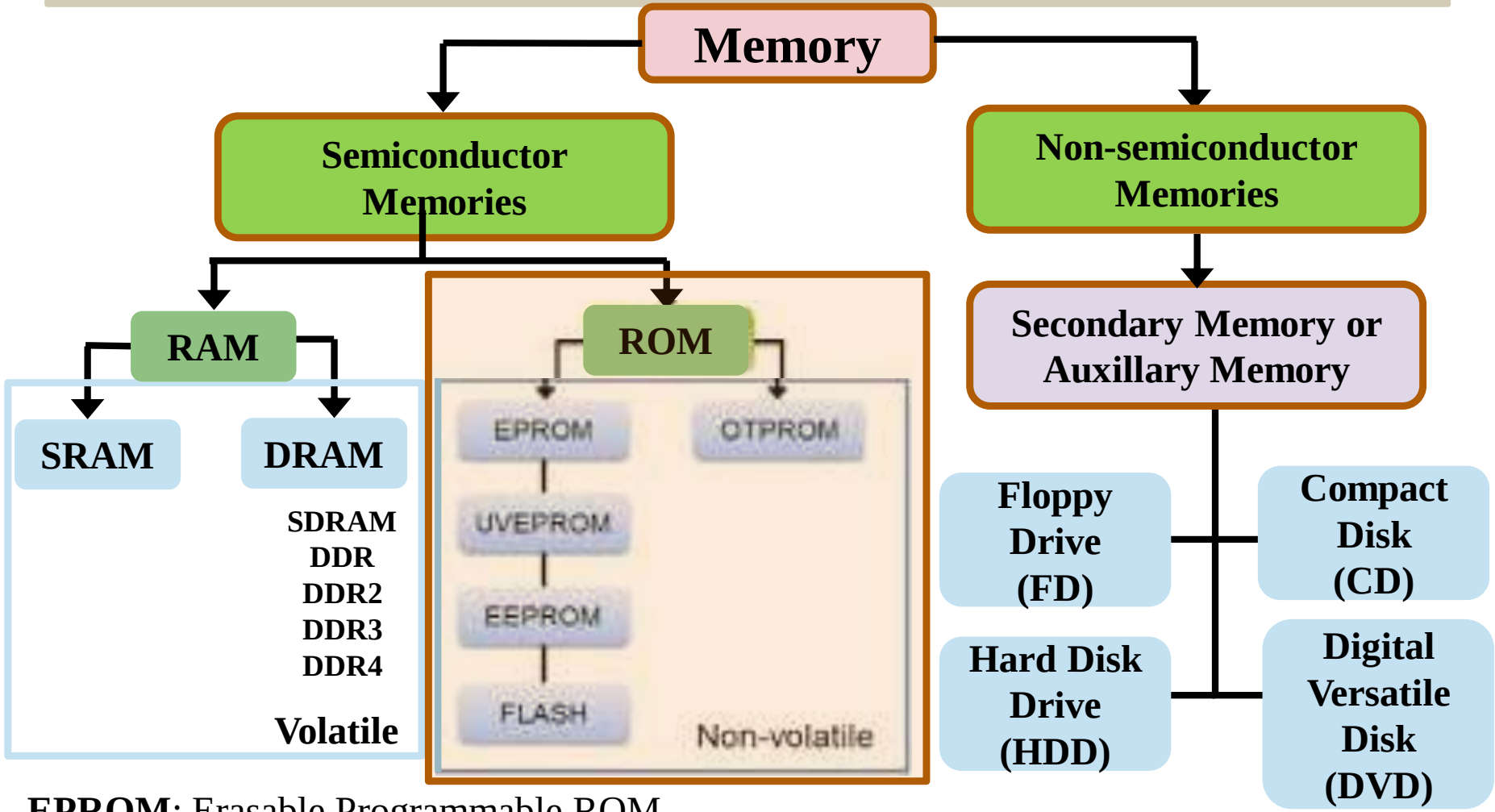
Embedded Systems -3

ES 3: Focus

- Introduction to different Memory Types
  - Volatile and Non-volatile Memory
  - Semiconductor Vs Non-semiconductor Memories
- Read Only Memories (ROM)
  - ROM and PROM
  - EPROM and EEPROM
- Random Access Memory (RAM)
  - SRAM Memory Design
  - Dynamic Random Access Memory (DRAM)

Different Memory Types

Memory Classification



EPROM: Erasable Programmable ROM

UVEPROM: Ultra-violet Erasable PROM

EEPROM: Electrically Erasable PROM

FLASH: Flash Memories

OTPROM: One Time PROM

Volatile and Non-volatile Memory

Volatile:

- **Contents** stored in the **memory** are **lost** when the **power** is **withdrawn**
- The data in it will be cleared as soon as the laptop is shutdown
- The advantage of RAM is that, it can be directly accessed by CPU, much faster than non-volatile memories

Non-volatile:

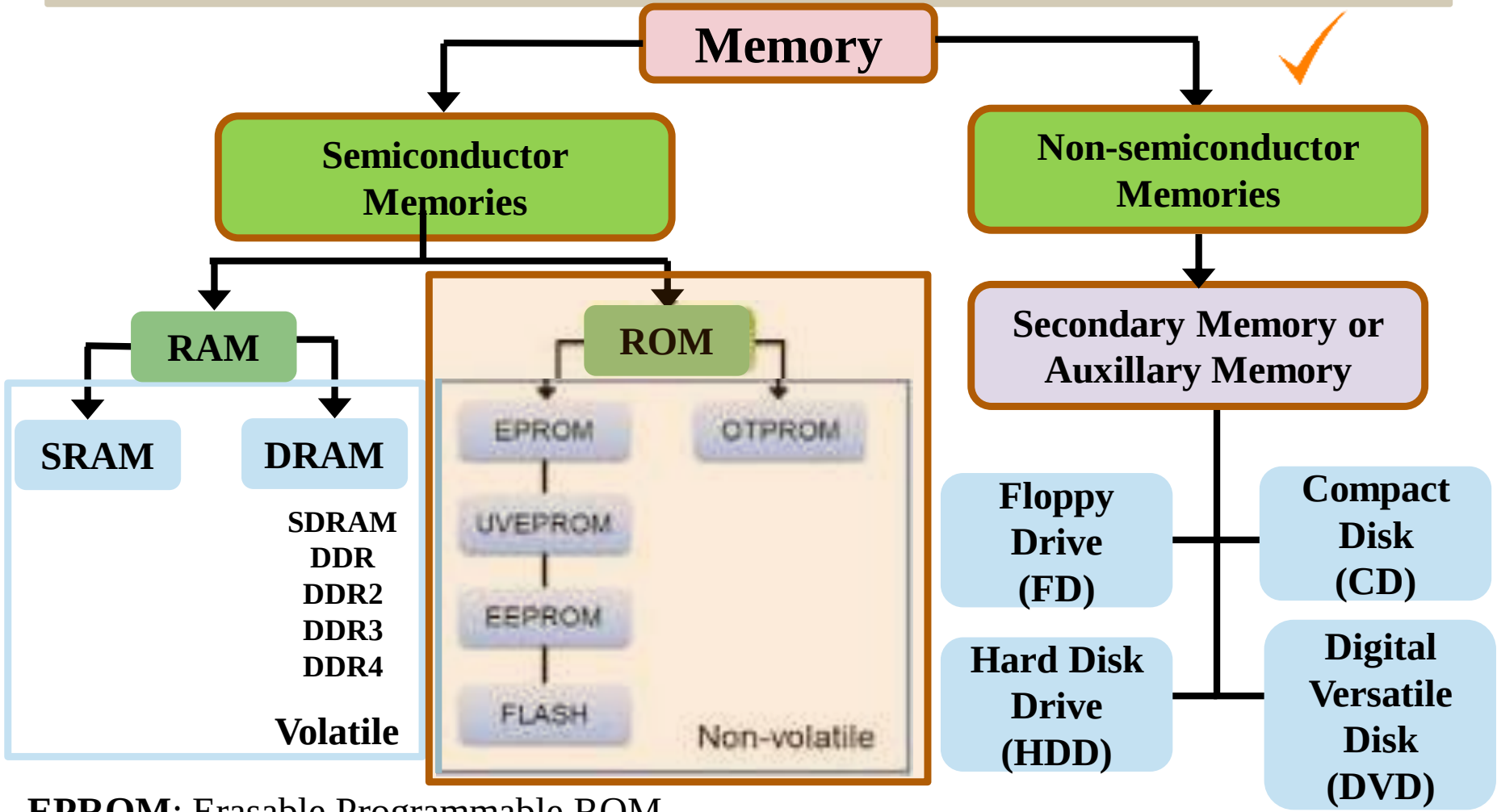
- **Retains** the **contents stored** even if **power** is **switched off**
- **Non-volatile memory** is typically used for the task of **secondary storage**, or **long-term** persistent **storage**.

Semiconductor Vs Non-semiconductor Memories

Semiconductor Memories	Non-semiconductor Memories
• **Semiconductor technologies** are **used** for building it • **Random access** capability • **Mostly Volatile** in nature ◦ **Flash drives** are an exception, which are **non-volatile** and **erasable** • **Access time** is much **lower** ◦ It take less time to read from it • **Closer to CPU**, whereas **SRAM** is built within the CPU chip itself • The size is in terms of **G Bytes** (**10^9**)	• **Magnetic** and **optical** technologies used • Normally **sequential access** • **Non-volatile** • Has **higher access time** ◦ It takes more time to read from it • **Farther** from **CPU** and built as a separate unit • Can be of much larger sizes, in terms of **Tera Bytes** (**10^{12}**) and more

Read-Only Memories (ROM)

Memory Classification



EPROM: Erasable Programmable ROM

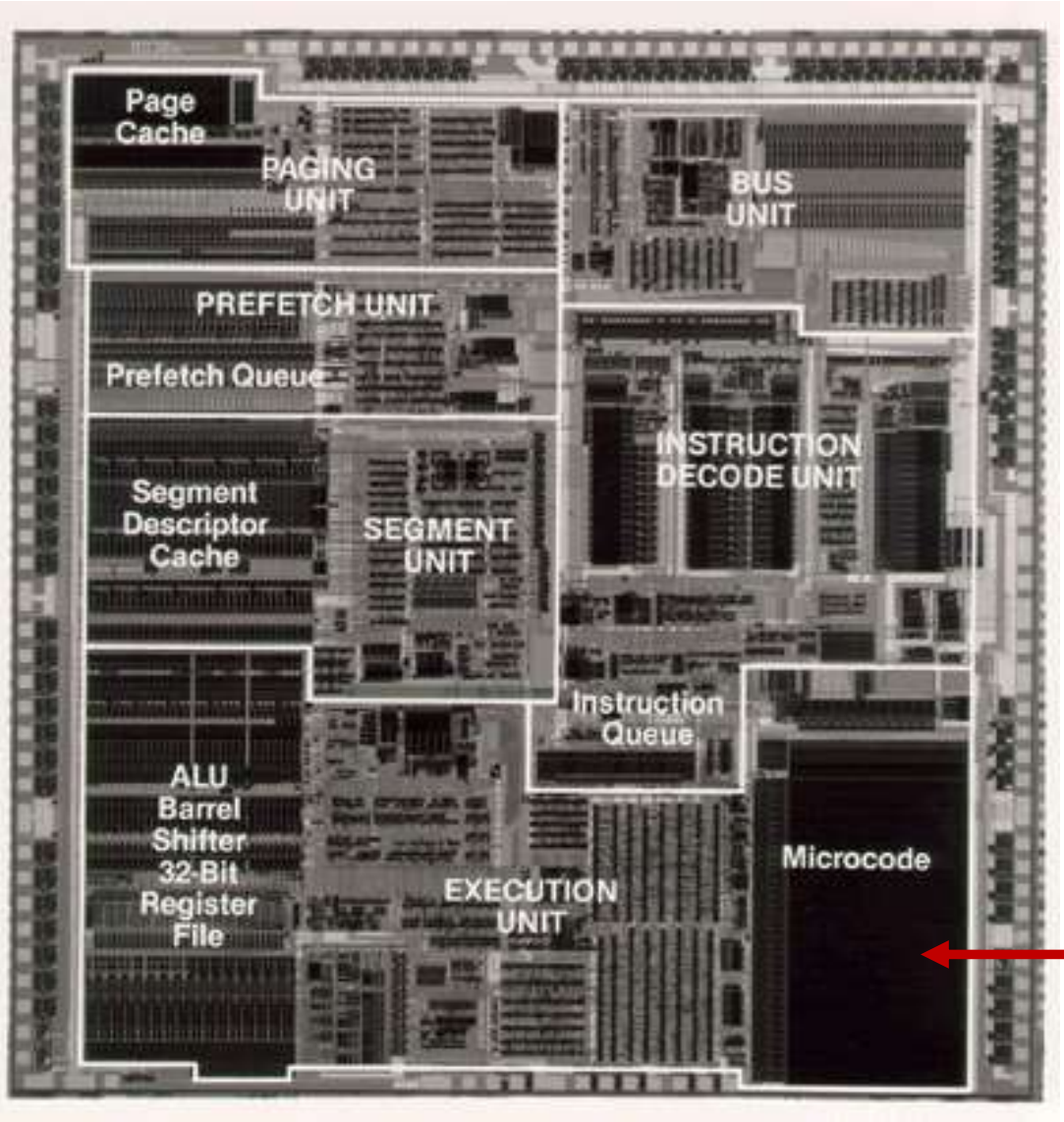
UVEPROM: Ultra-violet Erasable PROM

EEPROM: Electrically Erasable PROM

FLASH: Flash Memories

OTPROM: One Time PROM

Inside a Microprocessor: Sample

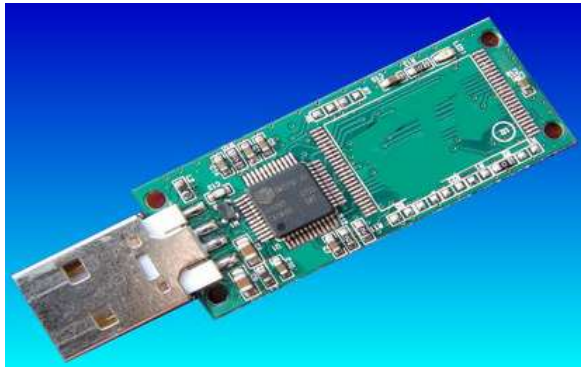


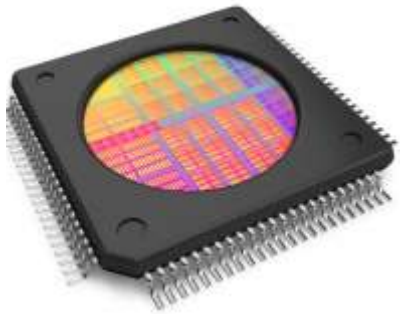
Types of ROM Memories

UV Erasable PROM

8 pin, 2-wire Serial Interface 64kx8 bits size

Electrically Erasable PROM (EEPROM)





Flash memory: USB Memory Stick

ROM and PROM

What is a ROM?

- **ROM** is a Programmable Logic Device (**PLD**)
- **Binary** information is **hard-wired** inside a **ROM**, during the manufacturing of the device
  - This process is referred to as **programming** the **device**
  - Process is a set of **hardware connections** to store binary values
  - Once it is done it cannot be modified

Types of ROM: 1. ROM

- Here, the data is **wired** into the chip during the **HW fabrication process**
- As the name suggests the **first versions** of **ROM** were **read-only**
  - While it was **possible to read** a **ROM**, it was **not possible** to **update** them or **write** any **new data** into them
- An important **application** of **ROMs** is to **store microprograms** or **microcode** inside **microprocessors**
  - Which we studied in the **CPU design**, to **store** a sequence of **µ-ops** to **execute** various **assembly instructions** in a CPU

Types of ROM: 2. PROM

- Like the ROM, the **PROM** is **non-volatile** and may be **written** into **only once**
- For the **PROM** the **writing process** is done **electrically**
  - Not wired into the chip during manufacture, like in ROM
- Writing (once) may be performed by the **supplier** or the **user** at a time later than the original chip fabrication
  - The contents of PROM cannot be written more than once
- Special equipment is required for the writing or "programming" process
- They are also called **One Time Programmable**
  - **OTP NVM** (non-volatile memories)
- PROMs provide flexibility and convenience and expensive than ROM
- ROM remains attractive for high volume production runs

Quiz 1: ROM Vs PROM – Yes or No

- **ROM**: Data is wired into it during manufacture: **YES**
- **ROM**: Writing into this is possible by the user: **NO**
- **PROM**: Writing is same as programming the chip: **YES**
- **PROM-OTP**: Cannot be written more than once: **YES**
- **PROM**: Programming is done during manufacture: **NO**
- **PROM**: It is more expensive than ROM: **YES**
- **ROM**: It can be used for storing µ-ops in CPUs: **YES**

UV Erasable PROM

160D0WQ
EEPROM Chip

EPROM and EEPROM

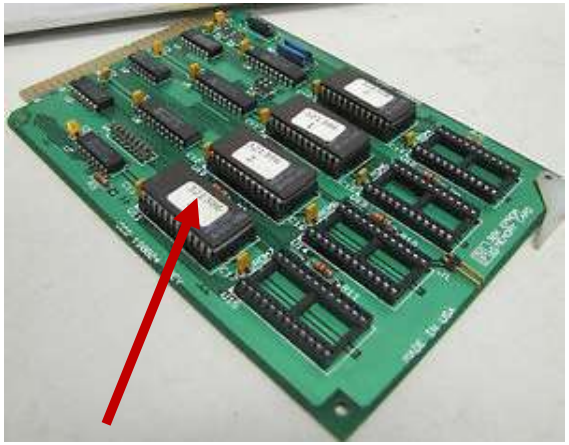
EPROM: Programming Sequence

1. ERASE the entire chip

EPROM programmer Connected to a **PC**

2. PROGRAM it

3. USE it



- Keep the **EPROM** in an **UV eraser** for **20 minutes** to **erase all** the contents of it. After erasing, the whole memory will be **filled** with **1s**
- Then, using a **PROM programmer, write specific contents** into **PROM**
- **Put** it back into the **socket** on the **board** to **use** it as **normal memory chip** to read the program and /or data from it

PROMs are **covered** with **stickers** to **avoid accidental erasing** due to **exposure to light**

Types of ROM: 3. EPROM

- **Erasable programmable read-only memory** is **optically (UV) erasable**
- EPROMs are **read** and **written electrically**, similar to PROM (used as normal memories on the circuit board)

- However, before a write operation, all the storage cells must be erased to the same initial state (**all 1s**) by **exposing** the packaged **chip** to **ultraviolet radiation**
- **Erasure** is performed by **shining UV** through the window on the chip for a period of around **20 minutes**
- Thus, **EPROM** can be **erased multiple times**
- **EPROM** is **more expensive** than **PROM** and has **multiple update** capability

Types of ROM: 4. EEPROM

- A more attractive form of ROM is **electrically erasable programmable read-only memory** (**EEPROM**)
- This is a read-mostly memory that can be written into at any time, without erasing the entire prior contents
  - Updating only a byte or a set of bytes is possible
- The **write operation** takes considerably **longer time** than the read operations, in the order of **several 100 µsecs**
- EEPROM combines the advantage of non-volatility with the flexibility of being updatable in place (in-circuit)
  - Using ordinary bus control, address and data lines
- EEPROM is more expensive than EPROM and less dense
  - i.e., EEPROM supports fewer bits per chip than EPROM

Quiz 2a: EPROM Vs EEPROM – Yes or No

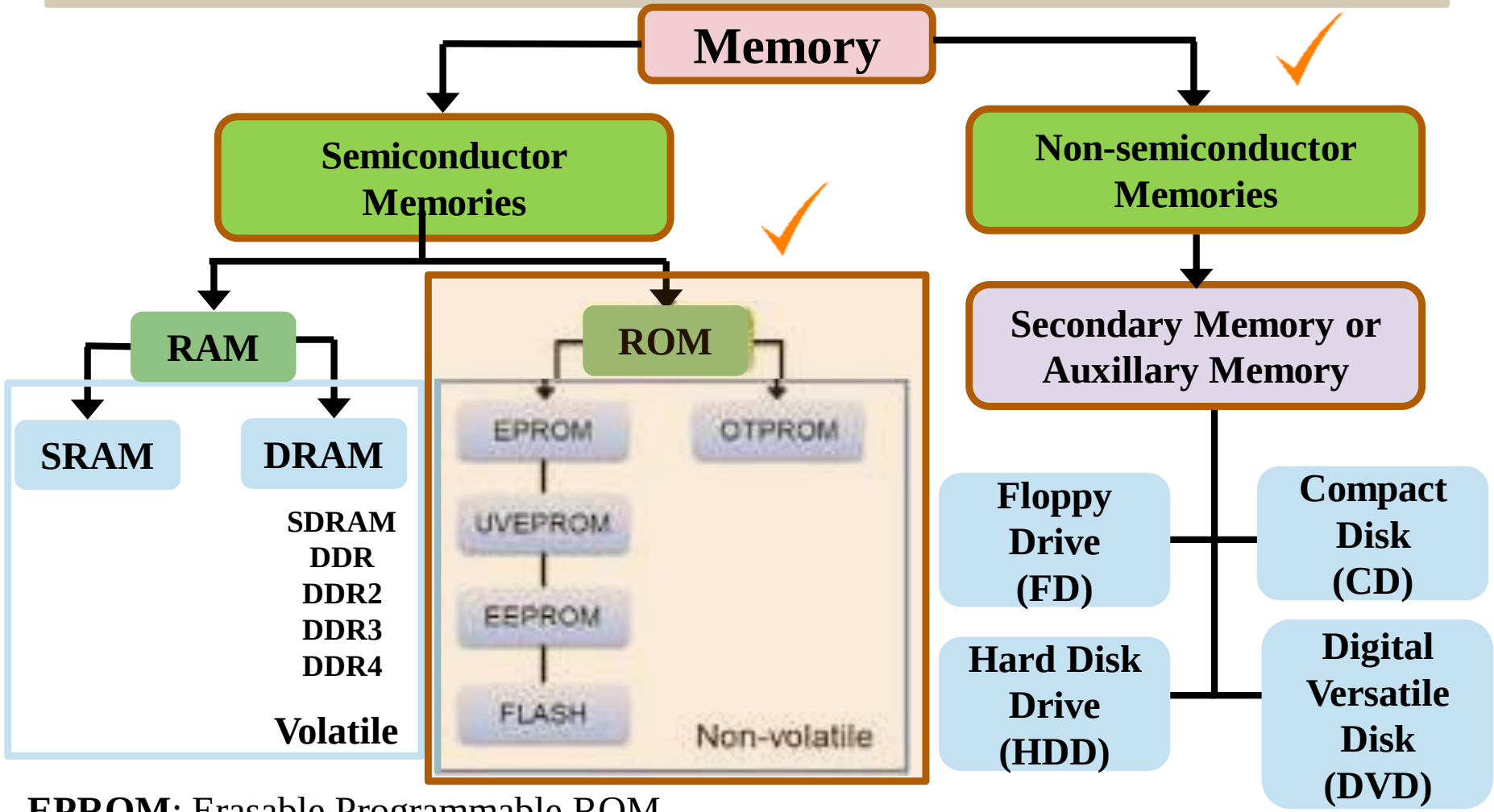
- **EPROM**: UV is used to erase the contents: **YES**
- **EPROM**: It is possible to erase part of the chip: **NO**
- **EEPROM**: Erasing is done optically: **NO**
- **EPROM**: Reading and writing are done electrically: **YES**
- **EPROM**: Can be written only after erasing all data: **YES**
- **EEPROM**: Erasing is easier than EPROM: **YES**
- **EEPROM**: Writing is faster than reading from it: **NO**
- **EEPROM**: It is more expensive than EPROM: **YES**
- In general **writing** is **slower than reading** from **them** on **all types of ROMs**: **YES**

Quiz 2b: EPROM Vs EEPROM – Yes or No

- **EPROM**: Can be erased without removing it from the circuit where it is used as memory: **NO**
- **EPROM**: Programming them actually involves only writing zeros into specific locations: **YES**

Random Access Memory (RAM)

Memory Classification



EPROM: Erasable Programmable ROM

UVEPROM: Ultra-violet Erasable PROM

EEPROM: Electrically Erasable PROM

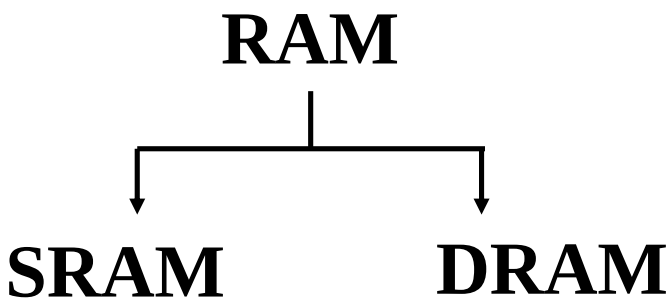
FLASH: Flash Memories

OTPROM: One Time PROM

What is Random Access Memory (RAM)?

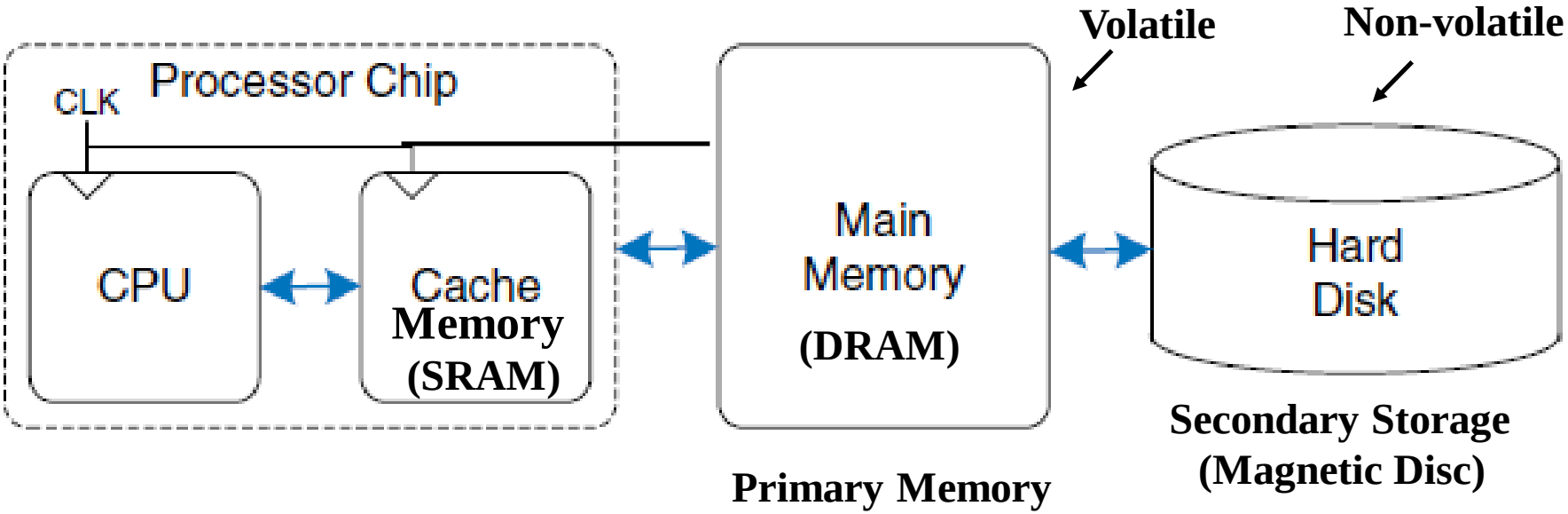
- The time taken to **read** or **write** a data into **any location** of the **RAM** takes the **same amount of time**
  - It **does not vary** with respect to the **location of the data**
- All locations of a RAM memory takes the same amount of time for read/write
- In contrast, with other direct-access data storage media such as hard disks, CD, DVD and the older magnetic tapes and drum memory,
  - The time required to read and write data items varies significantly depending on their physical locations on the recording medium

Major Types of RAM



- Computer memories are primarily built from D**ynamic RAM** (**DRAM**) and **Static RAM** (**SRAM**)
- **They** have **different construction** and **properties** but **both** types are **volatile**
  - Which means that they **lose** the **stored information** after the **power** is **shut off**
- **Let** us **study** the **construction** of **each**

Use of SRAM and DRAM



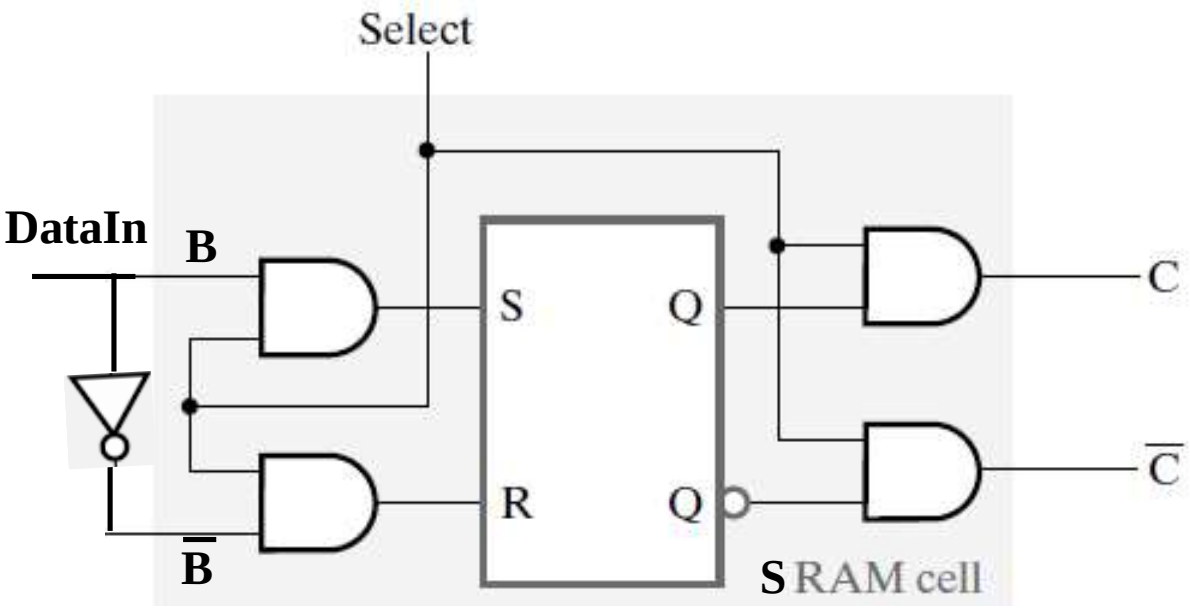
- The processor first seeks data in a small but fast cache that is usually located on the same chip.
- If the data is not available in the cache, the processor then looks in main memory.
- If the data is not there either, the processor fetches the data from hard disk

Quiz 1: Hibernate or Sleep modes in Laptop

- What **happens** when the **laptop** is on **Hibernate** or **Sleep mode?**
- **Both** are **power-saving states**
- **Which** takes **more time** to **power ON?** **Hibernate**
- **What happens** when you put **laptop** to **sleep?**
- **Sleep: Power** to **display** and other **peripherals** is **switched off,** but the **main memory** is **powered.** It **retains** the **contents** of the **currently running programs,** so it can **restart quickly** from the **sleep** mode**.**
- **Hibernate:** The **contents** of the **main memory (RAM)** is **saved into** the **hard disk** and **RAM** is also **powered off.** It **takes more time** because **OS** has to **copy back** the **program contents** from the **hard disk back to RAM** before **start running** them**.**

SRAM Memory Design

SRAM Cell Construction: Using SR Latch



Truth Table of SR Latch

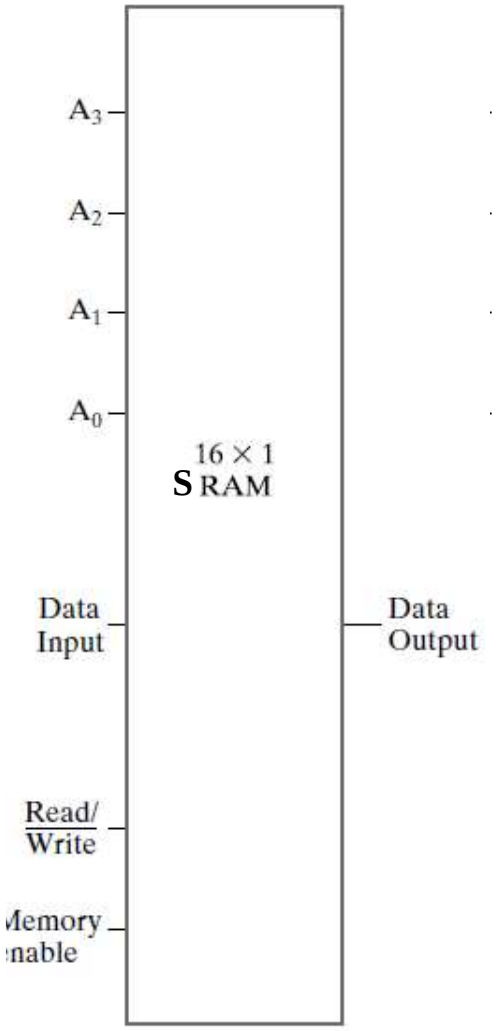
S	R	Q	State
0	0	Previous State	No change
0	1	0	Reset
1	0	1	Set
1	1	?	Forbidden

- When **Select** is **0**, then both ***S*** and ***R*** are **0**, and the **cell latch** content **remains unchanged**
- If the **Select** is **1**, then the **DataIn** will be **loaded** into the **latch**

16 x 1-bit SRAM Chip

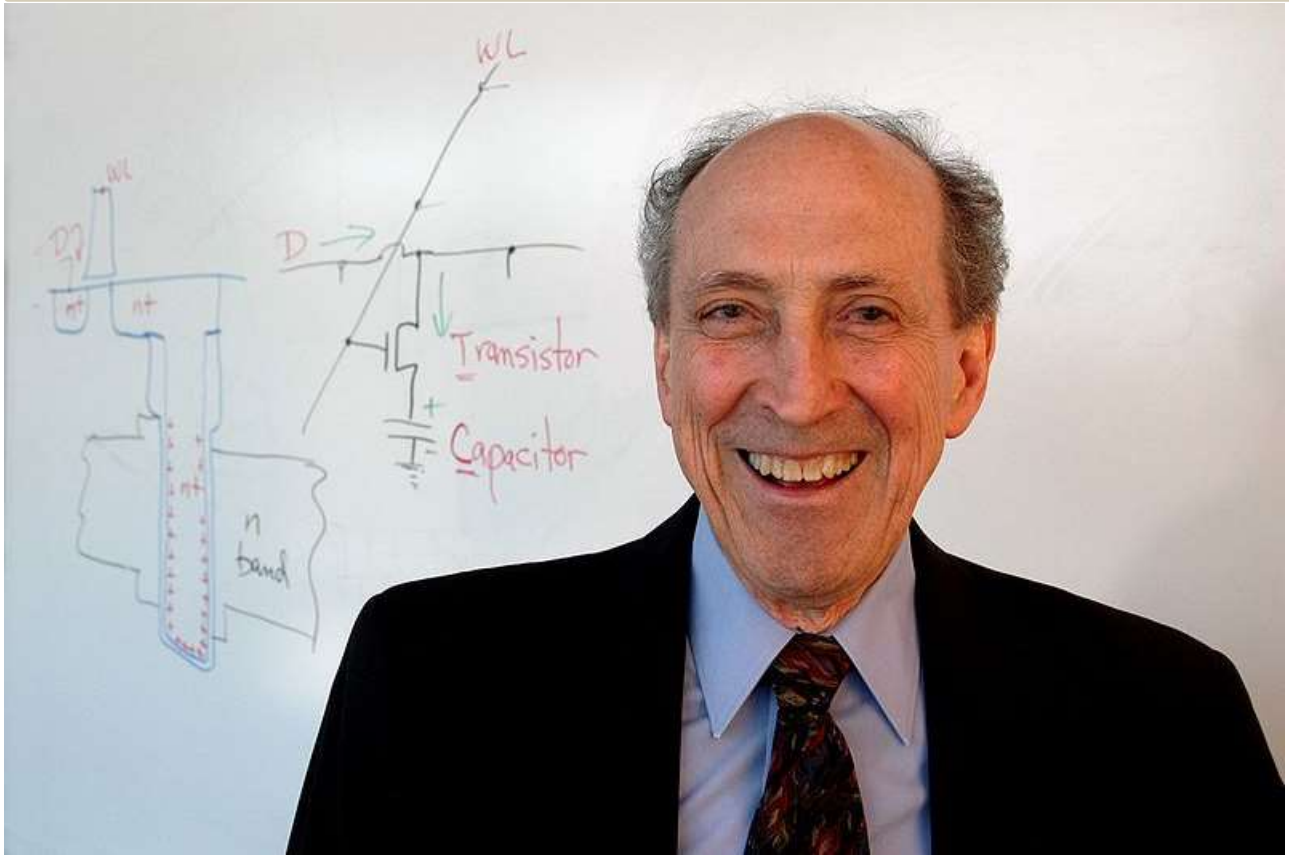
- How many bits are stored? **16 bits**
- How many address lines are required? **4**
- How many data input lines? **1**
- How many data output lines? **1**
- What are the control signals?

  Memory Enable or Select and Read/$\overline{\text{Write}}$



Dynamic Random Access Memory (DRAM)

Inventor of DRAM: Robert Dennard

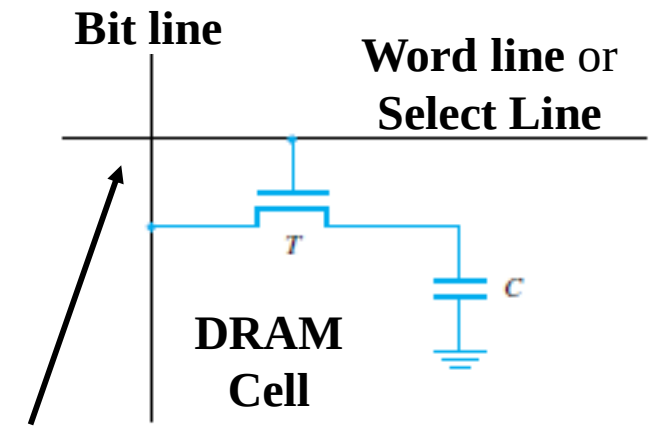


Dr. Robert H. Dennard, IBM Fellow, **beside his drawing** of **a DRAM cell** (circuit schematic)

In **1966** he invented **DRAM**

Scaling theory: **Dennard** and his engineering colleagues observed that as **transistors** are **reduced** in **size**, their **power density** stays **constant**. Meaning, **power use stays** in **proportion** with **area**, as both **voltage** and **current scale** (**downward**) with circuit **length.** This is called **Dennard Scaling.**

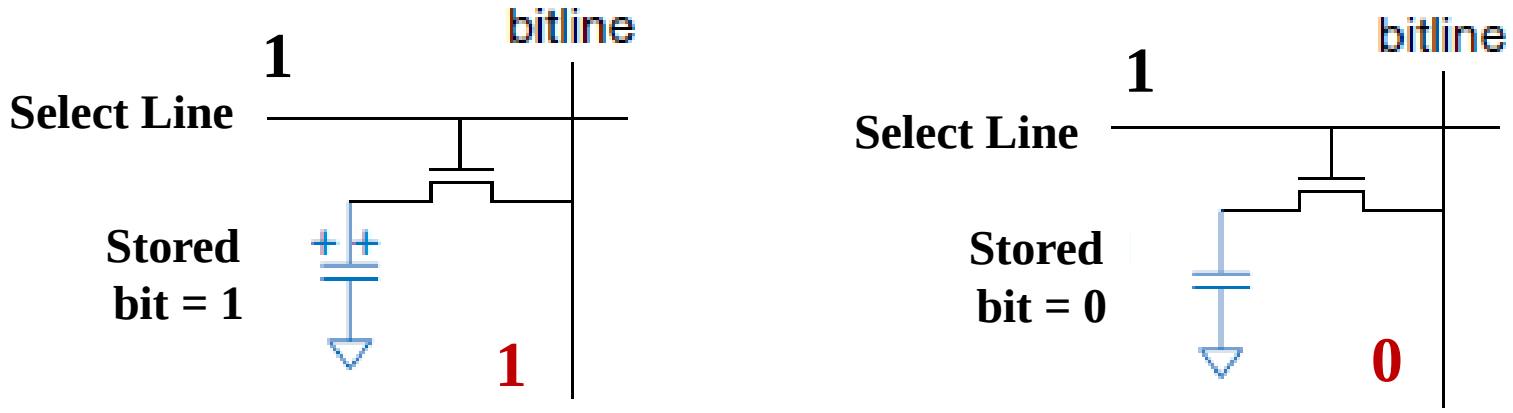
DRAM Circuit Schematic and Functioning



- **T** is a **transistor**, which **functions** like a **switch**
- **C** is a **capacitor** which **stores charges**
- **Select line** is used to **select** the **cell** to be **read/written into**
- **Each cell** can **hold** a **value (0 or 1)**
  - If **C** holds **charges**: 1 is **stored**
  - If **C** has **no charges** in it: **0** is **stored**

- **Bit line** gives the **stored value** out while **reading** from the **cell**
  - The **value** to be **written** into the **cell** is given at **Bit line** to **store** a **0** or **1** into the **cell**
- Note that the same **Bit line** is used to **read** the **stored value** from the cell as well as to **write** a **new value** into the **cell**

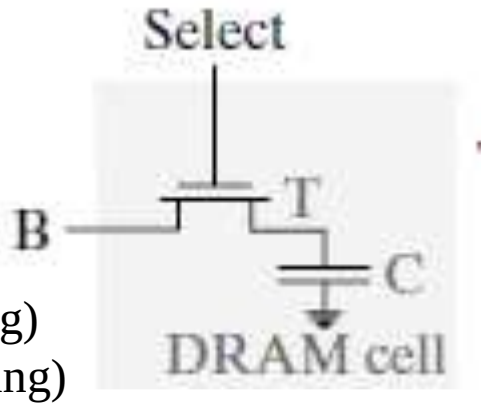
DRAM Cells: What are the stored values?



- If these **cells** are to be **selected** for **reading**, what should the **values** of **Select lines** be?) **They both need to be one (1)**
- What would be the **values** coming **out** of the **cells**, on the **bit lines**?
  - The first cell will give out 1 and the second will give out a zero (0)
- In the **first cell**, the **capacitor** is shown to have **sufficient charges stored** on it, with **++ sign** drawn on it, which has **one** stored in **it**
  - The **second cell** on the right have **insufficient** or **no charges** in them, so it has **zero** in **it**

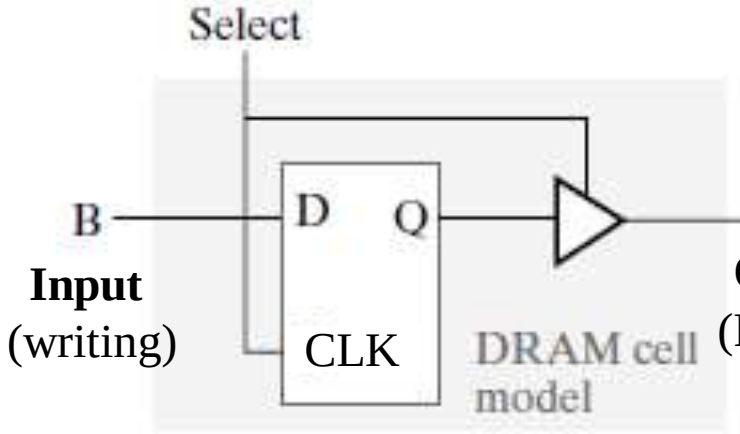
DRAM Cell Model

Actual Implementation



Input (writing)
Output (Reading)

Model of the DRAM Cell



Input
(writing)

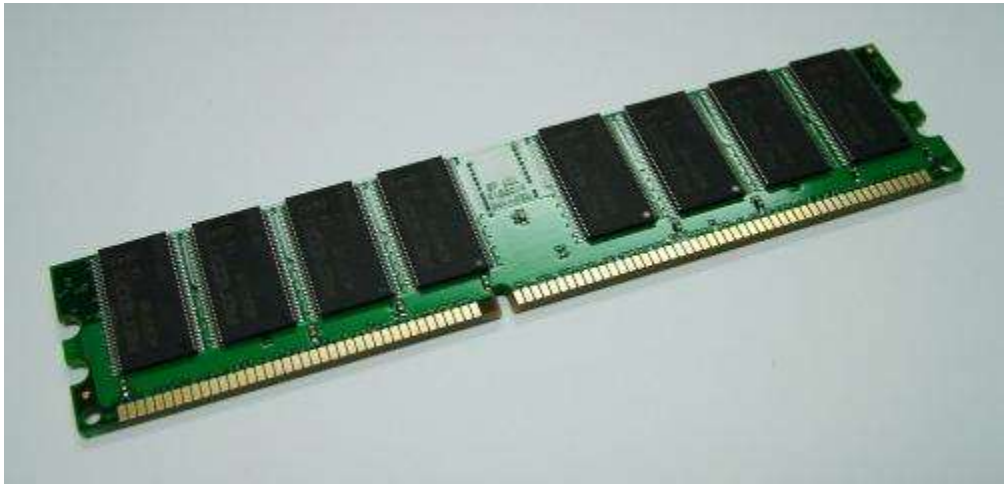
Output
(Reading)

- Model explains the functioning of one transistor (T) and a Capacitor (C) when the control signal (Select) and the input data are given
- In the actual implementation when the data is read, the value stored in the Capacitor also comes out through B

Bit-Slice Model of RAM

DRAM Chips
On PCI interface

PCI: Peripheral Component Interconnect



- In Bit-slice model, each chip stores a bit of the word.
- In the above picture, each bit of a byte (8 bits) is stored in each chip in arranged in an array
- When a byte has to be read from the RAM, all the 8 chips are accessed and
  - Each of them give one bit of the byte value stored in them

ES 3: Summary

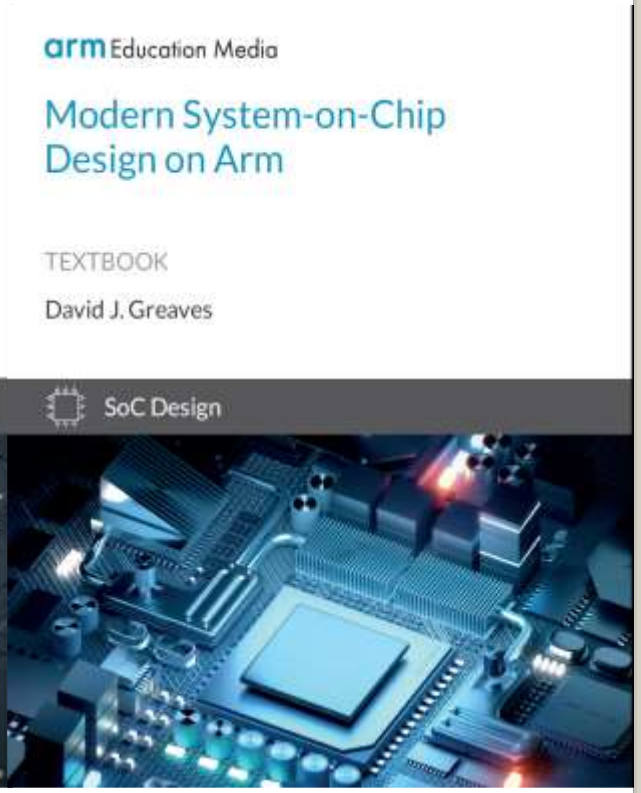
- Introduction to different Memory Types
  - Volatile and Non-volatile Memory
  - Semiconductor Vs Non-semiconductor Memories
- Read Only Memories (ROM)
  - ROM and PROM
  - EPROM and EEPROM
- Random Access Memory (RAM)
  - SRAM Memory Design
  - Dynamic Random Access Memory (DRAM)

References - 1

Ref 0

Ref 1

Ref 2

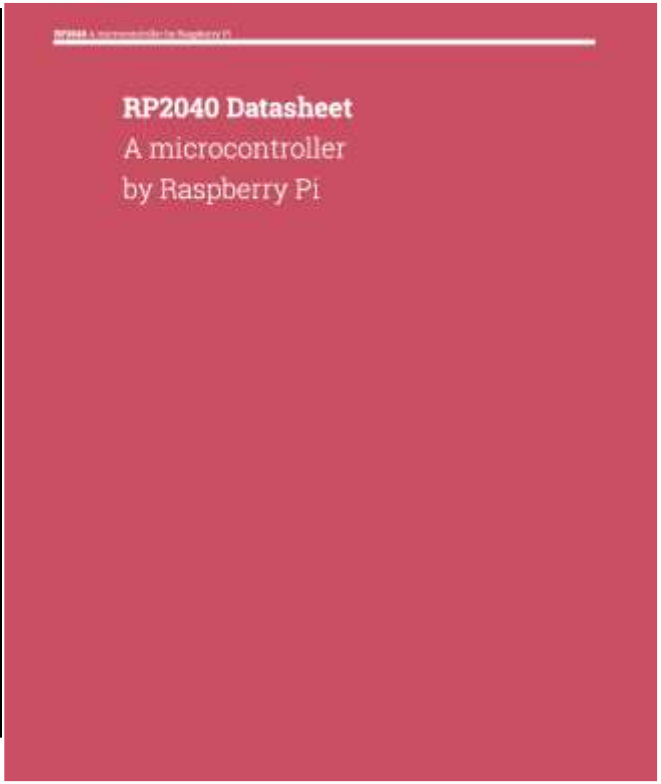


References - 2

Ref 3

Ref 4

Ref 5



Ref 6