Computer Programming (66111)

An-Najah N. University
Computer Engineering Department
Luai Malhis, Ph.D,

Introduction To Computer System

Computer System

Definition of a computer

- The computer is an electronic machine that performs the following four general operations:

1. Input
2. Storage
3. Processing
4. Output.

Computer Components

- A computer consists of two main components :
- **Hardware** the mechanical, magnetic, electronic, and electrical components making up a computer system
- **Software**: which are written programs pertaining to the operation of a computer system and that are stored in read/write memory.
- Following is an overview of the main hardware and software components in a computer

Computer Hardware

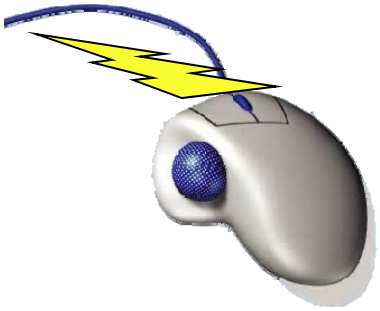
- Input devices
- System unit
- Output devices
- Storage devices
- Processing Unit:

  The CPU and

  Main Memory

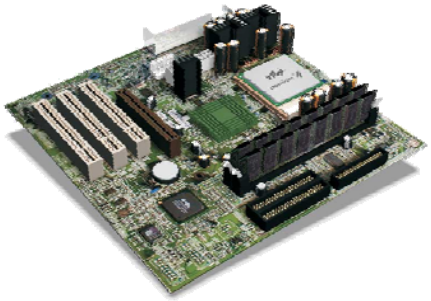
Input Devices

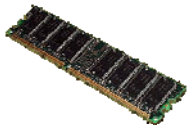
- Enter data to be processed
  - Keyboard
  - Scanners
  - Mouse
  - Trackball
  - Touch screen
  - Microphone
  - Game Controller
  - Digital camera



System Unit

- Cabinet that houses all components
- Motherboard
- CPU
- Memory modules





Output Devices

- Enable us to see or hear the processed information
  - Monitor
  - Speakers
  - Printers

Storage Devices

- Enable us to store data or information to be accessed again

Hard Disk Drive

Floppy Disk

CD / DVD Drive

Flash Drive

The Central processing Unit

- The CPU contains three parts:
- *Arithmetic Logic Unit* - ALU is where the "intelligence" of the computer is located. It performs all arithmetic operations such as addition, subtraction, multiplication and division. The ALU performs logical operations i.e. makes decisions by determining if a number is greater, less, or equal to the other number. An operation completes in nanoseconds, which is a billionth of a second.
- *Registers*: which are small storage devices holds instructions and operands needed by the ALU during operation execution.
- 3. *Control Unit* - This is the part of the unit, which directs information to the proper places in your computer, such as calculation of information by the ALU unit or to store and print material.

The Memory Unit

- The Main Memory:
- Two types of memory contained on a chip are ROM (Read Only Memory) or RAM (Random Access Memory).
- *ROM* memory is installed on a computer by the manufacturer and can not be altered. ROM is the memory that determines all the basic functions of the operation of rge computer such as startup, shut down, and placing a character on the screen.
- *RAM* is temporary memory, which stores programs during execution and also hold all information displayed on the monitor. RAM is read/write memory and it is much larger in size than ROM. Data disappears from the RAM when the computer is turned off or power is off.

Computer Software

- **Software** - programs that enable the hardware to perform different tasks
- Application software
  - Tools for getting things done

Computer Software

- System software
  - Essential for platform operation and support

Computer Platforms: PCs and Macs

PC

- CPU – Intel, AMD
- Operating system – Microsoft Windows

Mac

- CPU – Motorola
- Operating system – Apple Mac OS

Application Software

- Used to accomplish specific tasks other than just running the computer system.
- May consist of a single program, such as an image viewer;
- A small collection of programs (often called a software package) that work closely together to accomplish a task.
- Independent programs and packages that have a common user interface or shared data format, such as Microsoft Office.

Programming languages

- The machine language, which is the only languages understood by CPU. While easily understood by the CPU, the machine language is almost impossible for humans to use because they consist entirely of 0's and 1's.

- A assembly language contains the same instructions as a machine language, but the instructions and variables have names instead of being just 0's and 1's.

- High Level language Closely resemble human language Examples of high level languages are : Pascal, Fortran, Basic, Java, and C/C++. Programs written in high-level languages are translated into machine language by a compiler.

C/C++ Programming Lnaguage

- History of C
  - Evolved from two other programming languages
    - BCPL and B
      - “Typeless” languages
  - Dennis Ritchie (Bell Laboratories)
    - Added data typing, other features
  - Hardware independent
    - Portable programs
  - 1989: ANSI standard
  - The C Lnaguage is Then developed to contain classes and other object Oriented features and named as C++.
  - Many Other Lnaguages currently developed that uses a synatx and symantics like C.
  - C/C++ is traditionally the first language a programmer learns.

How C++ Works

Programs are written by humans.

Programs are run on computers.

C++ programs are written by humans and translates into machine language by the C++ compiler.

C++ Programmer

```
// Compute the area
// of a triangle
area =
    (base * height) /
```

C++ Compiler

Machine Language

```
0010 1101
1101 1000
1001 1000
0001 0010
0010 0000
```

What is a program

- A program is a set of instructions that a computer follows.

- Example: computing the Area of Rectangle
  Get base Get Height
  Area = 0.5 * base * height

- Steps to writing a program:
  Step 1. Think! (This is not optional.)
  Step 2. Organize your thoughts
  Step 3. Write them down in English
  Step 4. Translate them into C++

Program Construction

- **Text Editor**

This is used to create the program in C++ form. Since this is the start or source of the other forms this is called a *source file. (Source files end with* .cpp. -- also used C and .cc.)

- **Compiler**

This translates the *source file into a machine dependent file called an object file. The object file contains the instructions in a way that the* machine can understand. The source file is in the C++ language (high level code) while the object file is in machine language (low level code.)

- **Link**

  This is used to associate the object file with other necessary files to generate an EXE file Which contains the machine language.

Data vs. Information

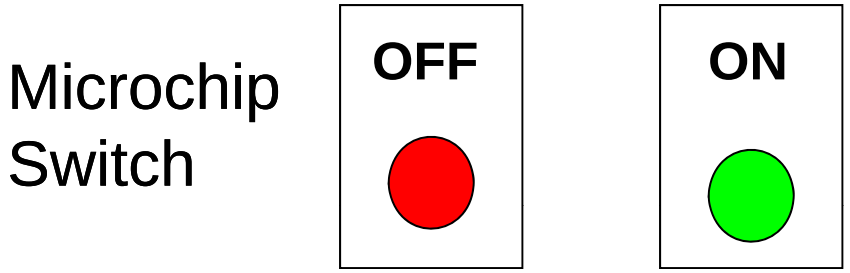
- Data vs. Information:
  - Data is a representation of a fact or idea
    - Number
    - Word
    - Picture
    - Sound
  - Information is data that has been organized or presented in a meaningful fashion.

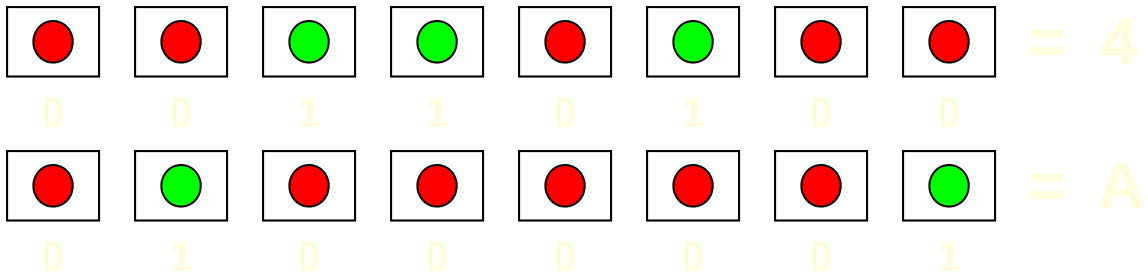
Computers are Data Processing Devices

- Four major functions:
  - Input data
  - Process data
  - Output information
  - Store data and information

Bits and Bytes:
The Language of Computers

- Bit
  - Binary digit
  - 0 or 1
- Byte
  - Eight bits
- ASCII
  - Each byte represents a letter, number or special character





How Much is a Byte?

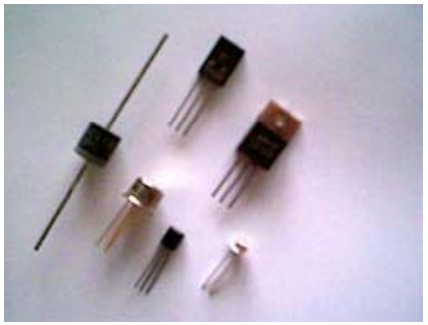
NAME	ABBREVIATION	NUMBER OF BYTES	RELATIVE SIZE
Byte	B	1 byte	Can hold one character of data.
Kilobyte	KB	1,024 bytes	Can hold 1,024 characters or about half of a typewritten page double-spaced.
Megabyte	MB	1,048,576 bytes	A floppy disk holds approximately 1.4 MB of data, or approximately 768 pages of typed text.
Gigabyte	GB	1,073,741,824 bytes	Approximately 786,432 pages of text. Since 500 sheets of paper is approximately 2 inches, this represents a stack of paper 262 feet high.
Terabyte	TB	1,099,511,627,776 bytes	This represents a stack of typewritten pages almost 51 miles high.
Petabyte	PB	1,125,899,906,842,624 bytes	The stack of pages is now 52,000 miles high, or about one-fourth the distance from the Earth to the moon.

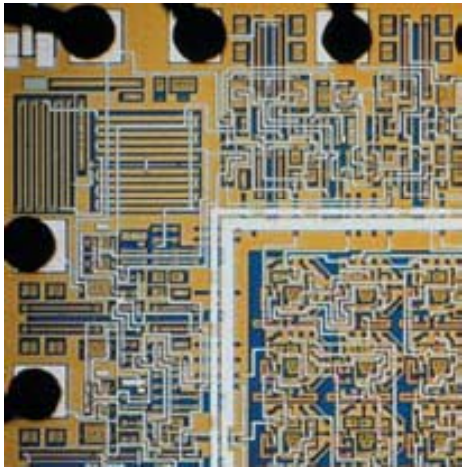
Binary Language

- Computers work in binary language
- Consists of two numbers: 0 and 1
- Everything a computer does is broken down into a series of 0s and 1s
- Switches: Devices inside the computer that can be flipped between these two states: 1 or 0, on or off

Switches

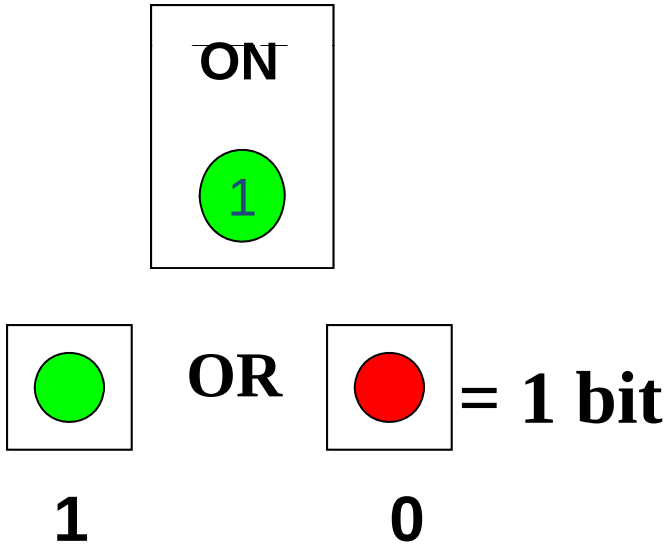
- Non-mechanical devices in computers that open and close circuits
- Types of electronic switches:
  - Vacuum tubes
  - Transistors:
    - Semiconductors
  - Integrated circuits





Switches Representing Data

- The on/off state of a switch represents one bit of data
- Bit (binary digit)
  - On = 1
  - Off = 0



The Binary Number System

- Describes a number as powers of 2
- Also referred to as base 2 numbering system
- Used to represent *every* piece of data stored in a computer: all of the numbers, letters, and instructions

The Binary Number System

- Number systems are organized ways to represent numbers
- Each number in one system has a corresponding number in another.

	128 2x64	**64** 2x32	**32** 2x16	**16** 2x8	**8** 2x4	**4** 2x2	**2** 2x1	**1**	
Binary	0	1	0	1	1	0	0	1	
Base 10	0 +	64 +	0 +	16 +	8 +	0 +	0 +	1 =	89

01011001 = 89

Binary Base 10

Understanding Decimal Numbers

- Decimal numbers are made of decimal digits: (0,1,2,3,4,5,6,7,8,9)
- But how many items does a decimal number represent?
  - $8653 = 8x10^3 + 6x10^{2+} 5x10^{1+} 3x10^0$
- What about fractions?
  - $97654.35 = 9x10^4 + 7x10^{3+} 6x10^{2+} 5x10^{1+} 4x10^0 + 3x10^{-1+} 5x10^{-2}$
  - In formal notation -> $(97654.35)_{10}$
- Why do we use 10 digits, anyway?

Understanding Binary Numbers

- Binary numbers are made of binary digits (bits):
  - 0 and 1
- How many items does an binary number represent?
  - $(1011)_2 = 1x2^3 + 0x2^{2+} 1x2^{1+} 1x2^0 = (11)_{10}$
  - $(1010010) = 64+16+2 = (82)_{10}$
  - $(100010001) = 256+ 16+1 = (273)_{10}$

Convert *from* Decimal *to* binary

For each digit position:

1. **Divide decimal number by the base (e.g. 2)**
2. **The *remainder* is the lowest-order digit**
3. **Repeat first two steps until no *divisor* remains.**

Example for $(13)_{10}$:

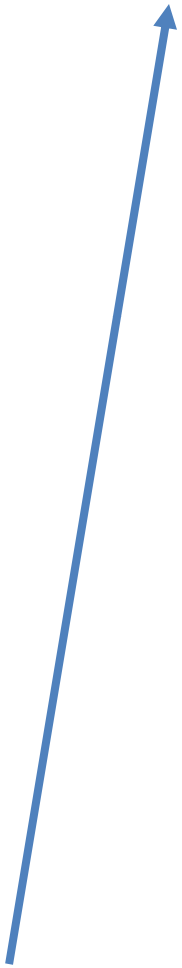
	Integer Quotient		Remainder	Coefficient
13/2 =	6	+	½	$a_0 = 1$
6/2 =	3	+	0	$a_1 = 0$
3/2 =	1	+	½	$a_2 = 1$
1/2 =	0	+	½	$a_3 = 1$

Answer $(13)_{10} = (a_3\, a_2\, a_1\, a_0)_2 = (1101)_2$

The Growth of Binary Numbers

N is the number of bits in the binary number

n	2^n
0	$2^0=1$
1	$2^1=2$
2	$2^2=4$
3	$2^3=8$
4	$2^4=16$
5	$2^5=32$
6	$2^6=64$
7	$2^7=128$



n	2^n	
8	$2^8=256$	
9	$2^9=512$	
10	$2^{10}=1024$	
11	$2^{11}=2048$	
12	$2^{12}=4096$	
20	$2^{20}=1M$	Mega
30	$2^{30}=1G$	Giga
40	$2^{40}=1T$	Tera

Understanding Octal Numbers

- Octal numbers are made of octal digits: (0,1,2,3,4,5,6,7)
- How many items does an octal number represent?
  - $(4536)_8 = 4x8^3 + 5x8^2 + 3x8^1 + 6x8^0 = (1362)_{10}$
- What about fractions?
  - $(465.27)_8 = 4x8^2 + 6x8^1 + 5x8^0 + 2x8^{-1} + 7x8^{-2}$
- Octal numbers don't use digits 8 or 9

Convert an Integer *from* Decimal *to* Octal

For each digit position:

1. **Divide decimal number by the base (8)**
2. **The *remainder* is the lowest-order digit**
3. **Repeat first two steps until no *divisor* remains.**

Example for $(175)_{10}$:

	Integer Quotient		Remainder	Coefficient
175/8 =	21	+	7/8	$a_0 = 7$
21/8 =	2	+	5/8	$a_1 = 5$
2/8 =	0	+	2/8	$a_2 = 2$

Answer $(175)_{10} = (a_2\, a_1\, a_0)_2 = (257)_8$

Understanding Hexadecimal Numbers

- Hexadecimal numbers are made of 16 digits:
  - (0,1,2,3,4,5,6,7,8,9,A, B, C, D, E, F)
- How many items does an hex number represent?
  - $(3A9F)_{16} = 3x16^3 + 10x16^2 + 9x16^1 + 15x16^0 = 14999_{10}$
- What about fractions?
  - $(2D3.5)_{16} = 2x16^2 + 13x16^1 + 3x16^0 + 5x16^{-1} = 723.3125_{10}$
- Note that *each* hexadecimal digit can be represented with four bits.
  - $(1110)_2 = (E)_{16}$

Converting Between Base 16 and Base 2

$3A9F_{16} = \underline{0011}\ \underline{1010}\ \underline{1001}\ \underline{1111}_2$

3 A 9 F

- **Conversion is easy!**
  - **Determine 4-bit value for each hex digit**
- **Note that there are $2^4 = 16$ different values of four bits**
- **Easier to read and write in hexadecimal.**
- **Representations are equivalent!**

Converting Between Base 16 and Base 8

$3A9F_{16}$ = 0011 1010 1001 1111_2

3 A 9 F

35237_8 = 011 101 010 011 111_2

3 5 2 3 7

Convert from Base 8 to Base 2

1. **Regroup bits into groups of three starting from right**
2. **Ignore leading zeros**
3. **Each group of three bits forms an octal digit.**

Number System Conversion Table

Dec	Bin	Oct	Hex
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F

Representing Letters and Symbols

- There are codes that dictate how to represent characters in binary format. Most of today's computers use the **American Standard Code for Information Interchange (ASCII code)** to represent each letter or character as an 8-bit (or 1-byte) binary code.
- The ASCII code represents the 26 uppercase letters and 26 lowercase letters used in the English language, along with a number of punctuation symbols and other special characters, using 8 bits. Eight bits is the standard length upon which computers are built.
- In the ASCII The representation for A is 41H (65) ,

  B is 42H (66), a is 61H (97) and b is 62h (98). The Complete table is shown next slide

ASCII Chart

Character	Decimal Number	Binary Number
blank space	32	$0010\ 0000_2$
!	33	$0010\ 0001_2$
"	34	$0010\ 0010_2$
#	35	$0010\ 0011_2$
$	36	$0010\ 0100_2$
A	65	$0100\ 0001_2$
B	66	$0100\ 0010_2$
C	67	$0100\ 0011_2$
D	68	$0100\ 0100_2$
E	69	$0100\ 0101_2$
F	70	$0100\ 0110_2$
G	71	$0100\ 0111_2$
H	72	$0100\ 1000_2$
I	73	$0100\ 1001_2$
J	74	$0100\ 1010_2$
K	75	$0100\ 1011_2$
L	76	$0100\ 1100_2$
M	77	$0100\ 1101_2$
N	78	$0100\ 1110_2$
O	79	$0100\ 1111_2$
P	80	$0101\ 0000_2$
Q	81	$0101\ 0001_2$
R	82	$0101\ 0010_2$
S	83	$0101\ 0011_2$
T	84	$0101\ 0100_2$
U	85	$0101\ 0101_2$
V	86	$0101\ 0110_2$
W	87	$0101\ 0111_2$
X	88	$0101\ 1000_2$
Y	89	$0101\ 1001_2$
Z	90	$0101\ 1010_2$
[	91	$0101\ 1011_2$
\	92	$0101\ 1100_2$
]	93	$0101\ 1101_2$

Character	Decimal Number	Binary Number
^	94	$0101\ 1110_2$
_	95	$0101\ 1111_2$
`	96	$0110\ 0000_2$
a	97	$0110\ 0001_2$
b	98	$0110\ 0010_2$
c	99	$0110\ 0011_2$
d	100	$0110\ 0100_2$
e	101	$0110\ 0101_2$
f	102	$0110\ 0110_2$
g	103	$0110\ 0111_2$
h	104	$0110\ 1000_2$
i	105	$0110\ 1001_2$
j	106	$0110\ 1010_2$
k	107	$0110\ 1011_2$
l	108	$0110\ 1100_2$
m	109	$0110\ 1101_2$
n	110	$0110\ 1110_2$
o	111	$0110\ 1111_2$
p	112	$0111\ 0000_2$
q	113	$0111\ 0001_2$
r	114	$0111\ 0010_2$
s	115	$0111\ 0011_2$
t	116	$0111\ 0100_2$
u	117	$0111\ 0101_2$
v	118	$0111\ 0110_2$
w	119	$0111\ 0111_2$
x	120	$0111\ 1000_2$
y	121	$0111\ 1001_2$
z	122	$0111\ 1010_2$
{	123	$0111\ 1011_2$
|	124	$0111\ 1100_2$
}	125	$0111\ 1101_2$
~	126	$0111\ 1110_2$