2 : ARITHMETIC

REPRESENTATION OF NUMBERS:-

There are a no of ways of representing numbers. The most widely accepted ways of representing the nos is

i) Fixed point numbers
ii) Floating point numbers.

FIXED POINT NUMBERS:-

In a computer, any number is stored as binary no. An integer or a fraction is called as fixed - point number.

In case of an integer, the point is assumed at the R.H.S. of the number. Ex: 31. → 0011000

In case of fractions, the binary point is to the LHS & its position is as follows.

Ex: .31 → .0011 0001

But, the limitation here is, the position of the point is fixed. Thus, the nos of this type are called FIXED POINT NUMBER. Also, we cannot handle large nos in fixed point no

The fixed point nos are represented in 2 forms. SIGNED Integers & UNSIGNED Integers.

The Unsigned integers represent +ve nos.

To represent negative nos the main techniques used are

- Signed - magnitude representation
- 1's - compliment representation
- 2's - compliment representation

Let the n-bit vector be represented as follows:

$B = B_0 B_1 B_2 \dots B_{n-1}$. In all the above 3 techniques, a bit B_n is added at the left end of the vector. The bit B_n is '0' for +ve nos & '1' for -ve nos.

SIGN - MAGNITUDE REPRESENTATION:

The representation shown shows sign-magnitude format for 8-bit signed nos.

8-bit

B_7	B_6	B_5	B_4	B_3	B_2	B_1	B_0

Sign: $0 \to +ve$, $1 \to -ve$ — Magnitude

In the sign-magnitude representation, the MSB is used to represent sign of the no. If MSB is zero, the no is positive & if MSB is 1, the no is negative. The remaining bits of the no represents the magnitude.

Ex:- $+6 = 0,000\ 0110$

$-6 = 1,000\ 0110$

$+24 = 0,001\ 1000$

$-64 = 1,100\ 0000$

$16 \overline{)24}$, $1-8 = 18H$

$16 \overline{)64}$, $4-0 = 40H$

The minimum number is $0,000\ 0000$

+ve maximum number is $0,111\ 1111 = +127_{(10)}$

-ve maximum number is $1,111\ 1111 = -127_{(10)}$

For Addition & Subtraction, it is necessary to consider sign's of both the nos. There are 2 representations for '0':

$+0 = 0,000\ 0000$

$-0 = 1,000\ 0000$

Scanned with CamScanner

* 6-BIT representation :-

B_5	B_4	B_3	B_2	B_1	B_0
Sign	Magnitude				

Max +ve no → +31

Max -ve no → -31

Ex: +8 = 0,01000

-24 = 1,11000

* 1's COMPLEMENT REPRESENTATION :

A 1's complement representation is simply a represe of -ve nos. For +ve nos, the 1's complement representa is same as unsigned no in sign-magnitude form.

In 1's complement representation, the weight of t MSB is -ve & only -ve nos are complemented.

To obtain the 1's complement of a no, complement each bit of the number.

Ex: Let us consider 6-bit representation of the following nos in one's complement.

i) -6 → 1,11001

+6 → 0,00110
1's → 1,11001 → -6

ii) +8 → 0,01000

iii) -24 → 1,00111

+24 → 0,11000
1's → 1,00111 → -24

iv) -18 → 1,01101

+18 → 0,10010
1's → 1,01101 → -18

Scanned with CamScanner

The addition in 1's complement representation is carried out by addition of sign bits & the carry-out of MSB is called END - AROUND CARRY & is added to LSB.

Ex: i) +9
-6

(Using 6-bit)

+9 $\xrightarrow{1's}$ 0,01001
-6 $\xrightarrow{1's}$ 1,11001

$\boxed{1}$0,00010
+1

0,00011 → +3

+6 → 0,00110
-6 → 1,11001

ii) +6
-9

+6 $\xrightarrow{1's}$ 0,00110
-9 $\xrightarrow{1's}$ 1,10110

1,11100 $\xrightarrow{1's}$ 0,00011
= -3.

+9 → 0,01001
-9 → 1,10110

Ans $\xrightarrow{1's}$ 0,00011

iii) -6
-9

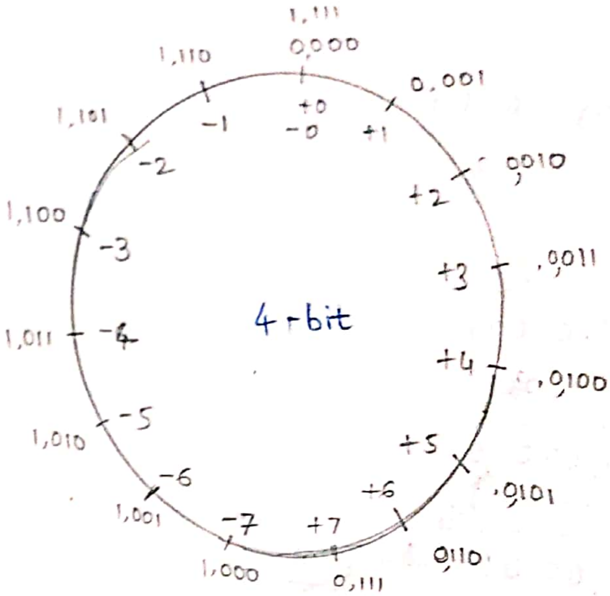
-6 $\xrightarrow{1's}$ 1,11001
-9 $\xrightarrow{1's}$ 1,10110

$\boxed{1}$1,01111
+1

1,10000 $\xrightarrow{1's}$ 0,01111
= -15

+6 → 0,00110
-6 → 1,11001
+9 → 0,01001
-9 → 1,10110

If the width of the no is 4-bit, the 1's complement integer is represented as follows:



A '0' is represented in both signed & unsigned manner. MSB represents the negative sign. The coding is continued till MSB is '1'. Till MSB is '1' the nos are positive integers. If MSB is '1', the nos are equivalent to 1's complement no.

The range of the nos for n-bits is

$-1 + 2^{n-1} \longrightarrow$ + ve integers

$+1 - 2^{n-1} \longrightarrow$ - ve integers

* 2's COMPLIMENT REPRESENTATION:

A 2's Compliment representation is simply the representation of -ve nos. For +ve nos, the 2's compliment representation is same as unsigned no in sign-magnitude form.

In 2's Compliment representation, the weight of MSB is -ve. Otherwise, it is identical to the no in unsigned representation & only -ve nos are complimented.

To form a 2's compliment no, compliment each bit & 1 is added to LSB of the result.

Q. Represent the following nos in 2's compliment (6 bit)

i) −6 → ~~1,00110~~
1,11010

| +6 → 0,00110
| −6 → 1,11001
| +1
| 1,11010

ii) −24 → 1,01000

| +24 → 0,11000
| 2's (24) → 1,01000

iii) +13 → 0,01101

iv) −18 → 1,01110

| +18 → 0,10010
| −18 $\xrightarrow{2's}$ 1,01110

The Addition in 2's compliment representation is carried out by addition of signed bits & the carry out of MSB is called END AROUND CARRY (EAC) & is neglected.

Ex. i) +9 $\xrightarrow{2's}$ 0,01001
−6 $\xrightarrow{2's}$ 1,11010

$\boxed{1}$0,00011
Ignore

0,00011 → +3(10)

| +6 → 0,00110
| −6 $\xrightarrow{2's}$ 1,11001

ii) +6 $\xrightarrow{2's}$ 0,00110
−9 $\xrightarrow{2's}$ 1,10111

1,11101

1,11101 $\xrightarrow{2's}$ 0,00011

∴ Ans. → −3(10)

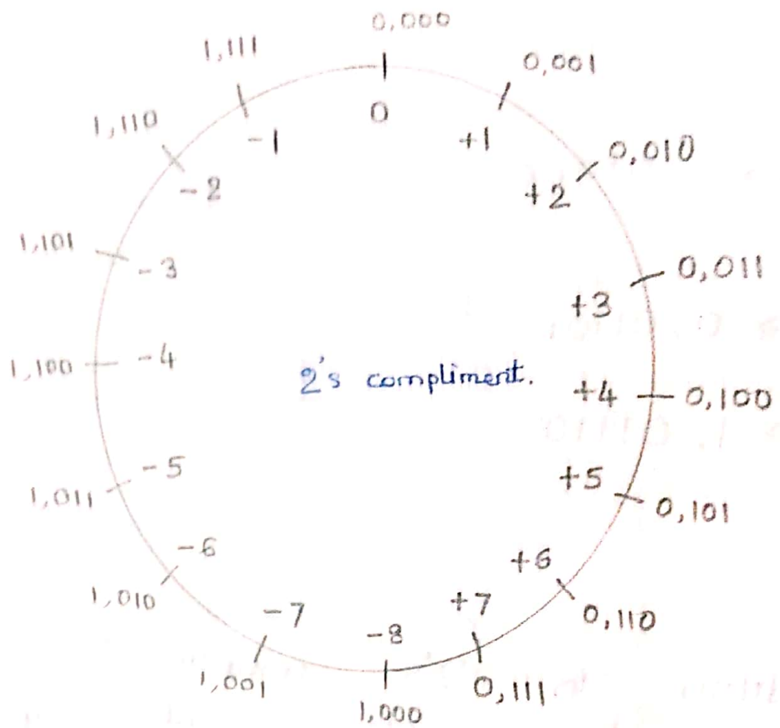
| +9 → 0,01001
| −9 $\xrightarrow{2's}$ 1,10111

| 0,11101
| $\xrightarrow{2's}$ 0,00011

iii) −6 $\xrightarrow{2's}$ 1,11010
−9 $\xrightarrow{2's}$ 1,10111

$\boxed{1}$1,10001
Ignore

1,10001 $\xrightarrow{2's}$ 0,01111
Ans → (−15)10



If the width of the no is 4-bit, then the 2's compliment integers with 4-bit is represented as above.

A 'zero' is represented in unsigned integer, the MSB represents the sign. The coding is continued till MSB is '1'. Till MSB is '1', the nos are +ve integers. If MSB is '1', the nos are equivalent of 2's compliment no.

With 4-bits, we can represent all the integers b/w -8 to +7. Here, it is seen that there are more negative values than +ve values. The range of the nos in 2's compliment representation is as follows :

$$2^{n-1} - 1 \longrightarrow \text{for +ve integer.}$$

$$-2^{n-1} \longrightarrow \text{for -ve integer.}$$