Chapter 1:
Data Storage

Computer Science: An Overview
Tenth Edition

by
J. Glenn Brookshear

Chapter 1: Data Storage

- 1.1 Bits and Their Storage
- 1.2 Main Memory
- 1.3 Mass Storage
- 1.4 Representing Information as Bit Patterns
- 1.5 The Binary System
- 1.6 Storing Integers
- 1.7 Storing Fractions
- 1.8 Data Compression
- 1.9 Communications Errors

The binary system

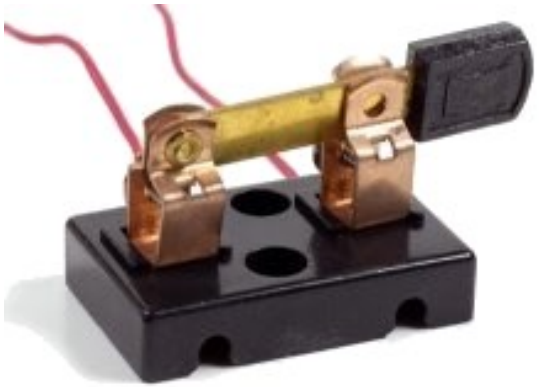
- The modern binary number system was invented by Gottfried Leibniz in 1679

Bits and Bit Patterns

- **Bit（位）：**Binary Digit (0 or 1)
- Bit Patterns are used to represent information.
  - Numbers
  - Text characters
  - Images
  - Sound
  - And others

Electric Switch （电气开关)

- What are the inputs and outputs?



Binary Operations and Logic Gates（逻辑门）

Basic operations for binary data and the physical devices to implement them

Binary and Logic (Sec. 1.1)

- Logic: concerns about true or false
- Logic operation:
  - If the room is dark AND someone is in the room, turn on the light.

Room is dark { Yes (1); No (0) }

Someone in the room { Yes (1); No (0) }

Turn Light on { Yes (1); No (0) }

- True/false can be represented by 0/1

Binary number system in computer ←→ logic

The AND Function

- We can use the AND function to represent the statement

Room is dark A	Someone in the room B	Turn on the light A .AND. B
0	0	0
0	1	0
1	0	0
1	1	1

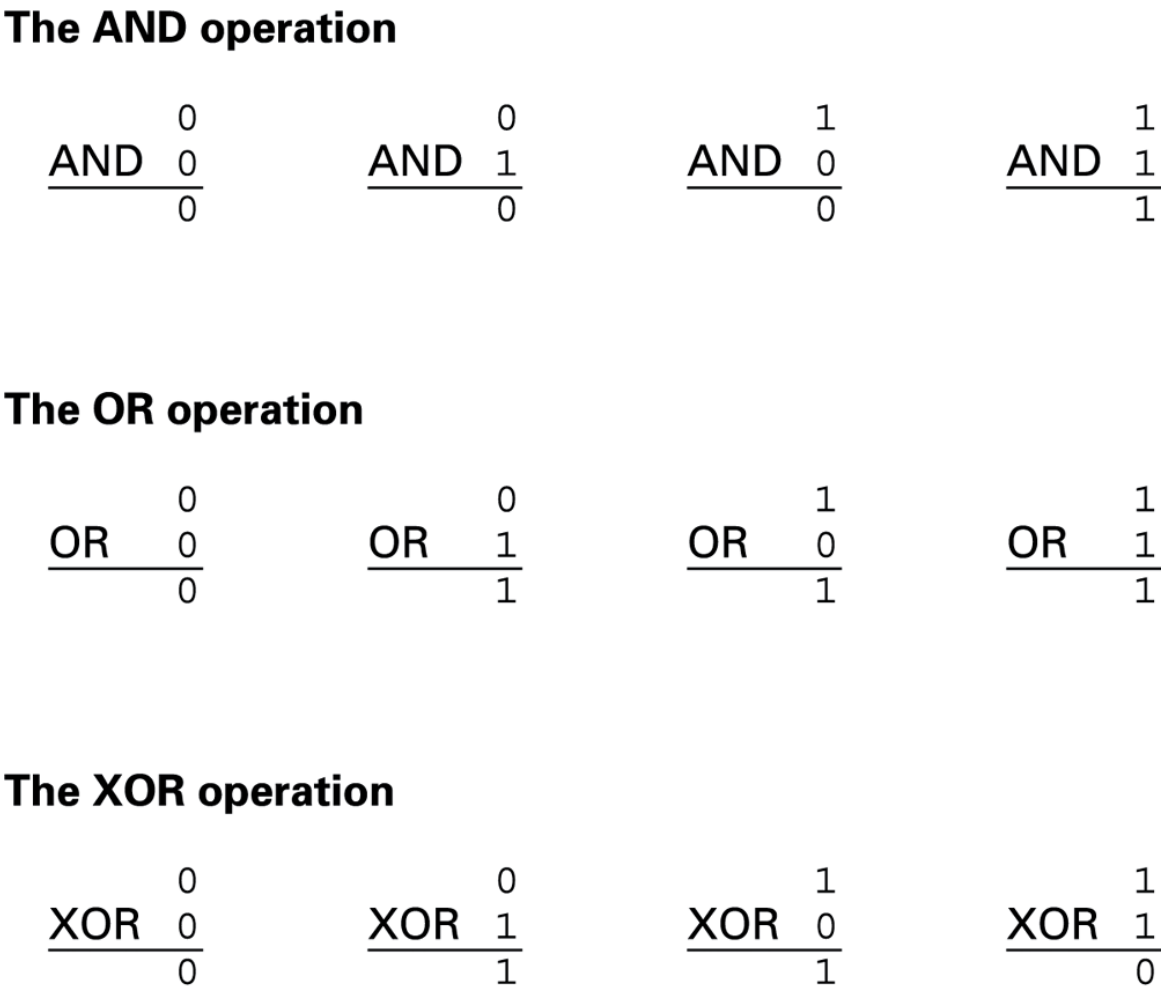
Input

Output

Boolean Operations （布尔运算)

- **Boolean Operation:** An operation that manipulates one or more true/false values
- Specific operations
  - AND
  - OR
  - XOR (exclusive or)
  - NOT

Figure 1.1 The Boolean operations AND, OR, and XOR (exclusive or)

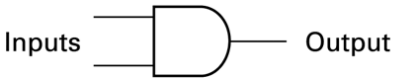


Gates

- **Gate:** A device that computes a Boolean operation
  - Often implemented as (small) electronic circuits
  - Provide the building blocks from which computers are constructed
  - VLSI (Very Large Scale Integration)

Figure 1.2 A pictorial representation of AND, OR, XOR, and NOT gates as well as their input and output values

AND



Inputs	Output
0 0	0
0 1	0
1 0	0
1 1	1

OR

Inputs	Output
0 0	0
0 1	1
1 0	1
1 1	1

XOR

Inputs	Output
0 0	0
0 1	1
1 0	1
1 1	0

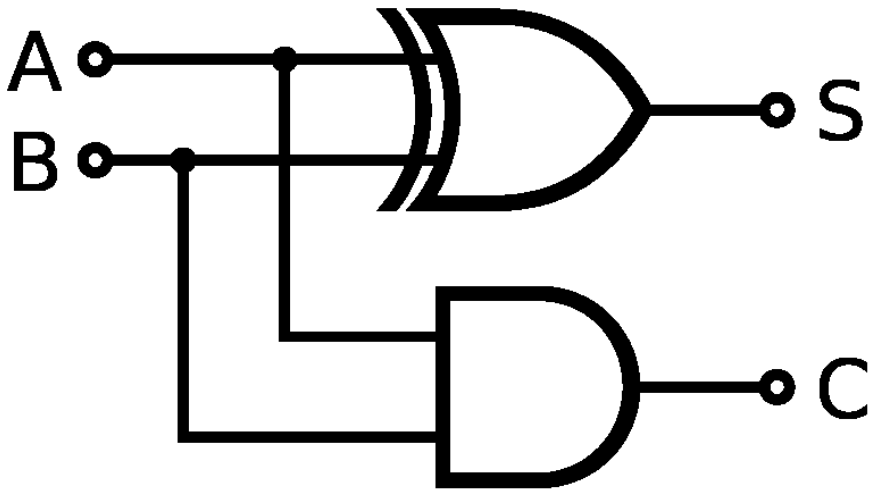
NOT

Inputs	Output
0	1
1	0

- Arithmetic

Adder

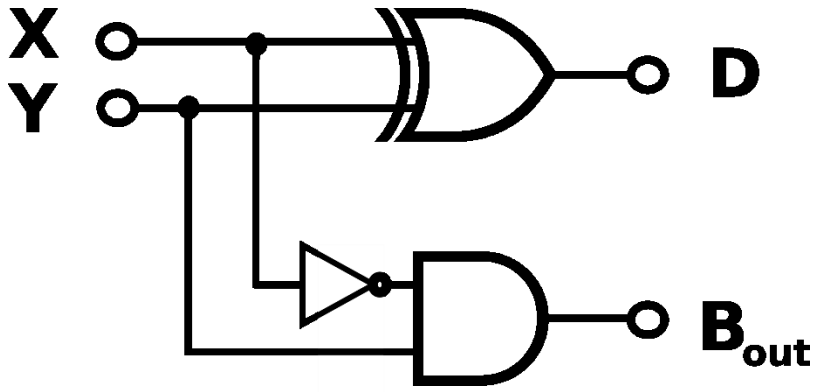
- Half adder
- Full adder



Inputs		Outputs	
A	*B*	*C*	*S*
0	0	0	0
1	0	0	1
0	1	0	1
1	1	1	0

Subtractor

- Half subtractor
- Full subtractor



X	Y	D	Bor
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

- Questions?

- Storage

Flip-flops

- **Flip-flop:** a circuit that has two stable states and can be used to store state information
  - The output will flip or flop between two values under control of external stimuli

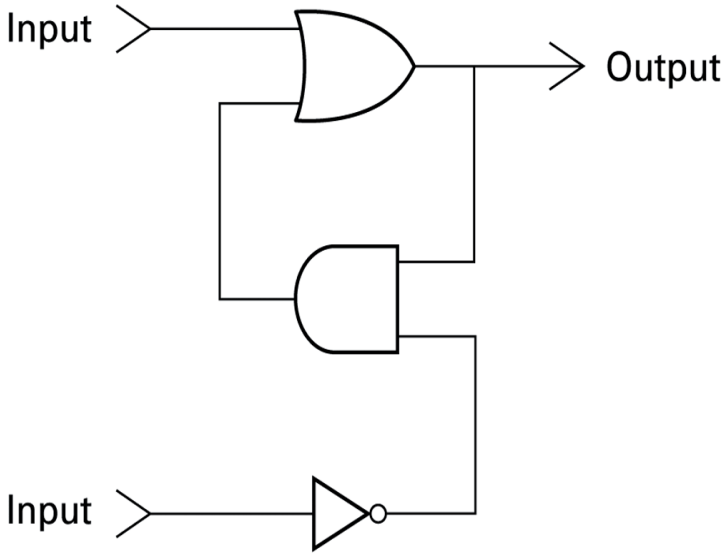
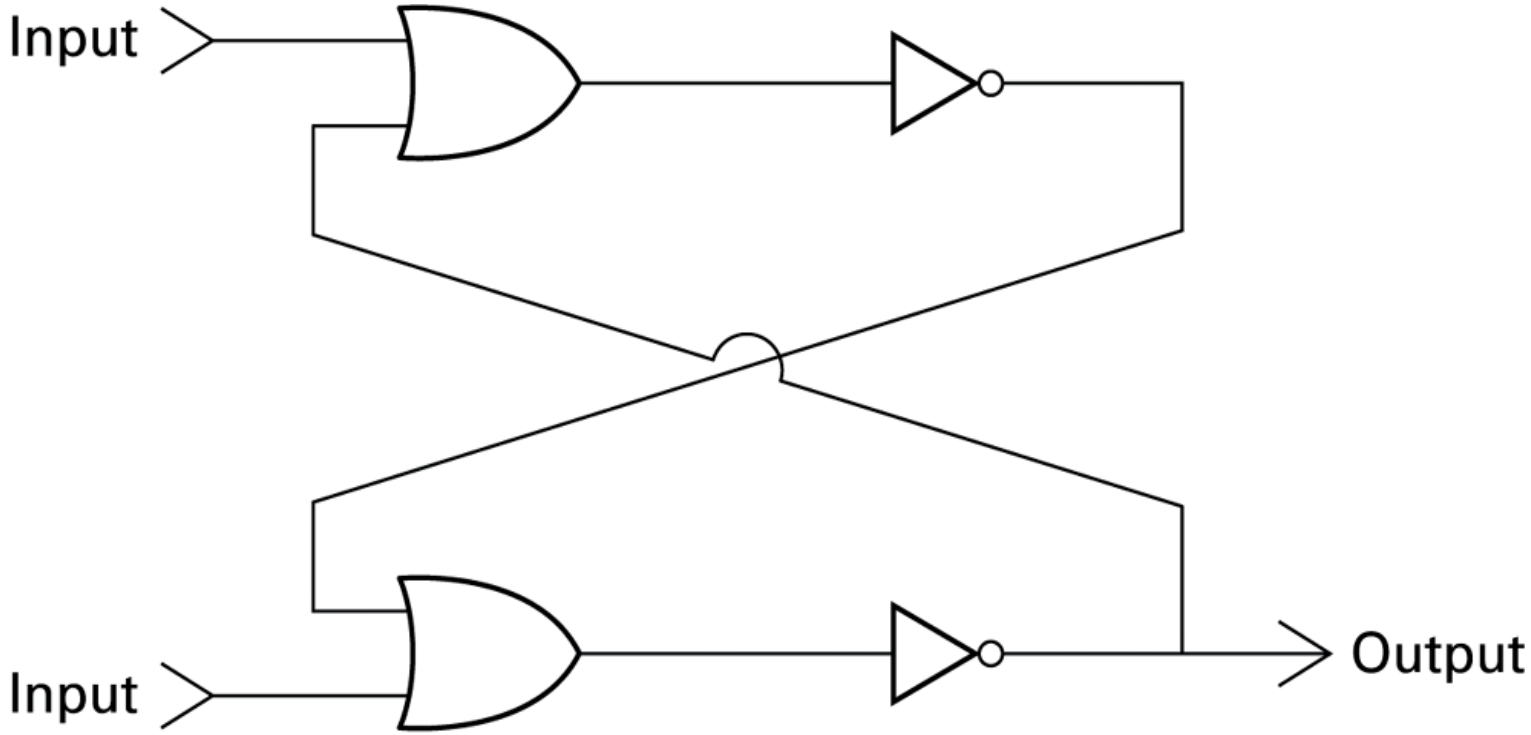
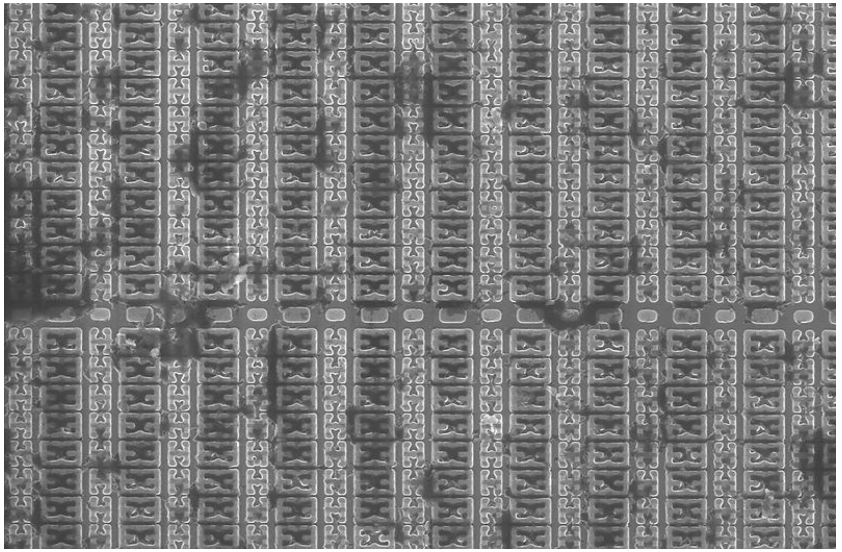


Figure 1.5 **Another way of constructing a flip-flop**

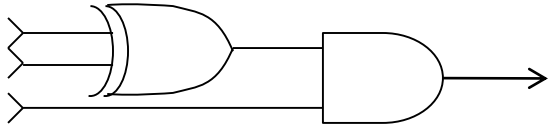


SRAM

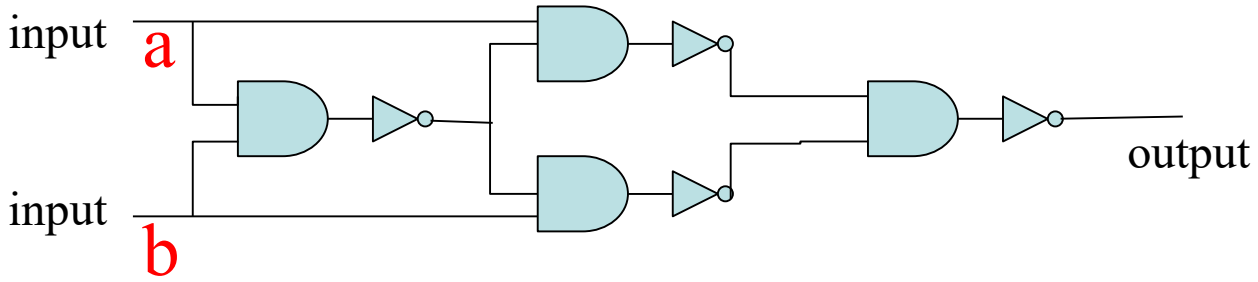


Exercises

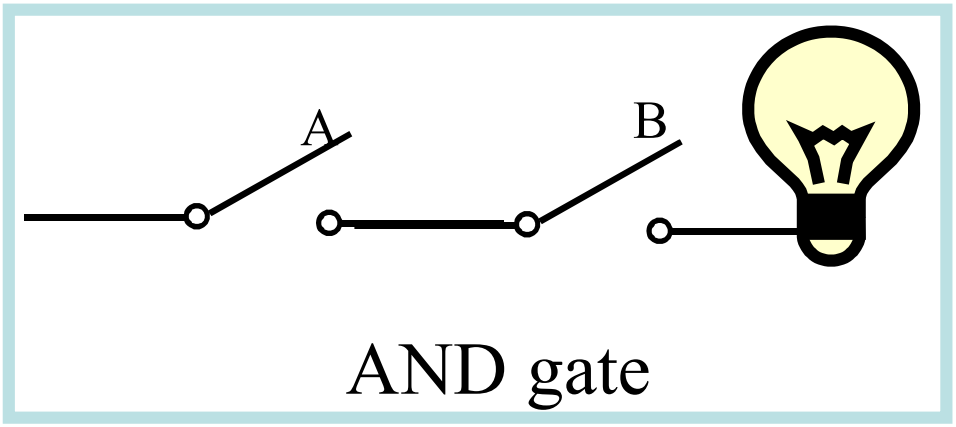
- What input bit patterns will cause the following circuit to output 1? And output 0?

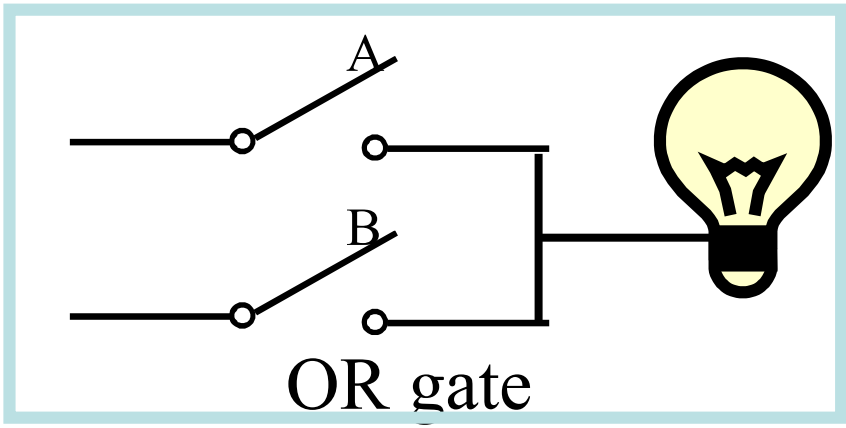


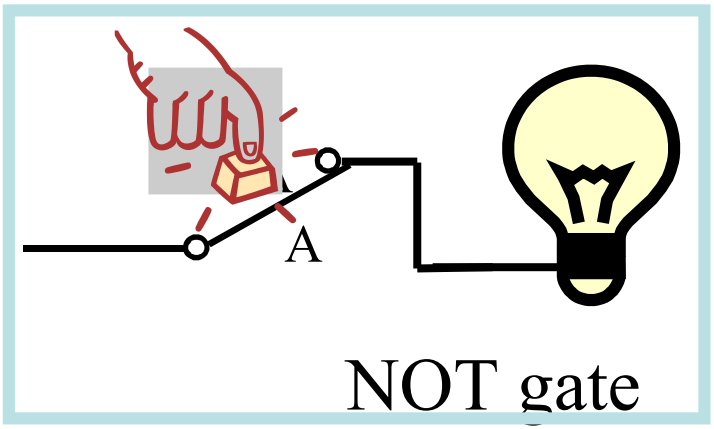
- What Boolean operation does the circuit compute?

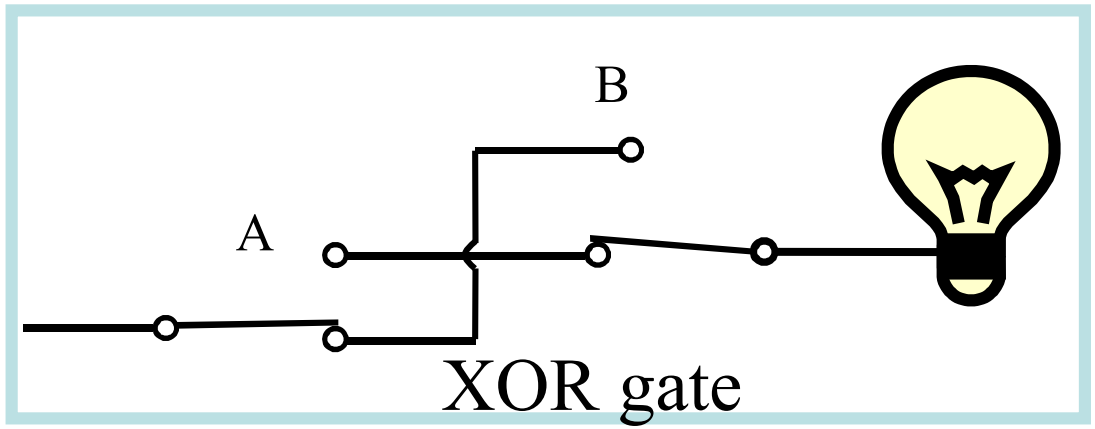


Implement Gate with Switch







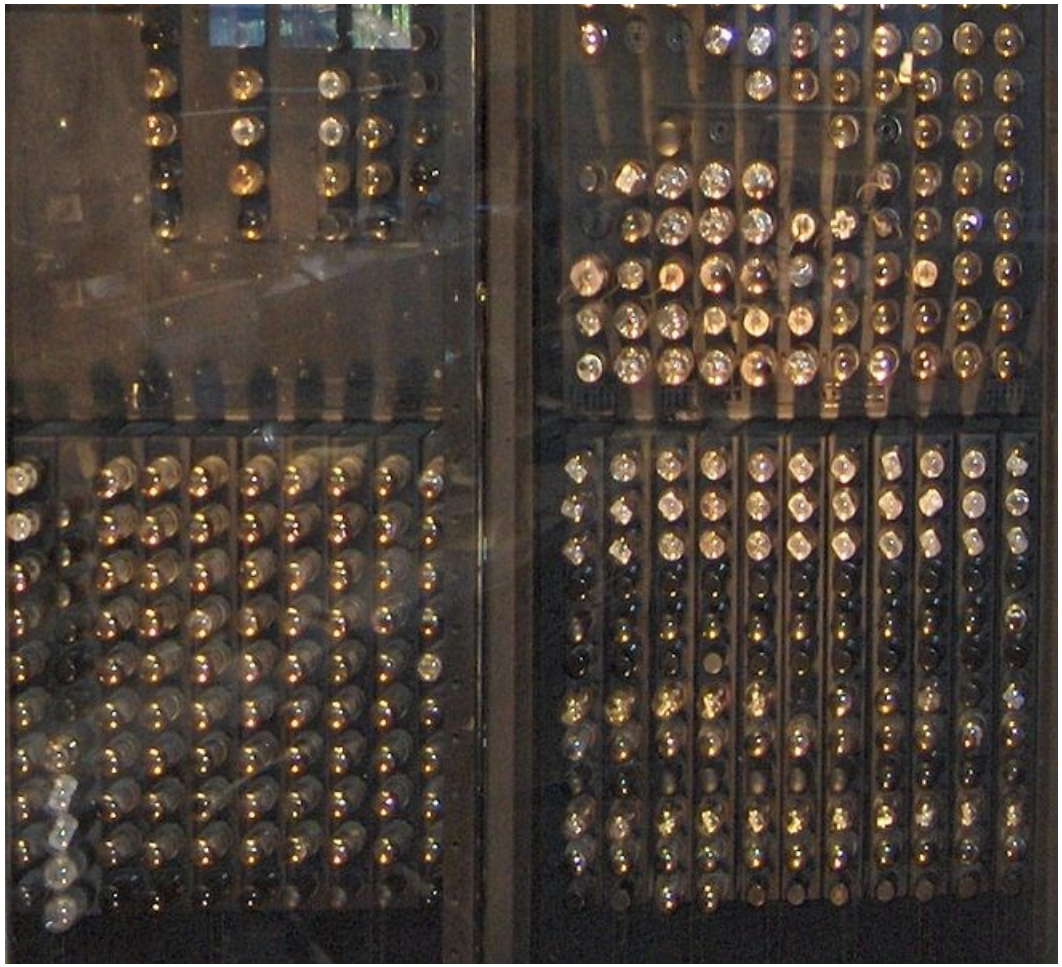


- Can we flip the switches without hands?

Electronic Switch

- The earliest one is the *vacuum tube*
  - 1884, Thomas Edison
  - 1901, Owen Richardson

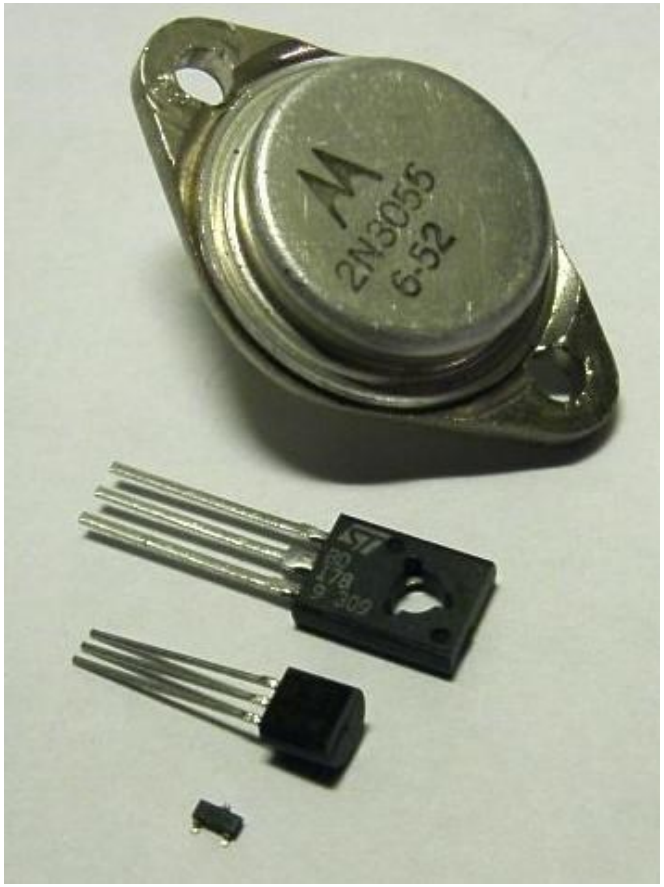
The 1946 ENIAC computer used 17,468 vacuum tubes and consumed 150 kW of power

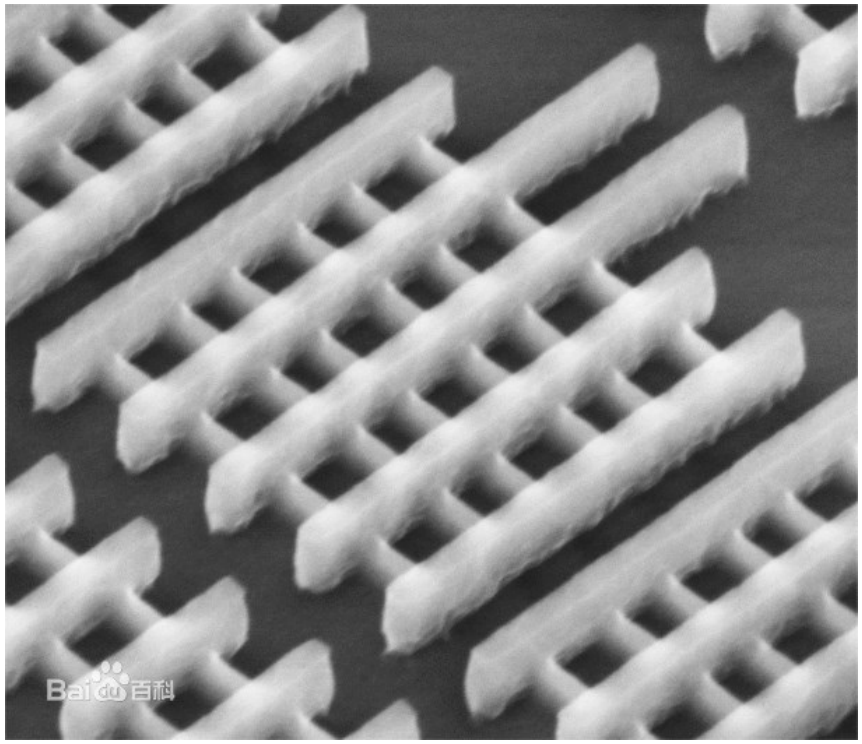


Electronic Switch

- Transistor can be faster, smaller, and more robust

John Bardeen, William Shockley and Walter Brattain at Bell Labs, 1948





Intel Core i7

- **How are patterns of bits stored in computer?**

Main Memory Cells

- **Cell:** A unit of main memory (typically 8 bits which is one **byte**)

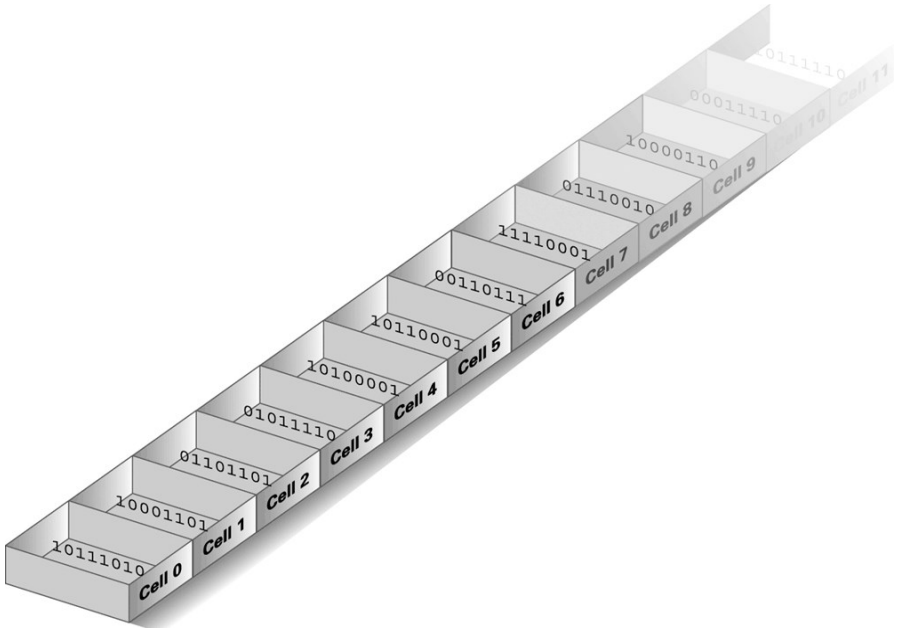
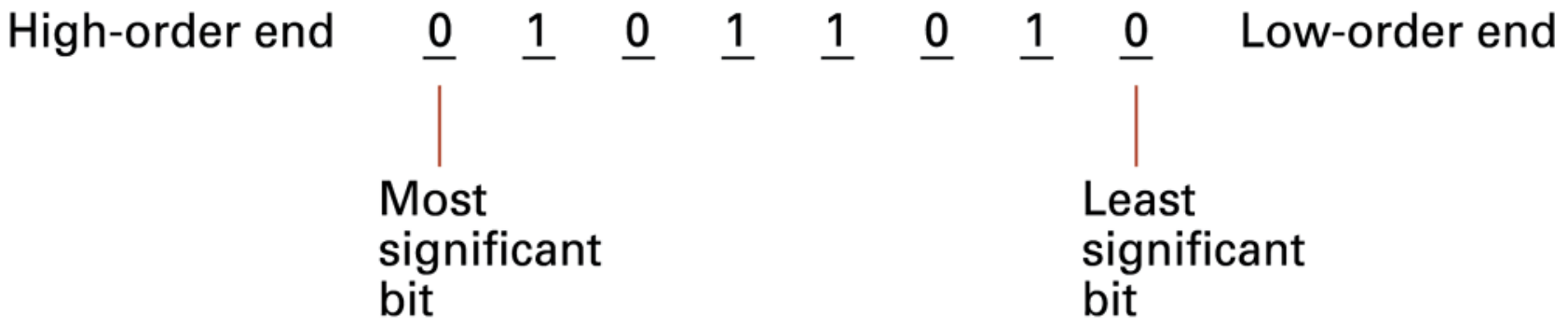


Figure 1.7 The organization of a byte-size memory cell



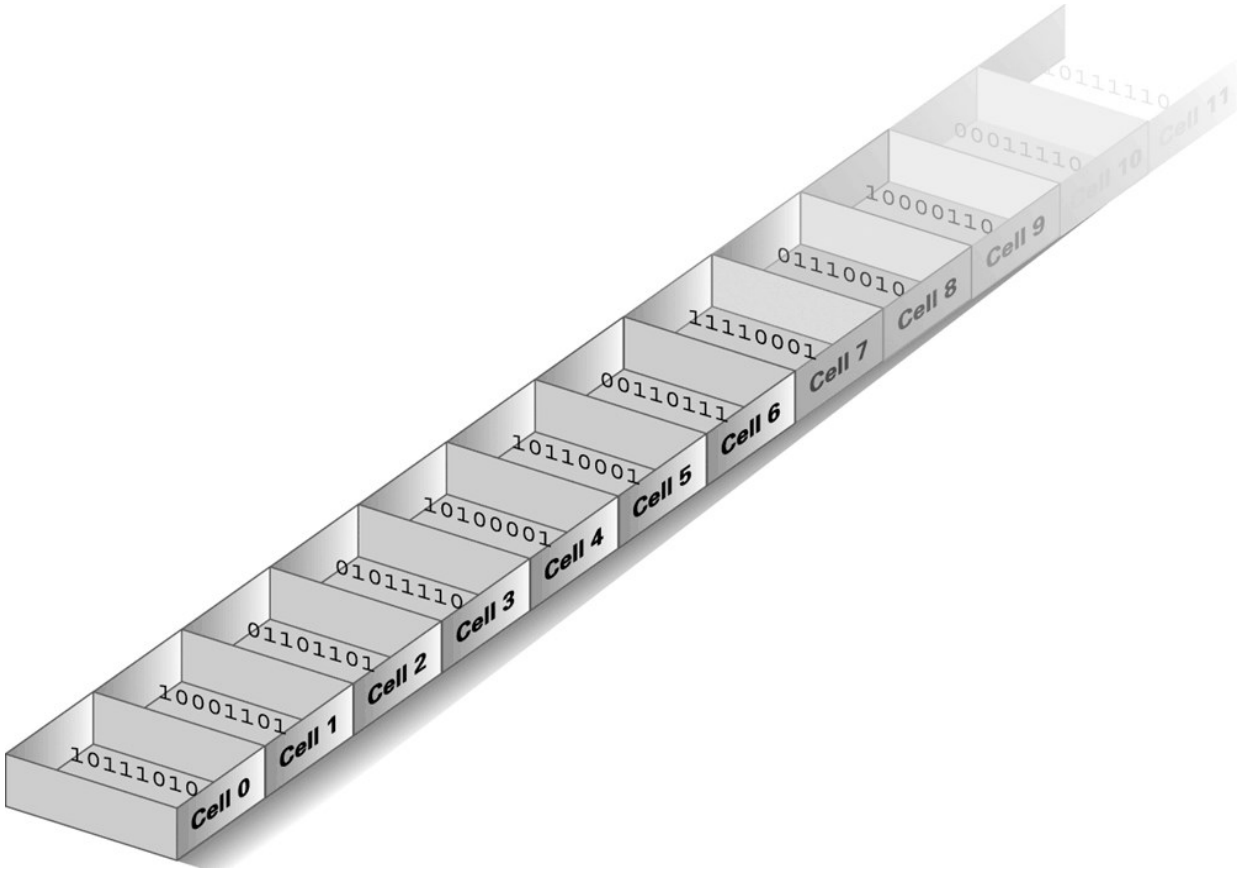
Main Memory Addresses（主存储器地址）

- **Address:** A “name” that uniquely identifies one cell in the computer’s main memory
  - The names are actually numbers.
  - These numbers are assigned consecutively starting at zero.
  - Numbering the cells in this manner associates an order with the memory cells.
- What will it be when the number of cells exceeds the maximum address?

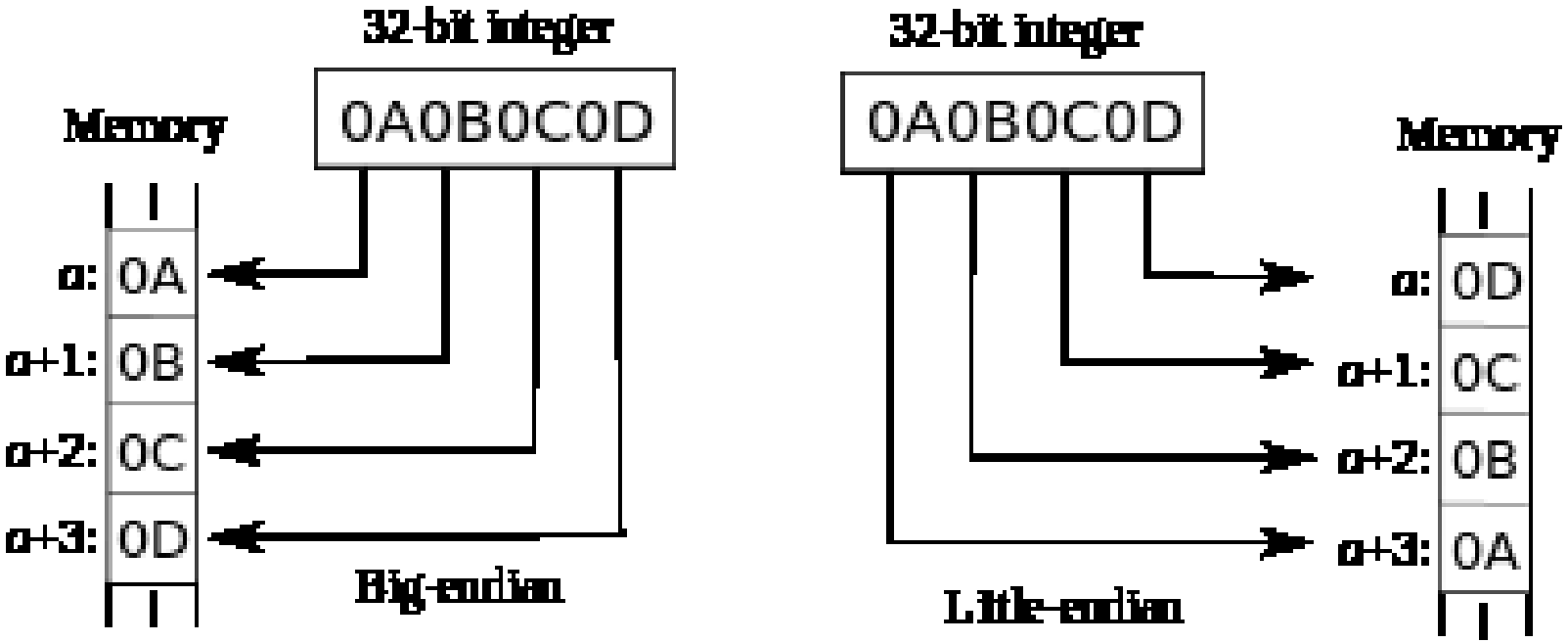
Hexadecimal Notation

- **Hexadecimal notation:** A shorthand notation for long bit patterns
  - Divides a pattern into groups of four bits each
  - Represents each group by a single symbol
- Example: 10100011 becomes A3

Figure 1.8 **Memory cells arranged by address**



Big-endian and Little-endian



- 32位
- 64位

- Questions?

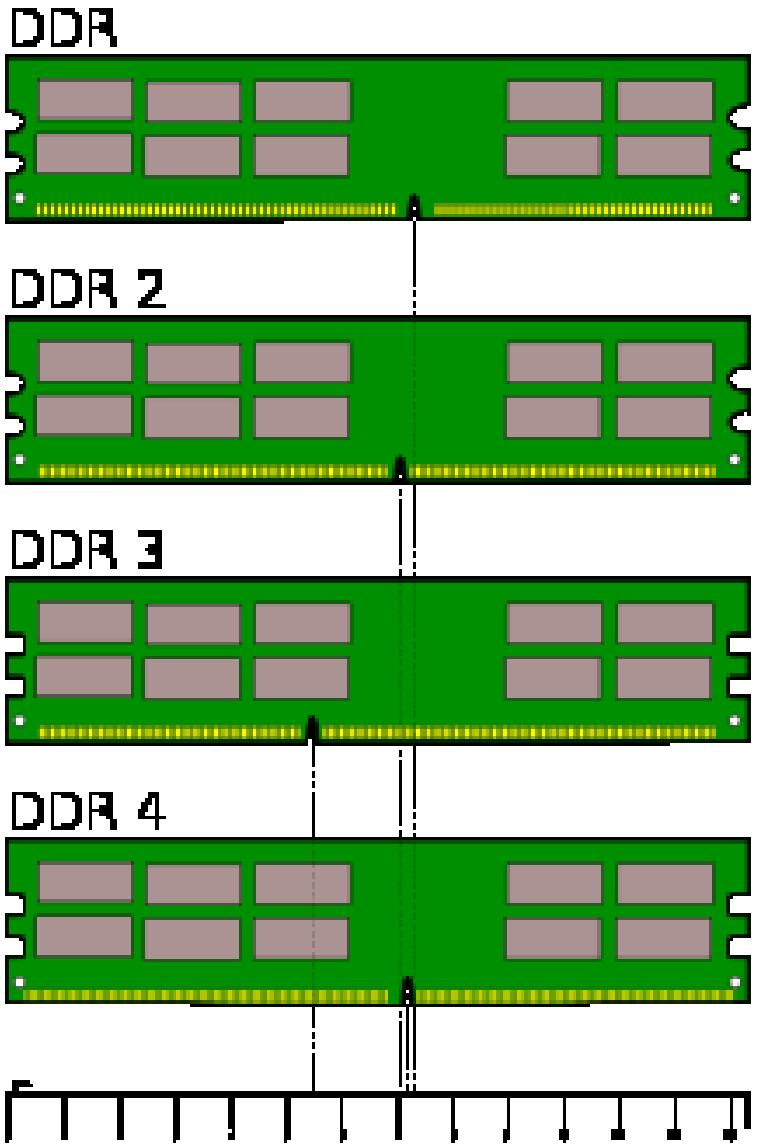
- The storage devices

Memory Categories

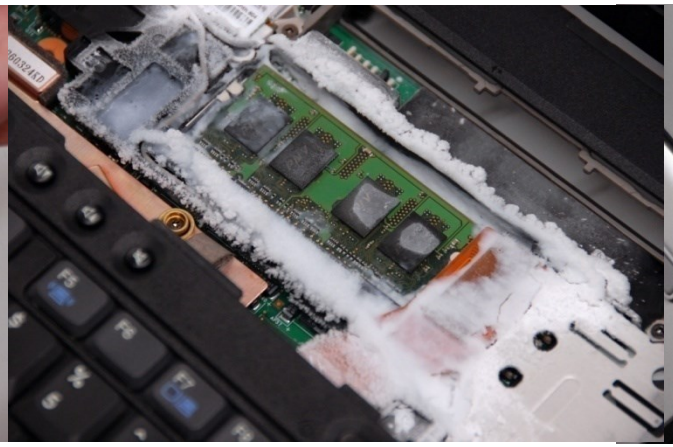
- Volatile memory
- Non-volatile memory
  - For example?

Memory Terminology

- **Random Access Memory (RAM):** Memory in which individual cells can be easily accessed in any order
  - **Dynamic Memory (DRAM)**
  - **Static Memory (SRAM)**


DDR
DDR 2
DDR 3
DDR 4

Freezing Memory Chips

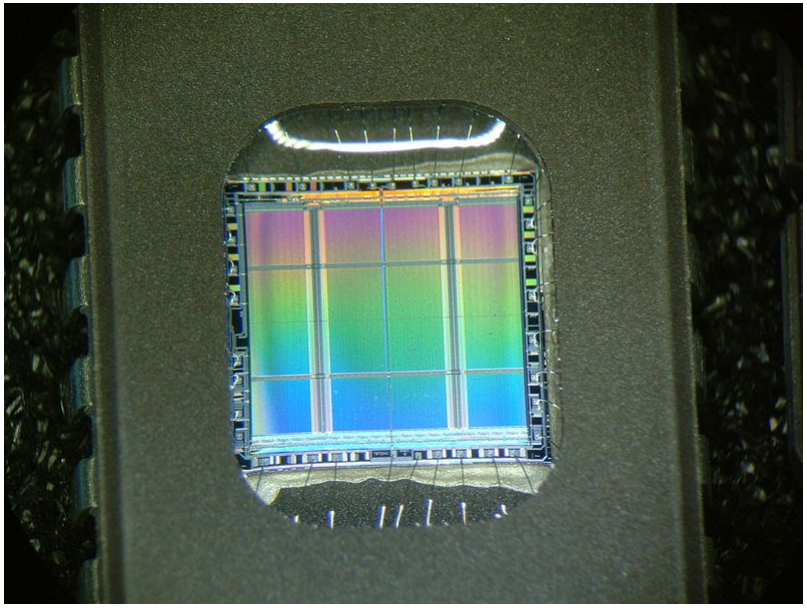


- Freeze a DRAM chip, cause it to retain data for minutes or even hours after the machine loses power

- Hackers can steal information stored in memory by rebooting the compromised machine with a simple program designed to copy the memory contents

Electrically

- Read only (mostly) memory (ROM)



Electrically

- Flash
  - Electrically erasable programmable read-only memory (EEPROM)

SSD solid-state drive

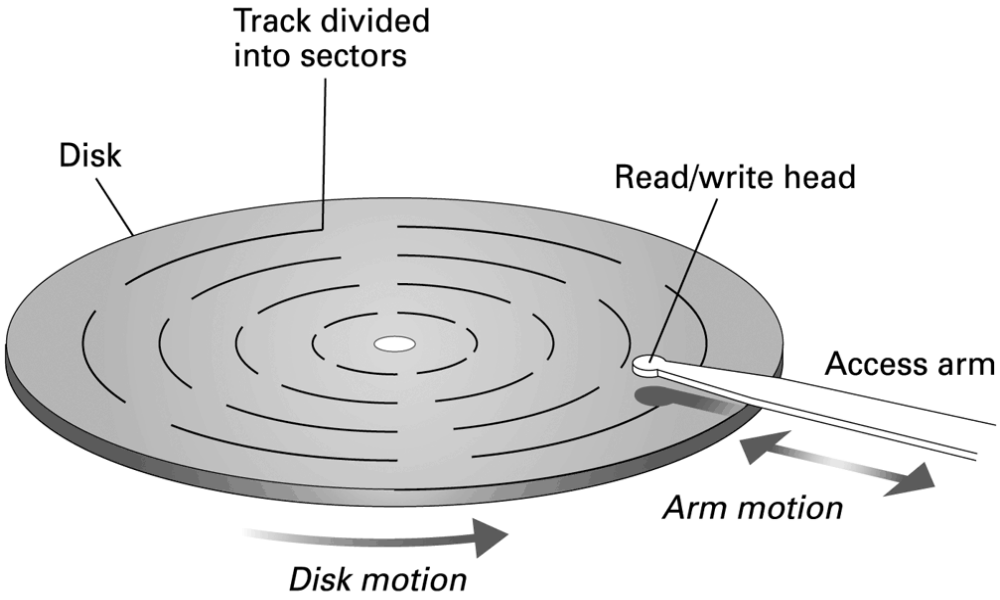
- NAND Flash

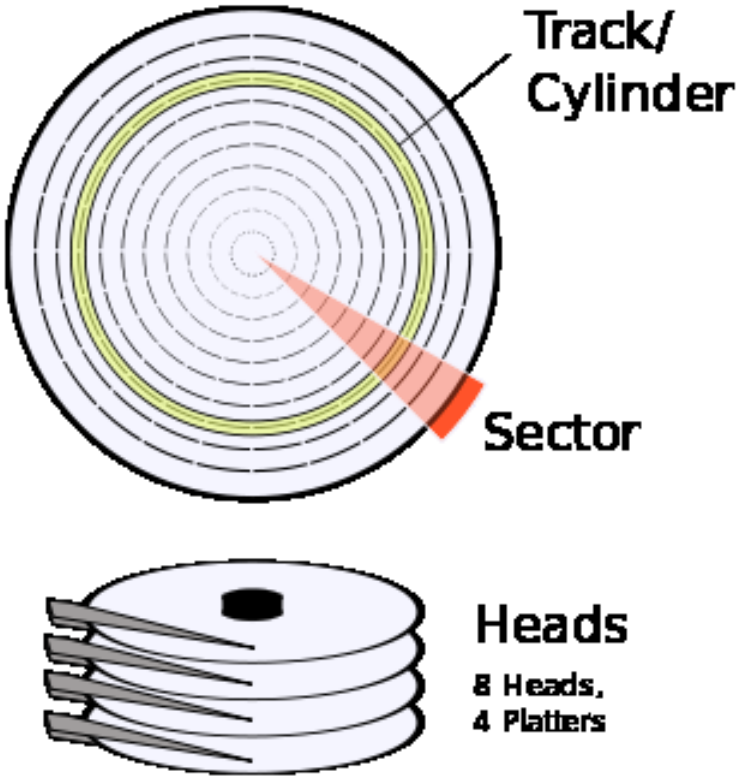
Magnetically or optical

- Magnetic Systems
  - Disk
  - Tape
- Optical Systems
  - CD
  - DVD
  - BD

Hard disk (HDD)

Figure 1.9 A magnetic disk storage system





- Questions?

Chapter 1: Data Storage

- 1.1 Bits and Their Storage
- 1.2 Main Memory
- 1.3 Mass Storage
- 1.4 Representing Information as Bit Patterns
- 1.5 The Binary System
- 1.6 Storing Integers
- 1.7 Storing Fractions
- 1.8 Data Compression
- 1.9 Communications Errors

Binary arithmetic

- Addition
- Subtraction
- Multiplication
- Division

Binary Bitwise Operations

- AND
- OR
- NOT
- XOR
- SHIFT
  - Left shift
  - Right shift
- Question: what is the equivalent of a shift?

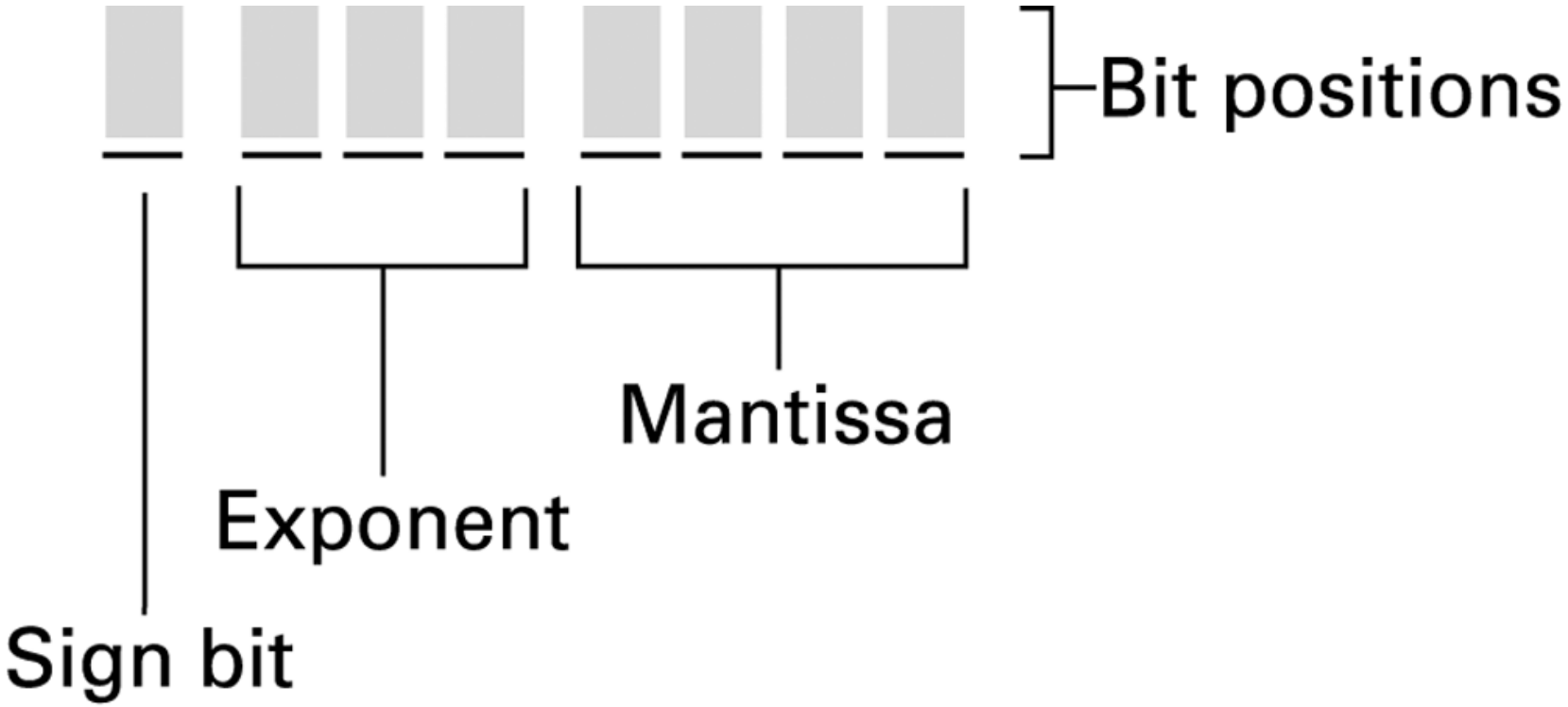
Signed Integer

- Signed magnitude representation (原码)
- 1‘s complement notation (反码)
- 2's complement notation (2补码)
  - 原码和反码的关系?
  - 补码和反码的关系?
  - 补码和原码的关系?

Storing fractions

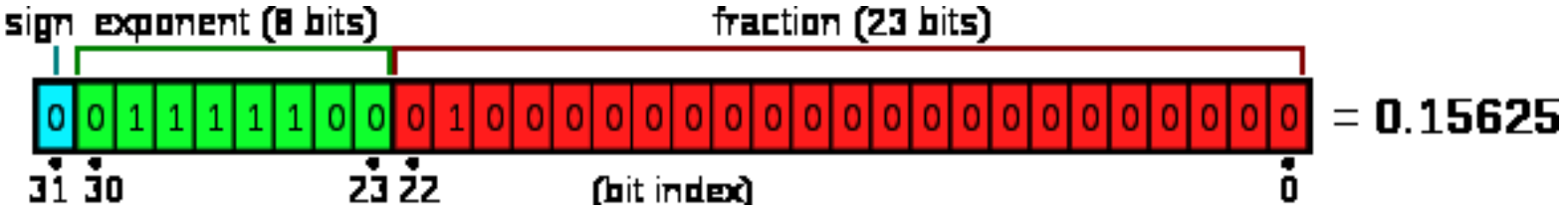
- Why not radix point?
- Can we fix the position of radix point?

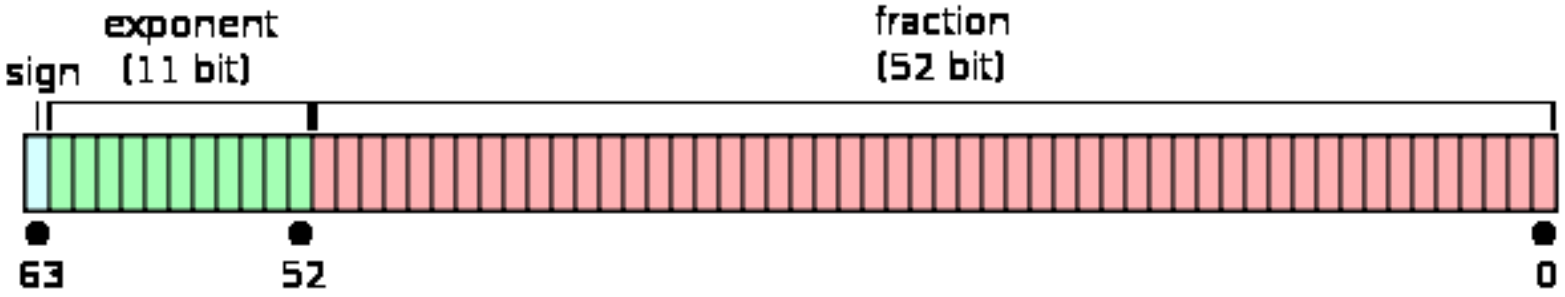
Floating point



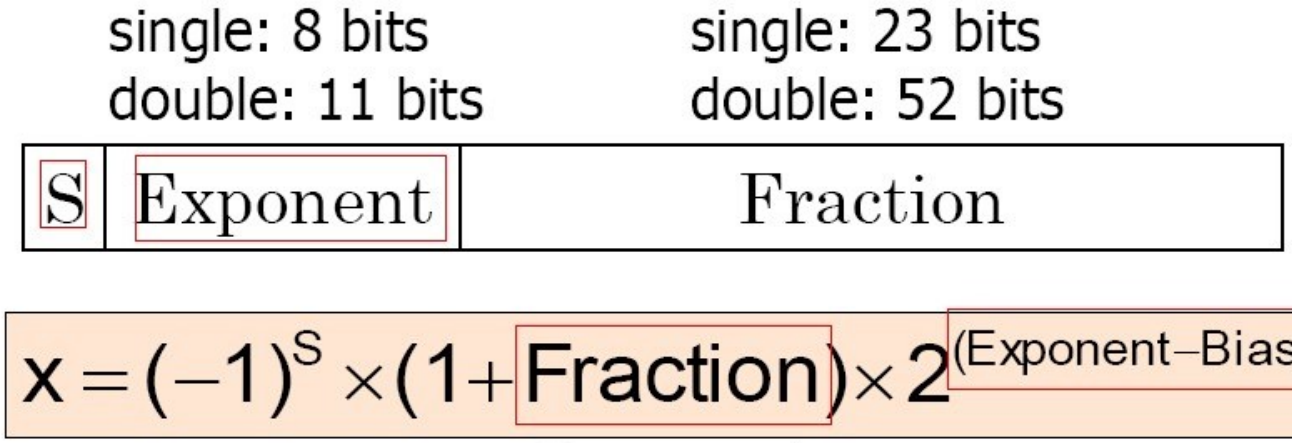
IEEE754

- Float
- Double





IEEE Floating-Point Format



- Exponent: excess representation: actual exponent + Bias
  - Ensures exponent is unsigned
  - Single precision: Bias = 127;
  - Double precision: Bias = 1023

IEEE754

参数	单精度浮点数	双精度浮点数	扩充精度浮点数
浮点数字长	32	64	80
尾数长度	23	52	64
符号位长度	1	1	1
阶码长度	8	11	15
指数偏移量	+127	+1023	+16383
可表示的实数范围	10^{-38}—10^{38}	10^{-308}—10^{308}	10^{-4932}—10^{4932}

Data Compression

- Lossy versus lossless
  - Run-length encoding
  - Frequency-dependent encoding
    (Huffman codes)
  - Relative encoding
  - Dictionary encoding (Includes adaptive dictionary encoding such as LZW encoding.)

Communication Errors

- During transmission, error could happen
  - For example, bit 0 $\rightarrow$1 or bit 1 $\rightarrow$0
- Parity bits (even versus odd) 奇偶校验位
- Checkbytes 校验字节
- Error correcting codes ECC 纠错码

Figure 1.28 The ASCII codes for the letters A and F adjusted for odd parity

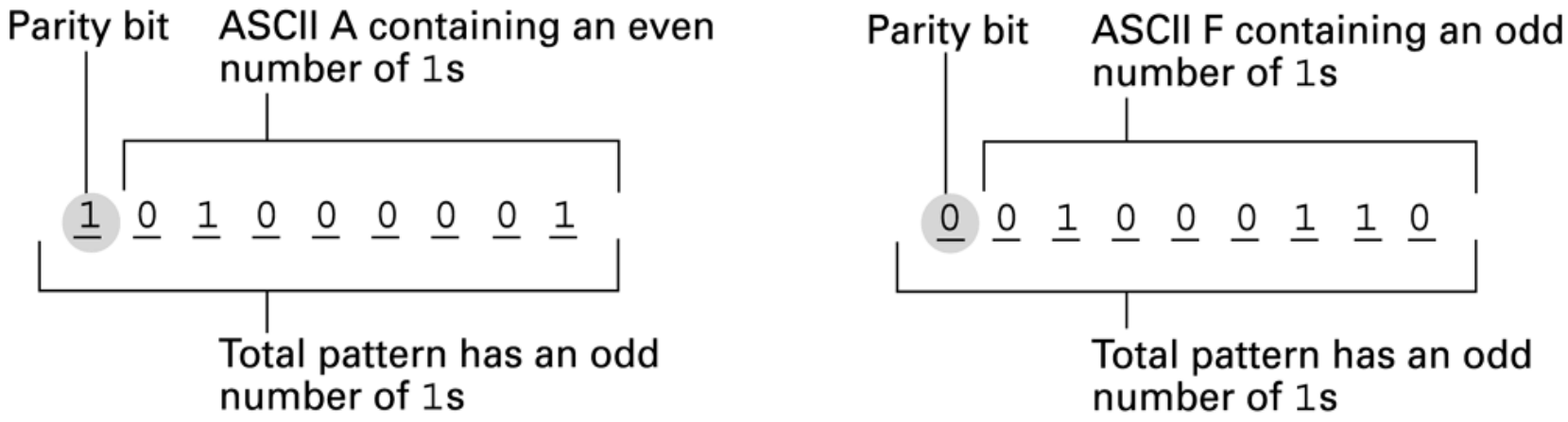


Figure 1.29 An error-correcting code Hamming code

Symbol	Code
A	000000
B	001111
C	010011
D	011100
E	100110
F	101001
G	110101
H	111010

Figure 1.30 Decoding the pattern 010100 using the code in Figure 1.30

Character	Code	Pattern received	Distance between received pattern and code
A	0 0 0 0 0 0	0 **1** 0 **1** 0 0	2
B	0 0 1 1 1 1	0 **1** **0** 1 **0** **0**	4
C	0 1 0 0 1 1	0 1 0 **1** **0** **0**	3
D	0 1 1 1 0 0	0 1 **0** 1 0 0	**1**
E	1 0 0 1 1 0	**0** **1** 0 1 **0** 0	3
F	1 0 1 0 0 1	**0** **1** **0** **1** 0 **0**	5
G	1 1 0 1 0 1	**0** 1 0 1 0 **0**	2
H	1 1 1 0 1 0	**0** 1 **0** **1** **0** 0	4

Hamming

Smallest distance

- Questions?