Communication Technical Engineering Department

1st Stage

Digital Logic- UOMU028021

Lecture 1 – Introduction to Number Systems

Dr. Mohammed Fadhil
PhD in Computer Networks
Email: mohammed.fadhil1@uomus.edu.iq

Overview Objectives

- By the end of this lecture, students will be able to:

  - Define digital logic and its applications
  - Understand numbering system and its type
  - Understand decimal, binary, octal and hexadecimal numbers.
  - Count in decimal, binary, octal and hexadecimal systems.

Introduction

- **Digital Computers**
  - Digital computers process and store data as discrete binary digits (bits). These bits represent states like:
    - **Magnetic markers (present/absent)**
    - **Switches (on/off)**
    - **Relay states**
    - **All data—including numbers, text, and commands—is converted to digital form.**
- **Digital Logic**
  - **The foundation of modern electronics (computers, phones, etc.), digital logic uses binary code (0s and 1s) to:**
    - **Design circuits (e.g., logic gates like AND, OR, NOT)**
    - **Process input signals into specific outputs**
    - **Enable computing, robotics, and electronic systems**
- **Digital Logic Design**
  - A core discipline in electrical and computer engineering, it involves:
  - Creating hardware (circuit boards, microchips)
  - Integrating electrical and computational traits (power, logic functions, protocols)
  - Processing inputs for computers, phones, navigation systems, and other tech

Number Systems

- **A numbering system** is a mathematical framework for representing and organizing numbers using a specific set of symbols and rules.
- It defines how numbers are expressed and manipulated within a given base or radix, determining the number of unique symbols representing values.
- Numbering systems are essential for various fields, including mathematics, engineering, and computer science, as they provide a structured way to handle numerical data.

Number Systems

- For digital information to be processed by a circuit, it must be represented in a suitable format for that circuit.
- To achieve this, **a base B number system** (B a natural number ≥ 2) needs to be chosen.
- Several number systems are used in digital technology, with the most commonly used systems are:
  - **Decimal** (base **10**)
  - **Binary** (base **2**)
  - **Octal** (base **8**)
  - **Hexadecimal** (base **16**).

Decimal number system

- **Key Features:**
  - **Base-10 system** using digits: 0,1,2,3,4,5,6,7,8,9
  - **Positional-value system:** Digit value depends on its place.
- **Examples:**
  - **453**
    - 4 (**MSD**): Hundreds (10^2)
    - 5: Tens (10^1)
    - 3 (**LSD**): Units (10^0)
- **Expanded Form:**
  - **9261** = $(9\times10^3) + (2\times10^2) + (6\times10^1) + (1\times10^0)$
  - **3267.317** = $(3\times10^3) + (2\times10^2) + (6\times10^1) + (7\times10^0) + (3\times10^{-1}) + (1\times10^{-2}) + (7\times10^{-3})$

Decimal number system

- **Visual Representation: 5432.387**

Position	10^3	10^2	10^1	10^0	.	10^{-1}	10^{-2}	10^{-3}
Example	5	4	3	2	.	3	8	7
Weight	1000	100	10	1	.	0.1	0.01	0.001

- **Key Terms:**
  - **MSD**: Most Significant Digit (leftmost)
  - **LSD**: Least Significant Digit (rightmost)
  - **Decimal point**: Separates integer (left) and fractional (right) parts.

Binary number system

- **Key Features:**
  - **Base-2 system** using digits: 0, 1 (bits)
  - **Positional-value system**: Each bit's value depends on its **power-of-2 position.**

- **Example**
  - 1101.011_2

Position	2^3	2^2	2^1	2^0	.	2^{-1}	2^{-2}	2^{-3}
Bit	1	1	0	1	.	0	1	1
Weight	8	4	2	1	.	0.5	0.25	0.125

- **Decimal Conversion:**
  - $(1\times8) + (1\times4) + (0\times2) + (1\times1) + (0\times0.5) + (1\times0.25) + (1\times0.125)$
  - $= 8 + 4 + 0 + 1 + 0 + 0.25 + 0.125 = 13.375_{10}$

Binary number system

- **Universal Rule:**
  - Any number (decimal, binary, octal, hexadecimal) equals the sum of each **digit ×** its positional value.
- **Key Terms & Notes**
  - **Bit**: Binary digit (0 or 1)
  - **Byte**: 8 bits
  - **MSB**: Most Significant Bit (leftmost)
  - **LSB**: Least Significant Bit (rightmost)
  - **Binary ↔ Decimal**:
    - Digital systems output binary → converted to decimal for display.
    - User inputs (decimal) → converted to binary for processing.

Octal Number System

- **Key Features:**
  - Uses digits (**Base-8 system**): 0, 1, 2, 3, 4, 5, 6, 7
  - Positional-value system: Each digit's value depends on its power-of-8 position.
- **Examples**:
  - Octal Number: $(\mathbf{1101.011})_8$
    - MSD (Most Significant Digit): Leftmost 1
    - LSD (Least Significant Digit): Rightmost 1
  - Expanded Form:
    - $(4372)_8 = (4\times8^3) + (3\times8^2) + (7\times8^1) + (2\times8^0)$
    - $(372.36)_8 = (3\times8^2) + (7\times8^1) + (2\times8^0) + (3\times8^{-1}) + (6\times8^{-2})$

Octal Number System

- **Visual Representation: $(1101.01)_8$**

Position	8^3	8^2	8^1	8^0	.	8^{-1}	8^{-2}
Example	1	1	0	1	.	0	1
Weight	512	64	8	1	.	0.125	0.0156

- **Key Notes:**
  - No digits 8 or 9 (only 0-7).
  - Common Uses:
    - Shorthand for binary (3 bits = 1 octal digit).
    - Legacy computing systems.

Hexadecimal Number System

- **Key Features:**
  - Uses digits (**Base-16 system**): 0, 1, 2, ..., 9, A(10), B(11), C(12), D(13), E(14), F(15)
  - Positional-value system: Each digit's value scales by powers of **16**.
- Examples:
  - Hexadecimal Expansion:
    - $\mathbf{(A2C9)_{16}} = (10\times16^3) + (2\times16^2) + (12\times16^1) + (9\times16^0)$
    - $\mathbf{(27.38)_{16}} = (2\times16^1) + (7\times16^0) + (3\times16^{-1}) + (8\times16^{-2})$
  - Decimal Equivalents:
    - **A → 10, F → 15**
    - $\mathbf{(1F)_{16} = 1\times16 + 15 = (31)_{10}}$

Hexadecimal Number System

- **Visual Representation: $(A2C9.38)_{16}$**

Position	16^3	16^2	16^1	16^0	.	16^{-1}	16^{-2}
Example	A	2	C	9	.	3	8
Weight	4096	256	16	1	.	0.0625	0.0039

- **Why Hexadecimal?**
  - Compact representation of binary (4 bits = 1 hex digit).
  - Commonly used in programming, memory addressing, and digital systems.

THANK YOU ☺