Digital Logic Design :

Lecture 3

Signed Numbers :

The sign Bit : The left most bit in a signal binary number is the sign bit.

- sign bit is '0' for positive numbers
  '1' for negative numbers

There are three ways in which signed numbers can be represented in binary form.

a) sign - magnitude system
b) 1's complement system
c) 2's complement system.

a) sign - magnitude system :

8 bit signed binary number representation of decimal number + 25

$$\underbrace{0}_{\text{Sign bit}}\ \underbrace{0011001}_{\text{magnitude bit}}$$

Decimal number -25 is expressed as,

10011001

b) 1's complement system :

Positive numbers in the 1's complement system are represented the same way as the positive sign-magnitude numbers.

Negative numbers are the 1's complement of the corresponding positive number.

∴ decimal -25 can be expressed by 8 bit in 1's complement system as

11100110 [1's complement of +25 (00011001)]

c) 2's complement system :

Positive numbers in the 2's complement system are represented the same way as in the sign magnitude and 1's complement systems.

Negative numbers are the 2's complement of the corresponding positive numbers.

In 2's complement system -25 can be represented by 8 bit as

11100111 [2's complement of +25 (00011001)]

Evaluate the value of 11001010 in

a) sign - magnitude system

b) 1's complement system

c) 2's complement system

a) sign - magnitude system :

sign bit = 1

magnitude bit = 100 10 10

$= 2^6 + 2^3 + 2^1$

$= (74)_{10}$

$\therefore$ 11001010 $= (-74)_{10}$

b) 1's complement system :

$11\,00\,10\,10 = (-2^7 + 2^6 + 2^3 + 2^1) + 1$

$= -128 + 64 + 8 + 2 + 1$

$= (-53)_{10}$

the left most bit is to be given negative weight and as 1 is to be added.

c) 2's complement system :

$11\,00\,10\,10 = -2^7 + 2^6 + 2^3 + 2^1$

$= -128 + 64 + 8 + 2$

$= (-54)_{10}$

the left most bit is to be given negative weight.

Arithmatic operations with signed numbers :

1) Addition :

0 0 0 0 0 1 1 1		7
0 0 0 0 0 1 0 0		+4
0 0 0 0 1 0 1 1		11

2) Subtraction : To subtract two signed numbers, the 2's complement of the subtrahend is added with the minuend and the carry bit is discarded.

a) 00001000 − 00000011 = ?

0 0 0 0 1 0 0 0	minuend +8
1 1 1 1 1 1 0 1	complement of subtrahend (−3)
1 0 0 0 0 0 1 0 1	

↳ Discard

∴ Difference is 5.

b) 00010000 − 00011000 = ? 16 − 24 = ?

0 0 0 1 0 0 0 0	minuend +16
1 1 1 0 1 0 0 0	2's complement of subtrahend (−24)
1 1 1 1 1 0 0 0	

[−128 + 64 + 32 + 16 + 8 = −8]

Two negative numbers are added :

11111011 + 11110111 = ?

11111011	-5
11110111	-9
111110010	-14

Discard

final carry bit is discarded. The sum is in 2's complement form.

Overflow condition :

Overflow can occur when both numbers are positive or both numbers are negative.

01111101 + 00111010 = ?

01111101	125
00111010	58
10110111	183

sign incorrect.

The sum 183 required 8 magnitude bit so result incorrect.

10000011 + 11000110 = ?

```
  10000011        -125
  11000110       - 58
----------      ------
 101001001       -183
```

↑ sign incorrect

The Gray Code :

The important feature of the Gray code is that, it exhibits only a single bit change from one code number to the next.

Four bit Gray Code :

Decimal	Binary	Gray code
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

▣ Binary to Gray Conversion :

1) The most significant bit (MSB) in Gray and Binary code is similar

2) Other bits are obtained by XOR operation of two adjacent binary bits

1	0	1	1	Binary
1	1	1	0	Gray

1	1	0	0	Binary
1	0	1	0	Gray

1	1	0	1	Binary
1	0	1	1	Gray

1	1	1	0	Binary
1	0	0	1	Gray

▣ Gray to Binary conversion :

i) The MSB of Binary and Gray code is similar.

ii) Each new Binary code bits are obtained by XOR operation of previous Binary code bit with new Gray code bit

1	0	0	0	Gray
1	1	1	1	Binary

1	0	0	1	Gray
1	1	1	0	Binary

1	0	1	1	Gray
1	1	0	1	Binary

1	1	0	0	Gray
1	0	0	0	Binary

BOOLEAN ALGEBRA AND LOGIC SIMPLIFICATION :

Boolen Addition : equivalent to OR operation

$0+0 = 0$

$0+1 = 1$

$1+0 = 1$

$1+1 = 1$

Some example of sum terms, $A+B$, $A+\bar{B}$, $A+B+\bar{C}$ etc.

Boolen multiplication : equivalent to AND operation

$0 \cdot 0 = 0$

$0 \cdot 1 = 0$

$1 \cdot 0 = 0$

$1 \cdot 1 = 1$

product terms, AB, $A\bar{B}$, $AB\bar{C}$ etc.

Laws of BOOLEAN Algebra :

① Commutative Laws :

$A+B = B+A$ commutative law of addition

$AB = BA$ commutative law of multiplication

(2) Associative Laws :

Associative Law of addition

$A+(B+C) = (A+B)+C$

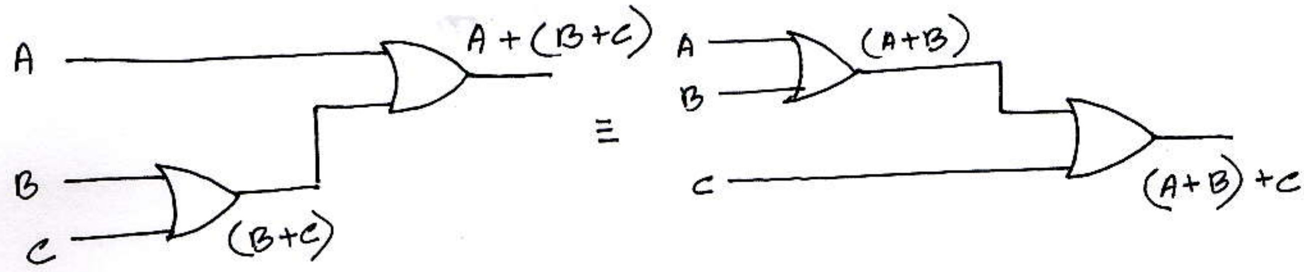

Fig. : Application of associative law of addition

Associative Law of multiplication

$A(BC) = (AB)C$

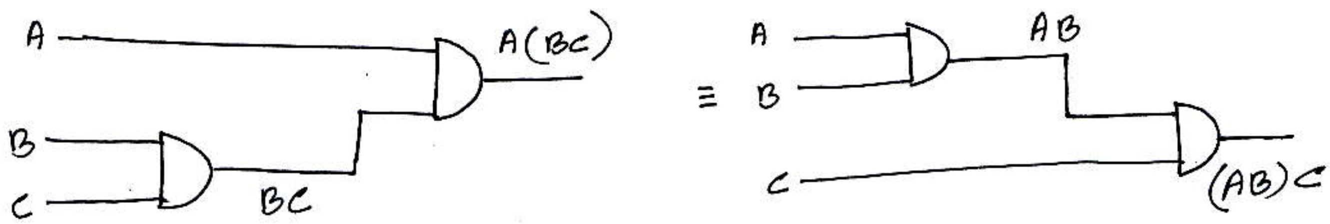

Fig. : Application of associative law of multiplication

(3) Distributive Law :

$A(B+C) = AB + AC$