College of Sciences
Intelligent Medical System Department

Lecture2:

Number systems and Binary codes

Subject: Logic Design
Level: First
Semester: second
Lecturer: Asst. Lect. Ali Saleem Haleem

Google Class Room

Introduction

Lecture 2 extends the study of digital quantities by introducing additional number systems and the coding methods used in digital electronics. Students move beyond decimal and binary representation to understand octal and hexadecimal notation, the role of complements in binary arithmetic, and the classification of binary codes into weighted, non-weighted, decimal, and alphanumeric forms. These topics are essential because engineers use them to represent data compactly, simplify hardware operations, and exchange information between human-readable symbols and machine-level binary patterns.

Learning Outcomes

- Define octal and hexadecimal number systems and explain why they are useful in digital systems.
- Convert numbers among binary, octal, hexadecimal, and decimal forms accurately.
- Distinguish between 1's complement and 2's complement and apply both in binary representation.
- Classify binary codes into weighted and non-weighted categories.
- Explain Binary Coded Decimal (BCD) and alphanumeric coding with practical examples.

Octal Number System

The octal number system has base 8, so it uses eight symbols only: 0, 1, 2, 3, 4, 5, 6, and 7. Each octal digit can be represented by exactly three binary bits. For this reason, octal notation provides a shorter and more readable way to write long binary numbers.

To convert a binary number to octal, the binary digits are grouped into sets of three starting from the right. Each group is then replaced by its equivalent octal digit. Likewise, each octal digit can be expanded back into a 3-bit binary group.

Octal Digit	Binary Equivalent	Decimal Value
0	000	0
3	011	3
5	101	5
7	111	7

Example: Binary 101111001_2 can be grouped as 101 111 001. Therefore, the octal equivalent is 571_8.

Hexadecimal Number System

The hexadecimal number system has base 16. It uses the symbols 0 to 9 and the letters A, B, C, D, E, and F, where A = 10, B = 11, C = 12, D = 13, E = 14, and F = 15. Hexadecimal is widely used in digital electronics and computing because one hexadecimal digit corresponds exactly to four binary bits.

To convert from binary to hexadecimal, the binary number is divided into groups of four bits from right to left. Each group is then replaced by one hexadecimal symbol. This makes hexadecimal notation especially suitable for representing addresses, machine codes, and compact digital values.

Hex Digit	Binary Equivalent	Decimal Value
9	1001	9
A	1010	10
C	1100	12
F	1111	15

Example: Binary 110101111010_2 can be grouped as 1101 0111 1010, so its hexadecimal form is $D7A_{16}$.

1's Complement and 2's Complement

Complements are alternative representations of binary numbers and are important in digital arithmetic, especially subtraction and signed number representation.

- 1's complement is obtained by changing every 0 to 1 and every 1 to 0.
- 2's complement is obtained by adding 1 to the 1's complement.
- 2's complement is more important in computer hardware because it simplifies subtraction circuits and signed arithmetic.

Example: For binary number 01011010, the 1's complement is 10100101 and the 2's complement is 10100110.

Binary Codes

A binary code is a group of bits used to represent symbols, numbers, letters, or instructions. Digital systems depend on codes because electronic circuits work naturally with two states only: 0 and 1. Different coding schemes are selected according to the purpose of the system, the ease of decoding, and the type of information being represented.

- Weighted codes assign a fixed positional weight to each bit.
- Non-weighted codes do not depend on fixed positional weights.

- Decimal codes represent decimal digits by binary patterns.
- Alphanumeric codes represent letters, numbers, punctuation marks, and control symbols.

Weighted Codes

A weighted code is a binary code in which each bit position has a specific numerical weight. The decimal value represented by a code word is obtained by summing the weights corresponding to the bits that contain 1. Weighted codes are useful because the represented value can be interpreted directly from the assigned weights.

The most common weighted code is 8421 BCD, where the weights of the four bits are 8, 4, 2, and 1.

Non-Weighted Codes

A non-weighted code does not assign a fixed arithmetic weight to each bit position. Its meaning depends on the code pattern itself rather than on positional value. These codes are used when reliability, error reduction, or specific logic properties are more important than direct arithmetic interpretation.

Examples of non-weighted codes include Gray code and Excess-3 code. Gray code is especially important because only one bit changes between two consecutive code words, which reduces ambiguity during transitions.

Binary Coded Decimal (BCD)

Binary Coded Decimal is a coding method in which each decimal digit is represented separately by a 4-bit binary code. In the 8421 BCD system, the decimal digits 0 to 9 are represented by 0000 to 1001. The six remaining combinations from 1010 to 1111 are invalid in standard BCD.

Decimal Digit	8421 BCD Code
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Example: Decimal 59 is written in BCD as 0101 1001 because the digit 5 is encoded separately from the digit 9.

Alphanumeric Codes

Alphanumeric codes are designed to represent letters, decimal digits, special characters, and control symbols. They allow computers and digital systems to process text as binary data. Common alphanumeric codes include ASCII and Unicode. ASCII uses standard binary patterns for English letters, digits, and symbols, while Unicode extends the concept to cover many world languages and additional characters.

Example: In ASCII, the capital letter A is represented by 65 in decimal, which is 01000001 in 8-bit binary form.

Comparison of Main Concepts

Concept	Key Idea	Typical Use
Octal	Groups binary bits into sets of 3	Compact binary representation
Hexadecimal	Groups binary bits into sets of 4	Addresses, machine code, digital systems
1's / 2's Complement	Represents inverse and signed binary forms	Binary arithmetic and subtraction
BCD	Represents each decimal digit separately	Digital displays and calculators
Alphanumeric Codes	Represents letters and symbols in binary	Text processing and communication

Homework

- Convert 745_8 to binary and decimal.
- Convert $2AF_{16}$ to binary and decimal.
- Find the 1's complement and 2's complement of 00110101.
- Write the decimal number 86 in 8421 BCD form.
- State the difference between weighted and non-weighted codes with one example for each.