There are two way of representing the numerical value of quantities :

Analog Representation

Digital Representation

In analog representation a quantity is represented by a voltage, current or meter movement that is proportional to the value of that quantity

Example :

- Automobile Speedometer
- Analog Voltmeter or Ammeter

Analog quantities can vary over a continuous range of values

In digital representation the quantities are represented not by proportional quantities but by symbols called digits.

Example : A Digital Watch

It provides the time of day in the form of decimal digits which represent hours and minutes (and sometimes seconds).

This digital representation of the time of day changes in discrete steps.

Advantages of Digital Techniques

1. Digital systems are easier to design as circuits used are only switching circuits with only HIGH and LOW range

2. Information storage is easy in digital form

3. Accuracy and precision are greater

4. Operation can be programmed

5. Digital circuits are less affected by noise. The spurious fluctuation in voltage (noise) are not as critical in digital systems because the exact value of a voltage is not important. *Noise does not have an effect unless it is high enough to make a HIGH become LOW or vice versa.*

6. More digital circuitry can be fabricated on IC chips

Limitation of Digital Techniques

The real world is mainly analog. The quantities we sense or use tend to be mostly continuous analog variables. In order to process these digitally, we need to use analog-to-digital converters to get digital representations of the analog signal. We may also need digital-to-analog converters to convert digital quantities to analog ones which we can output to the outside world.

Number Systems and Codes

Numbering System

Decimal	Base 10, using 0,1,….,8,9	
Binary	Base 2, using only 0 & 1	*used in computers and digital systems*
Octal	Base 8, using 0,1,….,6,7	
Hexadecimal	Base 16	

Any base can be used and numbers changed from one base to another to change representation. The Mayans used a Base 20 system, the Babylonians used Base 60!!!

Decimal Number System – Base 10

This uses ten numerals or symbols, i.e. 0, 1, 2, 3, 4, 5, 6, 7, 8 and 9

A positional value system is used with these symbols to represent any number.

Example: $a_N, a_{N-1}, \ldots\ldots, a_1, a_0 \,.\, a_{-1}, a_{-2}, \ldots\ldots, a_{-K}$
represents the following number

$$a_0 10^0 + a_1 10^1 + \ldots\ldots + a_{N-1} 10^{N-1} + a_N 10^N + a_{-1} 10^{-1} + \ldots\ldots + a_{-K} 10^{-K}$$

Binary Number System

The binary system uses only two symbols, 0 and 1

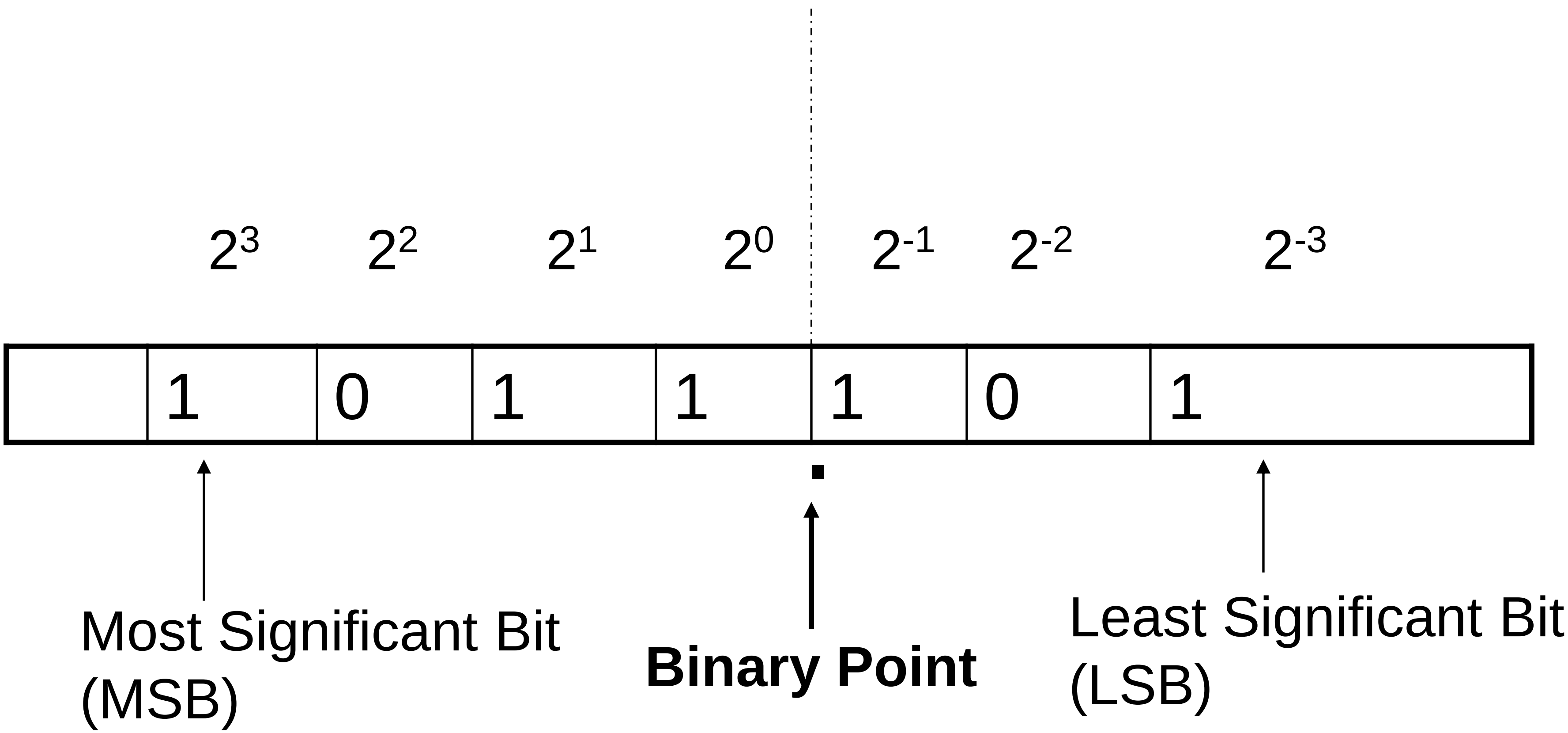
It is also a positional value system where each binary digit has its own value or weight expressed as a power of 2

- Places to the left of the binary point are positive powers of 2

- Places to the the right of the binary point are negative powers of 2

This is exactly what we did in the decimal system except that there we used powers of 10 instead of powers of 2!

Example : The Number 1011.101_2



BINARY TO DECIMAL CONVERSION

Example : Convert the number 1011.101_2 to its decimal equivalent

$$1011.101_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$$
$$+1 \times 2^{-1} + 0 \times 2^{-2} + 1 \times 2^{-3}$$
$$= 8 + 0 + 2 + 1 + 0.5 + 0 + 0.125$$
$$= 11.625_{10}$$

DECIMAL TO BINARY CONVERSION

Done by repeated divisions (see next slide)

(see next slide)

Example - Convert 456_{10} into binary

Answer = 111001000_2

2 | 456 = 111001000_2

2 | 228--------0 remainder

2 | 114------0

2 | 57----0

2 | 28--1

2 | 14---0

2 | 7----0

2 | 3----1

2 | 1---1

0---1

Write the remainder from bottom to top

Convert 42.9375_{10} to binary

2 | 42	0.9375
2 | 21----0	x 2
2 | 10----1	1.8750 ---1
2 | 5----0	0.8750
2 | 2----1	x 2
2 | 1----0	1.7500 --- 1
0---1	0.75
	x 2
	1.50------- 1
	0.5
	x 2
	1.0 -------1

Continue till 0 is obtained in fraction part

$$42.9375_{10} = 101010.1111_2$$

Example : Convert 25.012_{10} into binary

2 | 25
2 | 12------1
2 | 6-------0
2 | 3------0
2 | 1------1
0-----1

0.012
x 2
0.024----0
x 2
0.048----0
x 2
0.096----0
x2
0.192--------0

0.192
x 2
0.384----0
x 2
0.768--- 0
x 2
1.536----1
0.536
x 2
1.072----1

0.072
x 2
0.144---- 0
x 2
0.288----0

25.012_{10} =11001.0000001100_2

(approximately)

Octal Number System

It has a base of 8 and 8 possible digits : 0,1,2,3,4,5,6,7

Octal to decimal conversion

$24.6_8 = 2 \times 8^1 + 4 \times 8^0 + 6 \times 8^{-1} = 16 + 4 + 0.75 = 20.75_{10}$

Decimal to octal conversion

Example : Convert 49.21875_{10} into octal

$$\begin{array}{r|l} 8 & 49 \\ \hline 8 & 6\text{---}1 \\ \hline & 0\text{---}6 \end{array}$$

$$\begin{array}{r} 0.21875 \\ \times\ 8 \\ \hline 1.75000\text{----}1 \\ .75 \\ \times\ 8 \\ \hline 6.00\text{-------}6 \end{array}$$

$49.21875_{10} = 61.16_8$

Octal to Binary Conversion

Conversion from octal to binary is performed by converting each octal digit to its 3-bit binary equivalent

Ex : $472_8 = 100\ 111\ 010_2$

Ex : $642.71_8 = 110\ 100\ 010.111\ 001_2$

Binary to Octal Conversion

The bits of the binary number are grouped into groups of three bits starting at the LSB. Then each group is converted into its octal equivalent

Ex. $101100011001_2 = \underbrace{\underline{101}}_{5}\ \underbrace{\underline{100}}_{4}\ \underbrace{\underline{011}}_{3}\ \underbrace{\underline{001}}_{1} = 5431_8$

Ex. $\underline{100}.\underline{101}_2 = 4.5_8$

Hexadecimal Number System

It uses base 16 and 16 possible digit symbols 0 to 9 plus the letters A,B,C,D,E ,F

$A=10_D$, $B=11_D$, $C=12_D$, $D=13_D$, $E=14_D$, $F=15_D$

Hex to Decimal Conversion

Example: $356_{16} = 3 \times 16^2 + 5 \times 16^1 + 6 \times 16^0$

$= 854_{10}$

Decimal to Hex Conversion

Example: 567.1875_{10}

$= 237.3_{16}$

16 | 567
16 | 35----7
16 | 2----3
0----2

0.1875
x 16
3.000---3

Hex to Binary Conversion

Groups of 4 digits are taken

Example: $F9_{16} = 1111\ 1001_2$

Example: $20E.CA_{16} = 0010\ 0000\ 1110.1100\ 1010_2$

Binary to Hex Comversion

Grouped into groups of four bits, and each group is converted to its equivalent hex digit

Example: 1110100110_2 = $\underline{0011}$ $\underline{1010}$ $\underline{0110}$ = $3A6_{16}$

Example: 1101 **.** 0011 1100 = D**.**$3C_{16}$

Example : Convert $B2F_{16}$ to octal

First convert hex to binary, then convert the resultant binary to octal

$B2F_{16}$ = 1011 0010 1111_2 = $\underline{101}$ $\underline{100}$ $\underline{101}$ $\underline{111}$= 5457_8

CODES

When numbers, letters or words are represented by a special group of symbols, we say that they are being encoded and the group of symbols is called code.

Binary- Coded- Decimal Code (BCD)

If each digit of a decimal number is represented by its binary equivalent, the result is a code called binary-coded- decimal or BCD. Since a decimal digit can be as large as 9, 4 bits are required to code each digit.

Example: 874_D = 1000 0111 0100 (BCD)

Gray Code

This belongs to a class of codes called minimum-change codes, in which only one bit in the code groups changes when going from one stage to the next. The Gray code is an un-weighted code.

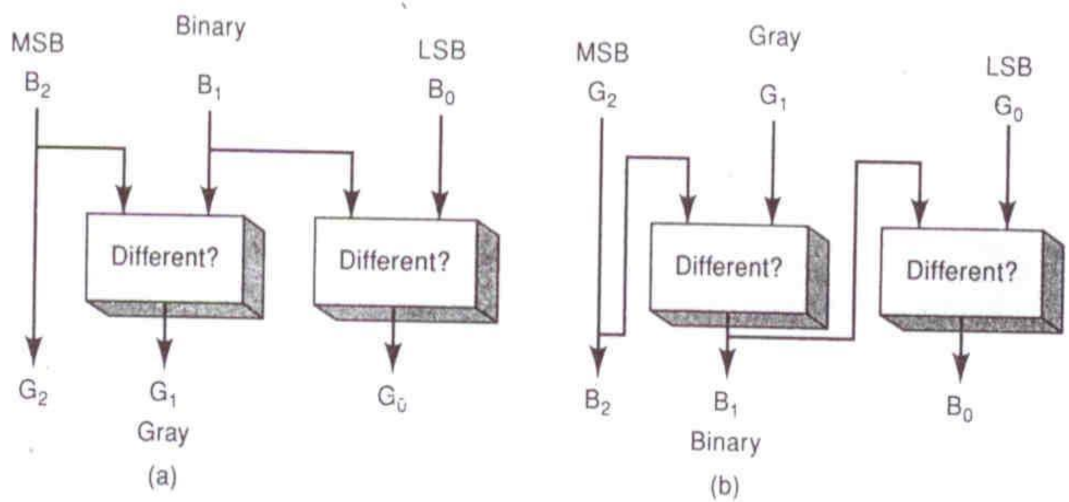
This code is not suited for arithmetic operation but finds application in input/output devices and some types of Analog to Digital Converters (ADCs).

The attractive feature of this code is that it minimizes the effect of any error made in converting from digital to analog!

Binary Equivalent	Gray Code
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 0
0 1 0 1	0 1 1 1
0 1 1 0	0 1 0 1
0 1 1 1	0 1 0 0
1 0 0 0	1 1 0 0
1 0 0 1	1 1 0 1
1 0 1 0	1 1 1 1
1 0 1 1	1 1 1 0
1 1 0 0	1 0 1 0
1 1 0 1	1 0 1 1
1 1 1 0	1 0 0 1
1 1 1 1	1 0 0 0

Three-bit binary and Gray code equivalents.

B_2	B_1	B_0	G_2	G_1	G_0
0	0	0	0	0	0
0	0	1	0	0	1
0	1	0	0	1	1
0	1	1	0	1	0
1	0	0	1	1	0
1	0	1	1	1	1
1	1	0	1	0	1
1	1	1	1	0	0



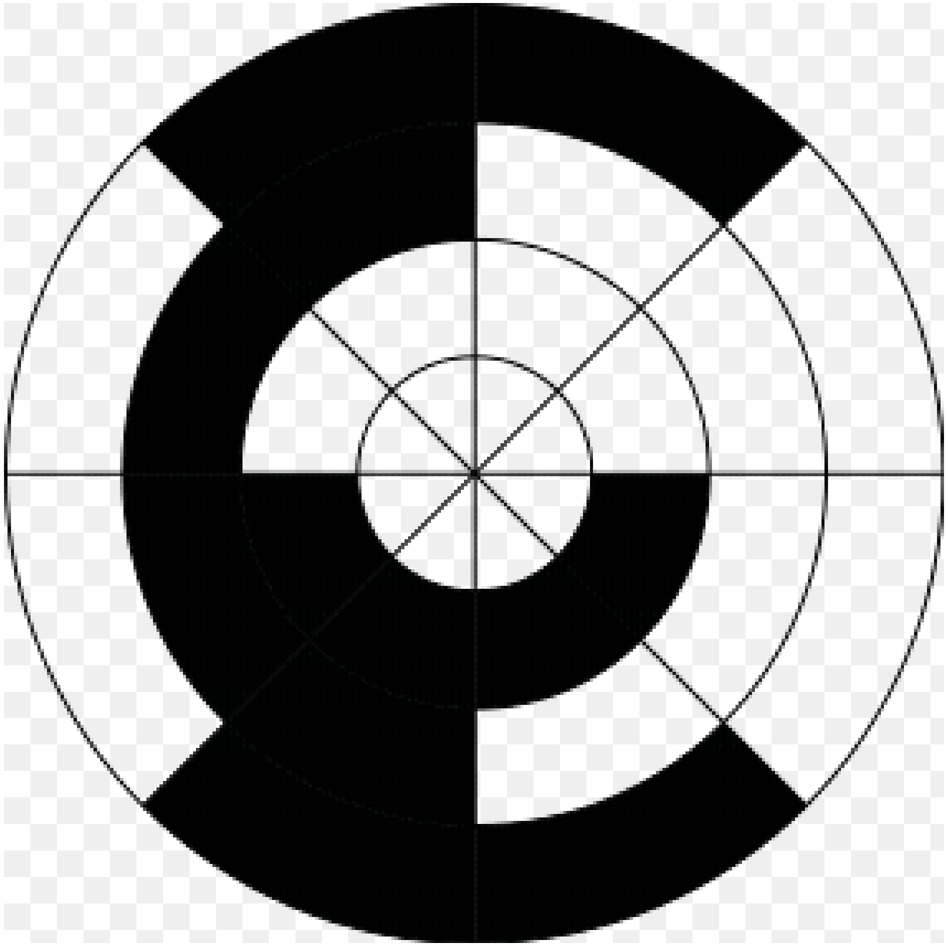
Converting (a) binary to Gray and (b) Gray to binary.

0 White

1 Black

3-bit Gray Code

000
001
011
010
110
111
101
100



3-bit Positional Encoder Disk

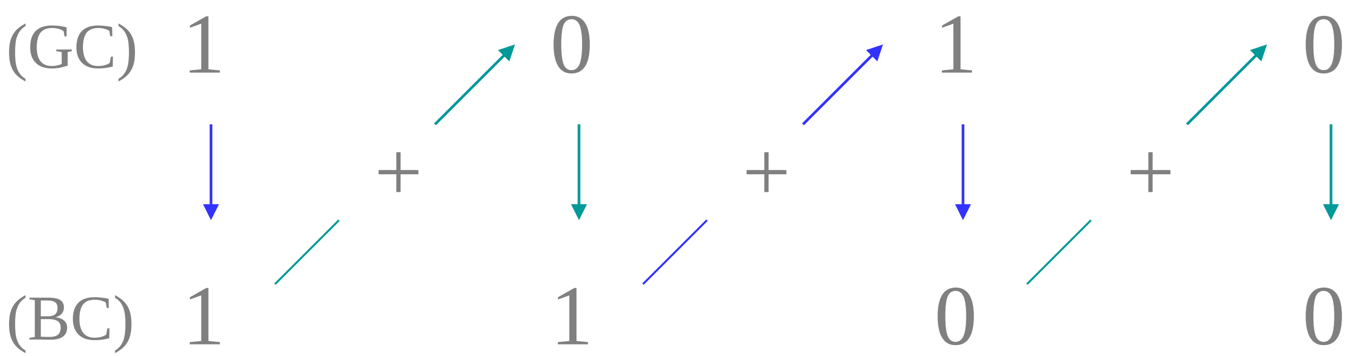
Binary to Gray Code Conversion

(BC) 1 + 1 + 0 + 0

(GC) 1 0 1 0

- MSB does not change as a result of conversion
- Start with MSB of binary number and add it to neighboring binary bit to get the next Gray code bit
- Repeat for subsequent Gray coded bits

Gray to Binary Code Conversion



- MSB does not change as a result of conversion
- Start with MSB of binary number and add it to the second MSB of the Gray code to get the next binary bit
- Repeat for subsequent binary coded bits

Alphanumeric Codes

An alphanumeric code represents all of the various characters and functions that are found in a standard typewriter (or computer) keyboard.

ASCII Code:

The most widely used alphanumeric code, the American Standard Code for Information Interchange used in computers.

The original ASCII code is a 7 bit code and so it has 2^7 =128 possible code groups.

The extended ASCII code has 8 bits and can represent 256 characters.

UNICODE can represent upto 1,114,112 characters and is now available for most scripts.

Character	7- Bit ASCII	Hex
A	1000001	41
B	1000010	42
C	1000011	43
…	….	…
Z	1011010	5A
0	0110000	30
…	…	…
9	0111001	39

a, b, … blank, etc.