Chapter 7 – Memory Tech introduction

- A register is storage for single value for quick access/readily available
- Memory is designed to store the code and data of programs during execution
- Storage memory technologies require less H/W than flip-flop
- Will go over commonly used memory technologies and their applications
- Storage size is defined in terms of bytes

Unit	Reads	Approximate Size
1kb	One Kilobyte	10^3 Bytes
1MB	One Megabyte	10^6 Bytes
1 GB	One Gigabyte	10^9 bytes
1 TB	One Terabyte	10^12 Bytes

- Memory technologies are categorized as Read only memory or Random-Access memory
- ROM memory is nonvolatile and retains their content even when they are not powered
- RAM are volatile and would lose their content when not powered
- Examples of Non-Volatile memory technologies are magnetic disks, flash memory and optical disk such as CD ROM's
- The content of a ROM cell is fixed at logic 0 or 1
- An Electrically erasable PROM (EEPROM) is most commonly used today
- EEPROM cells can be programmed only a certain number of times
- Ram's functions as the Main storage for programs and data

- SRAM vs DRAM
    - SRAM is preferable for register files and L1/L2 caches
        - Faster access time
        - No Refreshes needed
        - Simpler Manufacturing.
        - Lower density
- DRAM is preferable for stand alone memory chips
    - Much higher capacity per area
    - Higher density.
    - Lower cost – i.e. cost vs performance trade off.
    - Continuously refreshed to keep the capacitors charged up.
- A memory timing diagram precisely illustrates memory communication protocols.
- Time required to select target cells and perform a read or write operation is called a memory access time
- A memory cycle includes both an access time and data transfer time.
- The access time is directly proportional to the size of the memory cell array.
- The decoders in turn determine the time required to activate a target row and access target cells

A. Peak Memory Bandwidth Exercise

Example 1: Consider a 32-bit data bus SDRAM. Given that the clock frequency of the bus is 100MHz, what is the peak memory bandwidth in megabyte per second (MBs)?

32 -bit data bus:

1 byte = 8 bits, so 32 bits x 1 byte/8bits = 4Bytes per cycle

1 MB = 1,000,000 Bytes
1 Hz = 1 cycle/sec

1 MHz= 1 million Hz = 1,000,000 * 1 cycle/sec = 1,000, 000 cycles/sec

100 MHz = 100 Million Hz = 100 Million * 1 cycle/sec = 100,000,000 cycles/sec

100,000,000 cycle/sec * 4 Bytes/cycle = 400,000,000 Bytes/sec
400,~~000,000 Bytes~~/sec * 1MB/1,~~000,000 Bytes~~ = 400 MB/sec

7.5.1 SRAM

Figure 7.16 illustrates an SRAM read cycle from the memory point of view.

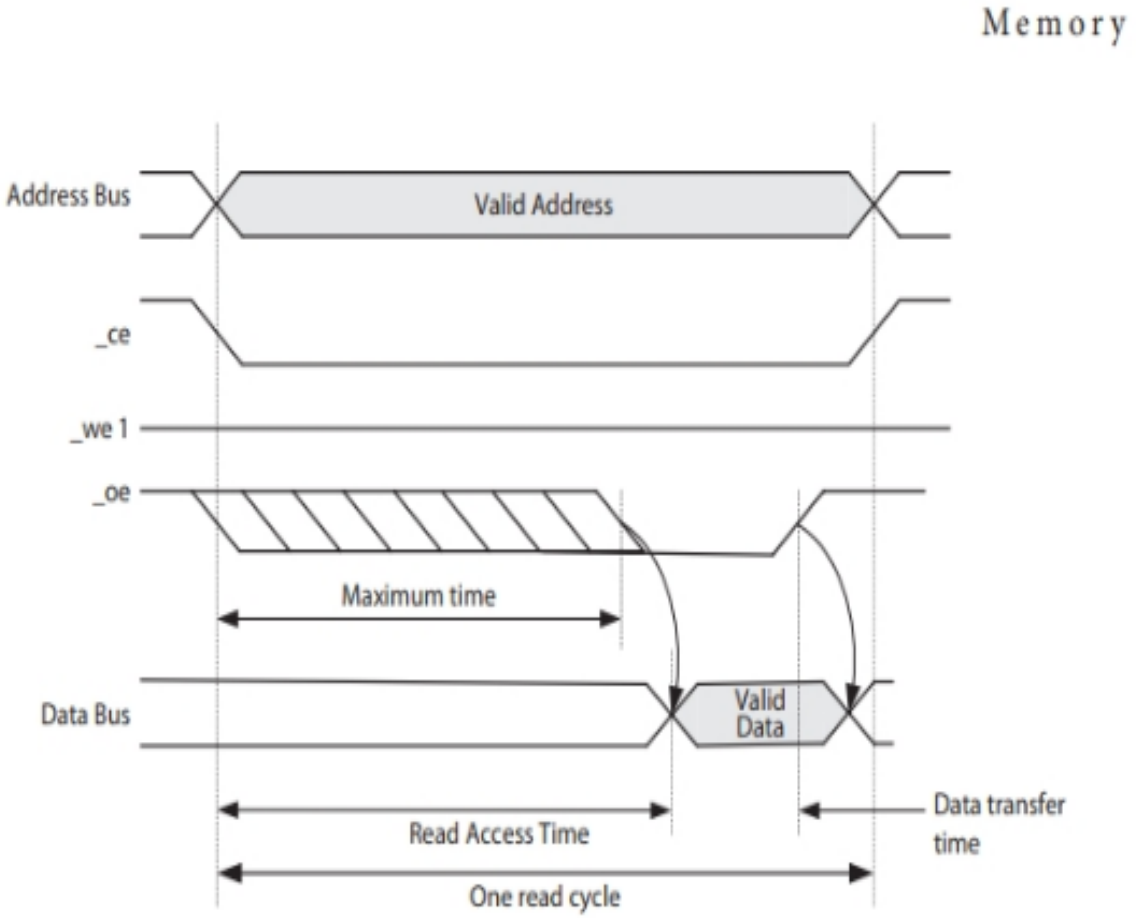


FIGURE 7.16 An SRAM read cycle from the memory point of view.

Figure 7.17 illustrates an SRAM memory write cycle.

A memory cycle is initiated by CPU and typically takes multiple CPU clock cycles to complete.

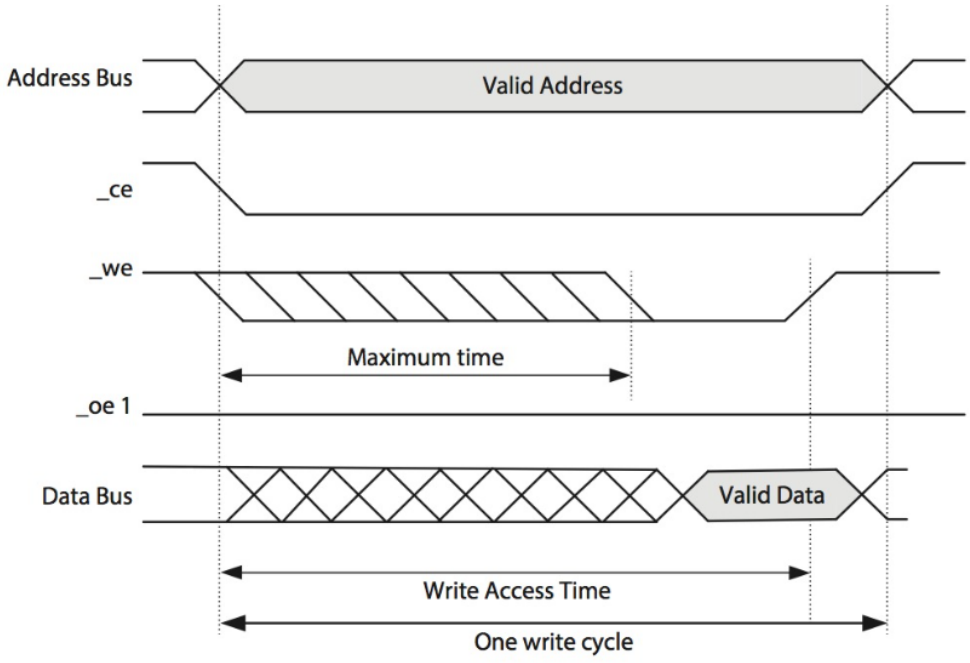


FIGURE 7.17 An SRAM write cycle from a memory point of view.

Tri-State Buffer

A tri-state buffer is similar to a buffer, but it adds an additional "enable" input that controls whether the primary input is passed to its output or not. If the "enable" inputs signal is **true**, the tri-state buffer behaves like a normal buffer. If the "enable" input signal is **false**, the tri-state buffer passes a *high impedance* (or hi-Z) signal, which effectively disconnects its output from the circuit.

Truth table for a tri-state buffer

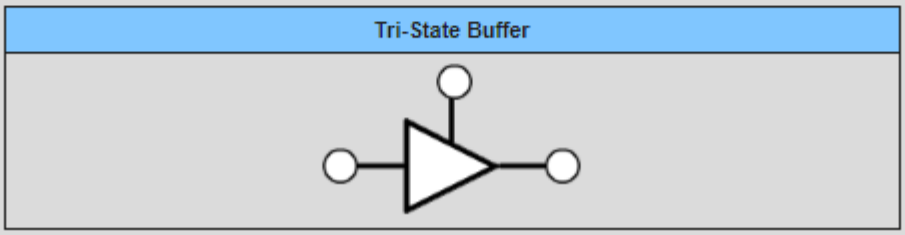
Enable Input	Input A	Output
false	false	hi-Z
false	true	hi-Z
true	false	false
true	true	true

Tri-state buffers are often connected to a *bus* which allows multiple signals to travel along the same connection.

The *truth table* for a tri-state buffer appears to the right.

Symbols

The symbol below can be used to represent a tri-state buffer.



tri-state buffer is similar to a buffer, but it adds an additional "enable" input that controls whether the primary input is passed to its output or not. If the "enable" inputs signal is **true**, the tri-state buffer behaves like a normal buffer. If the "enable" input signal is **false**, the tri-state buffer passes a *high impedance* (or hi-Z) signal, which effectively disconnects its output from the circuit.

Tri-state buffers are often connected to a *bus* which allows multiple signals to travel along the same connection.

The *truth table* for a tri-state buffer appears to the right.