Chapter 7
Memory

In this chapter

- Memory technologies
- Memory design
  - Memory cell
  - Memory chip internal organization
- Memory communication protocols
- Data storage schemes
- UMA vs. NUMA system architectures

Logical organization

- Number of addresses by word size
  - E.g., 1K × 8
  - E.g., 512 × 16
- Total size in bytes
  - 1KB
  - 1MB
  - 1GB
  - Etc.

Address Decimal	Address Binary (10 bits)	Data 8-bit Content
0	0000000000	00010001
1	0000000001	10000111
2	0000000010	00111100
3	0000000011	11000000
.	.	.
.	.	.
.	.	.
.	.	.
.	.	.
.	.	.
.	.	.
1023	1111111111	10000001

(a) 1K × 8 Memory

Address Decimal	Address Binary (9 bits)	Data 16-bit Content
0	000000000	0001000100010001
1	000000001	1000011111100001
2	000000010	0011110000001100
3	000000011	1100000011010100
.	.	.
.	.	.
.	.	.
511	111111111	1000000111001011

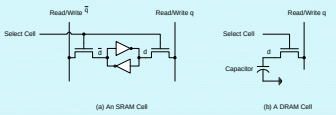
(b) 512 × 16 Memory

Memory Technologies

- Non-Volatile
  - Each memory cell retains 0 or 1 indefinitely
  - Word accessible
    - ROM
    - PROM
    - EEPROM
      - Can be written limited number (~ 100,000) of times
    - Older technologies EPROM
    - Applications: boot loader, LUT, firmware
  - Block accessible (as secondary memory storage)
    - Magnetic disk
    - Flash memory (EEPROM based)
- Volatile
  - Each memory cell retains 0 or 1 as long as powered
  - Word accessible only
    - SRAM
    - DRAM
      - SDRAM (DDR, DDR2, DRR3, etc.) as modern DRAM

RAM cells

- SRAM
  - Hardware
    - 6 transistors
  - Retains data while powered
  - fast
- DRAM
  - Hardware
    - One transistor
    - One small capacitor
  - Much smaller than SRAM cell
  - Cheaper per bit
  - Slow



(a) An SRAM Cell (b) A DRAM Cell

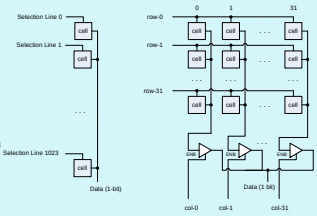
s	Action
0	a and b are disconnected (electrically isolated)
1	a and b are connected

(c) NMOS Transistor

Organization and Access

- 2D Organization (cell array)
  - Rectangular as the die
  - Requires fewer total number of wires
- Read/Write Operation
  1. Frist select a row
     - Also called row activation
  2. Then select one or more cells from activated row to either read or write
- Burst access
  - Access multiple cells in specific order typically from a single row
  - Cells form a block of data (e.g., 32B)
- Page access
  - Access many cells from one or more rows
  - Cells form a large block of data (e.g., 4KB)

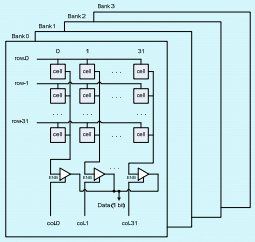
32 X 32 X 1



(a) One-dimensional organization (b) Two-dimensional organization (shown for read)

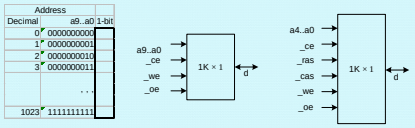
Multi-bank

- Allows seamless access
  - Cells read/written may belong to different banks
- Can overlap operations
  - Activating a row in one bank while read/writing cells from already activated row in another bank



Memory Interface

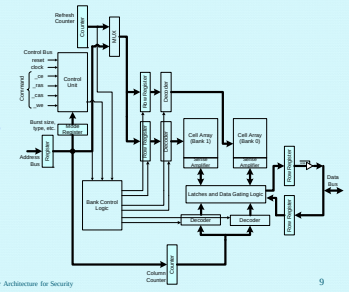
- Requires address lines (address bus)
  - Address for DRAM is provided in two cycles
- Requires control lines (control bus)
  - Indicating enabling, reading, and writing
- Requires data lines (data bus)
  - Bi-directional data bus
  - Separate input and output data lines



(a) Logical view (b) An SRAM block diagram (c) A DRAM block diagram

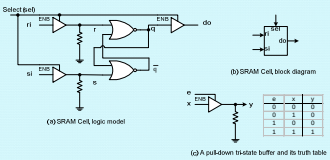
SDRAM (Synchronous DRAM)

- Interface signals form memory command
- Synchronous operation makes design of computers easier, cheaper
- Today SDRAM technologies are used for main memory



SRAM Cell Model

- Real RAM cells cannot be simulated with logic simulation tools
- It can be modeled with SR latch and tri-state buffers to mimic similar behavior
- Resister converts Hi-Z output to 0



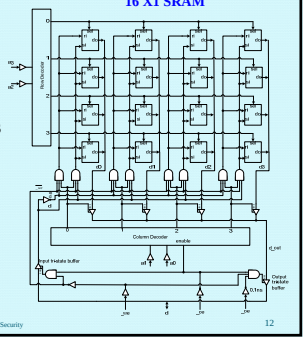
Memory Design

- Memory chip
  - Internal organization
    - Single or multi-banked
    - Bi-directional data bus
  - Access protocol defines signal timing
- Memory module
  - Wider data bus than memory chip
- Memory unit
  - Wider address bus than memory module

Memory Chip

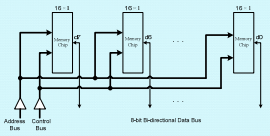
16 X1 SRAM

- Requires two decoders
  - Row decoder activates a row
  - Column decoder selects one or more cells
- Input and output tri-stated buffers to implement bi-directional data bus



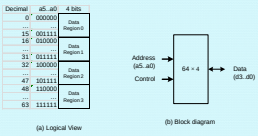
Memory Module

- Also called memory card
- 32- or 64-bit data bus
  - Wider if ECC
- For building memory unit(s) as main memory



Memory Unit

- Maps logical memory space to physical memory space
- Different mapping options
  - High-order interleaving
  - Low-order interleaving (later)
  - Hybrid
    - E.g., NUMA architectures

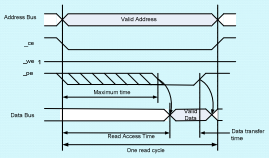


High-order Interleaving Example

Memory Access

- Follows specific communication protocol and signal timing
- Memory Cycle
  1. Starts when address decoding begins
  2. Waits to activate a row and select cell(s)
  3. Completes read or write operation
  4. Ends cycle
- Timing parameters
  - Access time
    - Read: From start until data appears on data bus
    - Write: From start until data is written to memory cells
  - Transfer time
    - Time to transfer data to/from memory
- Memory latency
  - Access time + transfer time

SRAM



SDRAM

- Concurrent memory operations
- Read Protocol:
  1. Issue burst size
  2. Issue row address
  3. Wait for row to activate (fixed number of clock cycles)
  4. Issue column address
  5. Repeat step 4 as needed
     - Timing depends on burst size
  6. Data placed on data bus, one per clock cycle, seamlessly

DDR SDRAM

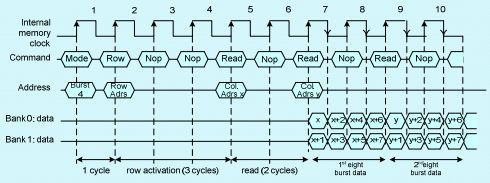
- Operation similar to SDRAM
- Data placed on data bus on rising as well as falling clock edges
  - Two data items per clock cycle
  - Doubling the bandwidth of SDRAM
    - Doubling number of data bytes per second

Data Interleaving

- High-Order Interleaving
  - Data for consecutive memory addresses are stored in the same memory module/unit
  - Advantage:
    - Divides memory space into two or more disjoint sub-spaces
    - Each sub-space may be accessed by a separate processor
- Low-Order (fine) Interleaving
  - Data for consecutive memory address are stored in different memory modules/units
  - Advantage:
    - Increases memory bandwidth

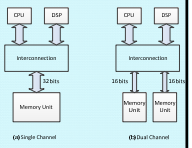
DDR2 SDRAM

- Read/write from two banks at the same time
  - Fine interleaving memory banks
- Doubling bandwidth of DDR SDRAM
  - Requires higher data transfer rate



Multi-Channel

- Organize data bus into two or more independent channels
  - Separate burst access in each channel
- Larger bursts to deliver same amount of data
  - More efficient channels
  - More continuous delivery of data
  - Better performance
    - E.g., for real-time processing
  - Application
    - Better performing embedded systems



Multi-Processor Memory Architecture

- Uniform memory access (UMA)
  - Memory latency about the same (uniform)
  - Good for small systems
    - E.g., multi-core processor system
- Non-uniform memory access (NUMA)
  - Memory latencies vary (non-uniform)
    - Small when accessing local memory
    - Long when accessing remote memory
  - Average latency < UMA
  - Better for multithreaded programs
    - Each threads mostly accesses its local memory
    - Only shared data (if any) accessed remotely
      - E.g., consider producer-consumer application
  - Nodes can be multi-core