DIGITAL Logic design

LECTURE 1

Reference Books

- **Digital Logic and Computer Design, *M. Morris Mano,***
- **Digital Systems Principles and Applications byJ. Tocci**
- ***Practical Digital Logic Design and Testing*, P K Lala, Prentice Hall, 1996**

NUMERICAL REPRESENTATIONS

- In science and technology we are constantly dealing with quantities.
- Quantities are measured, monitored, recorded, manipulated arithmetically, observed, or in some other way utilized in most physical systems.
- It is important when dealing with various quantities that we be able to represent their values efficiently and accurately.
- There are basically two ways of representing the numerical value of quantities:
- **Analog and Digital.**

Analog Representations

- In **analog representation** a quantity is represented by a continuously variable, proportional indicator.
- An example is an automobile speedometer from the classic cars of the 1960s and 1970s.
- The deflection of the needle is proportional to the speed of the car and follows any changes that occur as the vehicle speeds up or slows down.
- Analog quantities such as those cited above have an important characteristic, no matter how they are represented: ***they can vary over a continuous range of values***.
- The automobile speed can have *any* value between zero and, say, 100.
- Similarly, the microphone output might have any value within a range of zero to 10.

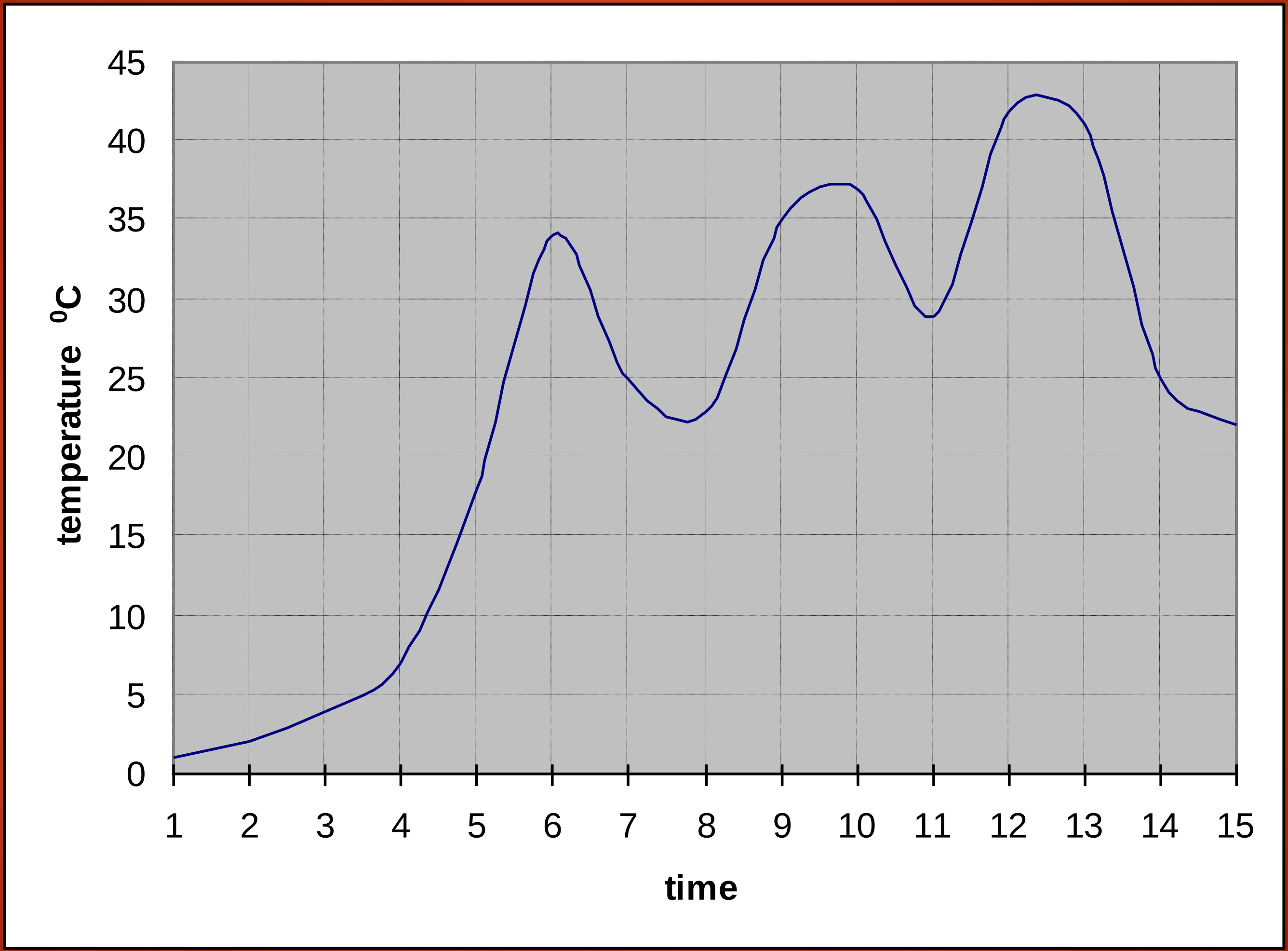
Analog Representations

- Most of the quantities in nature that can be measured are continuous. Examples include
- **Intensity of light during the day :** The intensity of light gradually increases as the sun rises in the morning; it remains constant throughout the day and then gradually decreases as the sun sets until it becomes completely dark.
- The change in the light throughout the day is gradual and continuous.
- **Rise and fall in temperature during a 24-hour period:** The temperature also rises and falls with the passage of time during the day and in the night.
- The change in temperature is never abrupt but gradual and continuous.
- **Velocity of a car travelling from A to B:** The velocity of a car travelling from one city to another varies in a continuous manner.

Analog Representations

- Sound is an example of a physical quantity that can be represented by an electrical analog signal.
- A microphone is a device that generates an output voltage that is proportional to the amplitude of the sound waves that strike it.
- Variations in the sound waves will produce variations in the microphone's output voltage.
- Tape recordings can then store sound waves by using the output voltage of the microphone to proportionally change the magnetic field on the tape.

Continuous Signal



Digital Representations

- In **digital representation** the quantities are represented not by continuously variable indicators but by symbols called *digits*.
- As an example, consider the digital clock, which provides the time of day in the form of decimal digits that represent hours and minutes.
- The major difference between analog and digital quantities, then, can be simply stated as follows:
- Analog → continuous
- Digital → discrete (step by step)
- Because of the discrete nature of digital representations, there is no ambiguity when reading the value of a digital quantity, whereas the value of an analog quantity is often open to interpretation.

Digital Representations

- Two Voltage Levels
- Two States
  - On/Off
  - Black/White
  - Hot/Cold
  - Stationary/Moving

difference between Analog and Digital quantities.

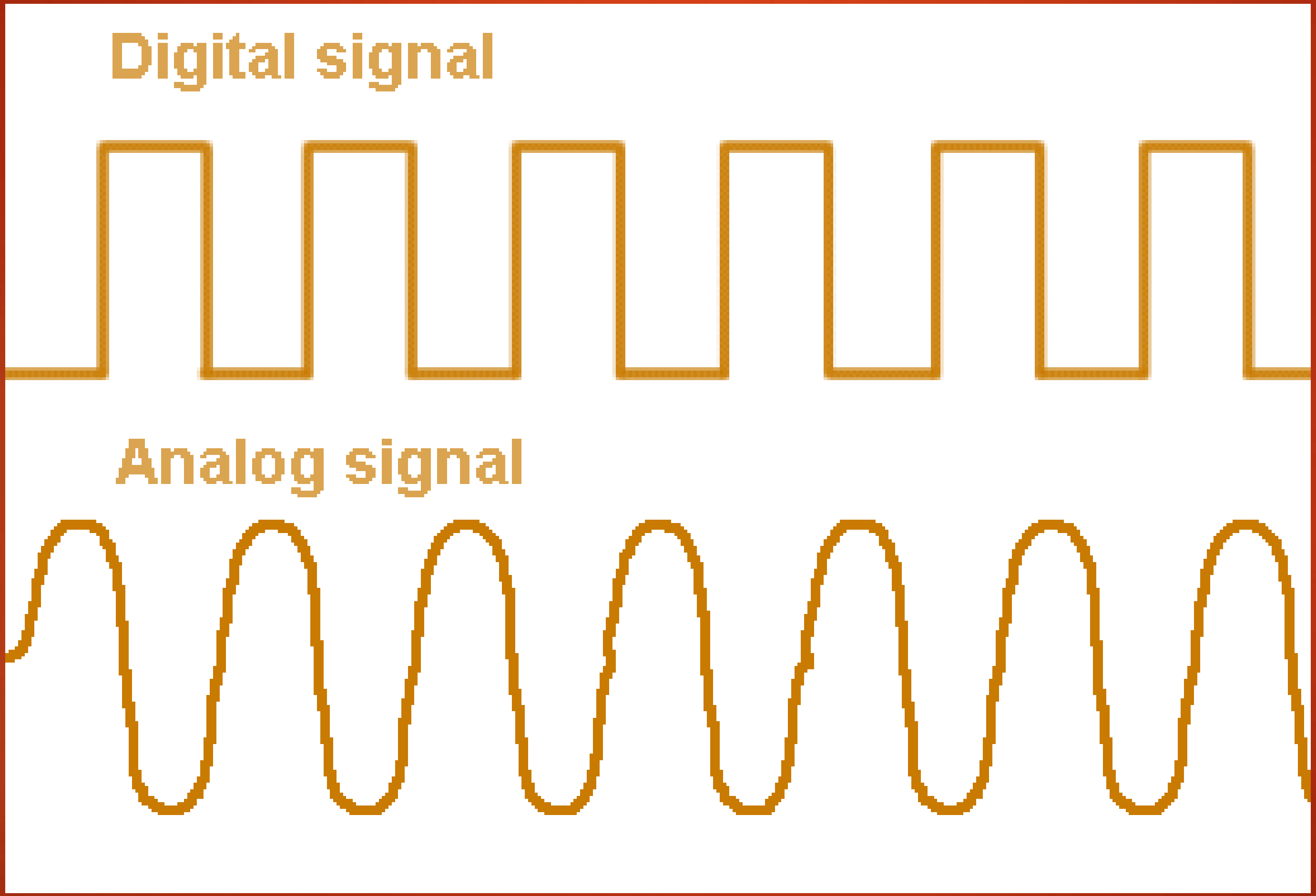
DIGITAL AND ANALOG SYSTEMS

- A **digital system** is a combination of devices designed to manipulate logical information or physical quantities that are represented in digital form; that is, the quantities can take on only discrete values.
- An **analog system** contains devices that manipulate physical quantities that are represented in analog form.
- In an analog system, the quantities can vary over a continuous range of values.

DIGITAL AND ANALOG SYSTEMS

- Digital systems are designed to work with two voltage values.
- A +5 volts represents a logic high state or logic 1 state and 0 volts represents a logic low state or logic 0 state.
- The Digital Systems which are based on two voltage values or two states can easily represent any two values.
- A Digital system such as a computer not only handles numbers but all kinds of information.
- Numbers
- Text
- Formula and Equations
- Drawings and Pictures
- Sound and Music

DIGITAL AND ANALOG SYSTEMS



Advantages of Digital Techniques

- ***Digital systems are generally easier to design***. The circuits used in digital systems are *switching circuits,* where *exact* values of voltage or current are not important, only the range (HIGH or LOW) in which they fall.
- ***Information storage is easy***. This is accomplished by special devices and circuits that can latch onto digital information and hold it for as long as necessary, and mass storage techniques that can store billions of bits of information in a relatively small physical space.
- ***Accuracy and precision*** *are easier to maintain throughout the system.*
- Once a signal is digitized, the information it contains does not deteriorate as it is processed.
- In analog systems, the voltage and current signals tend to be distorted by the effects of temperature, humidity, and component tolerance variations in the circuits that process the signal.

Advantages of Digital Techniques

- ***Operation can be programmed.*** It is fairly easy to design digital systems whose operation is controlled by a set of stored instructions called a *program*.
- ***Digital circuits are less affected by noise.*** Minute fluctuations in voltage (noise) are not as critical in digital systems because the exact value of a voltage is not important, as long as the noise is not large enough to prevent us from distinguishing a HIGH from a LOW.
- ***More digital circuitry can be fabricated on IC chips.*** It is true that analog circuitry has also benefited from the tremendous development of IC technology, but its relative complexity and its use of devices that cannot be economically integrated (high-value capacitors, precision resistors, inductors, transformers) have prevented analog systems from achieving the same high degree of integration.

Advantages of Digital Techniques

- Efficient Processing & Data Storage
- Efficient & Reliable Transmission
- Detection and Correction of Errors
- Easy Design and Implementation
- Occupy minimum space

Limitations of Digital Techniques

- **The real world is analog.**
- **Processing digitized signals takes time.**
- To take advantage of digital techniques when dealing with analog inputs and outputs, four steps must be followed:
- 1. Convert the physical variable to an electrical signal (analog).
- 2. Convert the electrical (analog) signal into digital form.
- 3. Process (operate on) the digital information.
- 4. Convert the digital outputs back to real-world analog form.

Limitations of Digital Techniques

- To illustrate a typical system that uses this approach Figure 1-1 describes a precision temperature regulation system.
- A user pushes up or down buttons to set the desired temperature in increments (digital representation).
- A temperature sensor in the heated space converts the measured temperature to a proportional voltage.
- This analog voltage is converted to a digital quantity by an **analog-to-digital converter (ADC)**.
- This value is then compared to the desired value and used to determine a digital value of how much heat is needed.
- The digital value is converted to an analog quantity (voltage) by a **digital-to-analog converter (DAC)**.
- This voltage is applied to a heating element, which will produce heat that is related to the voltage applied and will affect the temperature of the space.

Limitations of Digital Techniques

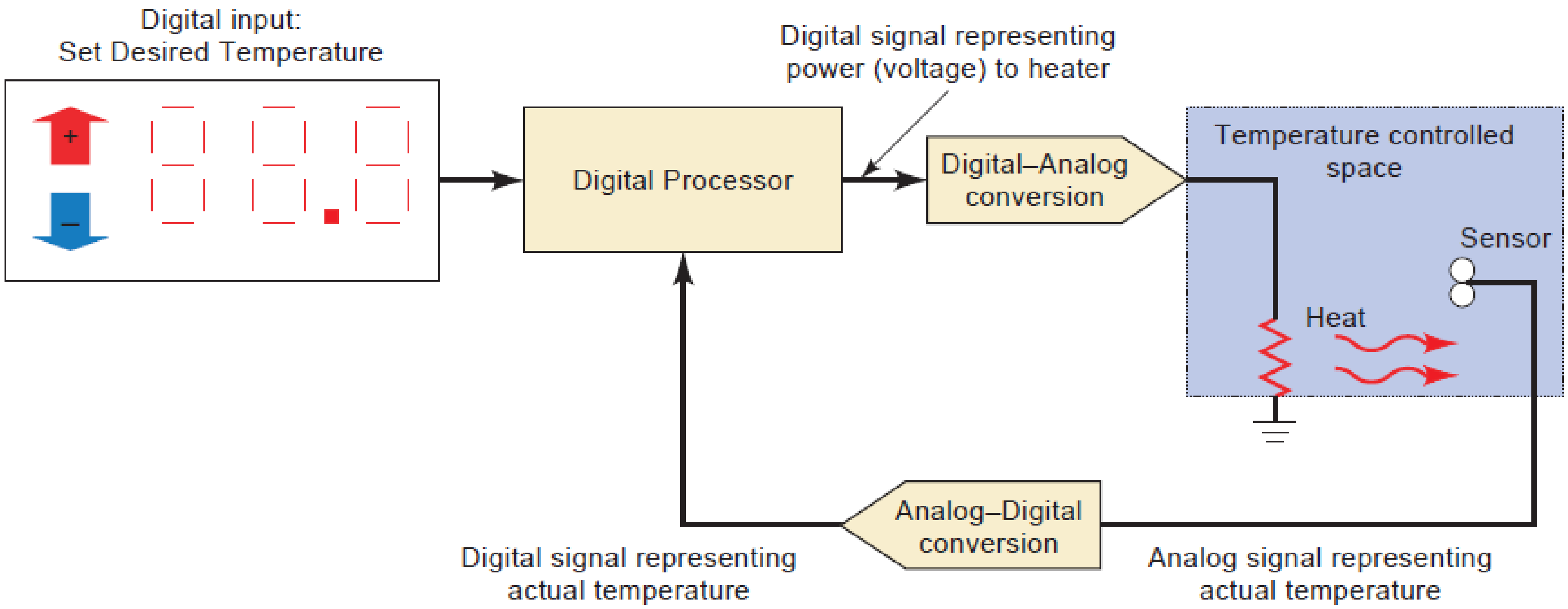
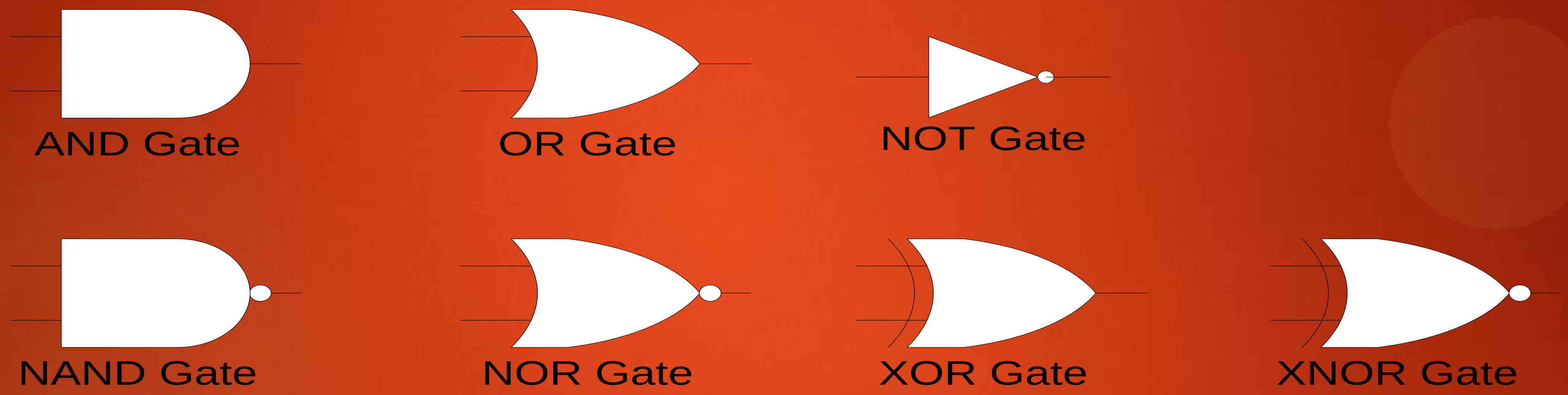


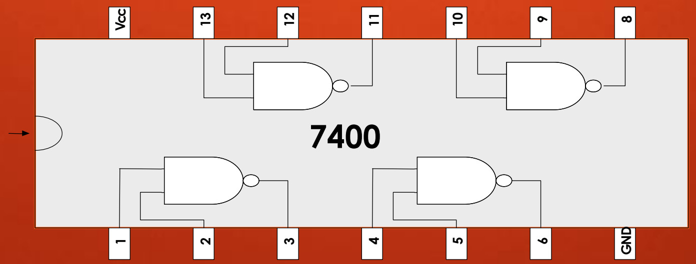
FIGURE 1-1 Block diagram of a precision digital temperature control system.

Digital System

- Digital system process binary information electronically through specialized circuits designed for handling digital information.
- These circuits operate with two voltage values of +5 volts and 0 volts.
- These specialized electronic circuits are known as Logic Gates and are considered to be the Basic Building Blocks of any Digital circuit.
- The commonly used Logic Gates are the AND gate, the OR gate and the Inverter or NOT Gate.
- Other gates that are frequently used include NOR, NAND, XOR and XNOR.
- Each of these gates is designed to perform a unique operation on the input information which is known as a logical or Boolean operation.

Logic Gate Symbol and ICs





NAND Gate IC

Digital Systems

- Discrete elements of information are represented in a digital system by physical quantities called signals.
- Electrical signals such as voltages and currents are the most common.
- Electronic devices called transistors predominate in the circuitry that implements these signals.
- The signals in most present-day electronic digital systems use just two discrete values and are therefore said to be *binary*.
- A binary digit, called a *bit,* has two values: 0 and 1.
- Discrete elements of information are represented with groups of bits called *binary codes*.

Digital Computer

- The computer deals with digital information, i.e., it deals with the information that is represented by binary digits
- Digital Computers function more reliably if only two states are used
- Digital Computers use binary number system (0,1)
- Information is represented in digital computers in group of bits
- Instructions are also represented in group of bits that performs different computations
- Suppose we have a binary group of bits $(1001011)_2=(75)$
- However the same code is also use to represent an alphabet 'K'
- It may also represent some control code in decision logic
- In other words group of bits are used to represent different things

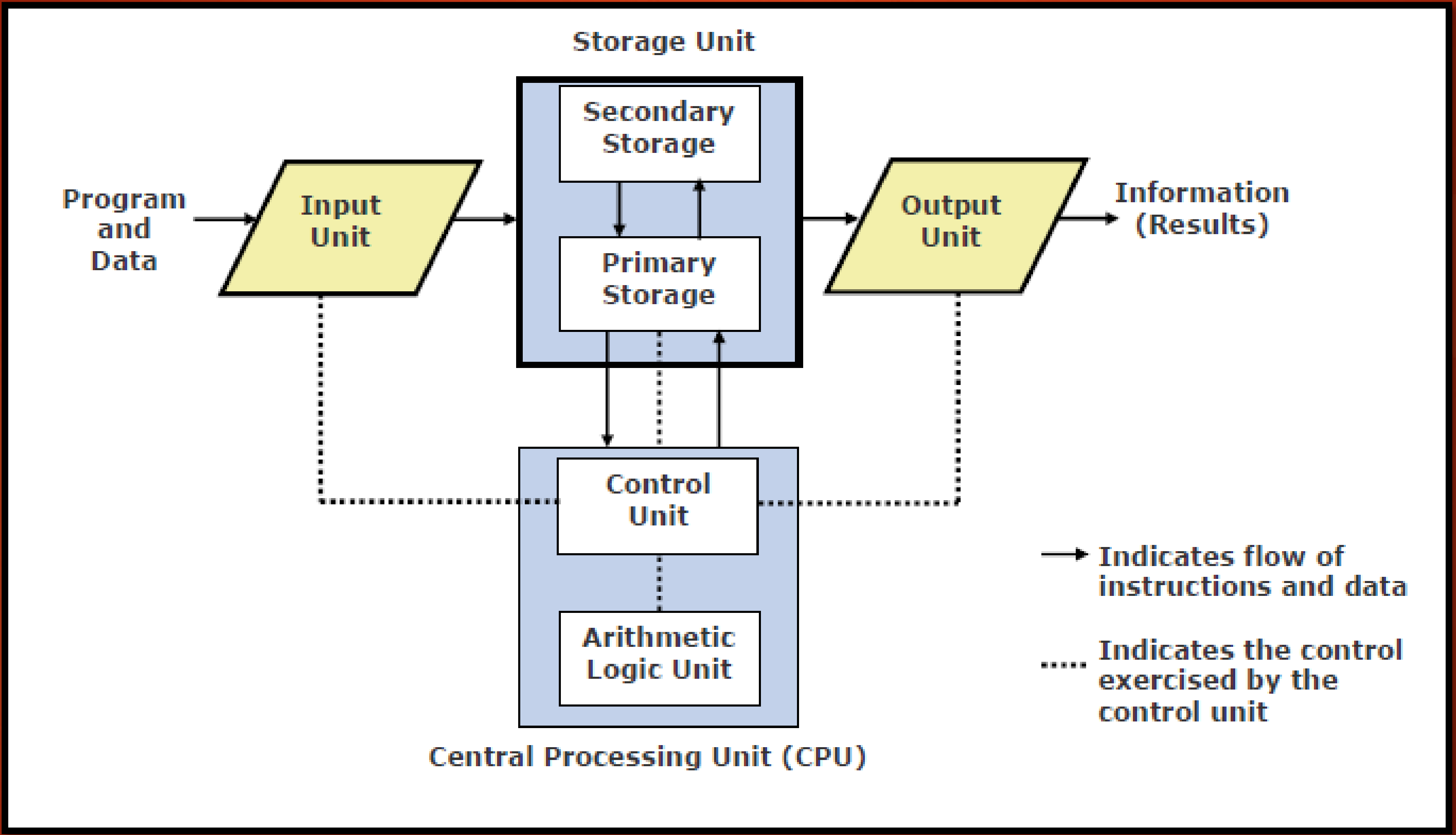
Digital Computer

- Computer system has two functional components:
- Hardware: electronic components and electro mechanical devices
- Software: instructions and data that computer manipulate for data processing
- A program is sequence of instructions for computer to perform some task
- System software are those group of softwares whose sole purpose is to control and effectively use the computer hardware resources e.g. OS
- Application softwares are for the facilitation of user not for system
- System software acts as an interface between your hardware and user

Digital Computer

- CPU contains:
- Arithmetic and logic unit (ALU) for manipulating data
- Registers for storing data
- Control unit to fetch and execute instructions
- Memory of computer contains storage for instructions and data called RAM
- Random access memory as CPU can access any location on memory within a fixed interval of time
- I/O processor contains electronic circuits for communicating and controlling transfer of information between computer and outside world

Digital Computer



NUMBER SYSTEM

- Two types of number systems are:
- Non-positional number systems
- Positional number systems

Non-positional Number Systems

- **Characteristics**
- Use symbols such as I for 1, II for 2, III for 3, IIII for 4, IIIII for 5, etc
- Each symbol represents the same value regardless of its position in the number
- The symbols are simply added to find out the value of a particular number
- **Difficulty**
- It is difficult to perform arithmetic with such a number system

Positional Number Systems

- **Characteristics**
- Use only a few symbols called digits
- These symbols represent different values depending on the position they occupy in the number

Positional Number Systems

- The value of each digit is determined by:
- 1:The digit itself
- 2:The position of the digit in the number
- 3:The base of the number system
- (**base** = total number of digits in the number system)
- Base is also called *radix.*
- The maximum value of a single digit is always equal to one less than the value of the base

Decimal Number System

- **Characteristics**
- A positional number system
- Has 10 symbols or digits (0, 1, 2, 3, 4, 5, 6, 7, 8, 9).
- Hence, its base = 10
- The maximum value of a single digit is 9 (one less than the value of the base)
- Each position of a digit represents a specific power of the base (10)
- We use this number system in our day-to-day life

Decimal Number System

- **Example**
- **258610 = (2 x 10^3) + (5 x 10^2) + (8 x 10^1) + (6 x 10^0)**
- **= 2000 + 500 + 80 + 6**

Binary Number System

- **Characteristics**
- A positional number system
- Has only 2 symbols or digits (0 and 1). Hence its base = 2
- The maximum value of a single digit is 1 (one less than the value of the base)
- Each position of a digit represents a specific power of the base (2)
- This number system is used in computers

Binary Number System

- **Example**
- **$101012 = (1 \times 2^4) + (0 \times 2^3) + (1 \times 2^2) + (0 \times 2^1) \times (1 \times 2^0)$**
- **$= 16 + 0 + 4 + 0 + 1$**
- **$= 2110$**

Representing Numbers in Different Number Systems

- **In order to be specific about which number system we are referring to, it is a common practice to indicate the base as a subscript.**
- **Thus, we write:**
- $(10101)_2 = (21)_{10}$

Octal Number System

- **Characteristics**
- **A positional number system**
- **Has total 8 symbols or digits (0, 1, 2, 3, 4, 5, 6, 7).**
- **Hence, its base = 8**
- **The maximum value of a single digit is 7 (one less than the value of the base**
- **Each position of a digit represents a specific power of the base (8)**

Octal Number System

- **Sincethere are only 8 digits, 3 bits (23 = 8) are sufficient to represent any octal number in binary**

- **Example**

- **20578 = $(2 \times 8^3) + (0 \times 8^2) + (5 \times 8^1) + (7 \times 8^0)$**
- **= 1024 + 0 + 40 + 7**
- **=107110**

Hexadecimal Number System

- **Characteristics**
- **A positional number system**
- **Has total 16 symbols or digits (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F).**
- **Hence its base = 16**
- **The symbols A, B, C, D, E and F represent the decimal values 10, 11, 12, 13, 14 and 15 respectively**
- **The maximum value of a single digit is 15 (one less than the value of the base)**

Hexadecimal Number System

- **Each position of a digit represents a specific power of the base (16)**
- **Since there are only 16 digits, 4 bits (24 = 16) are sufficient to represent any hexadecimal number in binary**
- **Example**
- **1AF16 = $(1 \times 16^2) + (A \times 16^1) + (F \times 16^0)$**
- **= 1 x 256 + 10 x 16 + 15 x 1**
- **= 256 + 160 + 15**
- **= 43110**

Table 1.2
Numbers with Different Bases

Decimal (base 10)	Binary (base 2)	Octal (base 8)	Hexadecimal (base 16)
00	0000	00	0
01	0001	01	1
02	0010	02	2
03	0011	03	3
04	0100	04	4
05	0101	05	5
06	0110	06	6
07	0111	07	7
08	1000	10	8
09	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

DECIMAL (BASE 10)	BINARY (BASE 2)	OCTAL (BASE 8)	HEXADECIMAL (BASE 16)
0	00000	0	0
1	00001	1	1
2	00010	2	2
3	00011	3	3
4	00100	4	4
5	00101	5	5
6	00110	6	6
7	00111	7	7
8	01000	10	8
9	01001	11	9
10	01010	12	A
11	01011	13	B
12	01100	14	C
13	01101	15	D
14	01110	16	E
15	01111	17	F
16	10000	20	10
Examples			
255	11111111	377	FF
256	100000000	400	100

Binary number table
0
1

Binary number table for 2 variables	
0	00
1	01
2	10
3	11

Binary Number table for 3 variables	
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Converting a Number of Another Base to a Decimal Number

- **Method**
  - **Step 1: Determine the column (positional) value of each digit**
  - **Step 2: Multiply the obtained column values by the digits in the corresponding columns**
- **Step 3: Calculate the sum of these products**

Converting a Number of Another Base to a Decimal Number

Example

$4706_8 = ?_{10}$

Common values multiplied by the corresponding digits

$$4706_8 = 4 \times 8^3 + 7 \times 8^2 + 0 \times 8^1 + 6 \times 8^0$$
$$= 4 \times 512 + 7 \times 64 + 0 + 6 \times 1$$
$$= 2048 + 448 + 0 + 6$$
$$= 2502_{10}$$

Sum of these products

Converting a Decimal Number to a Number of Another Base

Division-Remainder Method

Step 1: Divide the decimal number to be converted by the value of the new base

Step 2: Record the remainder from Step 1 as the rightmost digit (least significant digit) of the new base number

Step 3: Divide the quotient of the previous divide by the new base

Converting a Decimal Number to a Number of Another Base

Step 4: Record the remainder from Step 3 as the next digit (to the left) of the new base number

Repeat Steps 3 and 4, recording remainders from right to left, until the quotient becomes zero in Step 3

Note that the last remainder thus obtained will be the most significant digit (MSD) of the new base number

Converting a Decimal Number to a Number of Another Base

Example

$952_{10} = ?_8$

Solution:

8	952	Remainders
	119	0
	14	7
	1	6
	0	1

Hence, $952_{10} = 1670_8$

Converting a Decimal Number to a Number of Another Base

Method

Step 1: Convert the original number to a decimal number (base 10)

Step 2: Convert the decimal number so obtained to the new base number

Converting a Number of Some Base to a Number of Another Base

- **Example**
  - **(545)6 = ()10**
- **Solution:**
  - **Step 1: Convert from base 6 to base 10**
      - **$(545)6 = 5 \times 6^2 + 4 \times 6^1 + 5 \times 6^0$**
        - **$= 5 \times 36 + 4 \times 6 + 5 \times 1$**
        - **$= 180 + 24 + 5$**
        - **$= (209)_{10}$**

Converting a Number of Some Base to a Number of Another Base

Step 2: Convert 209_{10} to base 4

Divisor	Quotient	Remainders
4	209	
	52	1
	13	0
	3	1
	0	3

Hence, $209_{10} = 3101_4$

So, $545_6 = 209_{10} = 3101_4$

Thus, $545_6 = 3101_4$

Shortcut Method for Converting a Binary Number to its Equivalent Octal Number

Octal Symbol	Binary equivalent
0	000
1	001
2	010
3	011
4	100
5	101
6	110
7	111

Shortcut Method for Converting a Binary Number to its Equivalent Octal Number

Decimal	Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Shortcut Method for Converting a Binary Number to its Equivalent Octal Number

- **Method**
  - **Step 1: Divide the digits into groups of three starting from the right**
  - **Step 2: Convert each group of three binary digits to one octal digit using the method of binary to decimal conversion**

Shortcut Method for Converting a Binary Number to its Equivalent Octal Number

Example

$1101010_2 = ?_8$

Step 1: Divide the binary digits into groups of 3 starting from right

$\underline{001}$ $\underline{101}$ $\underline{010}$

Step 2: Convert each group into one octal digit

$001_2 = 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 1$
$101_2 = 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 5$
$010_2 = 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 = 2$

Hence, $1101010_2 = 152_8$

Shortcut Method for Converting an Octal Number to Its Equivalent Binary Number

Method

Step 1: Convert each octal digit to a 3 digit binary number (the octal digits may be treated as decimal for this conversion)

Step 2: Combine all the resulting binary groups (of 3 digits each) into a single binary number

Shortcut Method for Converting an Octal Number to Its Equivalent Binary Number

- **Example**
  - **(562)8 = (?)2**
    - **Step 1: Convert each octal digit to 3 binary digits (5)8= (101)2,(6)8= (110)2, (2)8= (010)2**
    - **Step 2:Combine the binary groups (562)8= 101 110 010**
      - **5 6 2**
  - **Hence, (562)8 = $(101110010)_2$**

Shortcut Method for Converting a Binary Number to its Equivalent Hexadecimal Number

Method

Step 1: Divide the binary digits into groups of four starting from the right

Step 2: Combine each group of four binary digits to one hexadecimal digit

Shortcut Method for Converting a Binary Number to its Equivalent Hexadecimal Number

Example

$111101_2 = ?_{16}$

Step 1: Divide the binary digits into groups of four starting from the right

<u>0011</u> <u>1101</u>

Step 2: Convert each group into a hexadecimal digit

$0011_2 = 0 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 = 3_{10} = 3_{16}$

$1101_2 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 3_{10} = D_{16}$

Hence, $111101_2 = 3D_{16}$

Shortcut Method for Converting a Hexadecimal Number to its Equivalent Binary Number

- **Method**
  - **Step 1: Convert the decimal equivalent of each hexadecimal digit to a 4 digit binary number**
- **Step 2: Combine all the resulting binary groups**
  - **(of 4 digits each) in a single binary number**

Shortcut Method for Converting a Hexadecimal Number to its Equivalent Binary Number

Example

$2AB_{16} = ?_2$

Step 1: Convert each hexadecimal digit to a 4 digit binary number

$2_{16} = 2_{10} = 0010_2$
$A_{16} = 10_{10} = 1010_2$
$B_{16} = 11_{10} = 1011_2$

Shortcut Method for Converting a Hexadecimal Number to its Equivalent Binary Number

Step 2: Combine the binary groups

$$2AB_{16} = \underset{2}{\underline{0010}} \quad \underset{A}{\underline{1010}} \quad \underset{B}{\underline{1011}}$$

Hence, $2AB_{16} = 001010101011_2$

Fractional Numbers

Fractional numbers are formed same way as decimal number system

In general, a number in a number system with base *b* would be written as:

$a_n \, a_{n-1} \dots a_0 \, . \, a_{-1} \, a_{-2} \dots a_{-m}$

And would be interpreted to mean:

$a_n \times b^n + a_{n-1} \times b^{n-1} + \dots + a_0 \times b^0 + a_{-1} \times b^{-1} + a_{-2} \times b^{-2} + \dots + a_{-m} \times b^{-m}$

The symbols a_n, a_{n-1}, ..., a_{-m} in above representation should be one of the *b* symbols allowed in the number system

Formation of Fractional Numbers in Binary Number System (Example)

						Binary Point ↓				
Position	4	3	2	1	0	.	-1	-2	-3	-4
Position Value	2^4	2^3	2^2	2^1	2^0		2^{-1}	2^{-2}	2^{-3}	2^{-4}
Quantity Represented	16	8	4	2	1		$1/2$	$1/4$	$1/8$	$1/16$

Formation of Fractional Numbers in Binary Number System (Example)

- **Example**
- **(110.101)2** $= 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$

  $= 4 + 2 + 0 + 0.5 + 0 + 0.125$

  $= (6.625)_{10}$

Formation of Fractional Numbers in Octal Number System (Example)

Octal Point

Position	3	2	1	0	.	-1	-2	-3
Position Value	8^3	8^2	8^1	8^0		8^{-1}	8^{-2}	8^{-3}
Quantity Represented	512	64	8	1		$^1/_8$	$^1/_{64}$	$^1/_{512}$

Formation of Fractional Numbers in Octal Number System (Example)

- **Example**

- **$(127.54)8 = 1 \times 8^2 + 2 \times 8^1 + 7 \times 8^0 + 5 \times 8^{-1} + 4 \times 8^{-2}$**
  - **$= 64 + 16 + 7 + 5/8 + 4/64$**
  - **$= 87 + 0.625 + 0.0625$**
  - **$= (87.6875)10$**

Convert (0.6875)10 to binary

- First, 0.6875 is multiplied by 2 to give an integer and a fraction.
- Then the new fraction is multiplied by 2 to give a new integer and a new fraction.
- The process is continued until the fraction becomes 0 or until the number of digits has sufficient accuracy

	Integer		Fraction	Coefficient
$0.6875 \times 2 =$	1	+	0.3750	$a_{-1} = 1$
$0.3750 \times 2 =$	0	+	0.7500	$a_{-2} = 0$
$0.7500 \times 2 =$	1	+	0.5000	$a_{-3} = 1$
$0.5000 \times 2 =$	1	+	0.0000	$a_{-4} = 1$

- Therefore, the answer is (0.6875)10= (0.1011)2

Convert $(0.513)_{10}$ to octal.

$$
\begin{aligned}
0.513 \times 8 &= 4.104 \\
0.104 \times 8 &= 0.832 \\
0.832 \times 8 &= 6.656 \\
0.656 \times 8 &= 5.248 \\
0.248 \times 8 &= 1.984 \\
0.984 \times 8 &= 7.872
\end{aligned}
$$

The answer, to seven significant figures, is obtained from the integer part of the products:

$$(0.513)_{10} = (0.406517 \ldots)_8$$

The conversion of decimal numbers with both integer and fraction parts is done by converting the integer and the fraction separately and then combining the two answers. Using the results of Examples 1.1 and 1.3, we obtain

$$(41.6875)_{10} = (101001.1011)_2$$

From Examples 1.2 and 1.4, we have

$$(153.513)_{10} = (231.406517)_8$$

(54.6875)10=(------)2

(54)10→()2

- 54 (divide by 2) = 27 remainder **0**
- 27 (divide by 2) = 13 remainder **1**
- 13 (divide by 2) = 6 remainder **1**
- 6 (divide by 2) = 3 remainder **0**
- 3 (divide by 2) = 1 remainder **1**
- 1 (divide by 2) = 0 remainder **1**
- Thus the binary equivalent of 54_{10} is therefore: 110110_2

(0.6875)10→()2

- 0.6875 (multiply by 2) = **1**.375 = 0.375 carry **1**
- 0.375 (multiply by 2) = **0**.75 = 0.75 carry **0**
- 0.75 (multiply by 2) = **1**.50 = 0.5 carry **1**
- 0.5 (multiply by 2) = **1**.00 = 0.0 carry **1**
- Thus the binary equivalent of 0.6875_{10} is therefore: 0.1011_2 ← (LSB)
- **Hence the binary equivalent of the decimal number: 54.6875_{10} is 110110.1011_2**

EXERCISE PROBLEMS

1.7* Convert the hexadecimal number 64CD to binary, and then convert it from binary to octal.

1.8 Convert the decimal number 431 to binary in two ways: (a) convert directly to binary; (b) convert first to hexadecimal and then from hexadecimal to binary. Which method is faster?

1.9 Express the following numbers in decimal:

(a)* $(10110.0101)_2$
(b)* $(16.5)_{16}$
(c)* $(26.24)_8$
(d) $(DADA.B)_{16}$
(e) $(1010.1101)_2$

1.10 Convert the following binary numbers to hexadecimal and to decimal: (a) 1.10010, (b) 110.010. Explain why the decimal answer in (b) is 4 times that in (a).