XMUT202 Digital Electronics

Memory

Week 16 Lecture 1

Agatha Rachmat

School of Engineering and Computer Science

Victoria University of Wellington

Introduction to memory

- Memory is a common element of digital systems.
- Each type of memory has advantages and disadvantages. You need to select the appropriate memory for your application.
- This introduction will look at a few selected aspects of memory devices.

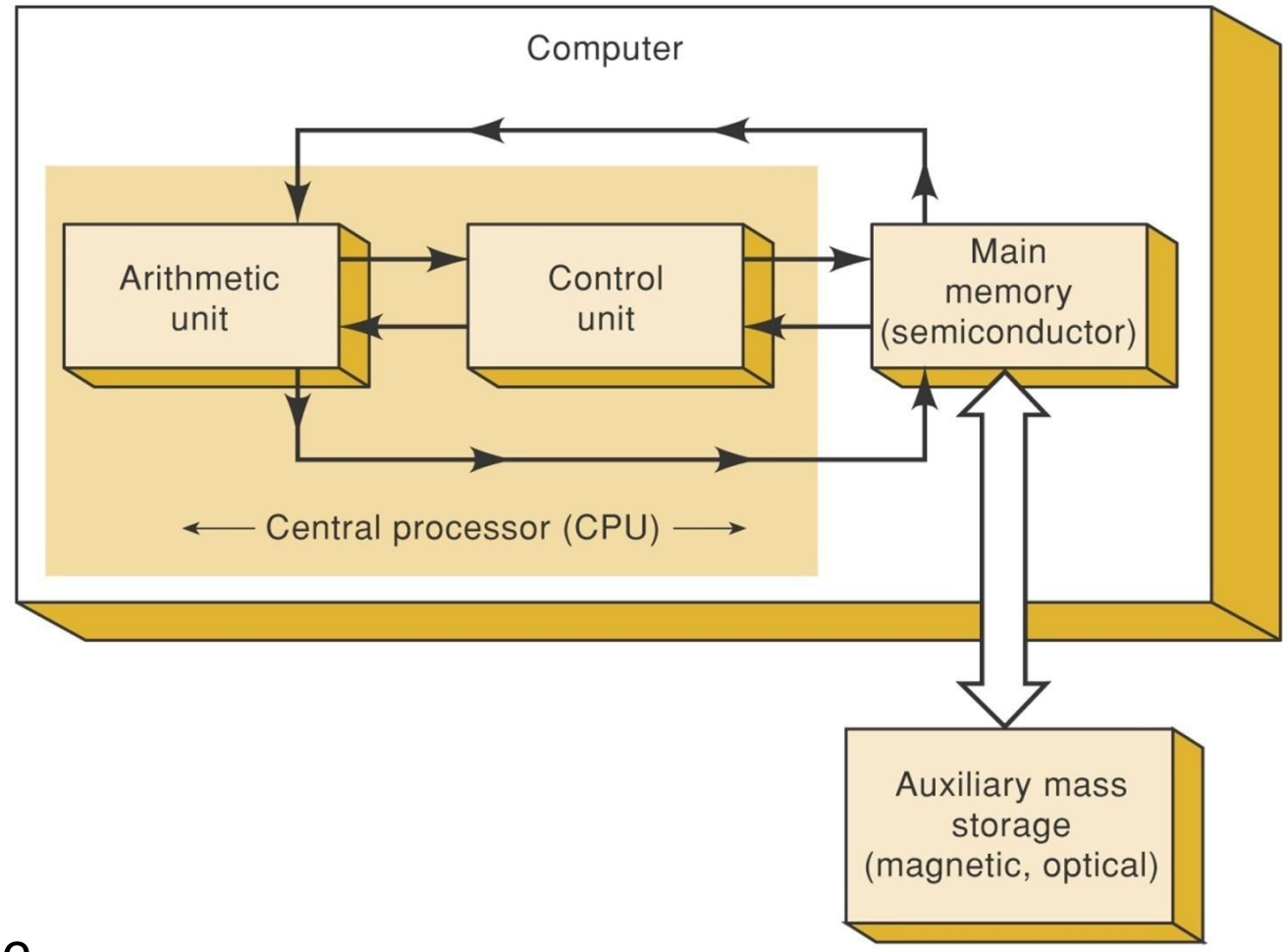
Introduction to memory

A computer system normally uses high-speed main memory and slower external auxiliary memory.

- The CPU contains a small extremely high speed memory space called cache.
- The main bulk memory of the computer is usually DRAM.
- The auxiliary mass storage memory is generally NAND flash (a solid-state hard drive or SSD).

Speed

Size



Capacity calculations:

A memory consists of **4096 words, each 20 bits long.**

The total capacity is 4096 x 20 = 81920 bits.

The number of words is typically represented as a multiple of 1024 (2^{10} = 1k). In the example above, we have 4k × 20 bits.

1024 x 1024 = 1048576 = **1M => 2^{20}**
1024 x 1024 x 1024 = 1073741824 = **1G => 2^{30}**

Which memory has the greatest capacity

1. A 5M x 8 device
2. A 3M x 16 device

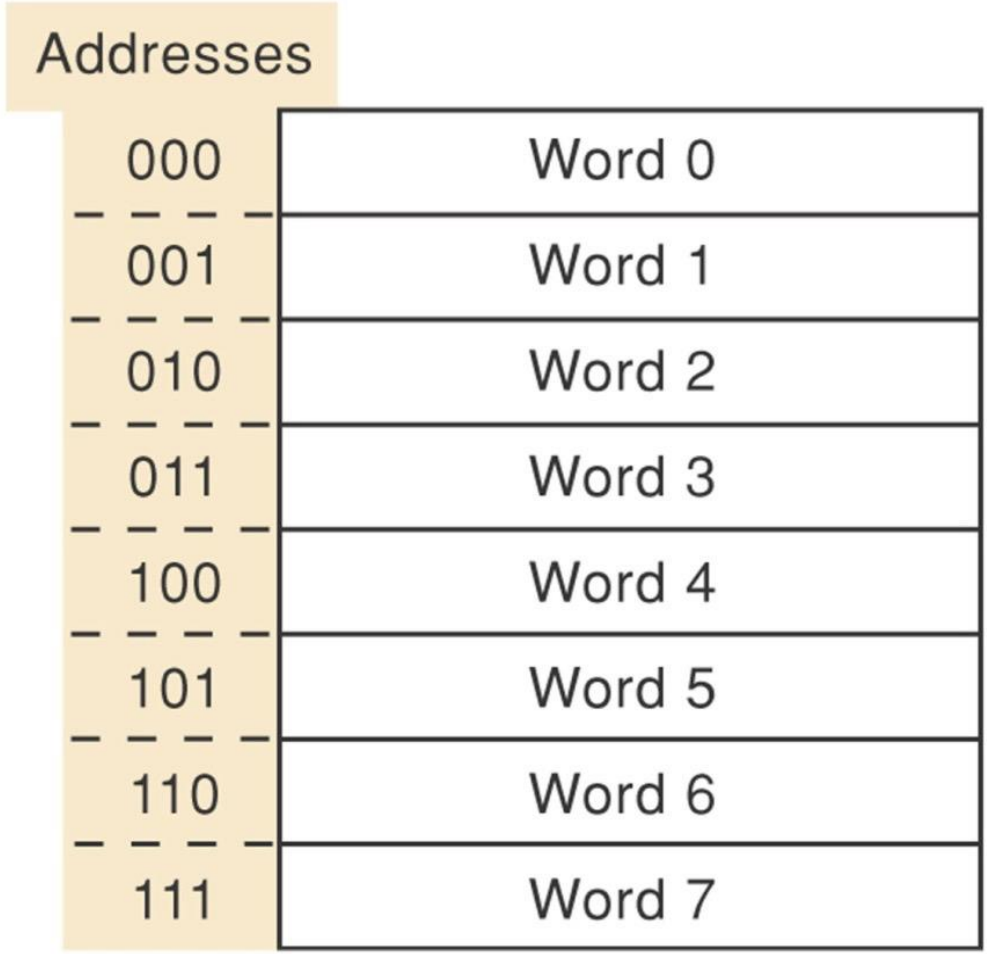
$5M \times 8 = 5 \times 2^{20} \times 8 = 41{,}943{,}040$ bits

$3M \times 16 = 3 \times 2^{20} \times 16 = 50{,}331{,}648$ bits

Memory addresses

- A unique number that identifies the location of a word in memory.

A 3-bit address can identify each word in an 8 word system.



Operations

Read: The operation where the binary word stored in a specific memory location is sensed and transferred to another device. Also called a fetch operation.

Write: The operation where a new word is placed at a memory location replacing the previous word stored there. Also referred to as a store operation.

Access time (t_{ACC}): The amount of time required to perform a read operation.

Types of memory

Volatile Memory: Memory that requires the application of electrical power to retain information.

Non-Volatile Memory: Memory that retains information when no electrical power is supplied.

Random Access Memory (RAM): Memory that can address and access any memory word in any order. The physical location of the memory word has minimal effect on how long it takes to access as the memory locations can be accessed in any order.

Sequential Access Memory (SAM): The memory access time is not constant, but depends on the address location. Used when words must be accessed in the same order every time. An example would be a DVD.

Types of memory

Read Write Memory (RWM): Memory that can be written to or read from with equal ease.

Read Only Memory (ROM): Designed for applications with many read cycles, typically write once read many (WORM) memory. All ROM is non-volatile.

Static Memory devices (SRAM): Semiconductor memory in which the stored data will remain permanently, as long as power is applied without the need to refresh the memory cells (re-write the data).

Dynamic Memory (DRAM): Semiconductor memory in which the stored data needs to be constantly refreshed (rewritten into memory again).

Accessing memory

Every memory system requires input and output lines to provide the following functions:

- Apply the address of the memory location to be accessed.
- Enable the memory device to respond to inputs/outputs.
- Select either a **Read** or a **Write** operation.
  - For a read operation, enable the outputs (tristate) which applies data to the output pins.
  - For a write operation, supply input data to be stored on the input pins.

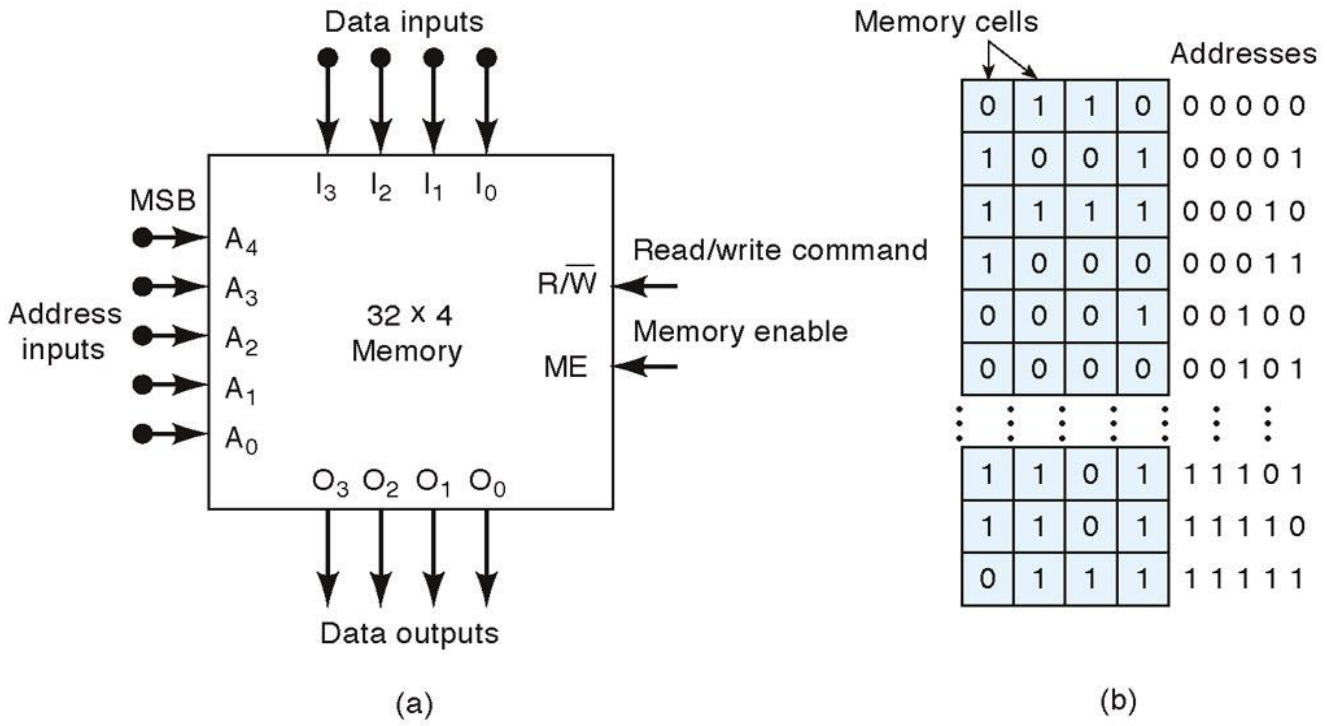
Accessing memory

Every memory system requires input and output lines to provide the following functions:

- Apply the address of the memory location to be accessed.
- Enable the memory device to respond to inputs/outputs.
- Select either a **Read** or a **Write** operation.
  - For a read operation, enable the outputs (tristate) which applies data to the output pins.
  - For a write operation, supply input data to be stored on the input pins. Enable the writing of the memory, which causes the input data to be moved to the address location.
- Disable the memory device.

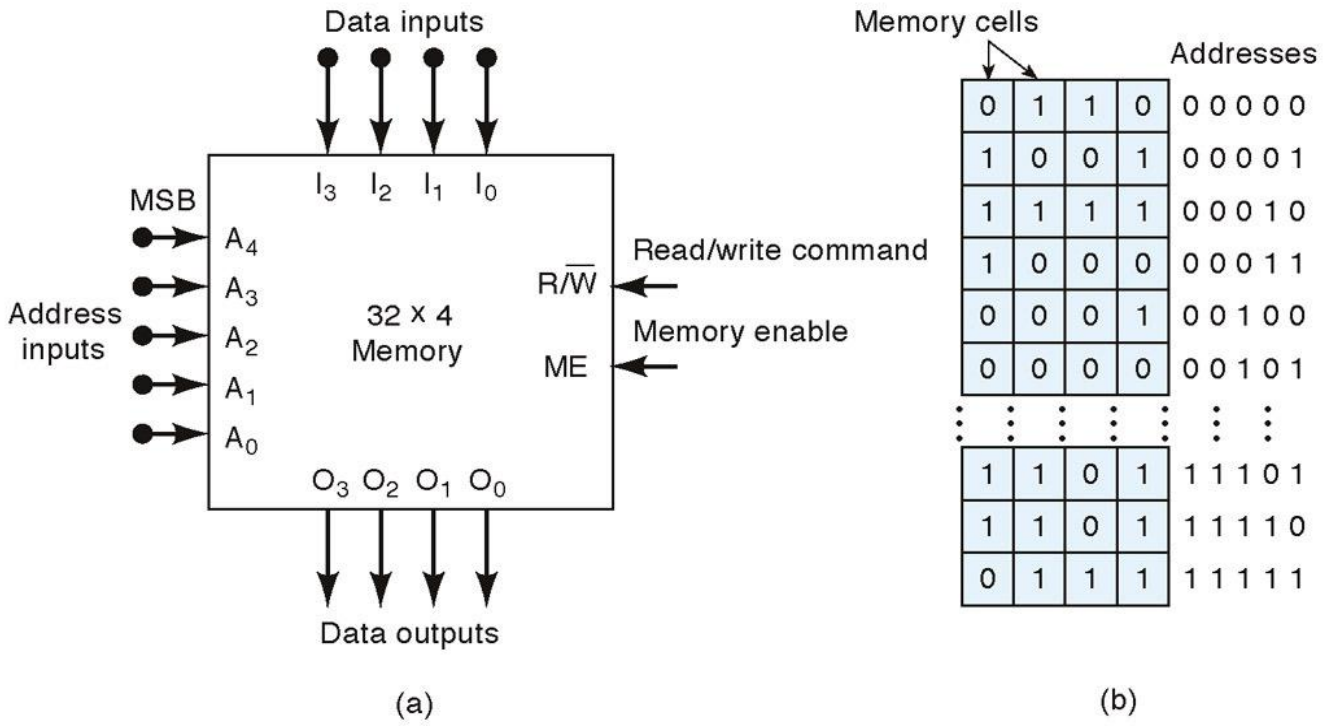
Accessing memory

The data inputs and outputs are usually connected to the same pins of the device. In some cases the address and data lines can be multiplexed (For example, video game cartridges).



Accessing memory

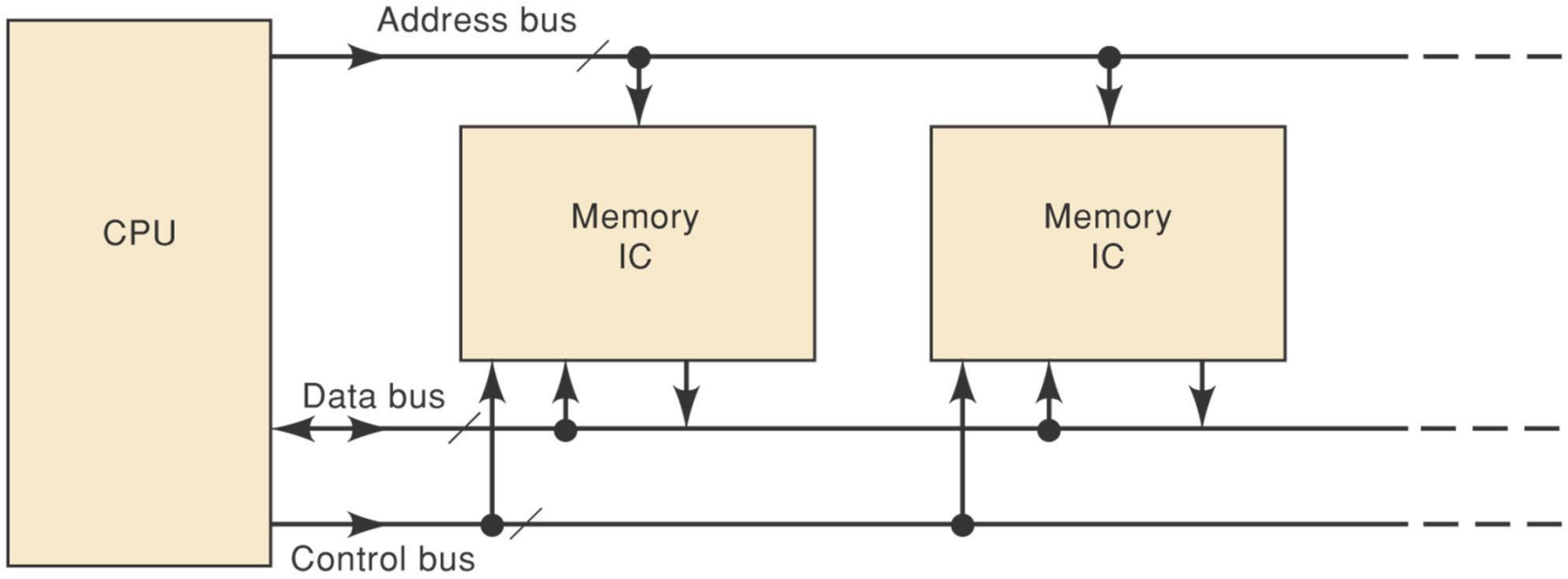
The data inputs and outputs are usually connected to the same pins of the device. In some cases the address and data lines can be multiplexed (For example, video game cartridges).



Accessing memory

Main memory (typically RAM) is interfaced to the CPU/microcontroller through:

- Address bus
- Data bus
- Control bus



Accessing memory

Read operation process:

1. CPU places the address of the memory location for data retrieval on the address bus.
2. An address decoder activates the enable input of the required memory device.
3. CPU activates the control signal lines for the read operation (high to low).
4. Memory ICs determine the location of the data being retrieved.
5. Memory IC places data from the memory location onto the data bus.
6. CPU latches the data and signals read cycle complete by changing the state of either the read or enable control lines (low to high).

Accessing memory

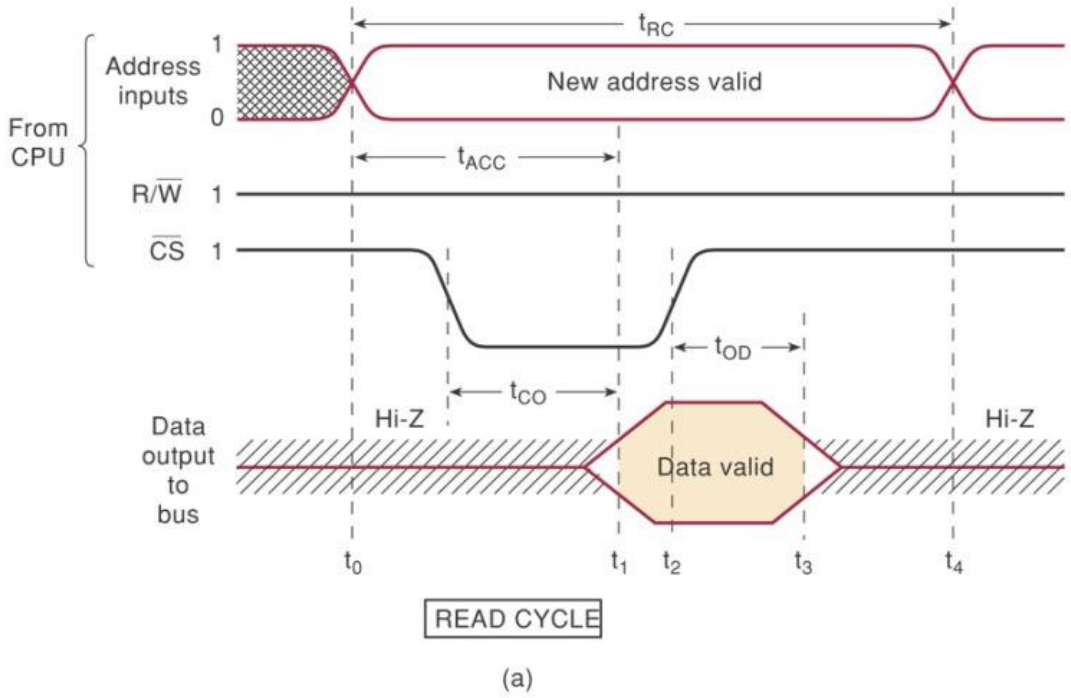
Write operation process:

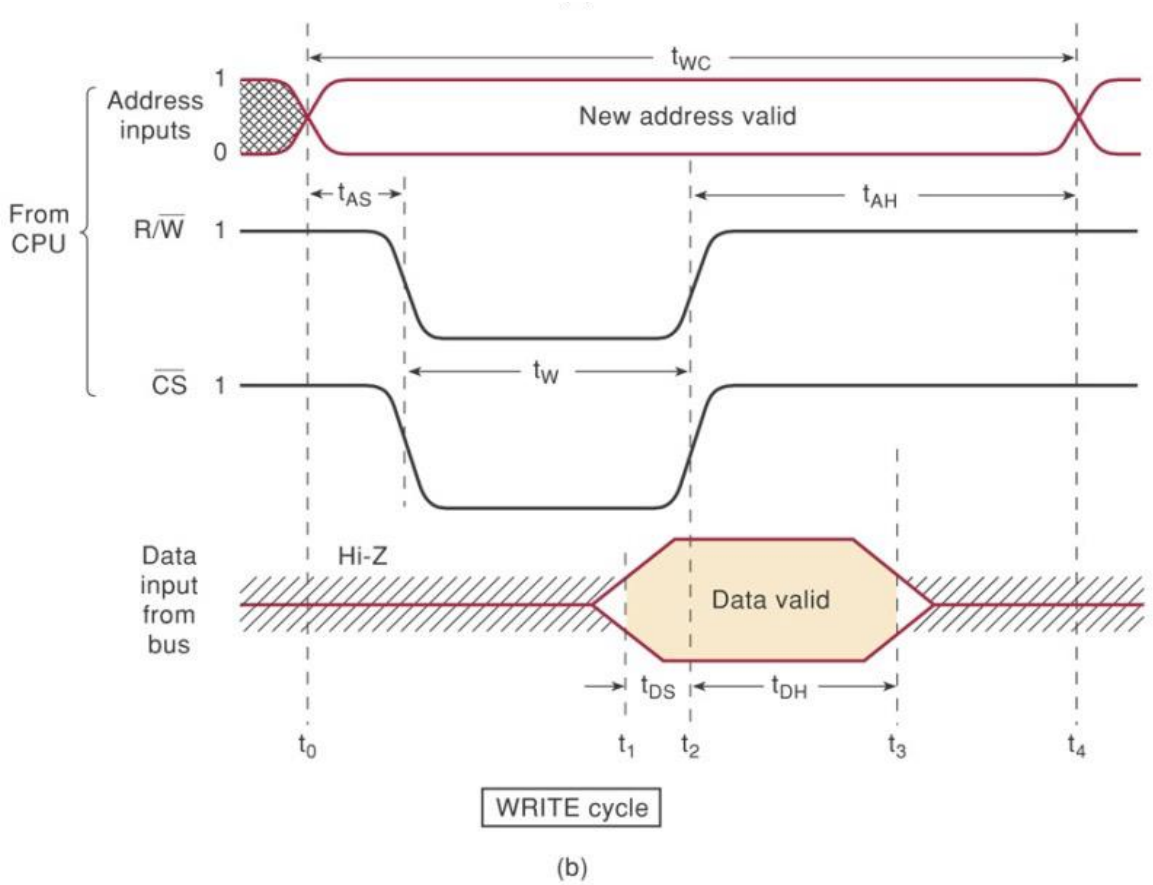
1. CPU places the address of the memory location for data storage on the address bus.
2. An address decoder activates the enable input of the required memory device.
3. The CPU places the data to be stored on the data bus.
4. CPU activates the control signal lines for the write operation (high to low).
5. Memory ICs determine the address location to store the data.
6. CPU signals to the memory when it should latch the data. This is normally done by changing the state of the write control signal (low to high).
7. Data on the data bus is transferred to the memory location.

Typical timing for static RAM:

(a) Read cycle.

(b) Write cycle.





Read Only Memory (ROM)

- A ROM is used to hold data that changes infrequently.

- It is non-volatile, data is not lost when the device is powered down.

Typical applications: Program or permanent data storage, boot programs, look-up tables.

Here we'll look at a couple of types of ROM and their differences.

Read Only Memory (ROM)

Control is similar to the RAM.

- Address inputs
- Data outputs
- Control input(s)

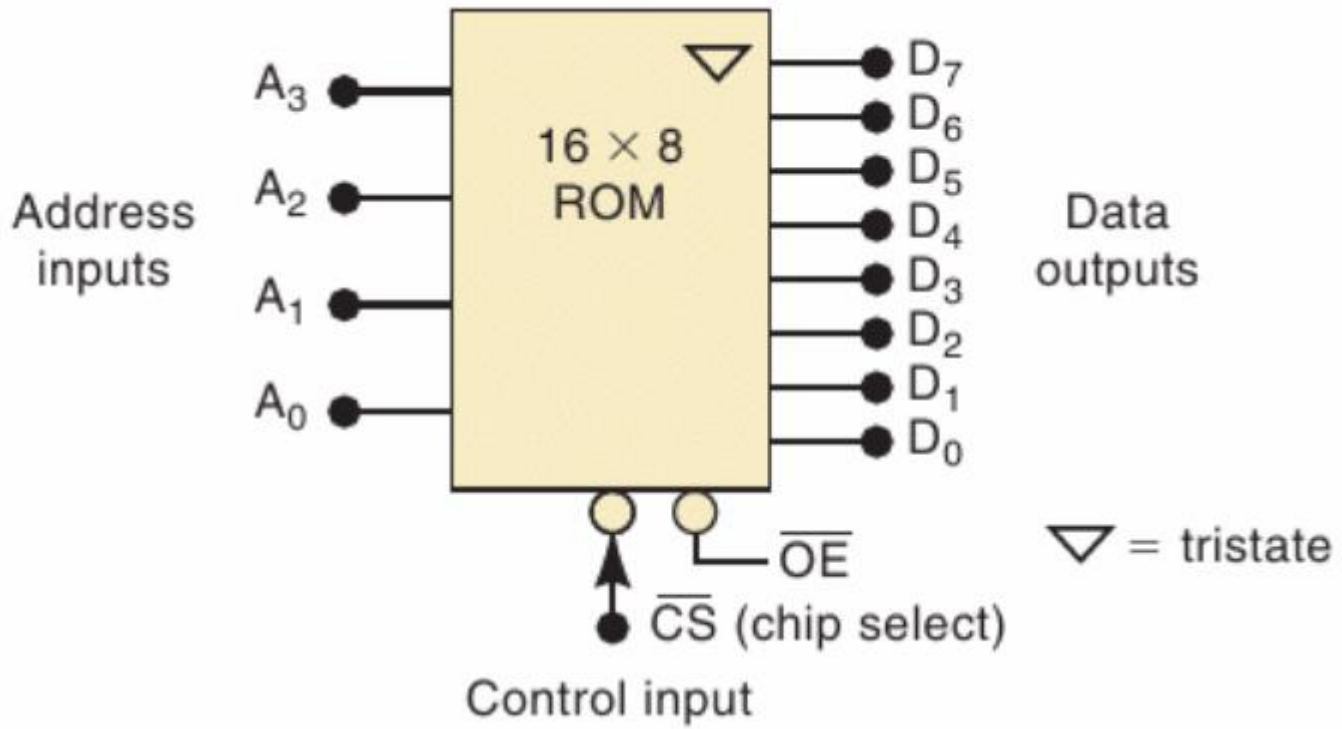
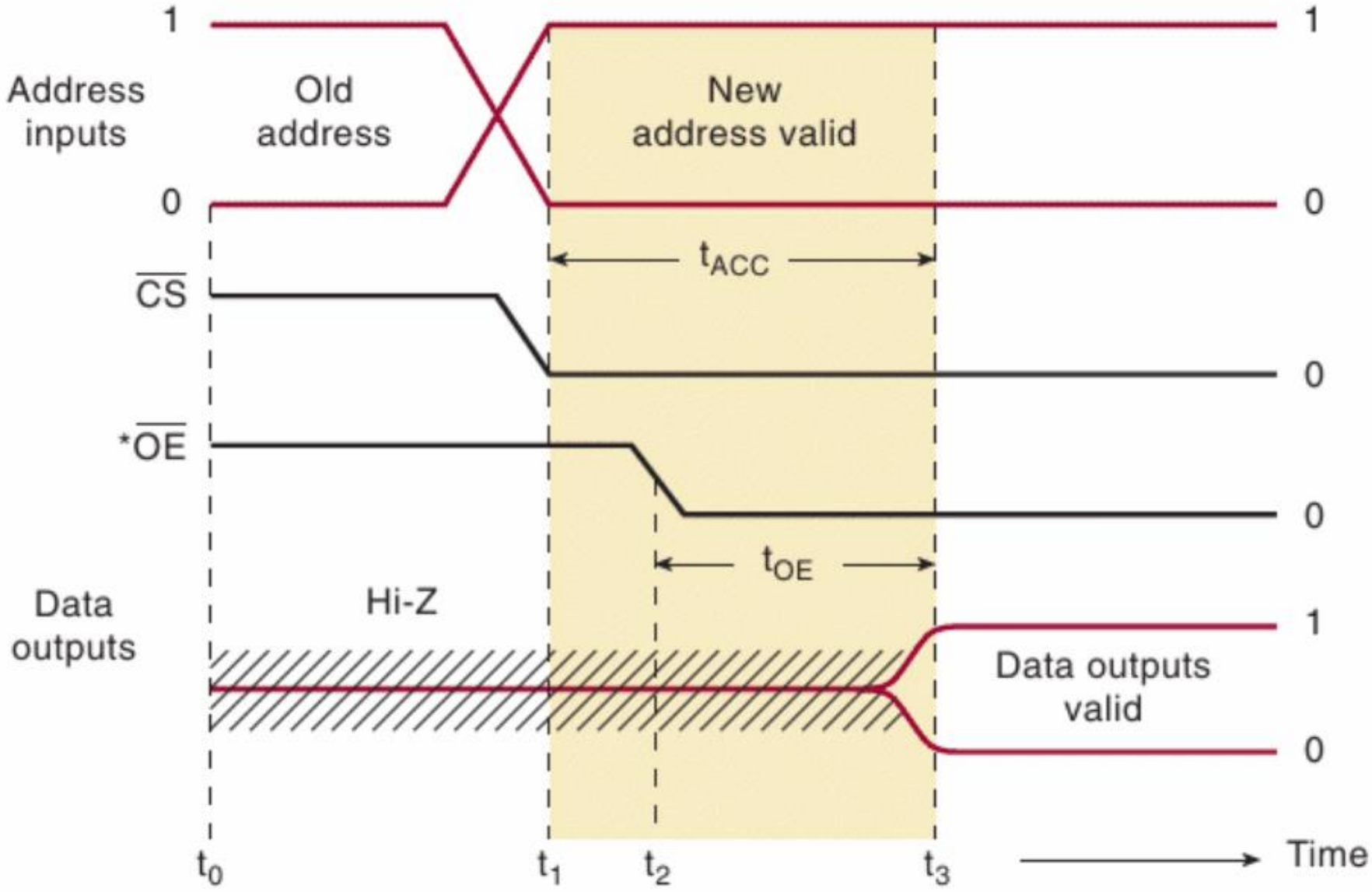


FIGURE 12-8 Typical timing for a ROM read operation.

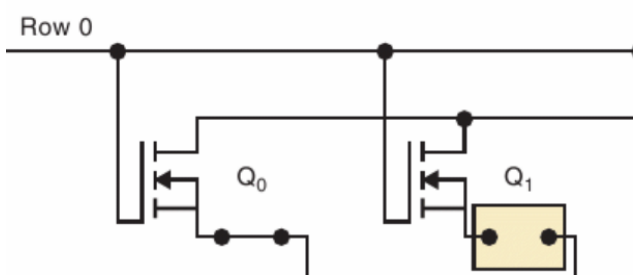


*t_{OE} is measured from the time $\overline{CS}$ and $\overline{OE}$ have both been asserted.

Mask programmed ROM (MROM)

A photolithographic mask (or pattern) establishes electrical interconnections.

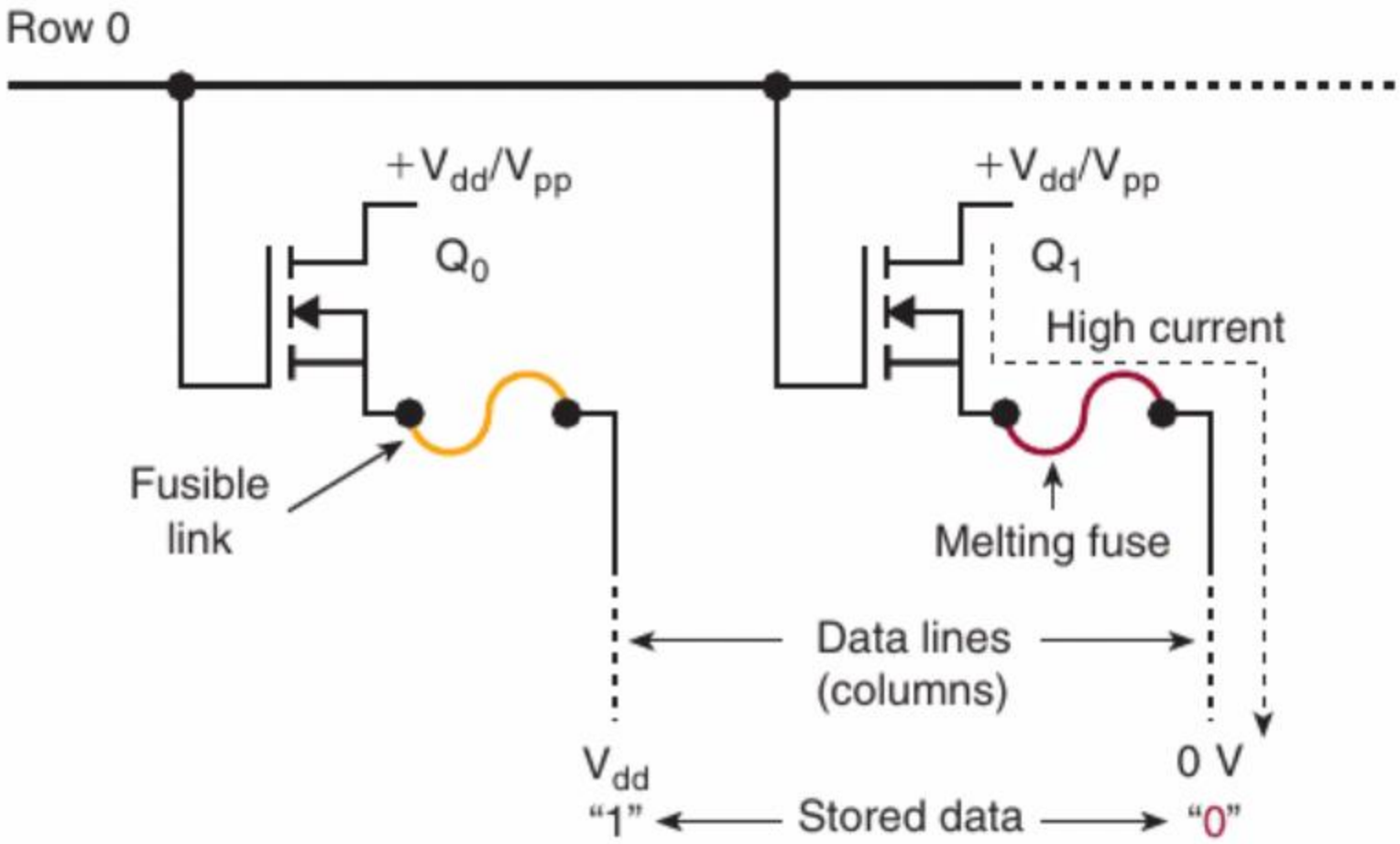
- The programming is done during IC manufacture. The ‘mask’ hardcodes the data into the MROM.
- Not user programmable.



- Economical only in high volume applications (mask and process cost).

Programmable ROM (PROM)

FIGURE 12-11 PROMs use fusible links that can be selectively blown open by the user to program a logic 0 into a cell.

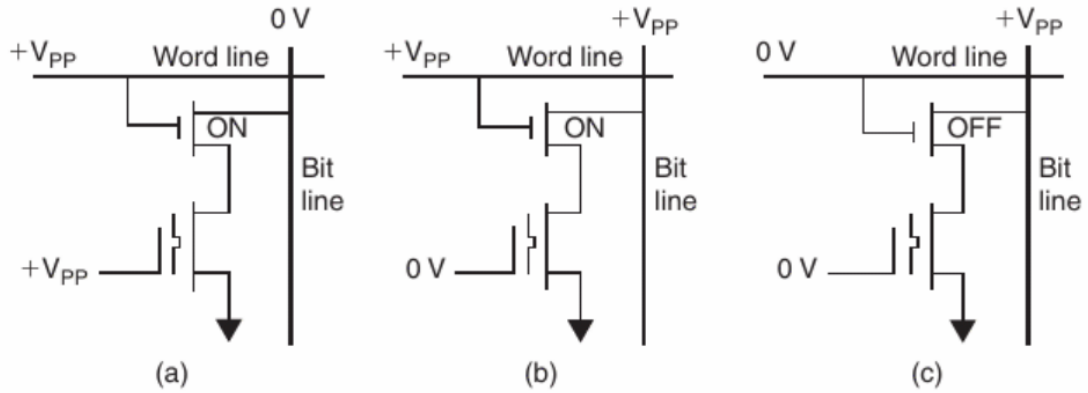


Electrically Erasable Programmable ROM (EEPROM)

Uses a special transistor where a permanent charge is optionally stored, that alters the behaviour of the transistor.

Voltage, often an elevated voltage, is used to charge or discharge the memory element.

FIGURE 12-13 (a) Erasing/precharging a cell; (b) writing a '0'; (c) writing a '1.'



Individual bytes (words) can be erased, but erasure is often done one sector (group of bytes) at a time.

Flash Memory (EEPROM variants)

Flash memory allows rapid in-circuit reprogramming of individual blocks. Combines the best features of EEPROM – all flash memory is an electrical variant of EEPROM.

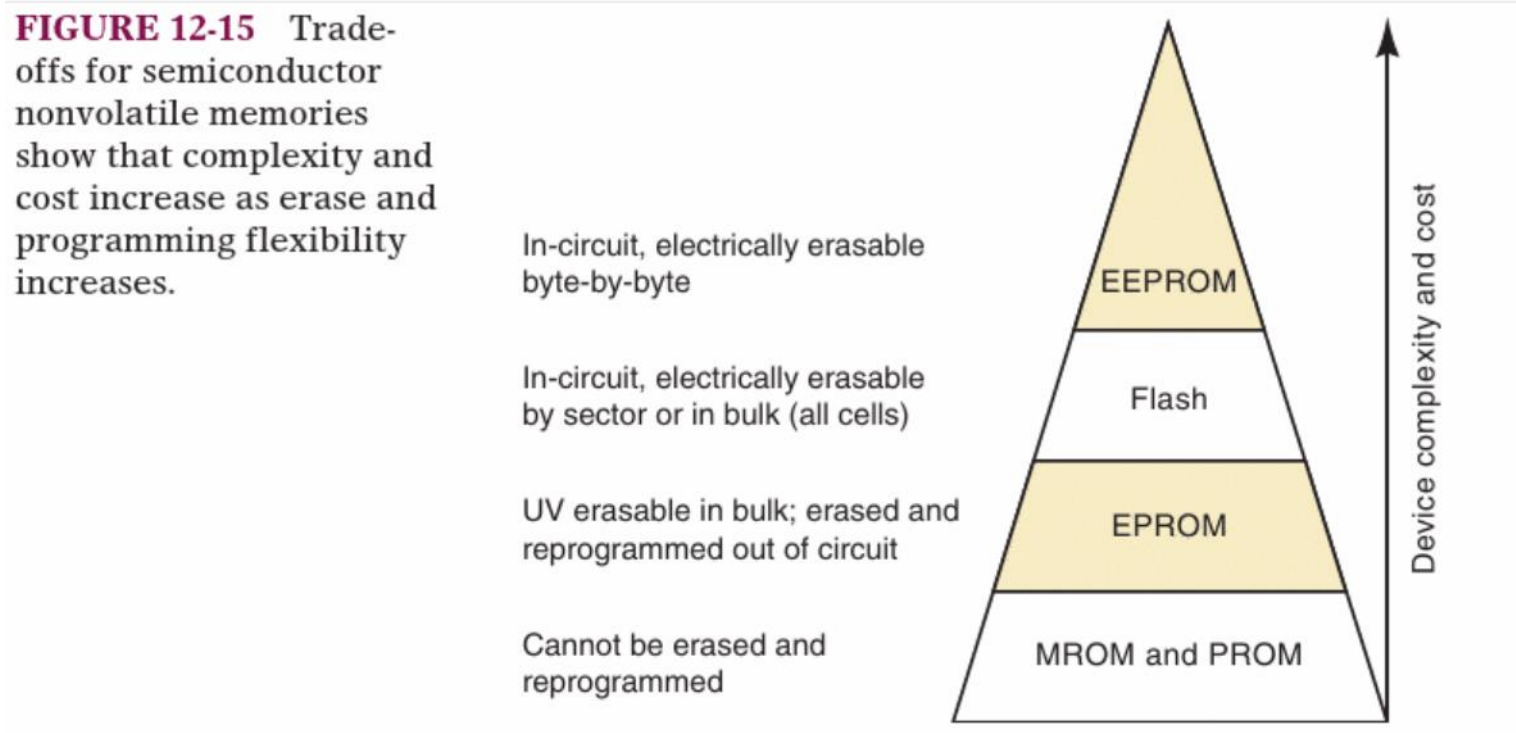


FIGURE 12-15 Trade-offs for semiconductor nonvolatile memories show that complexity and cost increase as erase and programming flexibility increases.

Optical ROM

- Light is used to store binary data

- Large quantities of data can be stored, good for back-up applications.

- DVDs, Blu-Rays, etc.

- Plasmonic assisted drives offer higher density storage.

    Multiple colours multiple bits per cell

    However, they’re difficult to fabricate.

ROM Applications

- Embedded microcontroller program memory
- Data transfer and portability
- Data tables
- Data converter
- Function generator
- Auxiliary storage

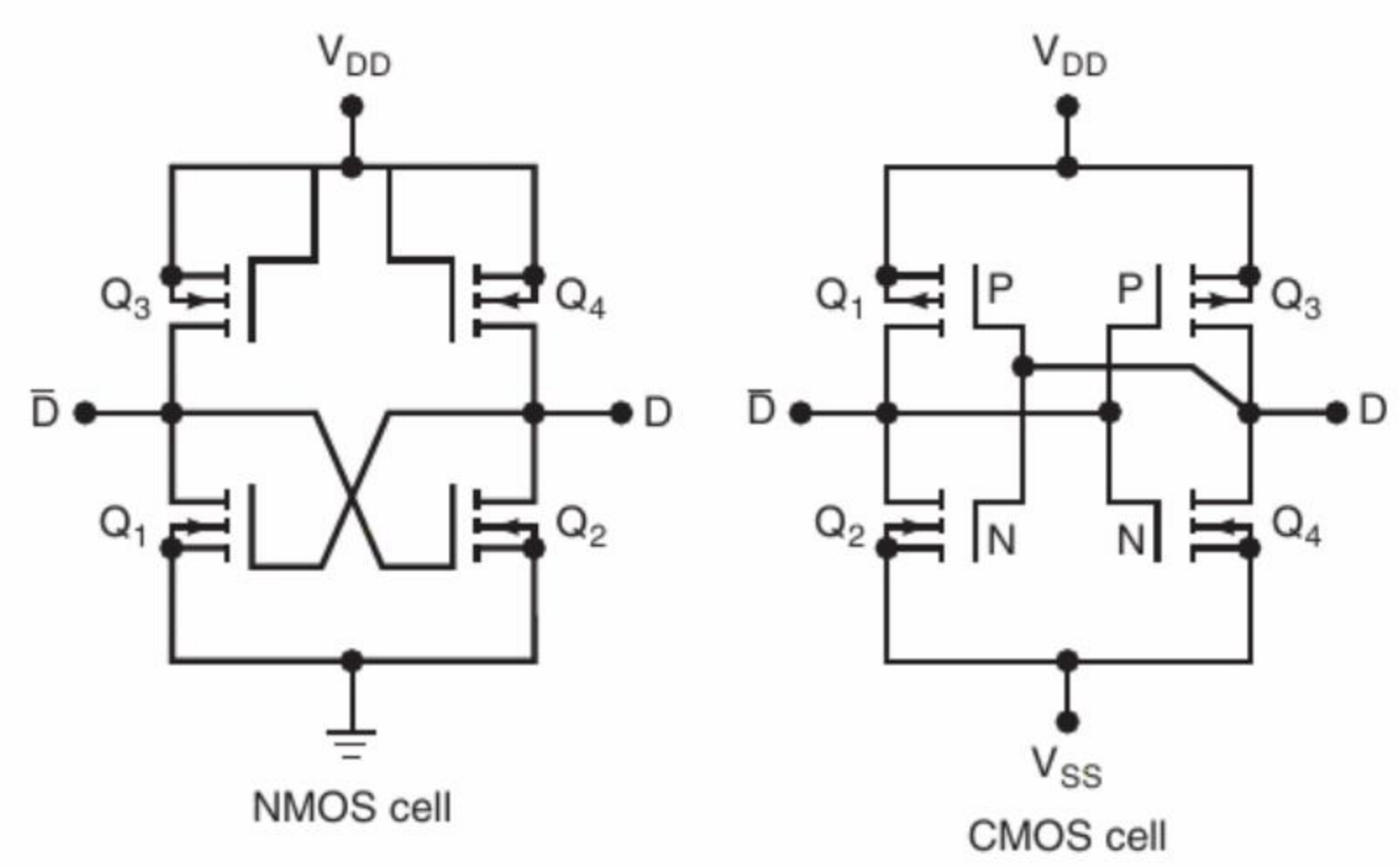
Semiconductor RAM

- Random Access Memory (RAM) all memory locations accessible at any time.
- Used for temporary storage
- Fast read and write times are necessary.
- RAM is volatile, data disappears when the power goes off.
- Many of the concepts that apply to ROM apply to RAM as well.

Static RAM (SRAM)

- Stores data as long as power is applied.
- Usually very fast, used as L2 cache memory.
- Flip-flop type storage element.
- Timings differ for read and write cycle.

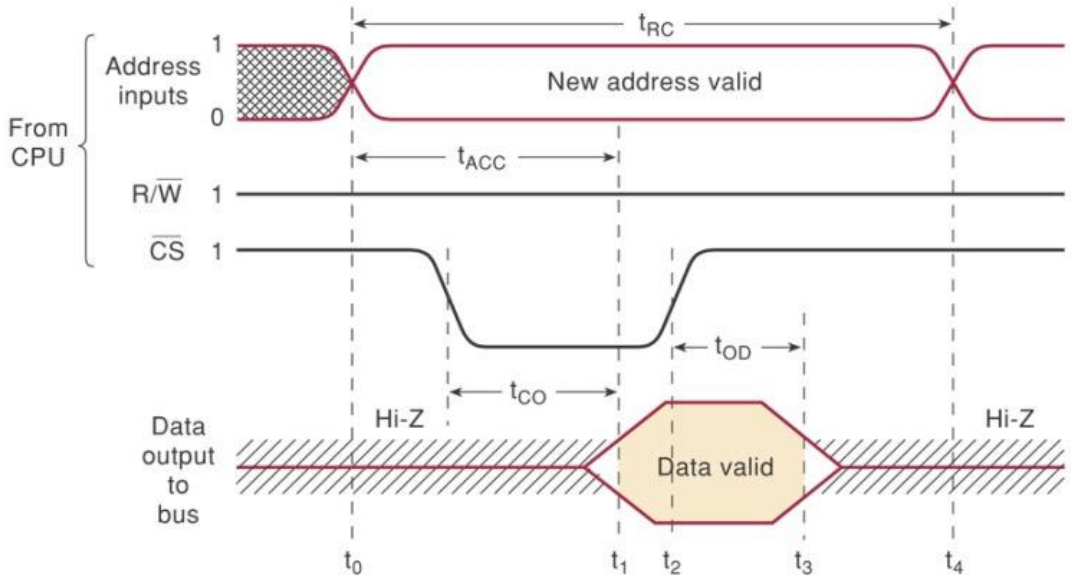
Typical NMOS and CMOS Static-RAM cells

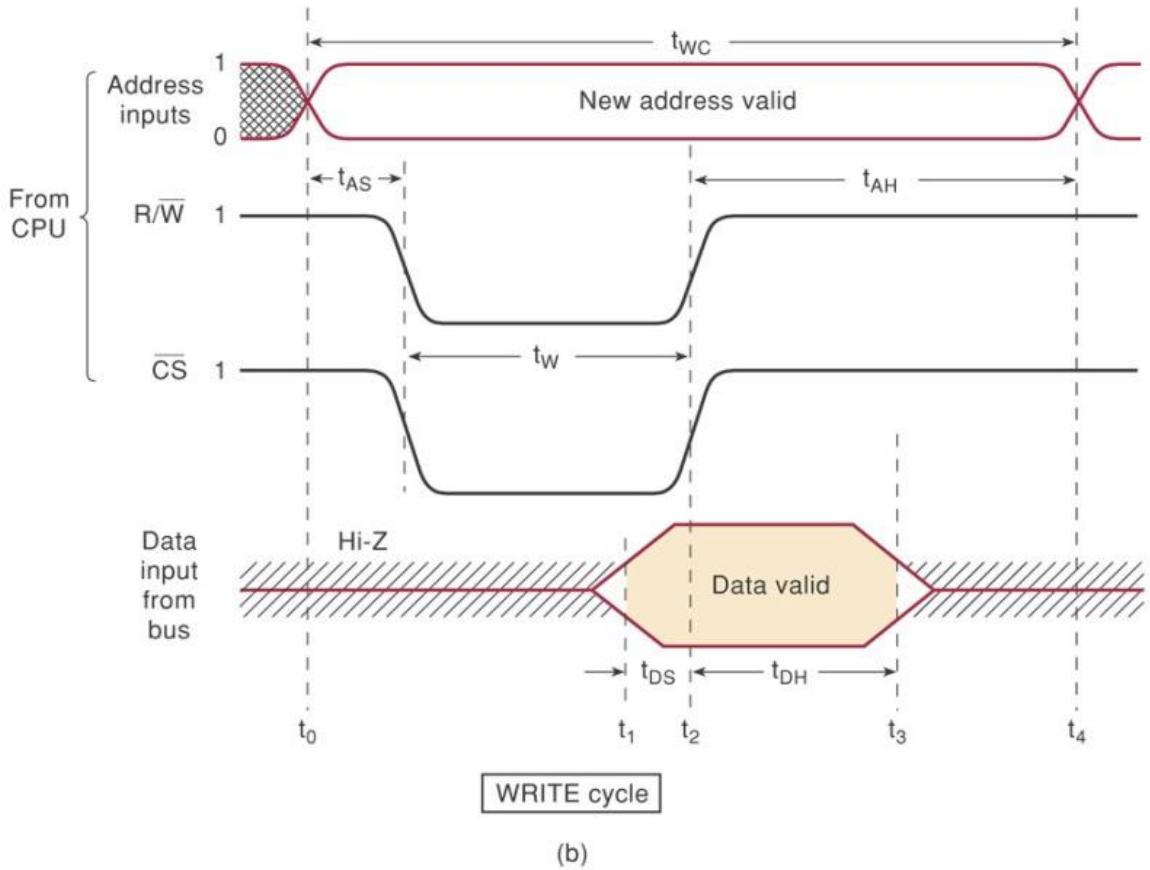


Typical timing for static RAM:

(a) Read cycle.

(b) Write cycle.





Dynamic RAM (DRAM)

- High capacity
- Low power requirements
- Moderate speed
- Small capacitors are used to store data (but capacitors discharge…)
- Must be periodically refreshed by DRAM controller
- Used for main internal memory in PCs
  - Double Data Rate - Synchronous DRAM (DDR-SDRAM)

DRAM Structure and Operation

- An array of cells with unique row and column positions

Read and write operations can be explained using the simplified representation below:

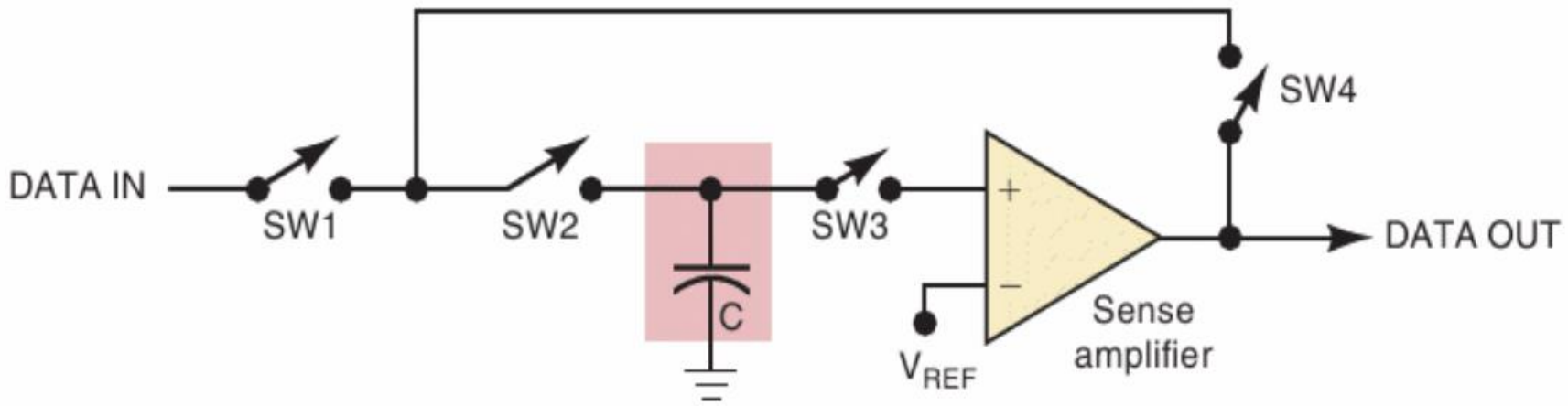
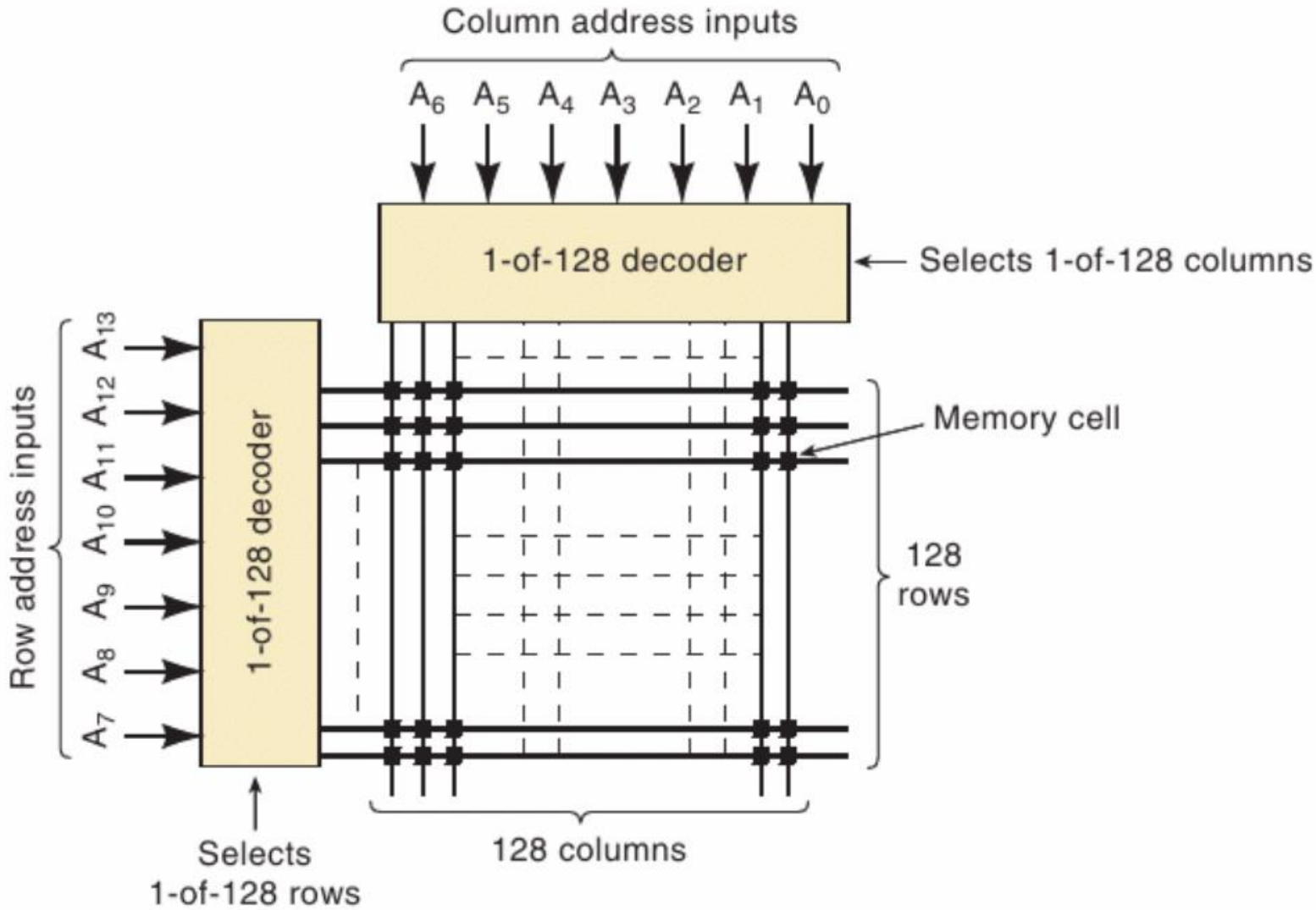


FIGURE 12-25
Symbolic representation of a dynamic memory cell. During a WRITE operation, semiconductor switches SW1 and SW2 are closed. During a read operation, all switches are closed except SW1.

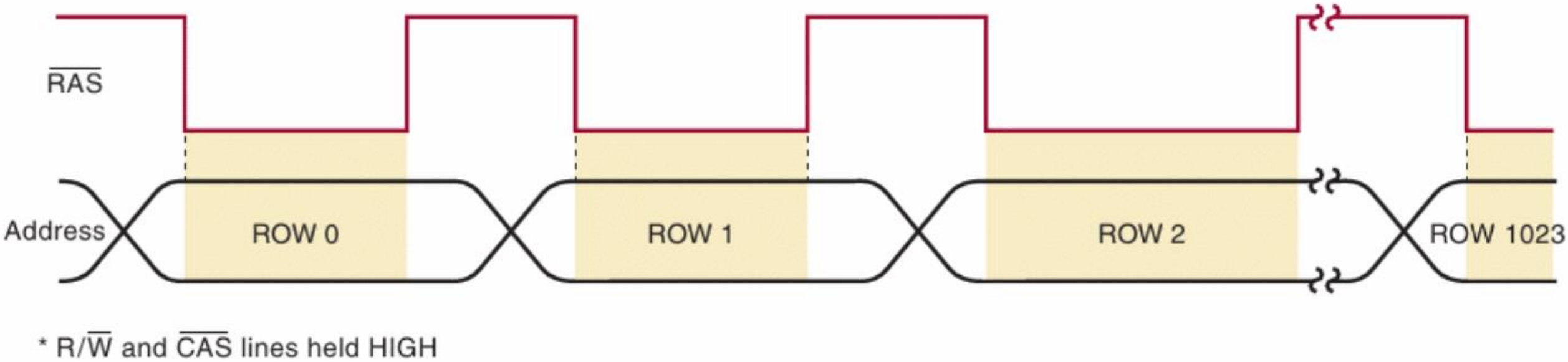
Dynamic RAM Structure and Operation

FIGURE 12-24 Cell arrangement in a 16K × 1 dynamic RAM.



DRAM Refreshing

Different ways to refresh DRAM, but all methods must periodically recharge the capacitor on each cell. This is managed by a DRAM controller. As an example: Row Address Select ($\overline{RAS}$) - only refresh.



* R/$\overline{W}$ and $\overline{CAS}$ lines held HIGH

FIGURE 12-31 The *RAS*-only refresh method uses only the *RAS* signal to load the row address into the DRAM to refresh all cells in that row. The *RAS*-only refresh can be used to perform a burst refresh as shown. A refresh counter supplies the sequential row addresses from row 0 to row 1023.

DRAM Read Cycle

- t_0: *MUX* is driven LOW to apply the row address bits (A_8 to A_{15}) to the DRAM address inputs.
- t_1: *RAS* is driven LOW to load the row address into the DRAM.
- t_2: *MUX* goes HIGH to place the column address (A_0 to A_7) at the DRAM address inputs.
- t_3: *CAS* goes LOW to load the column address into the DRAM.
- t_4: The DRAM responds by placing valid data from the selected memory cell onto the DATA OUT line.
- t_5: *MUX*, $\overline{RAS}$, $\overline{CAS}$, and DATA OUT return to their initial states.

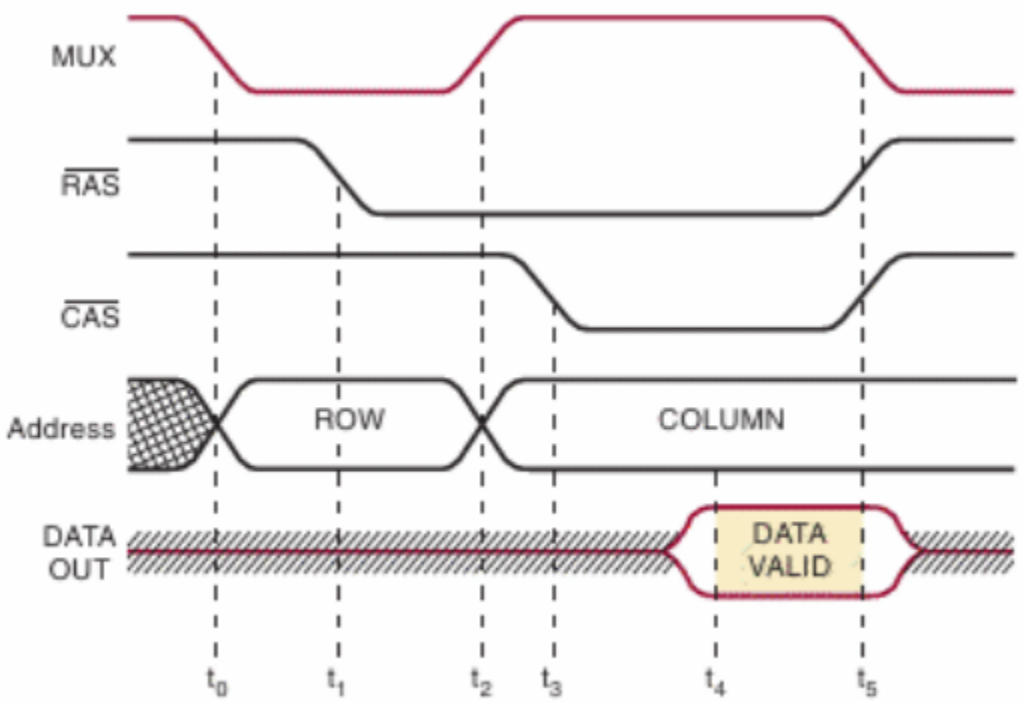


FIGURE 12-29 Signal activity for a read operation on a dynamic RAM. The $\overline{WE}$ input (not shown) is assumed to be HIGH.

DRAM Write Cycle

- t_0: The LOW at *MUX* places the row address at the DRAM inputs.
- t_1: The NGT at $\overline{RAS}$ loads the row address into the DRAM.
- t_2: *MUX* goes HIGH to place the column address at the DRAM inputs.
- t_3: The NGT at $\overline{CAS}$ loads the column address into the DRAM.
- t_4: Data to be written are placed on the DATA IN line.
- t_5: $\overline{WE}$ is pulsed LOW to write the data into the selected cell.
- t_6: Input data are removed from DATA IN.
- t_7: *MUX*, $\overline{RAS}$, $\overline{CAS}$, and $\overline{WE}$ are returned to their initial states.

FIGURE 12-30 Signal activity for a write operation on a dynamic RAM.