

# LECTURE 1: DIGITAL LOGIC CIRCUIT DESIGN

## Introduction to Digital Logic

The material covered in this class will be as follows:

- **Introduction.**
- **Digital and analog quantities.**
- **Digital system examples.**
- **Course material.**
- **Course syllabus.**
- **Policies on home works, quizzes, exams, and term design project.**

After finishing this class, you should be able to:

- **Identify the importance of this introductory course on digital logic.**
- **Appreciate the arrangement of the course material units.**
- **Become familiar with the assessment method through home works, quizzes, projects and exams.**

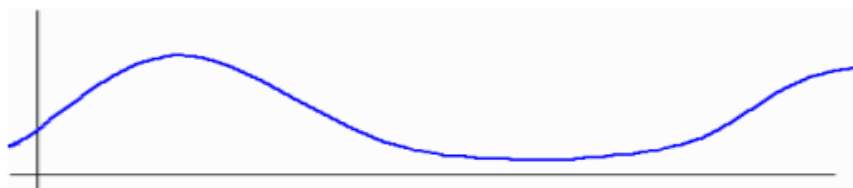
### Introduction:

- Prominence of digital systems in everyday life.
- The present time could be referred to as the Digital Age.
- Some examples are the digital computers, mobile phones, digital cameras, digital watches, control systems in cars and aeroplanes...etc.
- Digital systems are used in almost all aspects of life, in communications, business transactions, traffic control, medical treatment, the internet...etc.

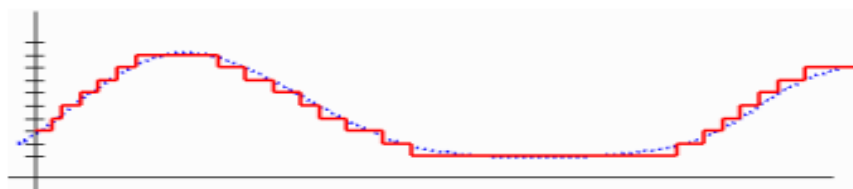
# LECTURE 1: DIGITAL LOGIC CIRCUIT DESIGN

- Digital logic is the first course in studying digital systems, which is an essential part of almost all branches of Electrical Engineering.
  - Digital Communications
  - Digital Electronics
  - Digital Control
  - Digital Signal and Image Processing
  - Power System Analysis
- Digital systems are able to manipulate discrete elements of information e.g. 10 decimal digits, 26 letters of the alphabet...etc.
- Discrete elements of information are represented in digital systems by physical quantities called signals. The most common are the voltage and current signals in electronic devices.
- The signals in most present day electronic digital systems use two discrete values (binary signal).
- A binary digit called bit has two values: 0 and 1.
- Discrete elements of information are represented with groups of bits called binary codes.
- A digital system is an interconnection of digital modules. Their operation can be understood from the basic knowledge of digital circuits and their logic functions.
- An important new trend in digital design is the use of hardware description language (HDL). It resembles a programming language and is suitable for describing digital circuits in textual form.
- HDL is used to simulate a digital system and verify its operation before hardware is built.

## Analog and Digital Signals



An analog signal



A digital signal

# LECTURE 1: DIGITAL LOGIC CIRCUIT DESIGN

## Course contents:

1. BINARY SYSTEMS
  - 1-1 Digital Systems
  - 1-2 Binary Numbers
  - 1-3 Number Base Conversions
  - 1-4 Octal and Hexadecimal Numbers
  - 1-5 Complements
  - 1-6 Signed Binary Numbers
  - 1-7 Binary Codes
  
2. BOOLEAN ALGEBRA AND LOGIC GATES
  - 2-1 Axiomatic Definition of Boolean Algebra
  - 2-2 Theorems and Properties of Boolean Algebra
  - 2-3 Boolean Functions
  - 2-4 Digital Logic Gates
  - 2-5 Canonical and Standard Forms
  
3. GATE-LEVEL MINIMIZATION
  - 3-1 Simplification of Boolean Functions Using K-Maps
  - 3-2 Product of Sums Simplification
  - 3-3 Don't-care Conditions
  - 3-4 NAND, NOR, and other two Level Implementation
  - 3-5 X-OR Function
  - 3-6 Introduction to HDL
  
4. COMBINATIONAL LOGIC
  - 4-1 Combinational Circuits
  - 4-2 Analysis and Design Procedures
  - 4-3 Code Conversion
  - 4-4 Adder-Subtractor Circuits
  - 4-5 Decimal Adder
  - 4-6 Binary Multiplier
  - 4-7 Magnitude Comparator
  - 4-8 Decoders
  - 4-9 Encoders and Multiplexers
  - 4-10 Example of HDL for Combinational circuit

# LECTURE 1: DIGITAL LOGIC CIRCUIT DESIGN

## 5. MEMORY AND PROGRAMMABLE LOGIC

- 5-1 Programmable Logic Devices
- 5-2 Read Only Memory
- 5-3 Programmable Logic Array
- 5-4 Programmable Array Logic

## 6. SYNCHRONOUS SEQUENTIAL LOGIC

- 6-1 Sequential Circuits
- 6-2 Latches
- 6-3 Flip-Flops
- 6-4 Characteristic Tables
- 6-5 Analysis of Clocked Sequential Circuits
- 6-6 Examples of HDL for Sequential Circuits
- 6-7 State reduction and Assignment
- 6-8 Design Procedure
- 6-9 Synthesis Using Different Flip-Flops

## 7. REGISTERS AND COUNTERS

- 7-1 Registers and Shift Registers
- 7-2 Ripple Counters
- 7-3 Synchronous Counters
- 7-4 Other Counters

## Number Systems

After finishing this class, you should be able to:

- Represent any number in any number system.
- Convert between the different number systems.
- Appreciate the ease of conversion between binary, octal and hexadecimal numbers.

Any number  $N$  can be represented in any base ( $r$ ). As  $\rightarrow$

.....  $a_3 a_2 a_1 a_0 . a_{-1} a_{-2}$  .....

$$\therefore N = \dots + a_3 r^3 + a_2 r^2 + a_1 r + a_0 r^0 + a_{-1} r^{-1} + a_{-2} r^{-2} + \dots$$

$r$  is the base (or the radix ) of the number system.

# LECTURE 1: DIGITAL LOGIC CIRCUIT DESIGN

## Examples:

- Decimal number  $\rightarrow 49.75_{10} = 4 \times 10^1 + 9 \times 10^0 + 7 \times 10^{-1} + 5 \times 10^{-2}$
- Octal number  $\rightarrow 236.05_8 = 2 \times 8^2 + 3 \times 8^1 + 6 \times 8^0 + 0 \times 8^{-1} + 5 \times 8^{-2}$
- Binary number  $\rightarrow 1101.01 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2}$
- Hexadecimal number  $\rightarrow r = 16$ , Digits are 0 - 9, A, B, C, D, E & F  
 $A6F.8 = A \times 16^2 + 6 \times 16^1 + F \times 16^0 + 8 \times 16^{-1}$

## Conversion From Base $r$ To Decimal:

Use the positional notation used in the previous examples to convert from any base to decimal.

## Examples:

- $236.05_8 = 2 \times 8^2 + 3 \times 8^1 + 6 \times 8^0 + 0 \times 8^{-1} + 5 \times 8^{-2} = 128 + 24 + 6 + (5/64) = 158.078125_{10}$
- $A6F.8 = A \times 16^2 + 6 \times 16^1 + F \times 16^0 + 8 \times 16^{-1} = 10 \times 256 + 96 + 15 + 0.5 = 2671.5_{10}$
- $1101.01 = 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 + 0 \times 2^{-1} + 1 \times 2^{-2} = 8 + 4 + 1 + 0.25 = 13.25_{10}$

## Conversion From Decimal To Base $r$ :

The number is separated into integer part and a fraction. The integer is successively divided by the new base, keeping track of the remainder, until the quotient is zero. The fraction is multiplied by the base, keeping track of the generated integer part, until the required accuracy is reached.

Example: Convert  $41.25_{10}$  to binary.

	0	1	2	5	10	20	41(/2)
Remainder $\rightarrow$	1	0	1	0	0	1	

$$\therefore 41_{10} = 101001_2$$

0.25	0.5	.0				
.	0	1	$\leftarrow$ Integer part			

$$\therefore 0.25_{10} = 0.01_2$$

$$\therefore 51.25_{10} = 101001.01_2$$

# LECTURE 1: DIGITAL LOGIC CIRCUIT DESIGN

The 16 Hexadecimal digits expressed in decimal, octal and binary are as follows:

Hex.	Decimal	Octal	Binary
0	00	00	0000
1	01	01	0001
2	02	02	0010
3	03	03	0011
4	04	04	0100
5	05	05	0101
6	06	06	0110
7	07	07	0111
8	08	10	1000
9	09	11	1001
A	10	12	1010
B	11	13	1011
C	12	14	1100
D	13	15	1101
E	14	16	1110
F	15	17	1111

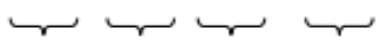

## Number Systems & Complements

After finishing this class, you should be able to:

- Convert numbers easily between binary, octal, and hexadecimal.
- Perform binary arithmetic operation and apply the rules to other number systems.
- Understand and be able to obtain the complements of numbers in any number system.
- The base of the octal number is  $8 = 2^3$  and the base of the hexadecimal number is  $16 = 2^4$ .
- Each octal digit { 0 - 7 } is equivalent to 3 binary digits (bits) { 000 - 111 }.
- Each hexadecimal digit { 0 - F } is equivalent to 4 bits { 0000 - 1111 }.
- Conversion between these systems become very easy by replacing each digit by the 3 bits or 4 bits without going through decimal.

# LECTURE 1: DIGITAL LOGIC CIRCUIT DESIGN

## Examples

- Binary number →  $111010110.1010$   

- Equivalent octal number →  $726.5$   

- Equivalent hexadecimal number →  $1D6A$

## Binary Arithmetic:

The rules used in decimal arithmetic operations are applied in any other number system. ( digit carry to the higher order position in addition and the digit borrow from higher order position in subtraction).

