DIGITAL LOGIC AND COMPUETR AECHITECTURE

TOPIC NO 2

CODES

Suvarna Bhat

Vidyalankar Institute of Technology
Accredited A+ by NAAC

CODES

When Numbers, letters, or text represented by group of symbols

- CODES
  - Weighted codes
    - Eg: Binary Coded Decimal codes (BCD)
  - Non-weighted codes
    - Eg. Grey Codes
    - Excess-3 codes
  - Alphanumeric codes
    - Eg. ASCII codes
  - Error Detecting Codes
    - Eg. Parity codes
  - Error Detecting & correcting codes
    - Eg. Hamming Codes

HOW BINARY CODES WORKS ?

WEIGHTED CODES

- **Weighted codes :**
  - The main characteristic of a weighted code is, each binary bit is assigned by a "weight" and values depend on the position of the binary bit.
  - The sum of the weights of these binary bits, whose value is 1 is equal to the decimal digit which they represent.
  - A sequence of binary bits which represents a decimal digit is called a "code word".
  - Example of these codes is: BCD, 8421, 6421, 4221, 5211, 3321 etc.
- Application :
  - ✓ Data manipulation during arithmetic operation.
  - ✓ For input/output operations in digital circuits.
  - ✓ Digital Displays like in calculators, digital volt meters etc.

BINARY CODED DECIMAL (BCD)

- Binary Coded Decimal (BCD) or 8421 codes or 4bit code or 1 Nibble code
- Each digit is represented by 4 bits of binary number
    - Eg 3 : 0011 , 8 : 1000

DECIMAL To BCD Conversion

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

- Conversion Steps:

Identify number of digits to be converted in given BCD number

↓

Split all the digits

↓

Represent each digit with four bits of binary number

↓

Combine all the bits

EXAMPLES:

- Convert $(6)_{10}$ into BCD code
  - $(6)_{10} \longrightarrow (0110)_{BCD}$

- Convert $(37)_{10}$ into BCD code

$(3 \mid 7)$

0011 0111

$$(37)_{10} = (0011\ 0111)_{BCD}$$

BCD To Decimal Conversion

Step 1: Start clubbing the group of 4 binary bits together from RHS to LHS (for Integer no)

Step 2: Represent each group of 4 binary bits into its equivalent decimal number

Step 3: If number of binary bits are not multiple of 4 then pad zero to LHS side

- Convert given BCD code into Binary
  - ➢ $(00010101)_{BCD}$

    (1 5)
  - ➢ $(\ 0010110)_{BCD}$

    Pad zero at LSB

    (1 6)

Practice Problems

- Convert $(17)_{10}$ into BCD
- Convert $(64.38)_{10}$ into BCD

Non- WEIGHTED CODES

- **Non-Weighted codes :**
  - ➢Non-weighted or un-weighted codes are those codes in which the digit value does not depend upon their position i.e., each digit position within the number is not assigned fixed value.
  - ➢ The positional weightage is not assigned
- **Examples:** Excess-3 codes, Grey codes
- Applications:
  - ✓To perform certain arithmetic operations.
  - ✓Shift position encodes.
  - ✓Used for error detecting purpose.

EXCESS-3 CODES

- The excess-3 code is a non-weighted and self-complementary BCD code used to represent the decimal numbers
- Only unweighted code which is self complementing

Advantage:

- This code plays an important role in arithmetic operations because it resolves deficiencies encountered when we use the 8421 BCD code for adding two decimal digits whose sum is greater than 9.

DECIMAL to excess-3 Conversion

Decimal	BCD (8421)	EXCESS-3 CODE
0	0000	0011
1	0001	0100
2	0010	0101
3	0011	0110
4	0100	0111
5	0101	1000
6	0110	1001
7	0111	1010
8	1000	1011
9	1001	1100

- Conversion Steps:

Identify number of digits to be converted in given EXCESS -3 number

↓

Split all the digits

↓

Represent each digit with four bits of binary number (BCD code)

↓

Add 0011 (3) to each digit separately

↓

Combine all the bits

EXAMPLES:

- Convert $(6)_{10}$ into Excess-3 code

  - $(6)_{10} \longrightarrow (1001)_{Excess\text{-}3}$

$$
\begin{array}{r}
0110 \\
+\ 0011 \\
\hline
1001
\end{array}
$$

- Convert $(35)_{10}$ into Excess-3 code

$(3|5)$

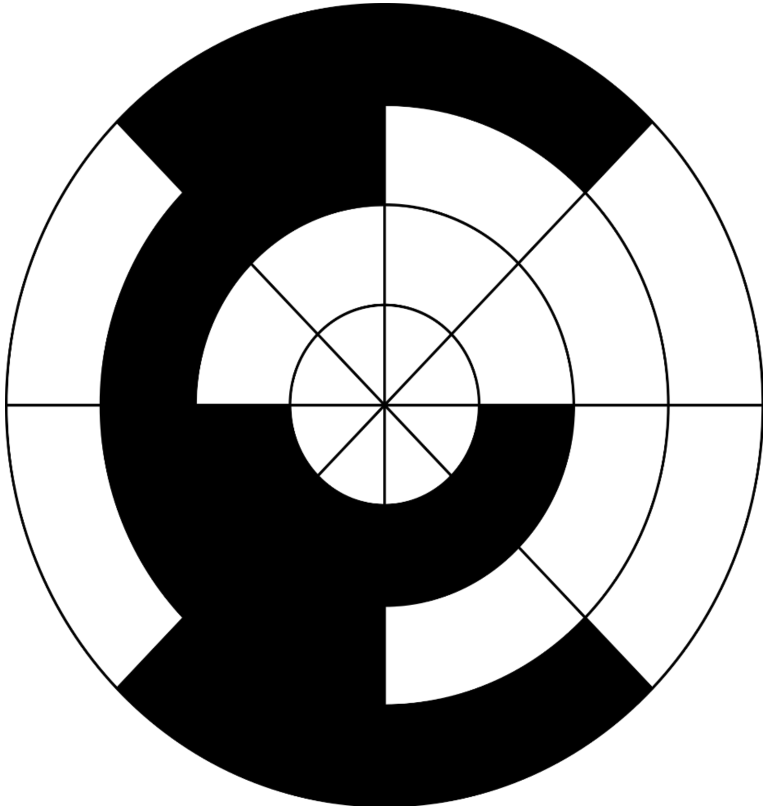
$$
\begin{array}{r}
0011 \\
+\ 0011 \\
\hline
0110
\end{array}
\qquad
\begin{array}{r}
0101 \\
+\ 0011 \\
\hline
1000
\end{array}
$$

$(35)_{10} = (0110\ 1000)_{Excess\text{-}3}$

Practice Problems

- Convert given $(57)_{10}$ into Excess-3 code

GRAY CODES



Gray code is a binary numeral system where two successive values differ in only one bit. This **code** was invented by Frank **Gray** in 1953

WHY GRAY CODES?

- Two successive values differ only in 1 bit
- Binary number is converted into Grey code to reduced switching operation
- Gray codes are also known as
  1. Unit Distance Code
  2. Minimum Error Code
  3. Reflection Code

How to Construct Gray codes

1 Bit Gray code

G0
0
1

How to Construct Gray codes

2 Bit

G1	G0
0	0
0	1
1	1
1	0

Suvarna Bhat

How to Construct Gray codes

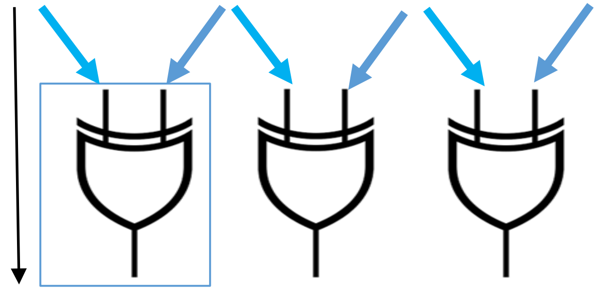
3 Bit

G2	G1	G0
0	0	0
0	0	1
0	1	1
0	1	0
1	1	0
1	1	1
1	0	1
1	0	0

Binary TO Gray Code Conversion

Eg: Convert Given $(1010)_2$ into Gray Code

BINARY → 1 0 1 0

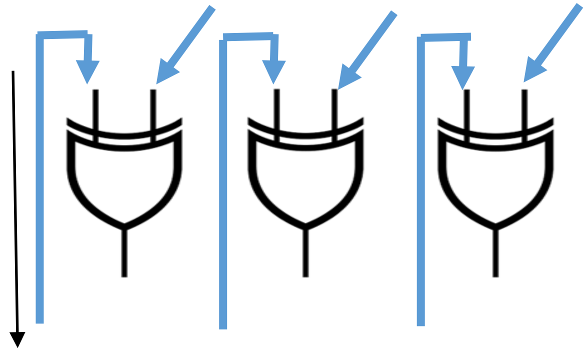


GRAY → 1 1 1 1

Gray To Binary Conversion

Eg: Convert Given Grey Code 1111 into Binary

Gray → **1 1 1 1**



Binary → **1 0 1 0**

Practice Problems

- Convert $(34)_{10}$ into Gray code
- Convert $(10101111)_2$ into Gray code.
- Convert given Gray code into binary equivalent.

  i) $(10001)_2$ ii) $(110001)_2$

ASCII CODES

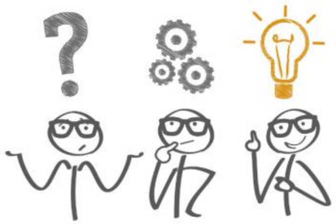
- American Standard Code for Information Interchange
- The ASCII code is used to give to each symbol / key from the keyboard a unique number called ASCII code.
- It can be used to convert text into ASCII code and then into binary code
- It can be used within your code to identify specific characters in a string or specific keys being pressed on the keyboard

ASCII Code Representation

- ASCII is a computer code which uses **128** different encoding combinations of a group of **seven** bits (2 = **128**)
- Includes:

ASCII CODE TABLE

SOLVE



- Based on ASCII table find equivalent binary code for following Characters:
  - A
  - J
  - 8
  - +
- Using AscII Table convert given word binary coded message
  - Hi
- Using AscII Table convert given binary coded message into word.
  - 1001000 1000101 1001100 1001100 1001111

Error Detecting codes

- What is Parity ?

- Types of parity:
  - EVEN Parity
  - ODD Parity

- Parity check

- Example:

I want to transmit binary data..

Error Detecting codes

Binary message bits : 1011110 , Transmit with EVEN parity

1 1 0 1 1 1 1 0

Parity Bit

Error Detecting codes : PARITY

At Receiver's End

CASE 1: Message Received correctly :

1 1 0 1 1 1 1 0

CASE 2: Message Received with 1 bit Error :

1 1 0 1 0 1 1 0

HOW PARITY BIT WILL HELP?

At Receiver's End

PARITY CHECK:

1 1 0 1 0 1 1 0

No of 1's : 5 (ODD Parity)

Received Message is wrong

Hence Proved Parity bit is error detecting code

Practice Problems

- Generate Even parity and Odd parity for following binary message bits
  - $(1011001)_2$