11.3 SC Memories: Types and Architectures

Reading Assignment: *pp. 1028-1030*

Digital Computers use **many** different types of **memory**.

Q:

A: HO: Computer Memory

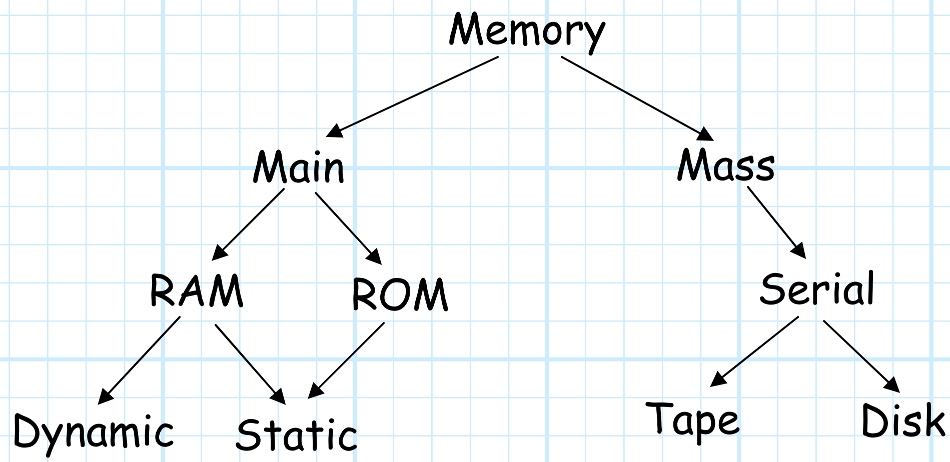
HO: Semiconductor Memories

Computer Memory

Digital Memory → Required for storing data and program instructions.

Memory Types:

Volatile	**OR**	Non-volatile
Main-Memory	**OR**	Mass-storage
Read/Write	**OR**	Read Only (ROM)
Random Access	**OR**	Serial
Dynamic (DRAM)	**OR**	Static (SRAM)



General performance parameters (Read/write):

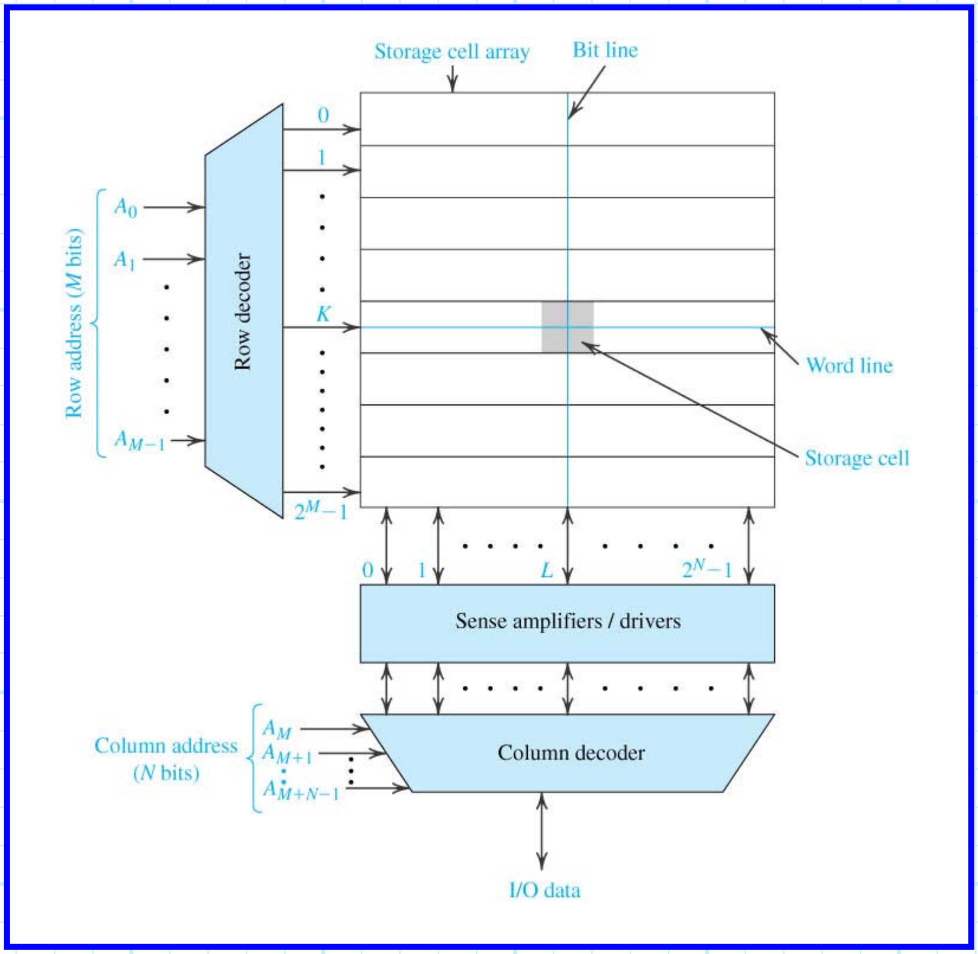
	Speed	Cost/bit	Volatility
SRAM	faster	$$$	Yes
DRAM	fast	$$	Yes
Serial	slow	$	No

As a result, there is **no** perfect computer memory—most computers will use **several** memory types!

Semiconductor Memories

Typically, **integrated circuit memory** is formed by creating multiple **storage cells**—each storing the value of just a **single** bit.

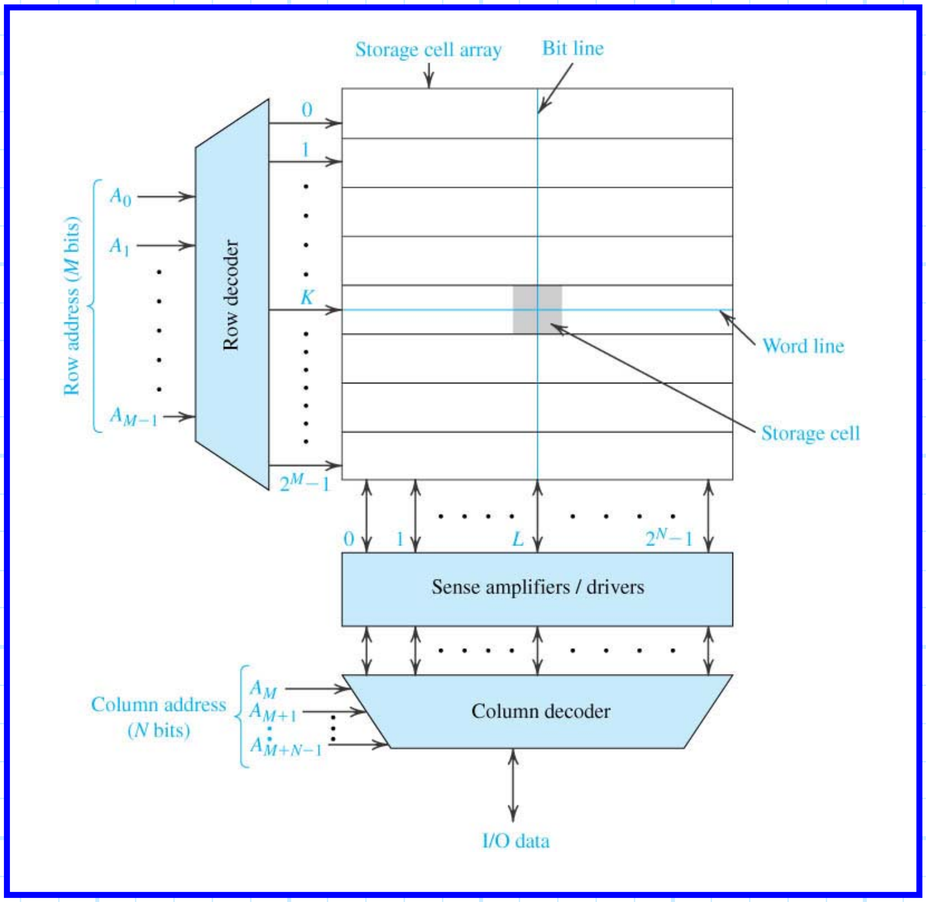
These storage cells are arranged into a **two dimensional matrix**. In other words, they are formed in **rows** and **columns**!



Typically, the **number** of rows is much **greater** than the number of columns. In other words, the **length** of each row (in bits) is much **smaller** then the length of each column.

Generally, the **length** of each row (i.e., the number of columns), is some small multiple of **1 byte** (1 byte= 8bits). For example, each row might be 1 byte (8 bits), 4 bytes (32 bits), or 16 bytes (128 bits) long!

Generally, the length of each column (i.e., the number of rows), is some **large** power of 2 (e.g., 2^{14}, 2^{16}, 2^{20}).



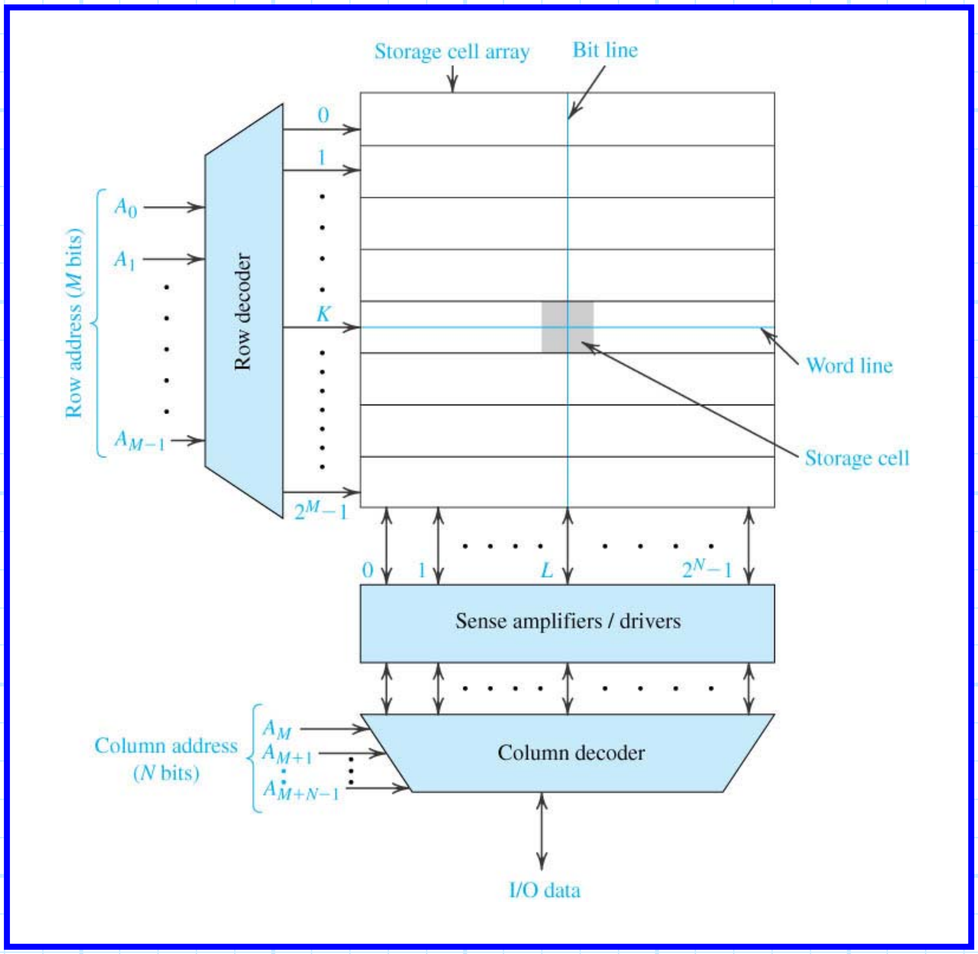
Thus, we can say that typically a semiconductor memory is an arrangement of **2^M rows**, with each row consisting of **2^N storage cells**.

Or, we can say that a typical semiconductor memory is a matrix with **2^N columns**, with each column consisting of **2^M storage cells**.

Note then that each **row** can be uniquely identified with a **M-bit** binary word (e.g., row 5 of 16 → 0101).

Likewise, each **column** can be uniquely identified with an **N-bit** binary word.

Thus, we can uniquely select a **storage cell** by selecting the **row** and the **column** that it resides in. In other words, each **storage cell** can be uniquely identified by a binary word of length **MN**.



By selecting a specific row and column (by setting the proper M-bit and N-bit words), a memory device is said to **enable** one specific memory cell.

We can do one of **two** things to an enabled memory cell:

1. **We can read it** - In other words, we **sense** the state of that memory cell. We answer the question, it that bit **high or low**?

2. **We can write to it** – In other words, we **place** the memory cell in a desired state—we **tell** the memory cell the value of the bit that we want it to remember.

We "read" the value of a memory cell using **sense amplifiers**, and we "write" to them using line **drivers**.