EEE205 – Digital Electronics (II)

Lecture 15-16

Memory Devices

Dr. Ming Xu

Dept of Electrical & Electronic Engineering

XJTLU

Introduction

- Digital information is easily stored
- Commonly used memory devices
  - Flip flops and Registers
  - VLSI and LSI memory devices

12-1 Memory Terminology

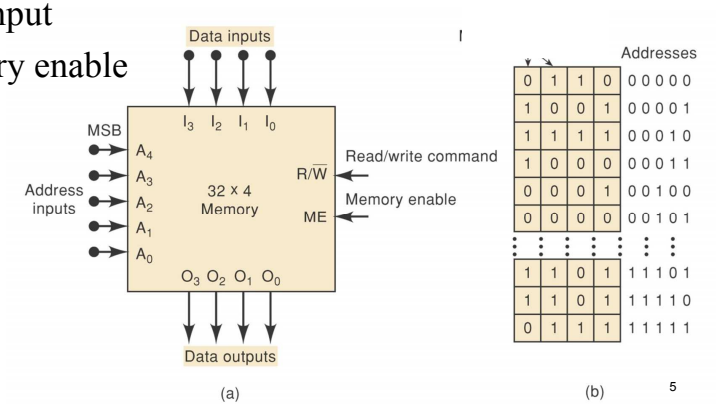
- **Volatile memory**: memory that requires electrical power to hold data.
- **Random access memory (RAM)**: the access time is the same for any address.
- **Read only memory (ROM)**: memory that can be written once (in factory) and can only be read afterwards.
- **Static memory**: data will remain stored as long as power is applied.
- **Dynamic memory**: data will not remain stored, even with power applied, unless the data are periodically rewrittem into memory.

General Memory Operation

- Every memory system requires I/O lines to provide the following functions
  - Select memory address being accessed for R or W operation
  - Select either a R or W operation
  - Supply input data to be stored during a W
  - Hold output data from memory during a R
  - Enable or disable memory so that it will or will not respond to read/write commands

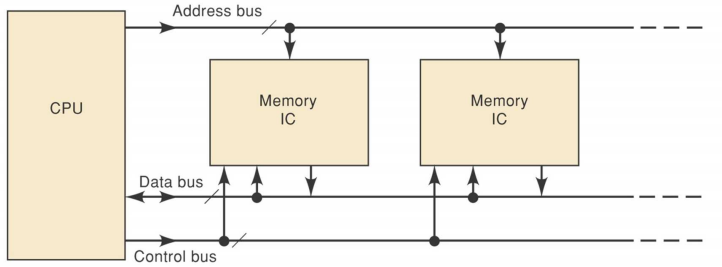
General Memory Operation

- Address inputs
- R/W input
- Memory enable



CPU Memory Connections

- Main memory is interfaced to the CPU through
  - address bus
  - data bus
  - control bus



CPU Memory Connections

- Write operation process:
  - CPU places the memory location address on the address bus
  - CPU places data to be stored on the data bus
  - CPU activates the control signal for the W operation
  - Memory ICs determine location for memory storage by decoding the binary address
  - Data on the data bus is transferred to selected memory location

CPU Memory Connections

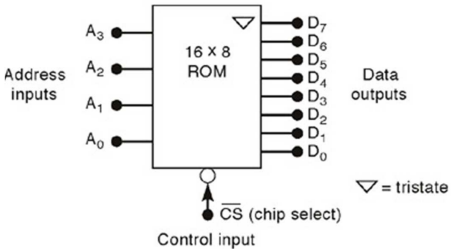
- Read operation process:
  - CPU places address of memory location for data retrieval on the address bus
  - CPU activates the control signal lines for the R operation
  - Memory ICs determine location of data being retrieved
  - Memory IC places data from the memory location onto the data bus

Read Only memories

- Holds data that does not change, or changes only infrequently
- Nonvolatile
- Applications: embedded microcontroller program memory, bootstrap memory, Data tables, etc.

Read Only memories

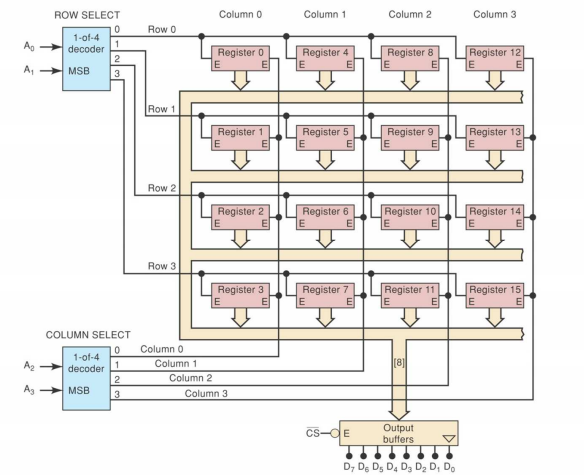
- Address inputs
- Data outputs
- Control input(s)
- The read operation



ROM Architecture

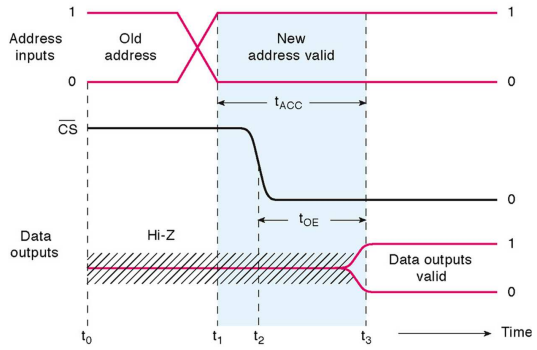
- Four basic parts
  - Register array – stores data programmed into the ROM
  - Address decoders – determines which register will be enabled by row and column
  - Output buffers - pass data to the external data outputs

ROM Architecture



ROM Timing

- Typical timing for a ROM read operation:



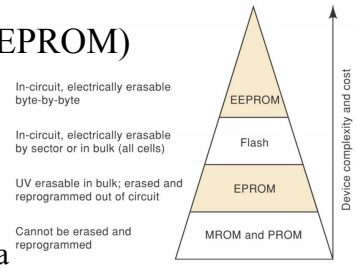
Types of ROMs

- Mask programmed ROM (MROM)
  - Photographic "mask" establishes electrical interconnections
  - Economical only in high volume applications
- Programmable ROMs (PROMs)
  - Fusible links allow end users to program the device
  - Can only be programmed once
  - Economical for small volume applications
- Flash Memory

  Allow rapid in-circuit reprogramming of individual bytes

Types of ROMs

- Erasable programmable ROM (EPROM)
  - Can be erased and reprogrammed by user
  - UV light is used to clear the device
  - Entire device is cleared
- Electrically erasable PROM (EEPROM)
  - Voltage is used to clear memory
  - Individual bytes can be erased
- CD ROM
  - Light is used to stores binary data
  - Large quantities of data are economically stored



Semiconductor RAM

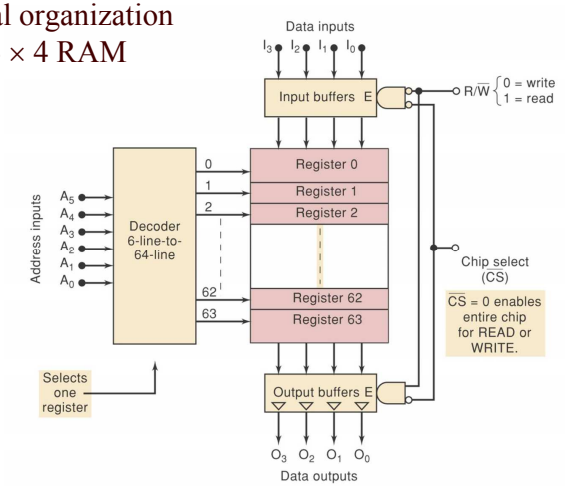
- Random access memory – all memory locations are equally accessible
- Used for temporary storage
- Fast read and write times are necessary
- RAM is volatile
- Many basic concepts that apply to ROM apply to RAM as well

RAM Architecture

- Consider RAM as a number of registers, each storing a single word and having a unique address
- Read operation
- Write operation
- Chip select
- Common I/O pins

RAM Architecture

Internal organization of a 64 × 4 RAM

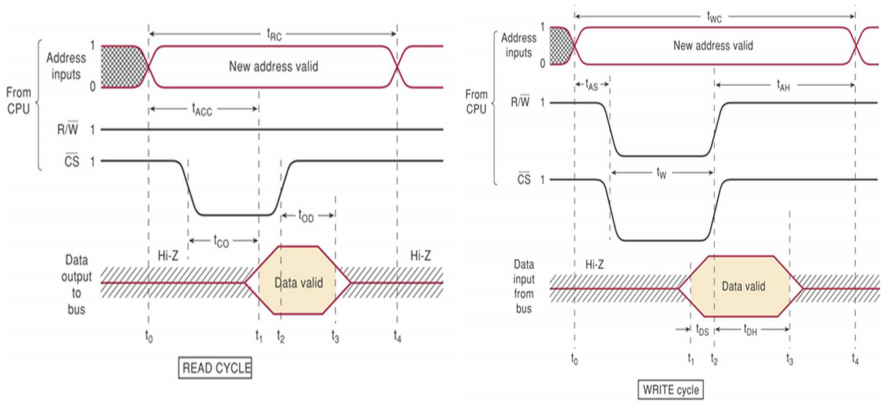


Static RAM (SRAM)

- Stores data as long as power is applied
- Available in bipolar, MOS and BiCMOS variations
- Static RAM timing
- Read cycle
- Write cycle
- Actual SRAM chip – the MCM6264C

Static RAM (SRAM)

Typical timing for static RAM

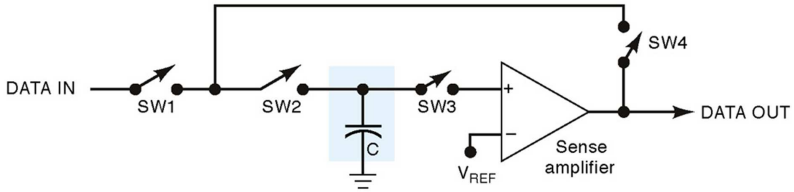


Dynamic RAM (DRAM)

- High capacity
- Low power requirement
- Moderate speed
- Small capacitors are used to store data
- Must be periodically refreshed
- Used for main internal memory in PCs or Macs

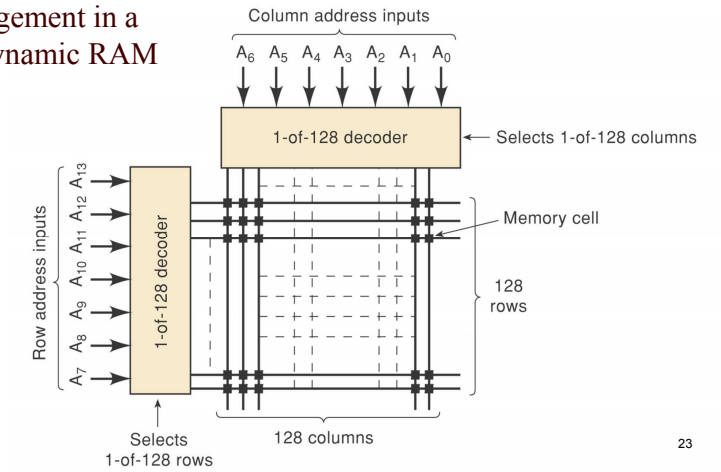
Dynamic RAM Structure and Operation

- An array of cells with unique row and column positions
- Analysis of read and write operations using the simplified representation below:



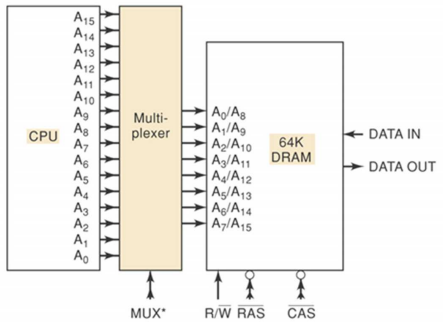
Dynamic RAM Structure and Operation

Cell arrangement in a 16k × 1 dynamic RAM



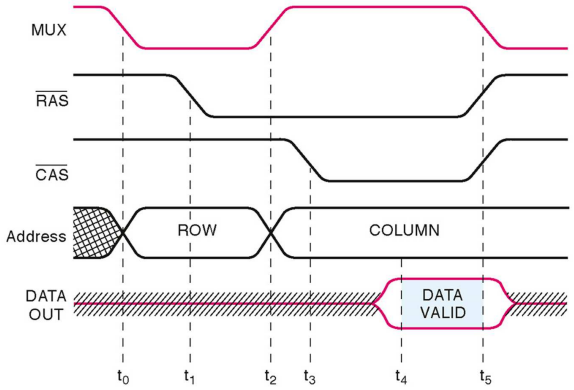
Dynamic RAM Structure and Operation

- Address multiplexing – each address pin can accommodate two different address bits



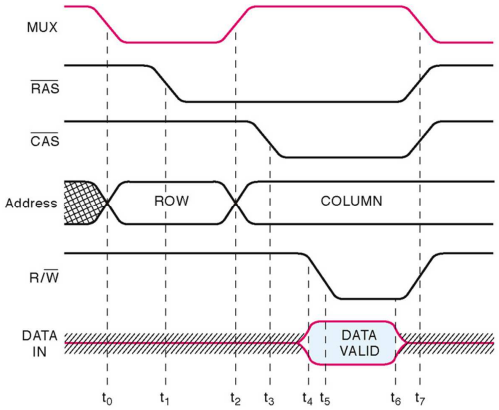
*MUX = 0 transmits CPU address A_8–A_{15} to DRAM. MUX = 1 transmits A_0–A_7 to DRAM.

DRAM Read/Write Cycles



A read operation on a dynamic RAM. The *R*/*W* input (not shown) is assumed to be HIGH.

DRAM Read/Write Cycles



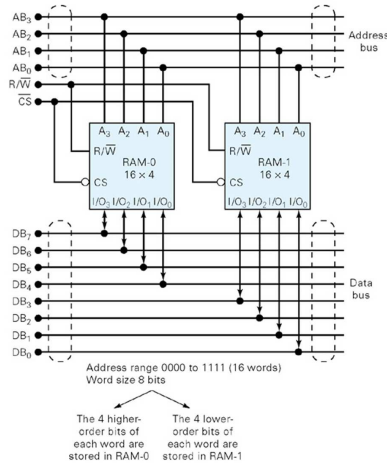
A write operation on a dynamic RAM.

DRAM Refreshing

- When a read operation is performed on a cell, all of the cells in the row will be refreshed
- Refresh control logic is used to make sure that each row is refreshed within the time limit
- Refresh modes
  - Burst
  - Distributed
- Most common method used is RAS only refresh
- The DRAM controller

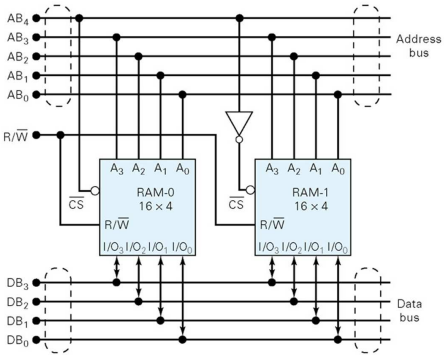
Expanding Word Size and Capacity

- Expanding word size
- Combining RAMs
  - At right two 16X4 RAMs are combined for a 16X8 module



Expanding Word Size and Capacity

- Expanding capacity
- Combining RAMs
  - At right two 16X4 chips are combined for a 32X4 memory



Special Memory Functions

- Cache memory
  - High speed memory that communicates directly with the CPU
  - Level 1 cache is on CPU
  - Level 2 cache is SRAM external to CPU
- First in first out (FIFO) memory
  - Useful as a buffer between systems with different data rates
- Circular buffers
  - Allow data to “wrap around” when the buffer is full