Subtraction Using 1's Complement

Binary subtraction $(X - Y)$ can be performed by adding the 1's complement of the subtrahend (Y) to the minuend(Y).

- If a carry is generated, remove the carry and add it to the result.
- If the subtrahend is larger than the minuend, then no carry is generated. The answer obtained is negative and it is 1's complement of the true result.

Ex) Subtract $(1001)_2$ from $(1101)_2$ using the 1's complement method.

			1's complement method	
	1 1 0 1		1 1 0 1	(+)
–	1 0 0 1	1's complement →	0 1 1 0	
		Carry →	1 0 0 1 1	
		Add Carry →	1	
			0 1 0 0	

Ex) Subtract $(0111)_2$ from $(1010)_2$ using the 1's complement method.

Ex) Subtract $(1100)_2$ from $(1001)_2$ using the 1's complement method.

		1's complement method	
1 0 0 1		1 0 0 1	(+)
– 1 1 0 0	1's complement →	0 0 1 1	
		1 1 0 0	
	1's complement →	0 0 1 1	
	True result	– 0 0 1 1	

Ex) Subtract $(1101)_2$ from $(1011)_2$ using the 1's complement method.

Subtraction Using 2's Complement

Binary subtraction $(X - Y)$ can be performed by adding the 2's complement of the subtrahend (Y) to the minuend (X).

- If a carry is generated, ignore it.
- If the subtrahend is larger than the minuend, then no carry is generated. The answer obtained is in 2's complement and it is negative. To get a true answer take the 2's complement of the number and change the sign.

Ex) Subtract $(0111)_2$ from $(1101)_2$ using the 2's complement method.

$$\begin{array}{r} 1\ 1\ 0\ 1 \\ -\ \underline{0\ 1\ 1\ 1} \end{array}$$

2's complement method

		1 1 0 1	(+)
2's complement	$\rightarrow$	1 0 0 1	
Carry	$\rightarrow$	1 0 1 1 0	
Discard Carry		0 1 1 0	(Result)

Ex) Subtract $(0101)_2$ from $(1011)_2$ using the 2's complement method.

Ex) Subtract $(1010)_2$ from $(1001)_2$ using the 2's complement method.

$$\begin{array}{r} 1\ 0\ 0\ 1 \\ -\ \underline{1\ 0\ 1\ 0} \end{array}$$

2's complement method

		1 0 0 1	(+)
2's complement	$\rightarrow$	0 1 1 0	
		1 1 1 1	
2's complement	$\rightarrow$	0 0 0 1	
True result		–0001	

Ex) Subtract $(1110)_2$ from $(1010)_2$ using the 2's complement method.

Comparison between 1's and 2's Complements

1. The 1's complement has the advantage of being easier to implement since the only thing to be done is to change the ones to zeros and vice versa while to implement 2's complement we need to find the 1's complement then add one to the answer.
2. During the subtraction of two numbers, the 2's complement has advantageous since only one arithmetic addition is required. The 1's complement requires two arithmetic since the end carry has to be added to the result.

Signed Binary Numbers

In decimal system, generally a plus (+) sign denotes a positive number whereas a minus (–) sign denotes a negative number. But in digital circuits, there is no plus or minus sign because everything in digital circuits is represented in terms of 0 and 1.

In digital system, there are three types of representations for signed binary numbers which are: sign-magnitude representation, 1's complement representation and 2's complement representation.

Sign-magnitude Representation

Generally, The MSB bit is used as the *sign bit* where a 0 is reserved for a positive number and a 1 is reserved for a negative number.

For example, an 8-bit signed binary number 01101001 represents a positive number whose magnitude is $(1101001)_2 = (105)_{10}$. The MSB is 0, which indicates that the number is positive.

On the other hand, in the signed binary form, 11101001 represents a negative number whose magnitude is $(1101001)_2 = (105)_{10}$. The 1 in the MSB position indicates that the number is negative and the other seven bits give its magnitude.

Ex) Find the decimal equivalent of the following sign-magnitude binary numbers.

(a) 0101100 *(b) 101000* *(c) 1111* *(d) 011011*

Solution.

(*a*) Sign bit is 0, which indicates the number is positive.

Magnitude $101100 = (44)_{10}$

Therefore $(0101100)_2 = (+44)_{10}$.

(*b*) Sign bit is 1, which indicates the number is negative.

Magnitude $01000 = (8)_{10}$

Therefore $(101000)_2 = (-8)_{10}$.

1’s Complement Representation

In 1’s complement representation, positive and negative numbers are a complement of each other. Also, in this type of representation, the MSB is 0 for positive numbers and 1 for negative numbers.

Ex) Represent the following numbers in 1’s complement form.

(a) +5 and –5 (b) +9 and –9 (c) +15 and –15

Solution.

(a) $(+5)_{10} = (0101)_2$

and $(-5)_{10} = (1010)_2$

(b) $(+9)_{10} = (01001)_2$

and $(-9)_{10} = (10110)_2$

(c) $(+15)_{10} = (01111)_2$

and $(-15)_{10} = (10000)_2$

2’s Complement Representation

In 2’s complement representation the resulting negative number is 2’s complement of the positive binary number. Also, in this type of representation, the MSB is 0 for positive numbers and 1 for negative numbers.

Ex) Represent the following numbers in 2’s complement form.

(a) +11 and –11 (b) +9 and –9 (c) +18 and –18

Solution.

(a) $(+11)_{10} = (01011)_2$

and $(-11)_{10} = (10101)_2$

(b) $(+9)_{10} = (01001)_2$

and $(-9)_{10} = (10111)_2$

(c) $(+18)_{10} = (010010)_2$

and $(-18)_{10} = (101110)_2$

Ex) Represent (–19) in

(a) Sign-magnitude, (b) one's complement, and (c) two's complement representation.

Signed Binary Arithmetic

Signed Binary Addition

In signed binary arithmetic, negative numbers must be initially represented in 2's complement form. Then, the addition of the two numbers is performed.

Any carry out of the sign-bit position is discarded, and negative results are automatically in 2's-complement form.

Ex) Add the following numbers:

$$a) -6+(+13) \qquad b) +6+(-13) \qquad c) -6+(-13)$$

In order to obtain a correct answer, we must ensure that the result has a sufficient number of bits to accommodate the sum.

If we start with two n bit numbers, then the sum occupies n + 1 bits. Thus we should have a sufficient number of bits to obtain a correct answer.

a)

$(+6)_{10} = (00000110)_2$

$(-6)_{10} = (11111010)_2$ 2's complement form

$(+13)_{10} = (00001101)_2$

$$\begin{array}{rrl} & 11111010 & \\ & 00001101 & + \\ \hline End\ Carry \rightarrow \quad 1 & 00000111 & \\ Discard\ Carry & 0111 & Result \end{array}$$

b)

$(+6)_{10} = (00000110)_2$

$(+13)_{10} = (00001101)_2$

$(-13)_{10} = (11110011)_2$

$$\begin{array}{rrl} & 00000110 & \\ & 11110011 & + \\ \hline 2's\ complement & 11111001 & Result \end{array}$$

c)

$(+6)_{10} = (00000110)_2$

$(-6)_{10} = (11111010)_2$

$(+13)_{10} = (00001101)_2$

$(-13)_{10} = (11110011)_2$

$$\begin{array}{rrl} & & 11111010 \\ & & 11110011 \quad + \\ \hline End\ Carry \rightarrow & 1 & 11101101 \\ Discard\ Carry & & 11101101 \quad Result\ in\ 2's\ complement \end{array}$$

Signed Binary Subtraction

Subtraction of two signed binary numbers when negative numbers are represented in **2's-complement form** is as follows:

Take the 2's complement of the subtrahend (including the sign bit) and add it to the minuend (including the sign bit). A carry out of the sign-bit position is discarded.

Ex) subtract the following number: $-6 - (-13)$

$(+6)_{10} = (00000110)_2$

$(-6)_{10} = (11111010)_2$

$(+13)_{10} = (00001101)_2$

$(-13)_{10} = (11110011)_2$

$$\begin{array}{r} 11111010 \\ 11110011 \\ \hline \end{array} \; - \qquad \begin{array}{rrl} & & 11111010 \\ & & 00001101 \quad + \\ \hline End\ Carry \rightarrow & 1 & 00000111 \\ Discard\ Carry & & 00000111 \quad Result \end{array}$$