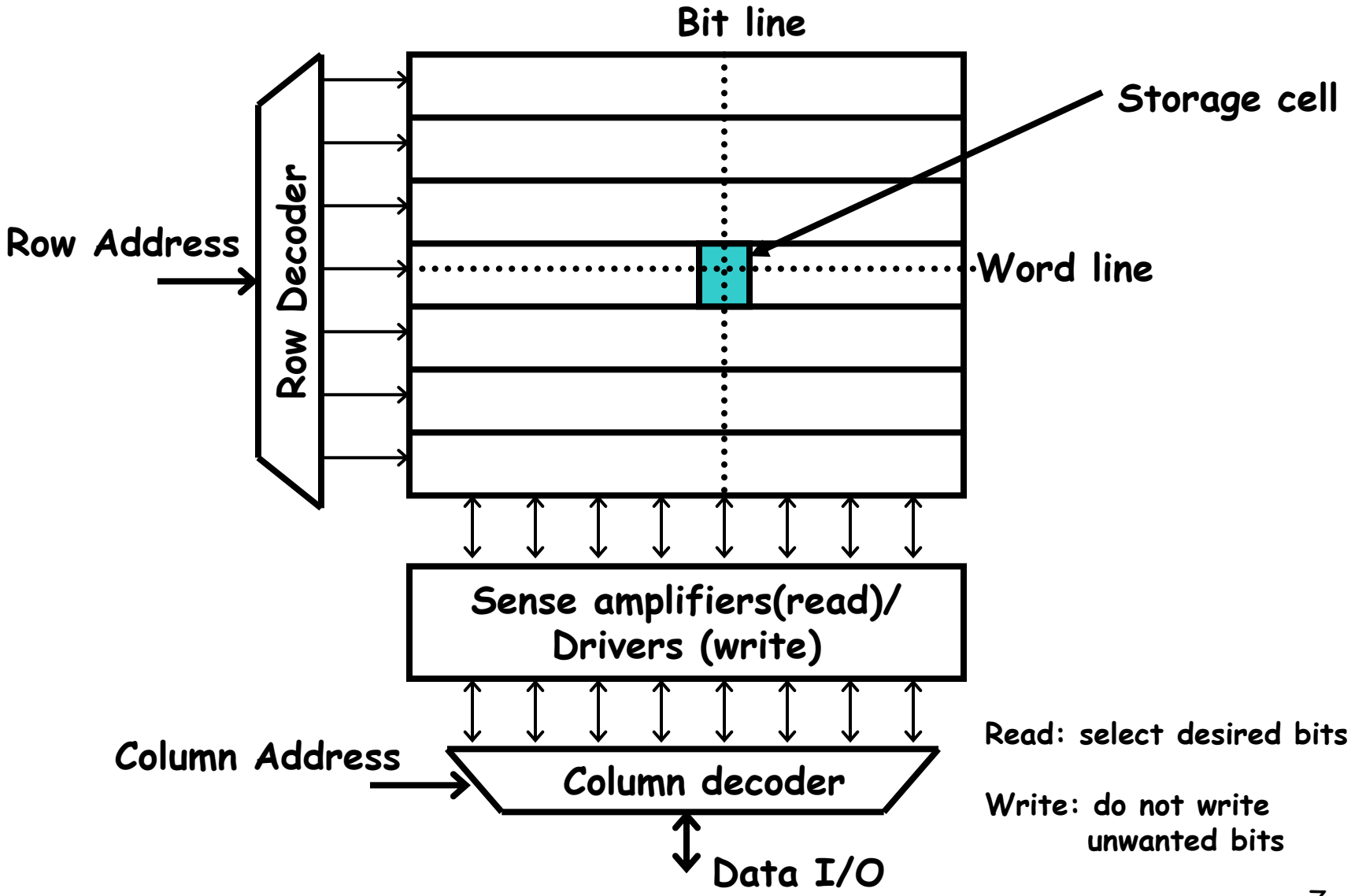


Fig. 7-2 Block Diagram of a Memory Unit

Αρχιτεκτονική Μνημών



Τα εσωτερικά της RAM

- Η διεύθυνση πάει στον decoder
  - Μόνο μια έξοδος ενεργή
- Η Word line επιλέγει μια γραμμή (row) από bits (word)
- Κάθε binary cell (BC) αποθηκεύει 1 bit
- Τα δεδομένα εισόδου αποθηκεύονται όταν το Read/Write είναι 0
- Τα δεδομένα εξόδου βγαίνουν όταν το Read/Write είναι 1

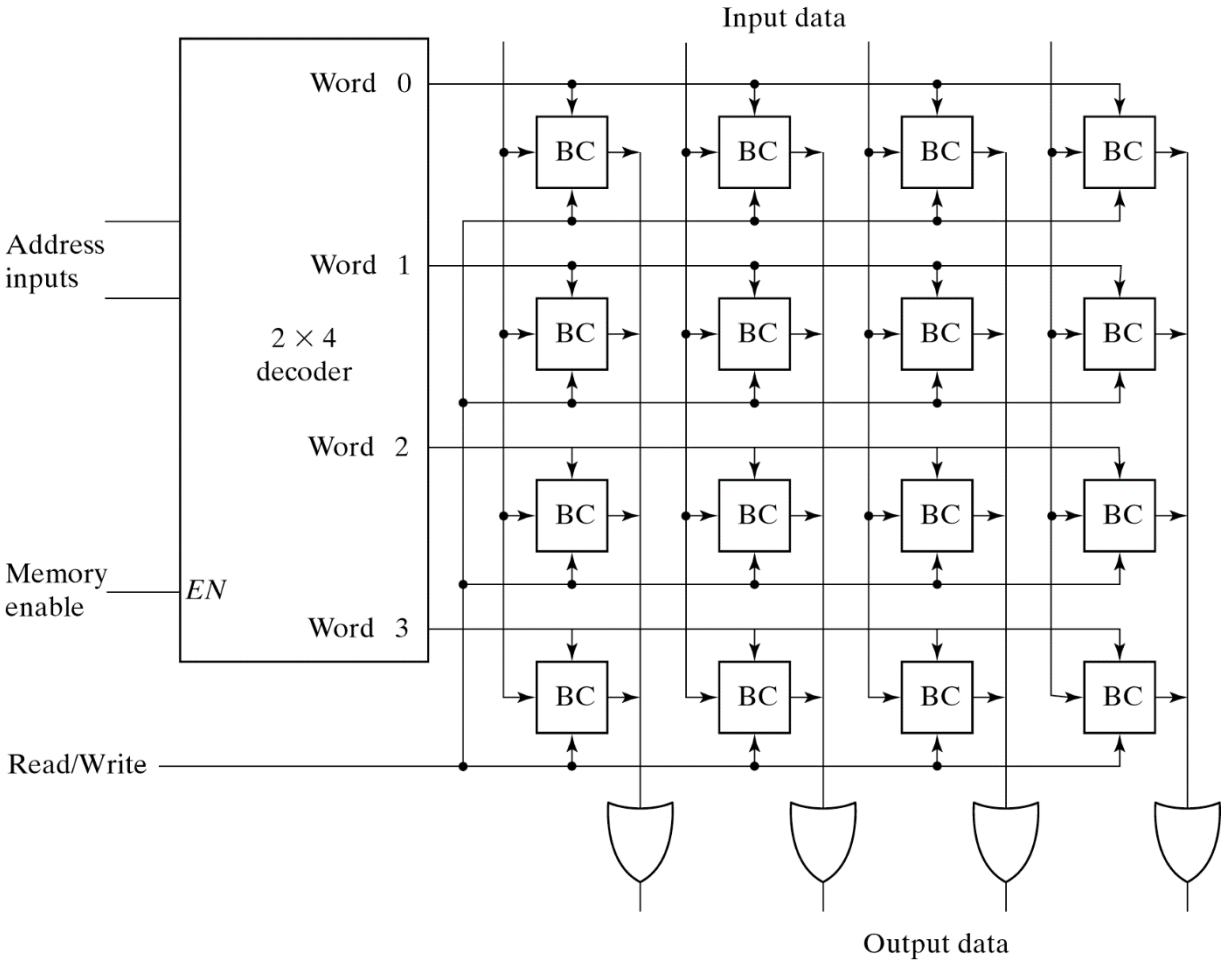


Fig. 7-6 Diagram of a 4 × 4 RAM

Τα εσωτερικά της SRAM

- Note: η καθυστέση εξαρτάται κυρίως από τον αριθμό των λέξεων
- Η καθυστέρηση δεν επηρεάζεται από το μήκος των λέξεων

° Πόσα address bits χρειαζόμαστε για 16 words?

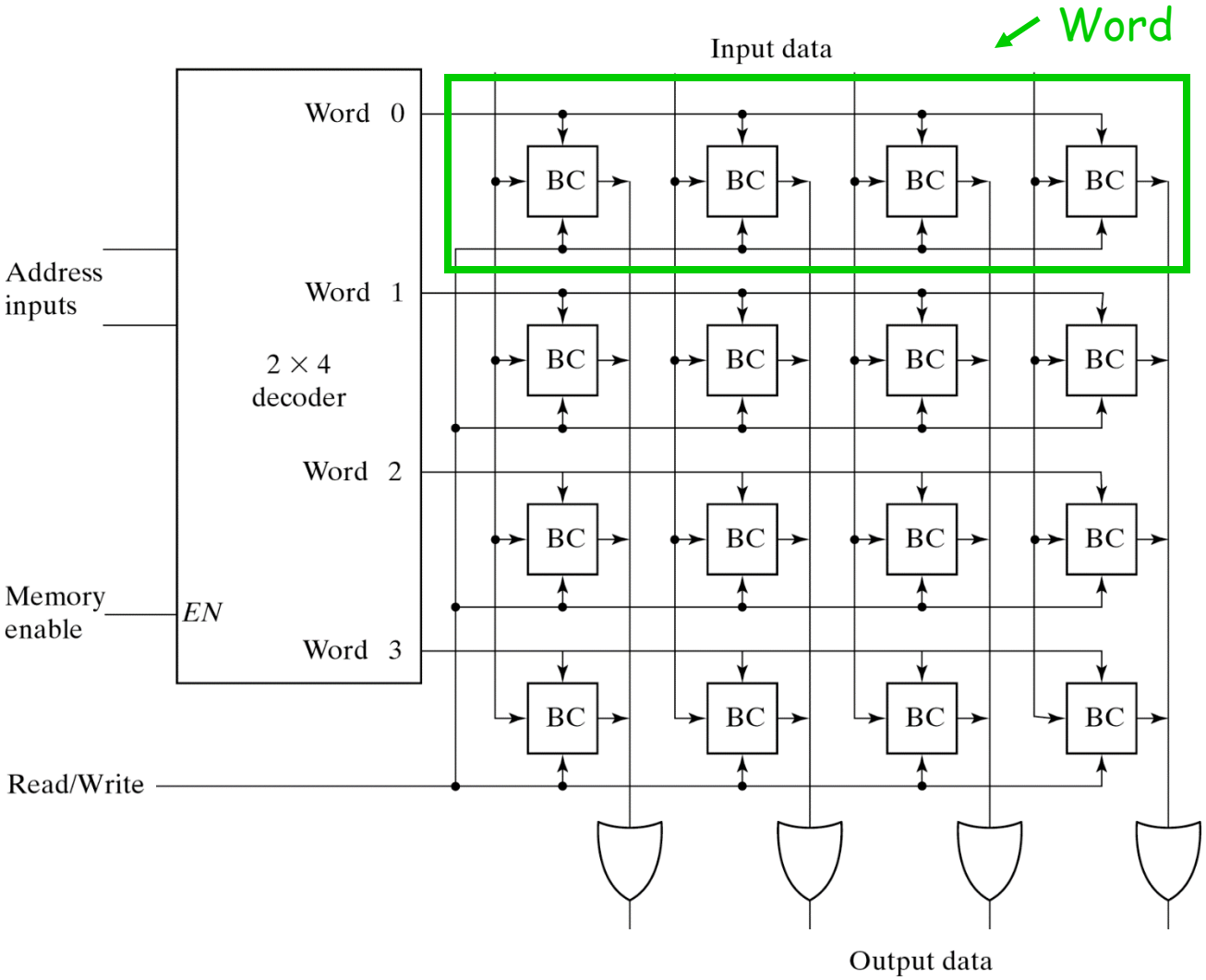
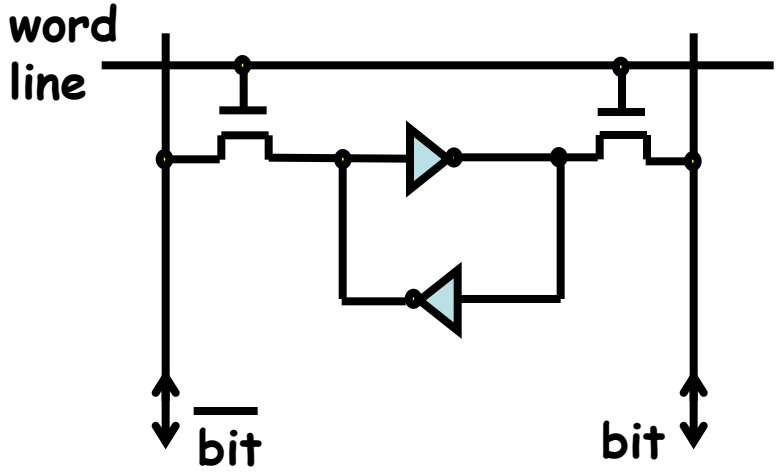


Fig. 7-6 Diagram of a 4 × 4 RAM

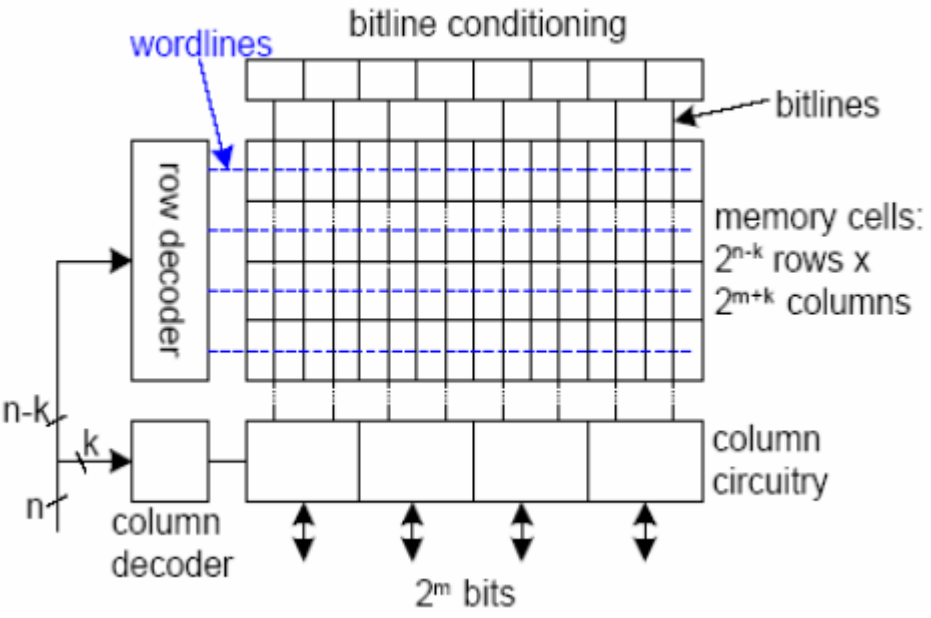
Ένα κελί SRAM – 6 transistors



Όταν η word line ενεργοποιείται (V_{DD}) τότε η τιμή του Latch διαβάζεται στα bit και $\overline{bit}$ κατά το διάβασμα της μνήμης ή η τιμή του Latch γράφεται από τα bit και $\overline{bit}$ κατά την εγγραφή της μνήμης.

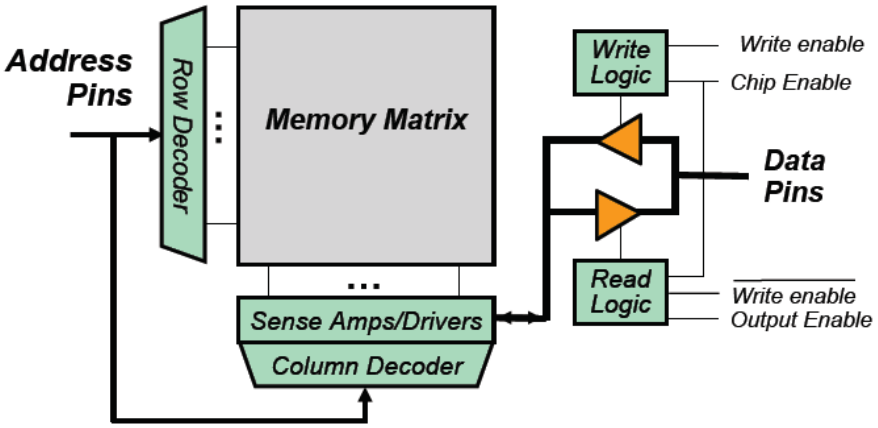
Αρχιτεκτονική για Μεγαλύτερες Μνήμες

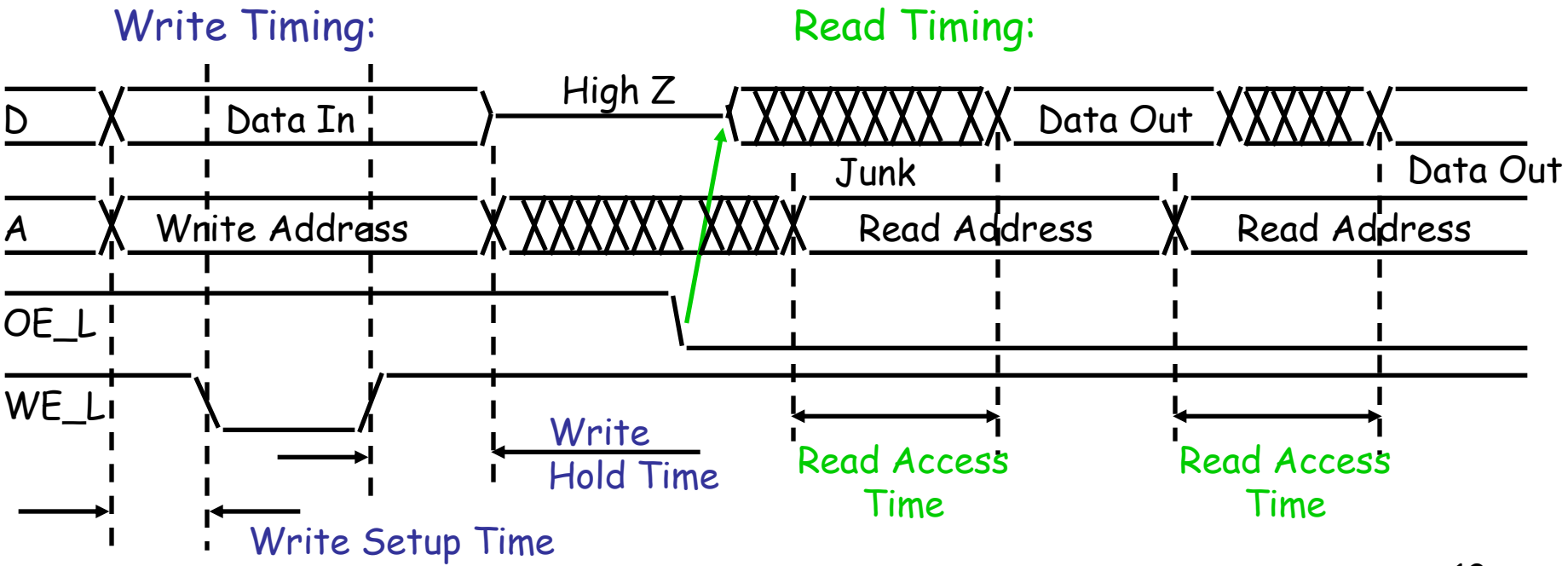
- 2^n λέξεις από 2^m bits η καθεμιά
- Εάν n >> m μπορούμε να την διπλώσουμε (fold) κατά $2^κ$ σε λιγότερες γραμμές και πιο πολλές στήλες.
  - Αποκωδικοποιητής στηλών !
- Κανονικότητα στη σχεδίαση – εύκολη σχεδίαση



Τυπικός χρονισμός Ασύγχρονης SRAM

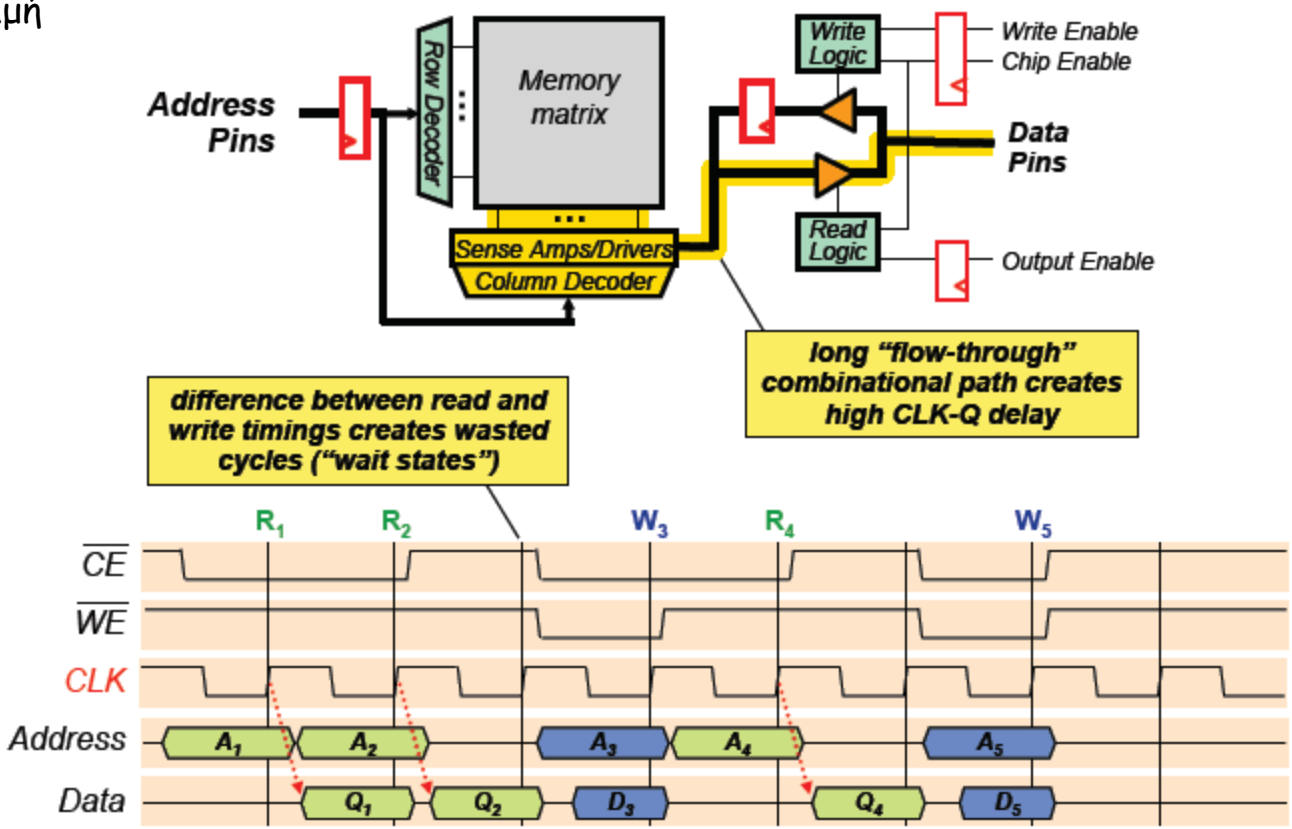
- Asynchronous SRAM
  - Χρονισμός μόνο βάση των σημάτων





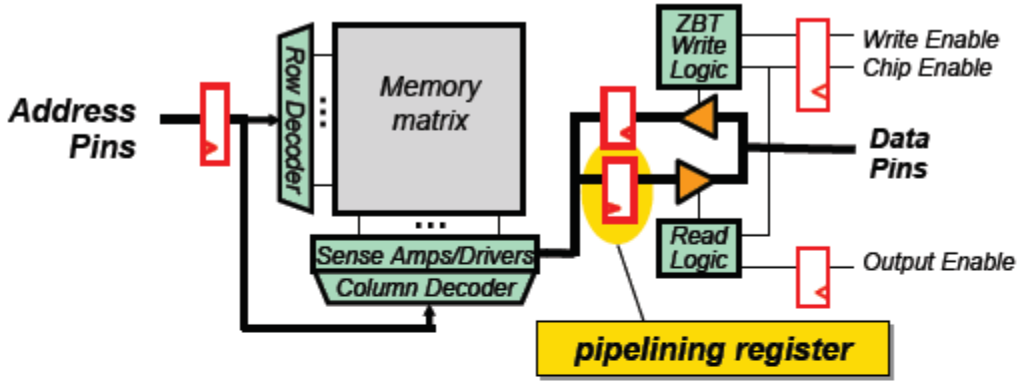
Τυπικός χρονισμός Σύγχρονης SRAM

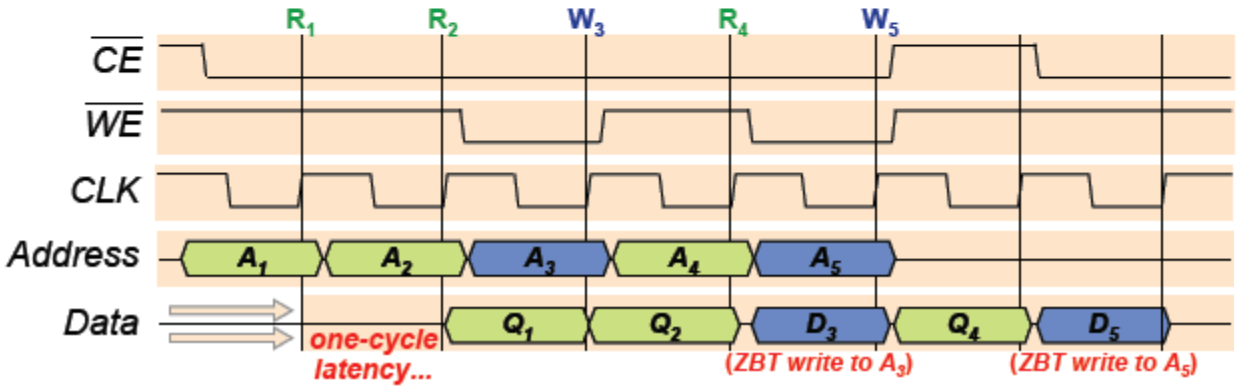
- Synchronous – SRAM
  - Χρονισμός με βάση τα σήματα στην ακμή του ρολογιού
- Write-after-Read πρόβλημα – Wait states – Bus Turnaround
  - Στα read τα data βγαίνουν μετά την ακμή ενώ στα writes τα data πρέπει να μπούν πρίν την ακμή



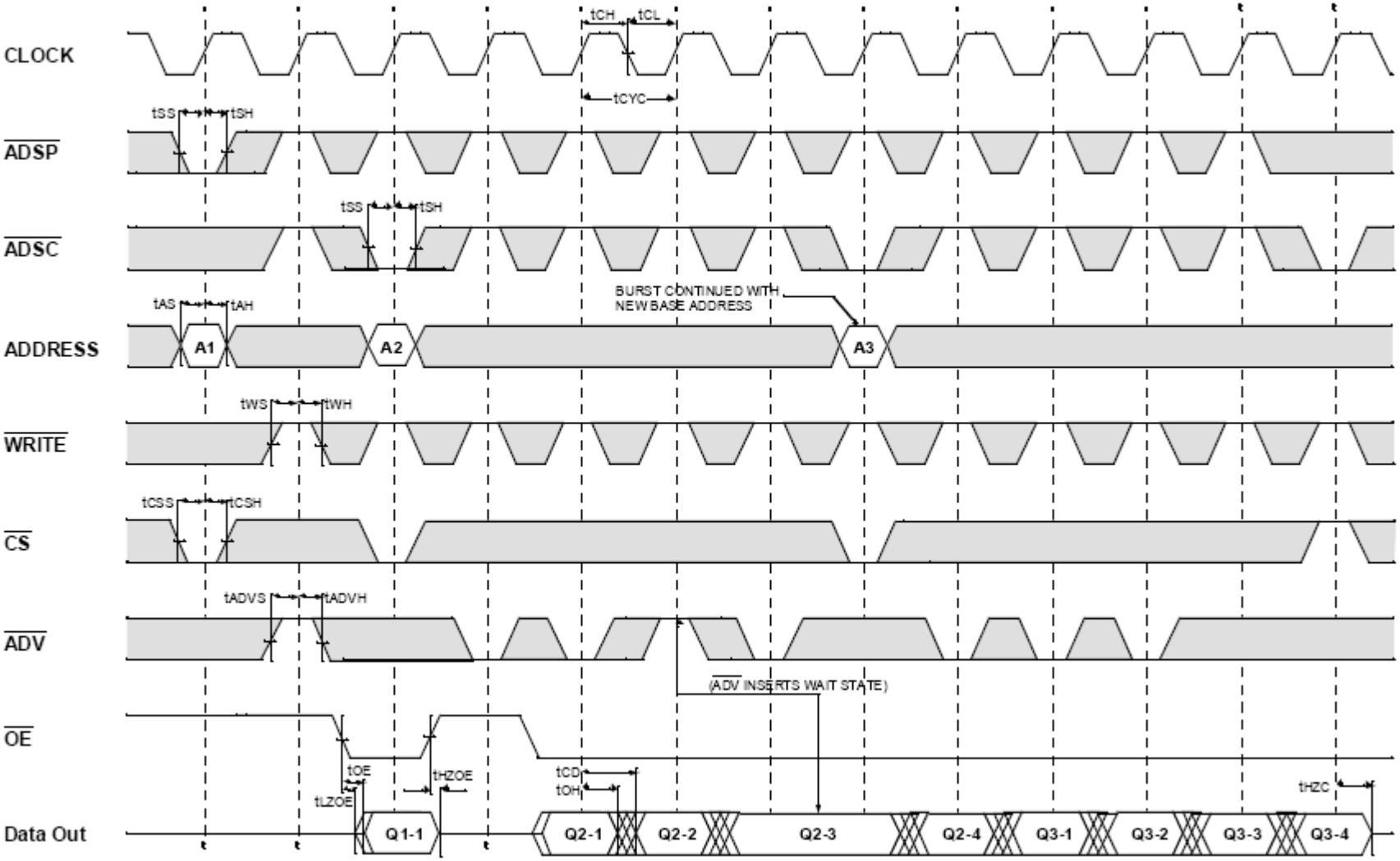
Τυπικός χρονισμός ZBT Synchronous SRAM

- Zero-Bus-Turnaround (ZBT) ή No Bus Latency (NoBL)
  - Pipelined έξοδος λύνει το πρόβλημα Write-after-Read
  - Μειώνει Clk2Q delay και επιτρέπει υψηλότερη συχνότητα ρολογιού
  - Extra κύκλος καθυστέρηση στην ανάγνωση

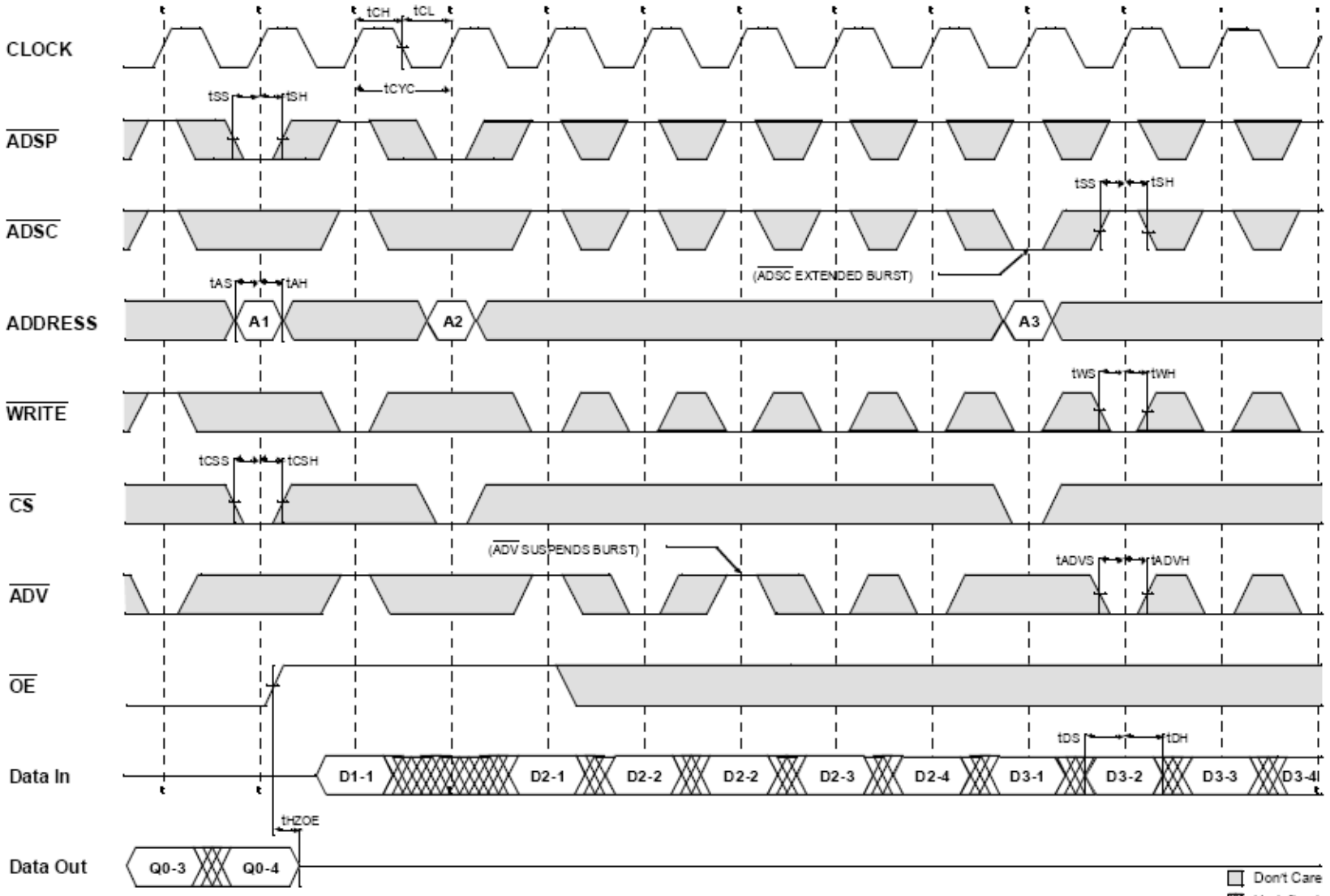




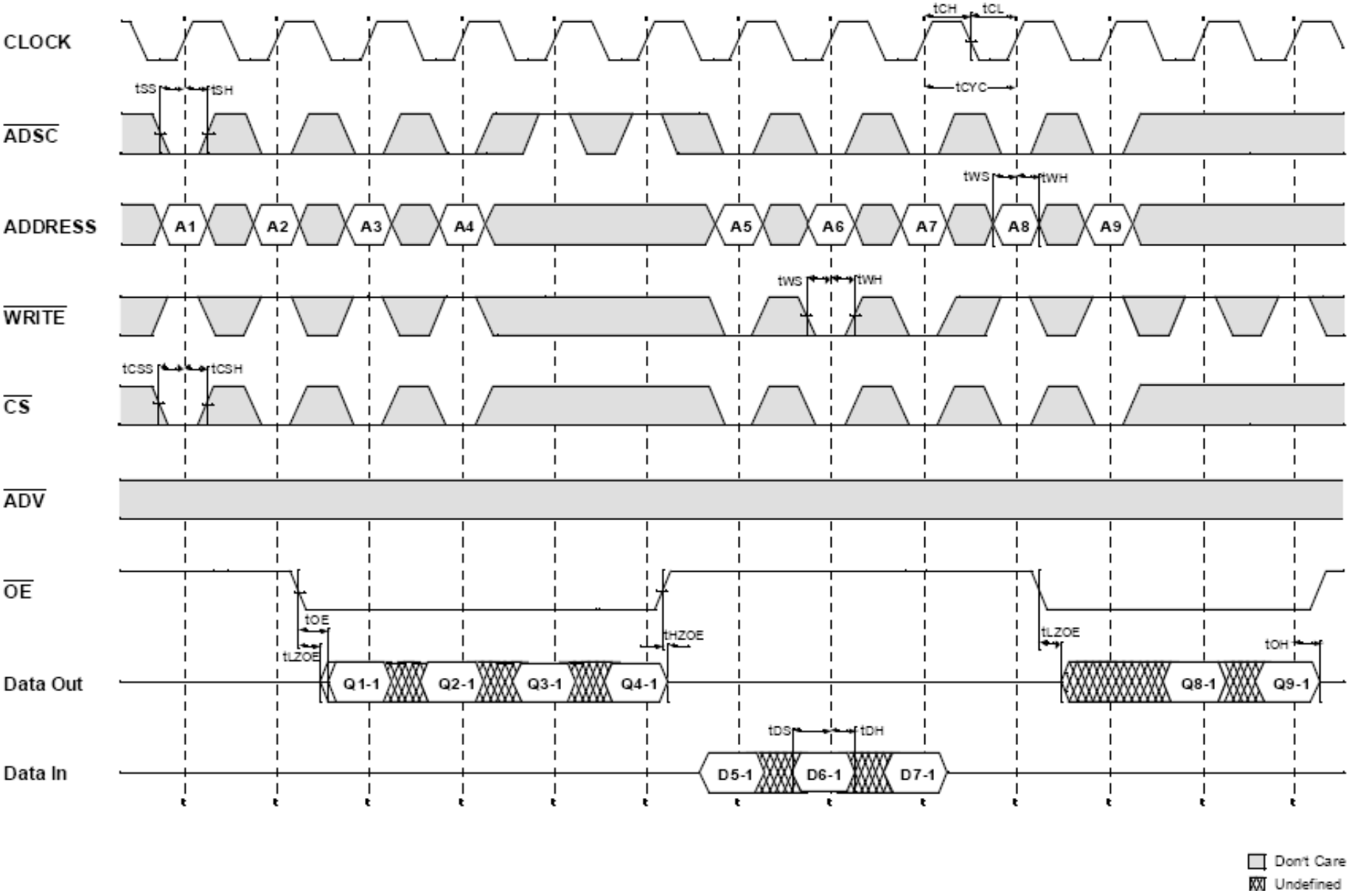
Χρονισμός SRAM για Read από Datasheet



Χρονισμός SRAM για Write από Datasheet

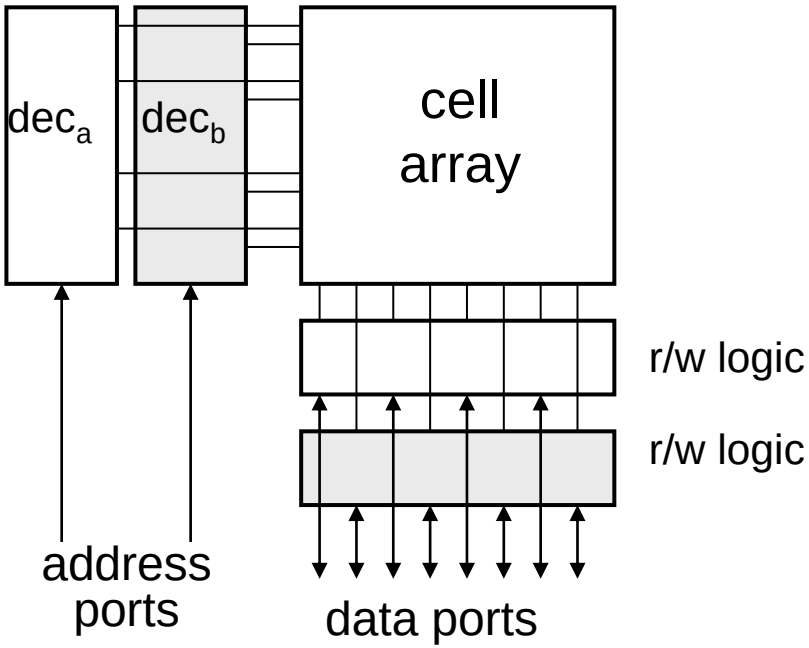


Χρονισμός SRAM για Read/Write από Datasheet

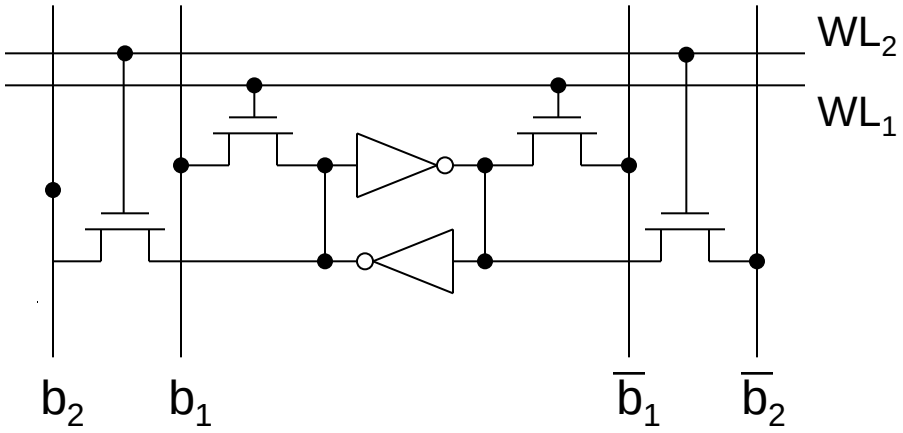


Dual-ported Memory Internals

- Add decoder, another set of read/write logic, bits lines, word lines:



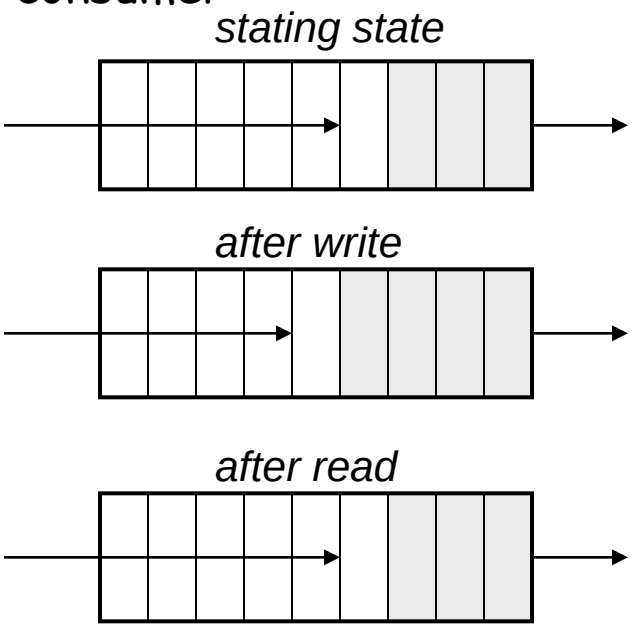
- Example cell: SRAM



- Repeat everything but cross-coupled inverters.
- This scheme extends up to a couple more ports, then need to add additional transistors.

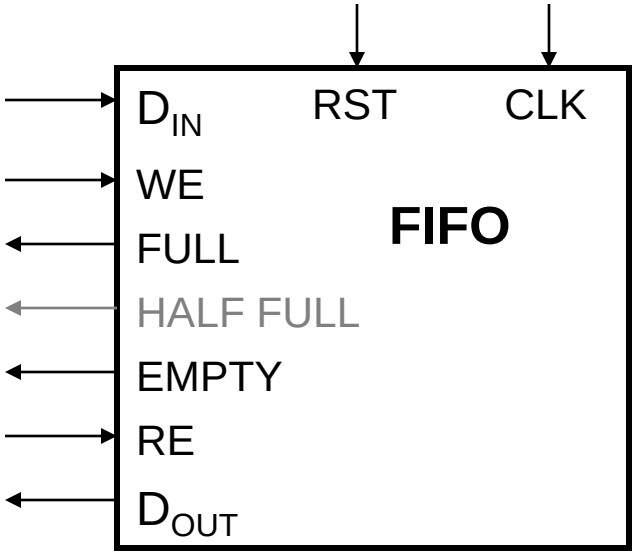
First-in-first-out (FIFO) Memory

- Used to implement *queues*.
- These find common use in computers and communication circuits.
- Generally, used for rate matching data producer and consumer:

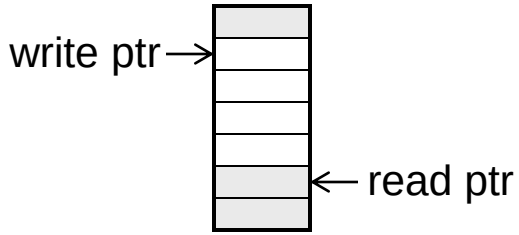


- Producer can perform many writes without consumer performing any reads (or vice versa). However, because of finite buffer size, on average, need equal number of reads and writes.
- Typical uses:
  - interfacing I/O devices. Example network interface. Data bursts from network, then processor bursts to memory buffer (or reads one word at a time from interface). Operations not synchronized.
  - Example: Audio output. Processor produces output samples in bursts (during process swap-in time). Audio DAC clocks it out at constant sample rate.

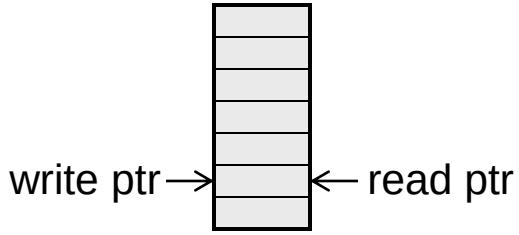
FIFO Interfaces



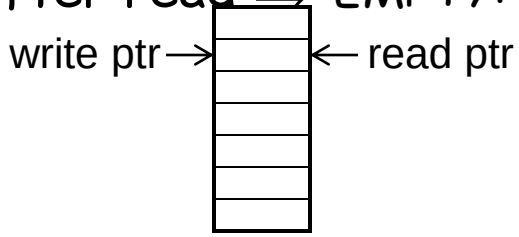
- Address pointers are used internally to keep next write position and next read position into a dual-port memory.



- If pointers equal after write ⇒ FULL:



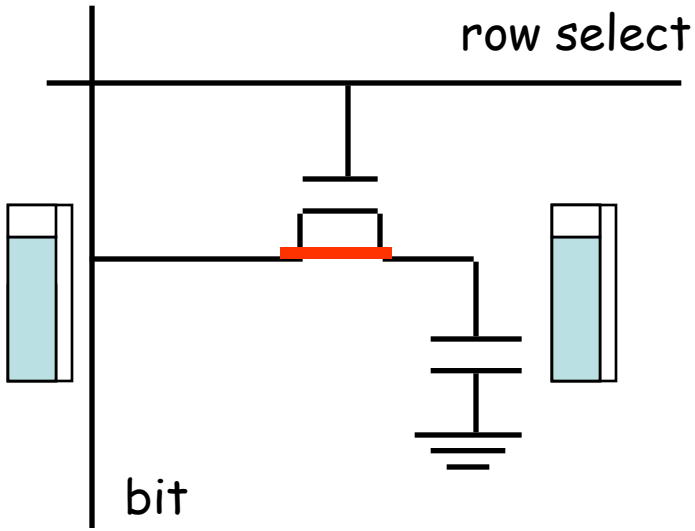
- If pointers equal after read ⇒ EMPTY:



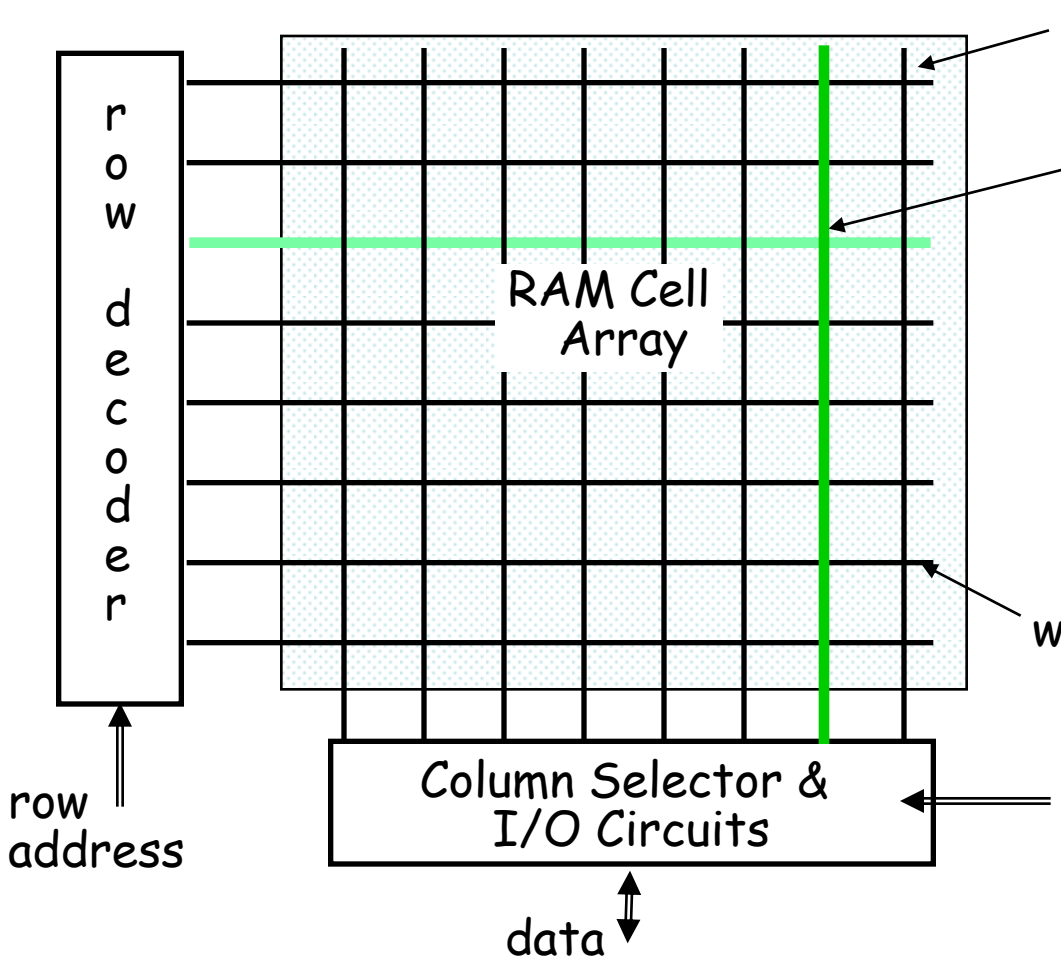
- After write or read operation, FULL and EMPTY indicate status of buffer.
- Used by external logic to control own reading from or writing to the buffer.
- FIFO resets to EMPTY state.
- HALF FULL (or other indicator of partial fullness) is optional.

Κελί μνήμης με 1 transistor (DRAM)

- Εγγραφή - Write:
  1. Οδηγούμε την bit line
  2. Επιλέγουμε γραμμή (row select)
- Ανάγνωση - Read:
  1. Προφορτίζουμε (precharge) την bit line σε Vdd/2
  2. Επιλέγουμε γραμμή (row select)
  3. Το κελί και η bit line μοιράζονται τα φορτία (charge sharing)
  4. Sense στην bit line (sense amplifier)
     Μπορεί να ανιχνεύει πολύ μικρές αλλαγές
  5. Write: επαναφέρουμε την τιμή
- Ανανέωση – Refresh :
  - Αρκεί ένα απλό read σε κάθε κελί



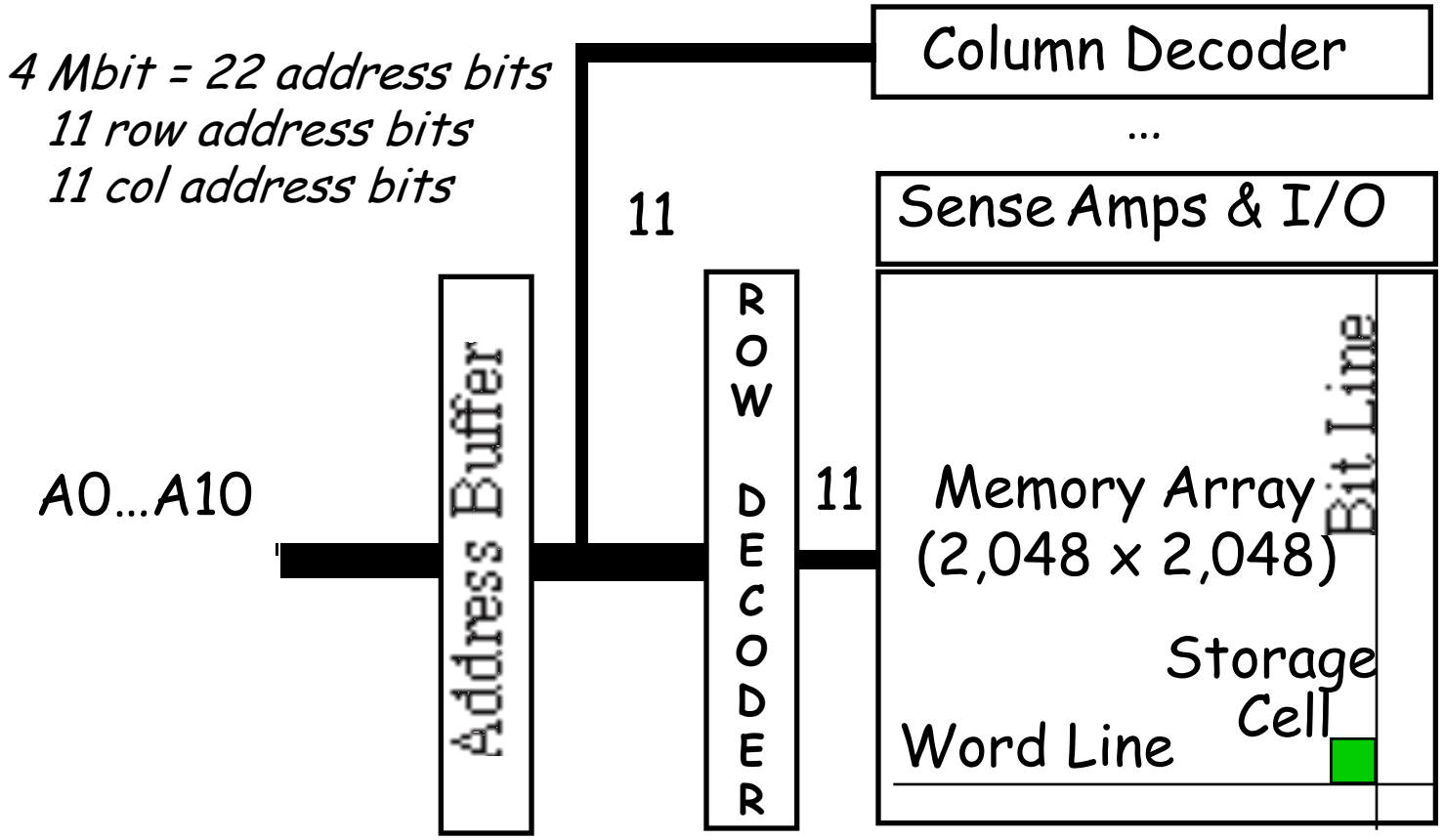
Κλασσική Οργάνωση DRAM (τετραγωνική)



Square keeps the wires short: Power and speed advantages Less RC, faster precharge and discharge is faster access time!

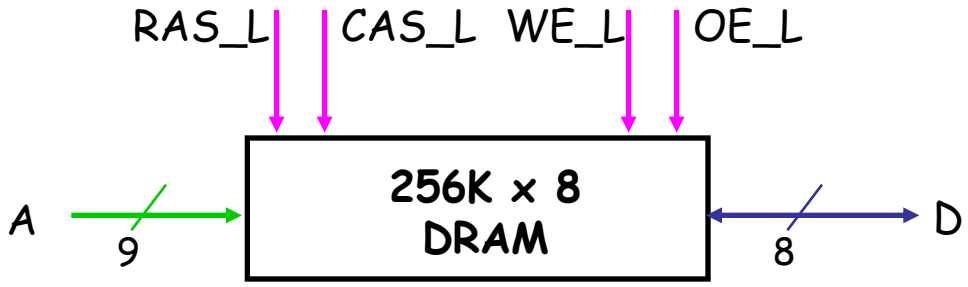
Row and Column Address together select 1 bit a time

Λογική Οργάνωση DRAM (4 Mbit)



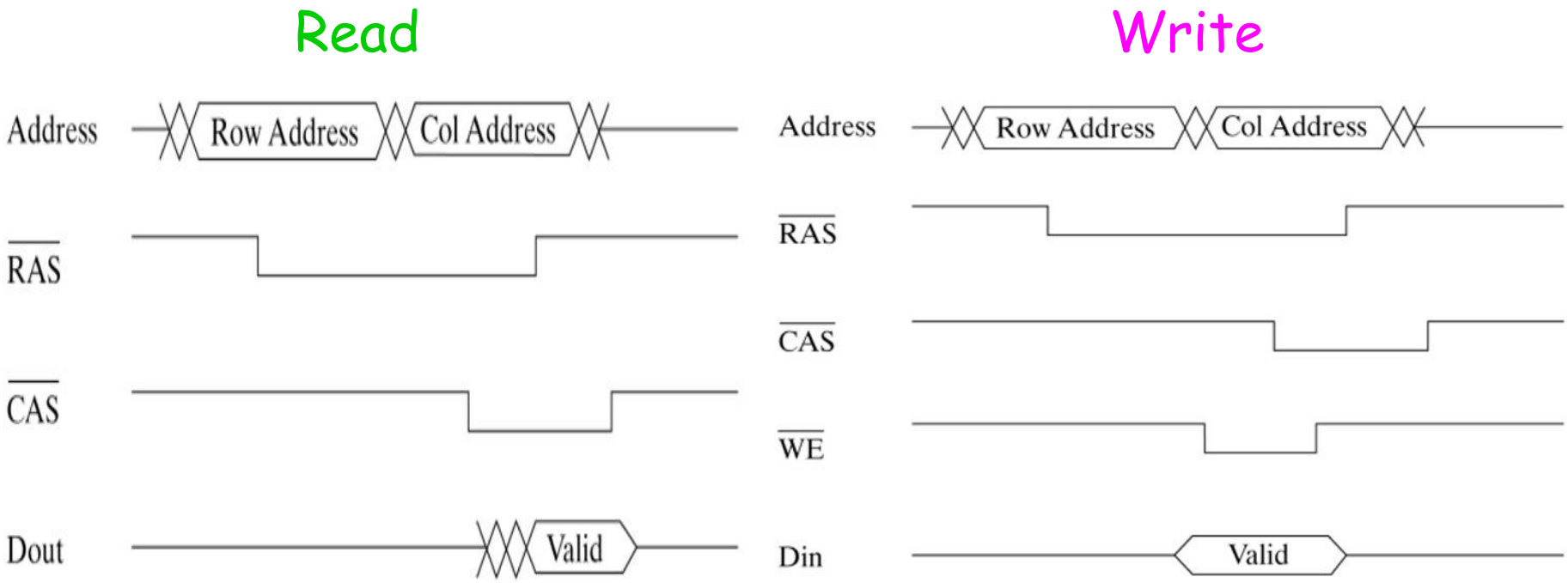
- Square root of bits per RAS/CAS
  - Row selects 1 row of 2048 bits from 2048 rows
  - Col selects 1 bit out of 2048 bits in such a row

Τα σήματα της DRAM



- Σήματα ελέγχου (RAS_L, CAS_L, WE_L, OE_L) όλα active low
- Κοινό bus δεδομένων D - εισόδου κα ι εξόδου (Din & Dout):
  - WE_L ενεργοποιείται (Low), OE_L απενεργοποιείται (High)
    - D χρησιμοποιείται σαν είσοδος στην DRAM
  - WE_L απενεργοποιείται (High), OE_L ενεργοποιείται (Low)
    - D χρησιμοποιείται σαν έξοδος από την DRAM
- Οι διευθύνσεις γραμμής και στήλης μοιράζονται τα ίδια pins (A)
  - RAS_L ενεργοποιείται (low) -> A αποθηκεύεται σαν row address
  - CAS_L ενεργοποιείται (low) -> A αποθηκεύεται σαν column address
  - RAS/CAS edge-sensitive

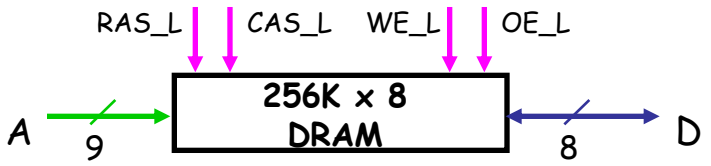
Απλοποιημένο διάγραμμα χρονισμού DRAM

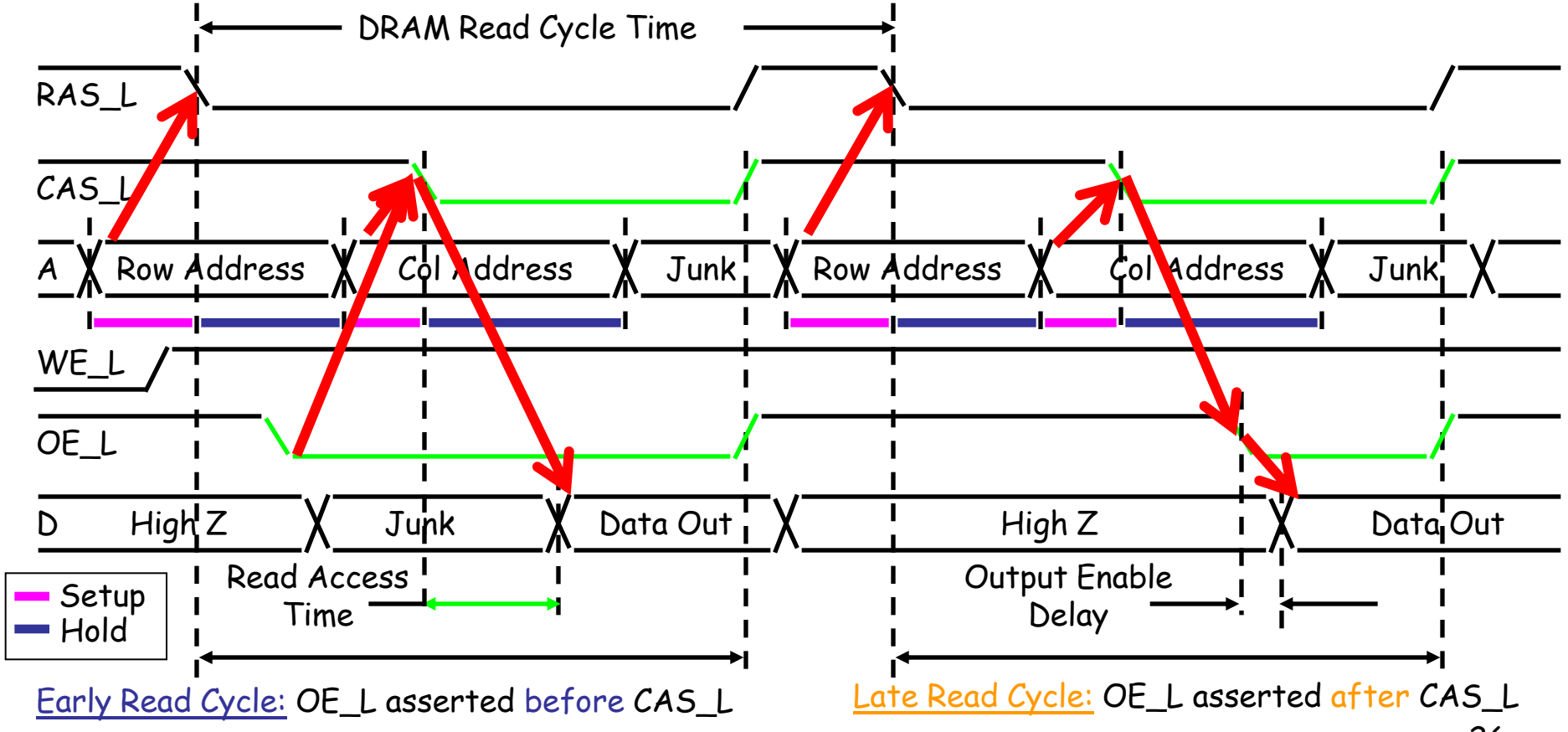


- Η διεύθυνση δίνεται σε 2 βήματα

Τυπικός χρονισμός Ανάγνωσης DRAM

- Κάθε προσπέλαση DRAM ξεκινάει με:
  - Ενεργοποίηση του RAS_L
  - 2 τρόποι ανάγνωση: early or late





Early Read Cycle: OE_L asserted before CAS_L

Late Read Cycle: OE_L asserted after CAS_L

Τα βήματα για Early Read

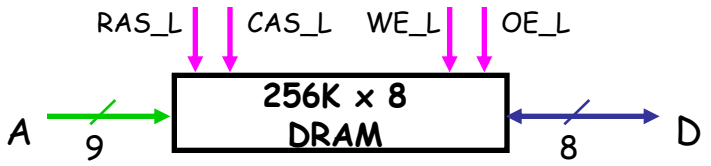
- Assert Row Address
- Assert RAS_L
  - Start read cycle
  - Meet Row Addr setup time before RAS/hold time after RAS
- Assert OE_L
- Assert Col Address
- Assert CAS_L
  - Meet Col Addr setup time before CAS/hold time after CAS
- Valid Data Out after access time
- Disassert OE_L, CAS_L, RAS_L to end cycle

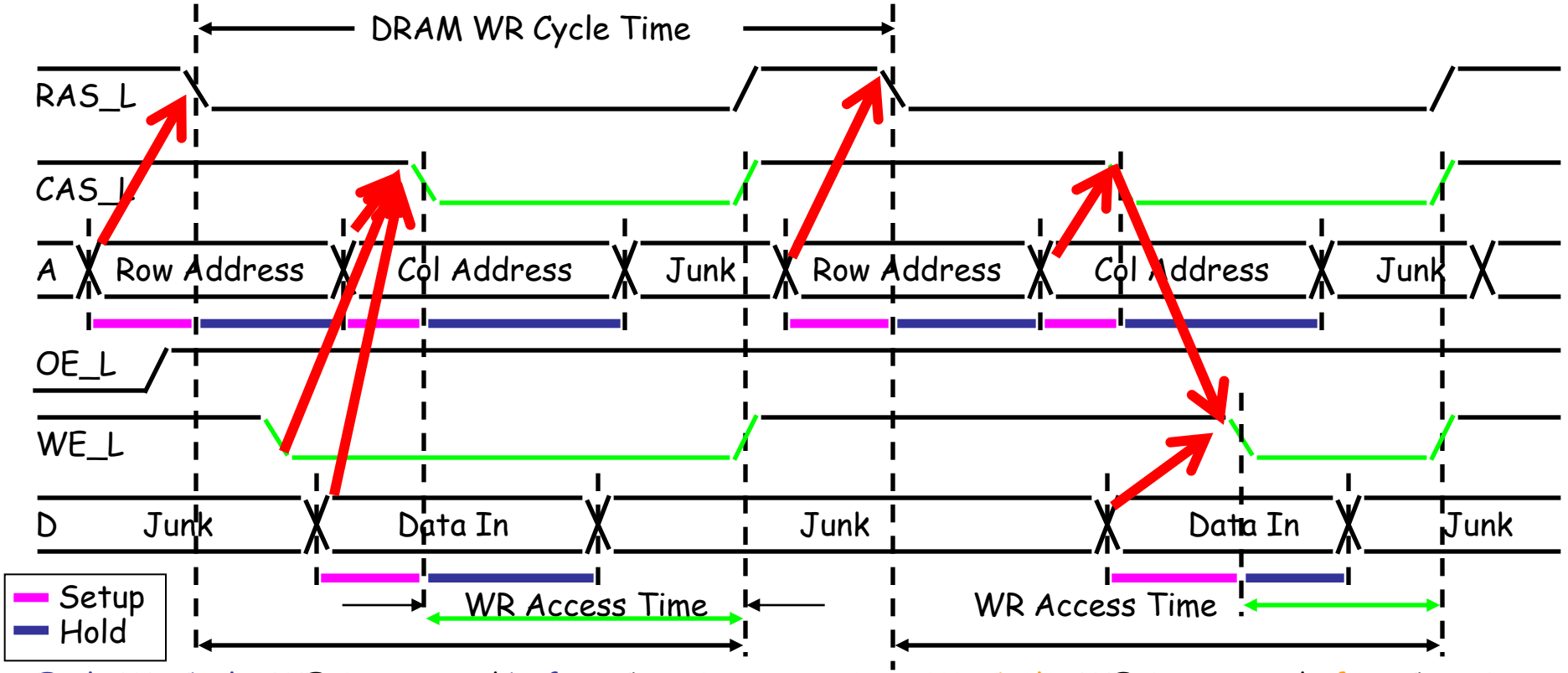
Τα βήματα για Late Read

- Assert Row Address
- Assert RAS_L
  - Start read cycle
  - Meet Row Addr setup time before RAS/hold time after RAS
- Assert Col Address
- Assert CAS_L
  - Meet Col Addr setup time before CAS/hold time after CAS
- Assert OE_L
- Valid Data Out after access time
- Disassert OE_L, CAS_L, RAS_L to end cycle

Τυπικός χρονισμός Εγγραφής DRAM

- Κάθε προσπέλαση DRAM ξεκινάει με:
  - Ενεργοποίηση του RAS_L
  - 2 τρόποι εγγραφής: early or late

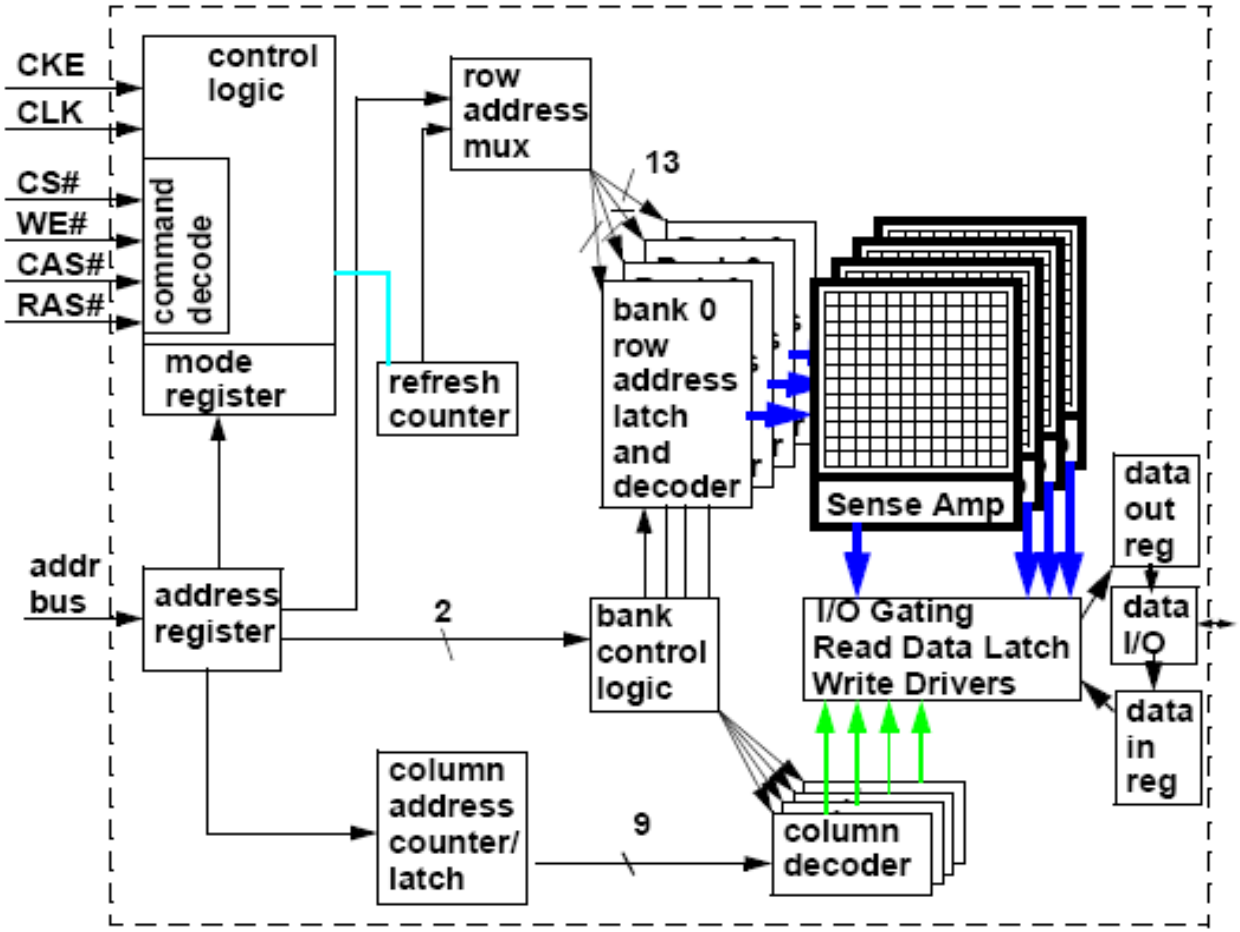




Early Wr Cycle: WE_L asserted before CAS_L

Late Wr Cycle: WE_L asserted after CAS_L

256 Mbit SDRAM Addressing

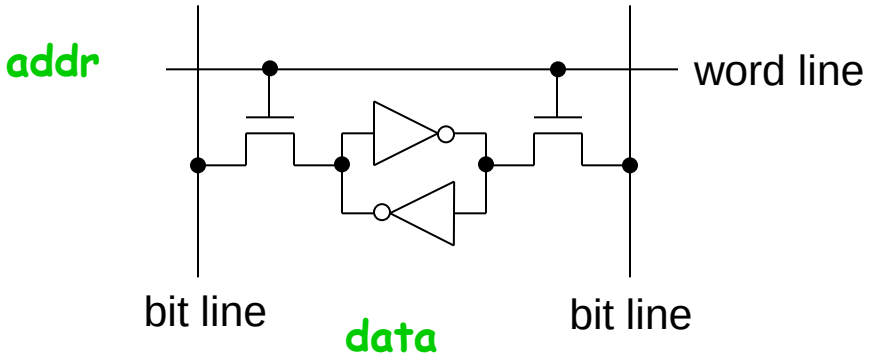


256 Mbit chip: 8192 rows, 512 columns, x16 data, 4 banks

Volatile Memory Comparison

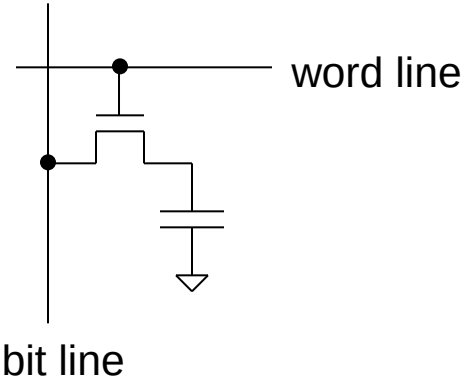
The primary difference between different memory types is the bit cell.

- SRAM Cell



- Larger cell ⇒ lower density, higher cost/bit
- No dissipation
- Read non-destructive
- No refresh required
- Simple read ⇒ faster access
- Standard IC process ⇒ natural for integration with logic

- DRAM Cell



- Smaller cell ⇒ higher density, lower cost/bit
- Needs periodic refresh, and refresh after read
- Complex read ⇒ longer access time
- Special IC process ⇒ difficult to integrate with logic circuits
- Density impacts addressing