Prerequisite

- Digital Signal
- Digital Circuits (AND, OR, NOT, NAND, NOR, EX-OR, EX-NOR)
- Number System (Decimal, Binary, Octal, Hexadecimal, Signed binary number)
- Codes

Codes

- When numbers, alphabets or words are represented by specific group of symbols means they are encoded.
-The group of symbols used to encode them are called ***Codes***.
- For example, the binary 1000001 represents,

  65 in Decimal

  41 in BCD code

  alphabet **A** in ASCII code
- Examples of Codes:

1. Binary Code
2. Natural BCD Code
3. Excess-3 Code
4. Gray Code
5. Octal Code
6. Hexadecimal Code
7. Alphanumeric Codes
8. Error detecting and correcting code

Codes – Binary Code

- The digital data is represented, stored and transmitted as group of bits. This group of bits is also called as **binary code**.
- Only 0 & 1 are being used to represent binary code.

Binary Power	2^3	2^2	2^1	2^0
Binary Weight:	8 (MSB)	4	2	1(LSB)

- Binary codes can be classified into two types.
  1. Weighted codes
  2. Un-weighted codes

If the code has positional weights, then it is said to be **weighted code**. Otherwise, it is an **un-weighted code**.

- Advantages: -> Binary codes are suitable for the computer applications and digital communications.
  -> Binary codes make the analysis and designing of digital circuits if we use the binary codes.

Decimal (Base 10)	Binary (Base 2)
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

Codes – BCD Code

- Each decimal digit is represented by a 4-bit binary number.
- BCD is a way to express each of the decimal digits with a binary code.
- In the Binary, with four bits we can represent sixteen numbers (0000 to 1111). But in BCD code only first ten of these are used (0000 to 1001). The remaining six code combinations i.e. 1010 to 1111 are invalid in BCD.

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

- For example,

$(754)_{10} = (0111\ 0101\ 0100)_{BCD}$

$(23)_{10} = (0010\ 0011)_{BCD}$

- Advantages : ->It is easy to covert from decimal to BCD and vice versa.

-Disadvantages : -> The addition and subtraction of BCD have different rules.

-> The BCD arithmetic is little more complicated.

-> BCD needs more number of bits than binary to represent the decimal number. So BCD is less efficient than binary.

Codes – Excess-3 Code

- This code doesn't have any weights. So, it is an **un-weighted code**.
- We will get the Excess 3 code of a decimal number by adding three 00110011 to the binary equivalent of that decimal number. Hence, it is called as Excess 3 code.
- It is a **self-complementing** code.

Decimal Number → 8421 BCD → (Add 0011) → Excess-3

Decimal	BCD 8 4 2 1	Excess-3 BCD + 0011
0	0 0 0 0	0 0 1 1
1	0 0 0 1	0 1 0 0
2	0 0 1 0	0 1 0 1
3	0 0 1 1	0 1 1 0
4	0 1 0 0	0 1 1 1
5	0 1 0 1	1 0 0 0
6	0 1 1 0	1 0 0 1
7	0 1 1 1	1 0 1 0
8	1 0 0 0	1 0 1 1
9	1 0 0 1	1 1 0 0

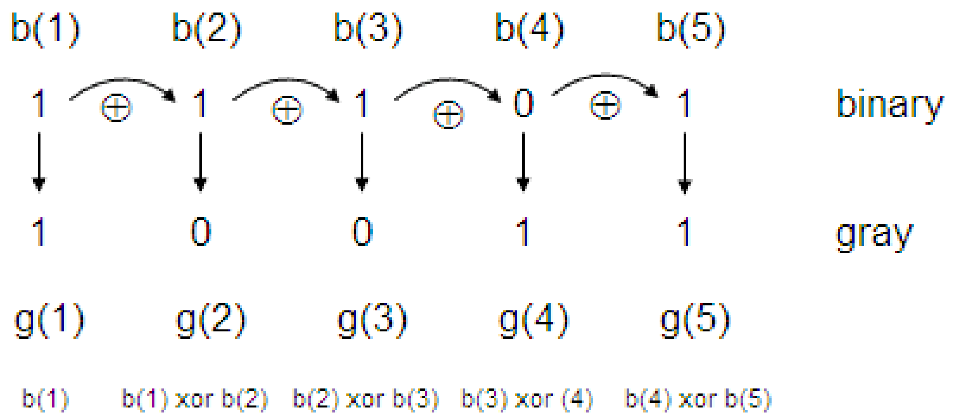
- Excess-3 for $(52)_{10}$ is,

```
      5                 2
     0101              0010
 +   0011          +   0011
  --------------    --------------
     1000              0101
```

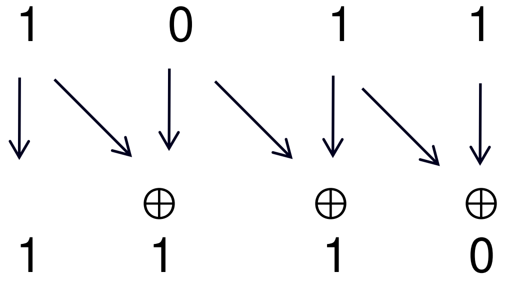
Excess-3 of $(52)_{10}$ = 1000 0101

Codes – Gray Code

- Gray code is not weighted that means it does not depends on positional value of digit.
-This cyclic variable code that means every transition from one value to the next value involves only one bit change i.e. **Unit Distance Code**.
- Binary to Gray Code conversion



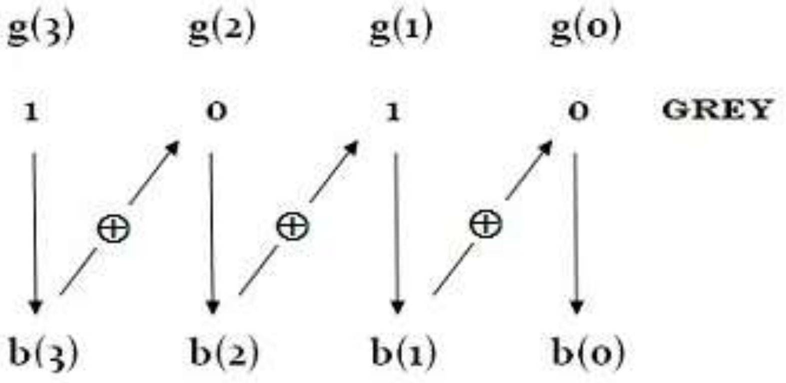
For example, Gray code of $(1011)_2$



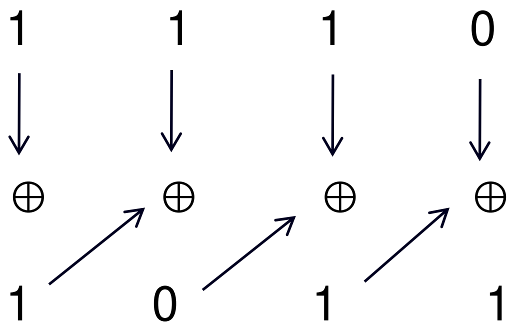
Decimal	Binary Code (input)	Gray Code (output)
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Codes – Gray Code

Gray to Binary Conversion



For example,
Convert (1110) gray code into binary.



Gray Code				Binary			
g_3	g_2	g_1	g_0	b_3	b_2	b_1	b_0
0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	1
0	0	1	0	0	0	1	1
0	0	1	1	0	0	1	0
0	1	0	0	0	1	1	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1
1	0	0	0	1	1	1	1
1	0	0	1	1	1	1	0
1	0	1	0	1	1	0	0
1	0	1	1	1	1	0	1
1	1	0	0	1	0	0	0
1	1	0	1	1	0	0	1
1	1	1	0	1	0	1	1
1	1	1	1	1	0	1	0

Codes – Gray Code - Applications

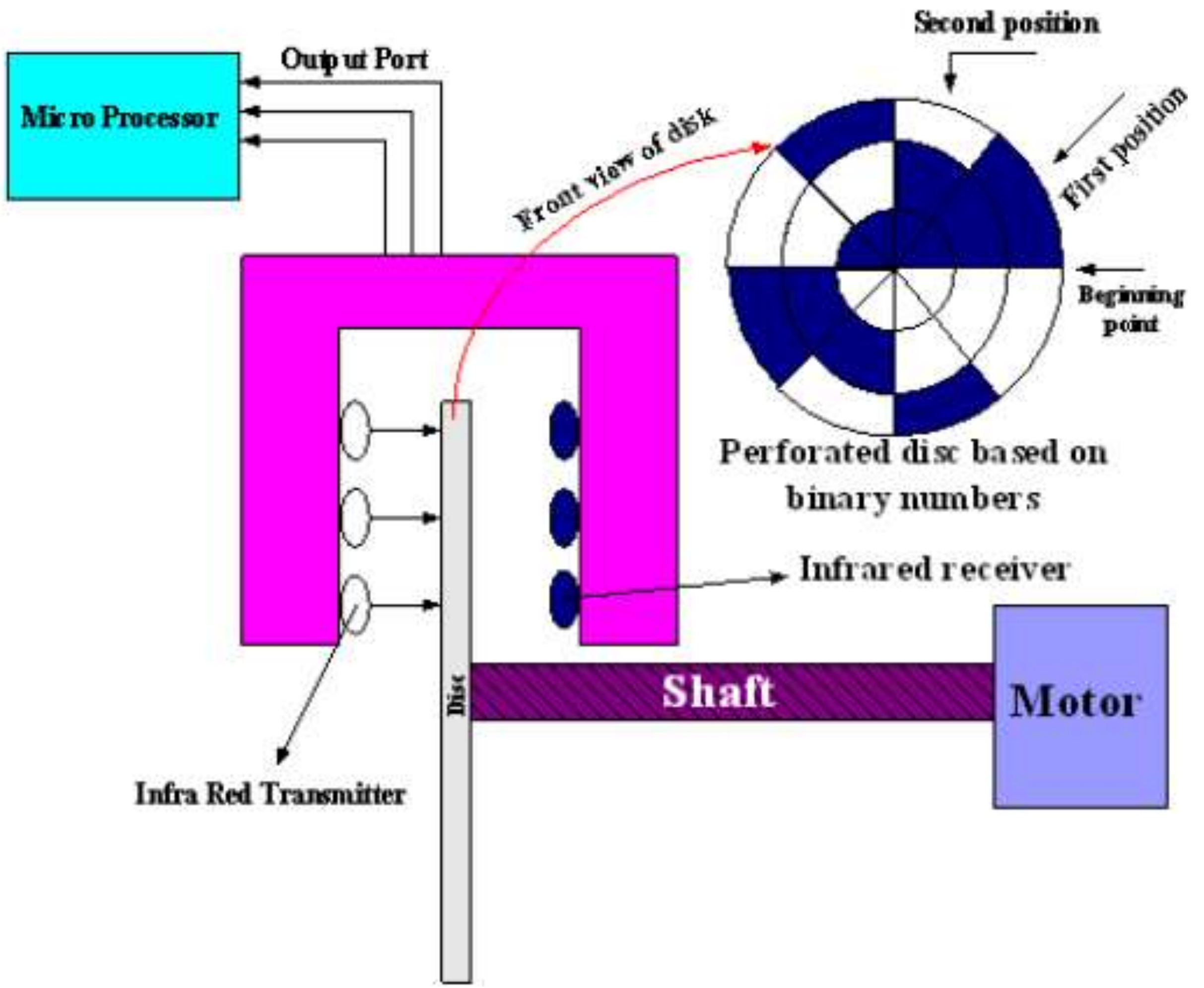
Gray code is a non-weighted code and is a special case of unit-distance code.

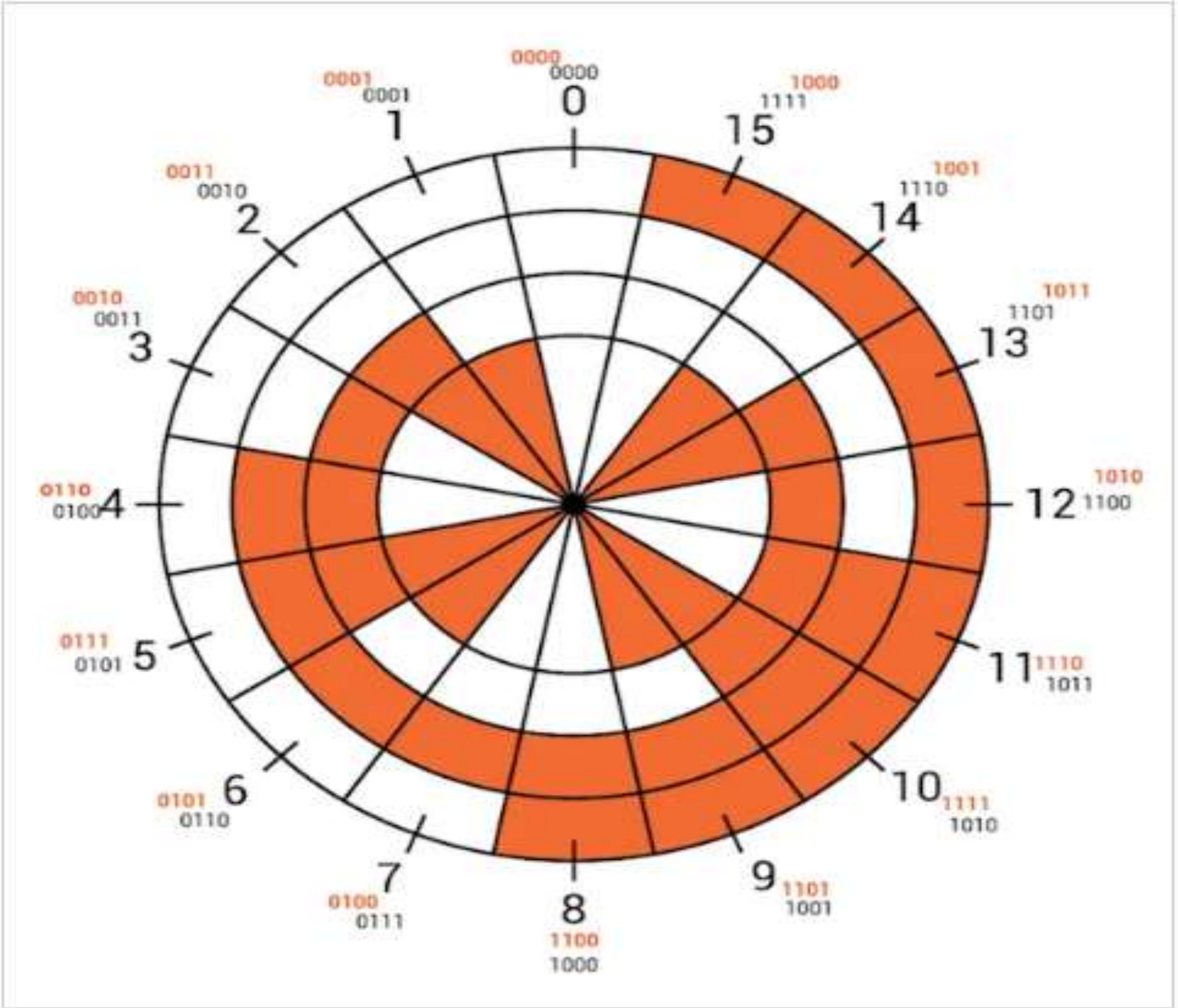
1. Karnaugh map (K-maps)

AB \ CD	00	01	11	10
00	X [0]	X [1]	0 [3]	X [2]
01	1 [4]	1 [5]	1 [7]	0 [6]
11	0 [12]	1 [13]	0 [15]	1 [14]
10	1 [8]	0 [9]	1 [11]	1 [10]

Codes – Gray Code - Applications

Gray code is a non-weighted code and is a special case of unit-distance code.





Codes – Alphanumeric Codes

- Computer is a digital system and can only deal with 1’s and 0’s. So to deal with letters and symbols they use alphanumeric codes.
- Alphanumeric codes, also called character codes, are binary codes used to represent alphanumeric data. The codes write alphanumeric data, including letters of the alphabet, numbers, mathematical symbols and punctuation marks, in a form that is understandable and process able by a computer.
- **American Standard-Code for Information Interchange** (**ASCII**)
- **Extended Binary Coded Decimal Interchange Code (EBCDIC)**

Codes – Alphanumeric Codes

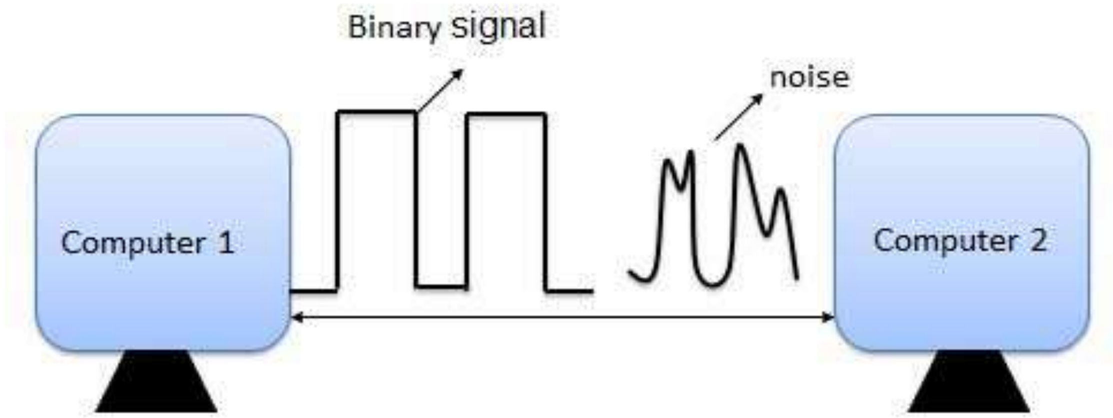
- **American Standard-Code for Information Interchange (ASCII 7- bits)**
- **Extended Binary Coded Decimal Interchange Code (EBCDIC 8- bits)**

ASCII Alphabet			
A	1000001	N	1001110
B	1000010	O	1001111
C	1000011	P	1010000
D	1000100	Q	1010001
E	1000101	R	1010010
F	1000110	S	1010011
G	1000111	T	1010100
H	1001000	U	1010101
I	1001001	V	1010110
J	1001010	W	1010111
K	1001011	X	1011000
L	1001100	Y	1011001
M	1001101	Z	1011010

Character	EBDIC Bit Configuration	Character	EBDIC Bit Configuration
A	1100 0001	S	1110 0010
B	1100 0010	T	1110 0011
C	1100 0011	U	1110 0100
D	1100 0100	V	1110 0101
E	1100 0101	W	1110 0110
F	1100 0110	X	1110 0111
G	1100 0111	Y	1110 1000
H	1100 1000	Z	1110 1001
I	1100 1001	0	1111 0000
J	1101 0001	1	1111 0001
K	1101 0010	2	1111 0010
L	1101 0011	3	1111 0011
M	1101 0100	4	1111 0100
N	1101 0101	5	1111 0101
O	1101 0110	6	1111 0110
P	1101 0111	7	1111 0111
Q	1101 1000	8	1111 1000
R	1101 1001	9	1111 1001

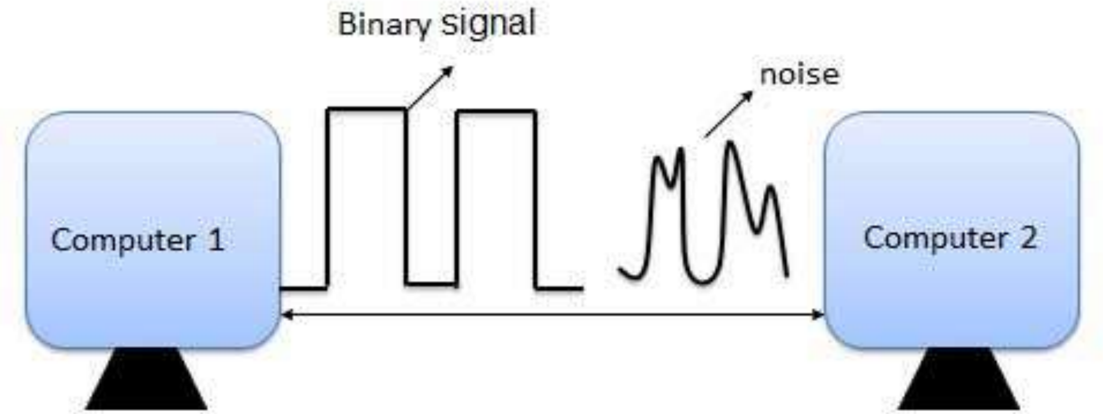
Codes – Parity Code

Error is a condition when the output information does not match with the input information. During transmission, digital signals suffer from noise that can introduce errors in the binary bits travelling from one system to other. That means a 0 bit may change to 1 or a 1 bit may change to 0.



Codes – Error detecting and error correcting

Error detecting (Parity Code): Error is a condition when the output information does not match with the input information. During transmission, digital signals suffer from noise that can introduce errors in the binary bits travelling from one system to other. That means a 0 bit may change to 1 or a 1 bit may change to 0.



Error correcting (Hamming Code): Hamming code is useful for both detection and correction of error present in the received data.