Memory Basics

- **RAM**: Random Access Memory
  - Historically defined as memory array with individual bit access
  - Refers to memory with both Read and Write capabilities

- **ROM**: Read Only Memory
  - No capabilities for “online” memory Write operations
  - Write typically requires high voltages or erasing by UV light

- **Volatility of Memory**
  - Volatile memory loses data over time or when power is removed
    - RAM is volatile
  - Non-volatile memory stores data even when power is removed
    - ROM is non-volatile

- **Static vs. Dynamic Memory**
  - Static: holds data as long as power is applied (SRAM)
  - Dynamic: will lose data unless refreshed periodically (DRAM)

SRAM/DRAM Basics

- **SRAM**: Static Random Access Memory
  - Static: holds data as long as power is applied
  - Volatile: can not hold data if power is removed
  - 3 Operation States: hold, write, read
  - Basic 6T (6 transistor) SRAM Cell
    - Bistable (cross-coupled) INVs for storage
    - Access transistors MAL & MAR
    - Word line, WL, controls access
      - WL = 0 (hold) = 1 (read/write)

- **DRAM**: Dynamic Random Access Memory
  - Dynamic: must be refreshed periodically
  - Volatile: loses data when power is removed
  - 1T DRAM Cell
    - Single access transistor; storage capacitor
    - Control input: word line (WL); data I/O: bit line

- **DRAM to SRAM Comparison**
  - DRAM is smaller & less expensive per bit
  - SRAM is faster
  - DRAM requires more peripheral circuitry
ROM/PROM Basics

- **ROM**: Read Only Memory
  - no capabilities for “online” memory Write operations
  - data programmed
    - during fabrication: ROM
    - with high voltages: PROM
    - by control logic: PLA
  - Non-volatile: data stored even when power is removed

- **PROM**: Programmable Read Only Memory
  - programmable by user -using special program tools/modes
  - read only memory -during normal use
  - non-volatile
  - Read Operation
    - like any ROM: address bits select output bit combinations
  - Write Operation
    - typically requires high voltage (~15V) control inputs to set data
      - stores charge to floating gate (see figure) to set to Hi or Low
  - Erase Operation
    - to change data
      - EPROM: erasable PROM: uses UV light to reset all bits
      - EEPROM: electrically-erasable PROM, erase with control voltage

Comparison of Memory Types

- **DRAM**
  - very high density → cheap data cache in computers
  - must be periodically refreshed → slower than SRAM
  - volatile; no good for program (long term) storage

- **SRAM** (basically a Latch)
  - fastest type of memory
  - low density → more expensive
    - generally used in small amounts (L2 cache) or expensive servers

- **EEPROM**
  - slow/complex to write → not good for fast cache
  - non-volatile; best choice for program memory

- **ROM**
  - hardware coded data; rarely used except for bootup code

- **Register** (flip flop)
  - functionally similar to SRAM but less dense (and thus expensive)
  - reserved for data manipulation applications
Memory Arrays

- N x n array of 1-bit cells
  - n = byte width: 8, 16, 32, etc.
  - N = number of bytes
  - m = number of address bits
    - max N = 2^m

Array I/O
- data (in and out)
  - D_{n-1} - D_0
- address
  - A_{m-1} - A_0
- control
  - varies with design
  - WE = write enable (assert low)
    - WE=1=read, WE=0=write
  - En = block enable (assert low)
    - used as chip enable (CE) for an SRAM chip

Memory Overview

- varies with design
- WE = write enable (assert low)
  - WE=1=read, WE=0=write
- En = block enable (assert low)
  - used as chip enable (CE) for an SRAM chip

Memory Array Addressing

- Standard Memory Addressing Scheme
  - m address bits are divided into x row bits and y column bits (x+y=m)
    - address bits are encoded so that 2^m = N
    - array physically organized with both vertical and horizontal stacks of bytes

Example byte:
- one word in an 8b-wide EPROM