One's Complement Method of Subtraction (Integers)

In computers, subtraction operation for binary integer numbers is done by the complement method

Computers perform this operation using an inverter logic gate

$$
\begin{array}{rl}
 & 10110011_2 \quad \text{(minuend)} \\
- & \underline{01101101_2} \quad \text{(subtrahend)}
\end{array}
$$

$$
\begin{array}{rl}
 & 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1 \\
- & \underline{0\ 1\ 1\ 0\ 1\ 1\ 0\ 1} \quad \text{(subtrahend)} \\
 & 1\ 0\ 0\ 1\ 0\ 0\ 1\ 0 \quad \text{(1's complement of subtrahend)}
\end{array}
$$

$$
\begin{array}{lcccccccccl}
1 & & 1 & 1 & & & 1 & & & & \text{(carry)} \\
 & 1 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & & \text{(minuend)} \\
+ & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & & \text{(1's complement of subtrahend)} \\
\hline
1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & & \\
+ & & & & & & & & 1 & & \text{(end around carry from line above)} \\
\hline
 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & & \text{(difference)}
\end{array}
$$

$$110010_2 - 100111_2$$

One's Complement Method of Subtraction (Fractional)

In computers, subtraction operation for real binary numbers is also done by the complement method

$$\begin{array}{rl} 1011.001_2 & \text{(minuend)} \\ - \quad 110.10_2 & \text{(subtrahend)} \\ \hline \end{array}$$

Very important that the number of digits in the inversion process for the subtrahend match the number of digits in the minuend

$$\begin{array}{rl} 1\,1\,1\,1\,.\,1\,1\,1 & \\ -\ 0\,1\,1\,0\,.\,1\,0\,0 & \text{(subtrahend)} \\ \hline 1\,0\,0\,1\,.\,0\,1\,1 & \text{(1's complement of subtraheld)} \end{array}$$

$$\begin{array}{rl} 1\quad 1\,1\quad\quad 1\,1\quad & \text{(carry)} \\ 1\,0\,1\,1\,.\,0\,0\,1 & \text{(minuend)} \\ +\ 1\,0\,0\,1\,.\,0\,1\,1 & \text{(1's complement of subtrahend)} \\ \hline 1\,0\,1\,0\,0\,.\,1\,0\,0 & \\ + \qquad\qquad\qquad 1 & \text{(end around carry from line above)} \\ \hline 0\,1\,0\,0\,.\,1\,0\,1 & \text{(difference)} \end{array}$$

$$10.110_2 - 1.11_2$$

Two's Complement Method of Subtraction (Integers)

Computers use the two's complement to perform the binary subtraction operation for integer numbers

$$\begin{array}{rl} 01100111_2 & \text{(minuend)} \\ -\ \underline{01001010_2} & \text{(subtrahend)} \end{array}$$

$$\begin{array}{rl} 1\,1\,1\,1\,1\,1\,1\,1 & \\ -\ \underline{0\,1\,0\,0\,1\,0\,1\,0} & \text{(subtrahend)} \\ 1\,0\,1\,1\,0\,1\,0\,1 & \text{(1's complement of subtrahend)} \\ +\ \underline{\qquad\qquad\quad 1} & \text{(add 1)} \\ 1\,0\,1\,1\,0\,1\,1\,0 & \text{(2's complement)} \end{array}$$

$$\begin{array}{rrl} & 1\,1\,1\;\;\;\;\;\;\;\;1\,1\;\;\;\;\; & \text{(carry)} \\ & 0\,1\,1\,0\,0\,1\,1\,1 & \text{(minuend)} \\ & +\ \underline{1\,0\,1\,1\,0\,1\,1\,0} & \text{(2's complement)} \\ \text{(ignore carry)} & \cancel{1}\,0\,0\,0\,1\,1\,1\,0\,1 & \text{(difference)} \end{array}$$

$$110010_2 - 100111_2$$

Two's Complement Method of Subtraction (Fractional)

Computers use the two's complement to perform the binary subtraction operation for real numbers

$$
\begin{array}{rl}
1010.11_2 & \text{(minuend)} \\
-\quad \underline{100.1_2} & \text{(subtrahend)}
\end{array}
$$

$$
\begin{array}{rl}
1\,1\,1\,1\,.\,1\,1 & \\
-\ \underline{0\,1\,0\,0\,.\,1\,0} & \text{(subtrahend)} \\
1\,0\,1\,1\,.\,0\,1 & \text{(1's complement of subtrahend)} \\
+\ \underline{1} & \text{(add 1)} \\
1\,0\,1\,1\,.\,1\,0 & \text{(2's complement)}
\end{array}
$$

$$
\begin{array}{rrl}
 & 1\quad 1\,1\,1 & \text{(carry)} \\
 & 1\,0\,1\,0\,.\,1\,1 & \text{(minuend)} \\
 & +\ \underline{1\,0\,1\,1\,.\,1\,0} & \text{(2's complement)} \\
\text{(ignore carry)} & \cancel{1}\ 0\,1\,1\,0\,.\,0\,1 & \text{(difference)}
\end{array}
$$

$1011.1_2 - 10.11_2$

Nine’s Complement Method of Subtraction

$$\begin{array}{rl} & 731_{10} \quad \text{(minuend)} \\ - & \underline{542_{10}} \quad \text{(subtrahend)} \end{array}$$

Nine’s complement subtraction for base 10 numbers is an alternative method

$$\begin{array}{rll} & 9\ 9\ 9 & \\ - & \underline{5\ 4\ 2} & \text{(subtrahend)} \\ & 4\ 5\ 7 & \text{(9's complement of subtrahend)} \end{array}$$

$$\begin{array}{cccccl} & 1 & & & & \text{(carry)} \\ & & 7 & 3 & 1 & \text{(minuend)} \\ & + & 4 & 5 & 7 & \text{(9's complement)} \\ \hline & 1 & (11) & 8 & 8 & \\ - & & 10 & & & \text{(subtract value of carry)} \\ \hline & 1 & 1 & 8 & 8 & \\ + & & & & 1 & \text{(end around carry from line above)} \\ \hline & & 1 & 8 & 9 & \text{(difference)} \end{array}$$

Ten's Complement Method of Subtraction

$$
\begin{array}{rl}
835_{10} & \text{(minuend)} \\
- \quad 676_{10} & \text{(subtrahend)} \\
\hline
\end{array}
$$

Ten's complement subtraction for base 10 numbers is an alternative method

$$
\begin{array}{rl}
9\;9\;9 & \\
-\quad 6\;7\;6 & \text{(subtrahend)} \\
\hline
3\;2\;3 & \text{(9's complement of subtrahend)} \\
+\quad\quad\; 1 & \text{(add 1)} \\
\hline
3\;2\;4 & \text{(10's complement)}
\end{array}
$$

$$
\begin{array}{lccccl}
 & 1 & & & & \text{(carry)} \\
 & & 8 & 3 & 5 & \text{(minuend)} \\
 & + & 3 & 2 & 4 & \text{(10's complement)} \\
\hline
 & 1 & (11) & 5 & 9 & \\
 & - & 10 & & & \text{(subtract value of carry)} \\
\hline
\text{(ignore carry)} & \cancel{1} & 1 & 5 & 9 & \text{(difference)}
\end{array}
$$

Eight’s Complement Method of Subtraction

Eight’s complement subtraction for base 8 numbers is an alternative method

	1713_8	(minuend)
-	1147_8	(subtrahend)

	7 7 7 7	
-	1 1 4 7	(subtrahend)
	6 6 3 0	(7’s complement of subtrahend)
+	1	(add 1)
	6 6 3 1	(8’s complement)

	1	1				(carry)
		1	7	1	3	(minuend)
+		6	6	3	1	(8’s complement)
	1	(8)	(13)	4	4	
-		8	8			(subtract value of carry)
(ignore carry)	~~1~~	0	5	4	4	(difference)

Sixteen's Complement Method of Subtraction

Sixteen's complement subtraction for base 16 numbers is an alternative method

$$\begin{array}{rl} 3CB_{16} & \text{(minuend)} \\ -\ 267_{16} & \text{(subtrahend)} \\ \hline \end{array}$$

$$\begin{array}{rl} F\ F\ F & \\ -\ 2\ 6\ 7 & \text{(subtrahend)} \\ \hline 13\ 9\ 8 = D\ 9\ 8 & \text{(15's complement of subtrahend)} \\ +\qquad\qquad 1 & \text{(add 1)} \\ \hline D\ 9\ 9 & \text{(16's complement)} \end{array}$$

$$\begin{array}{lcccccl} & 1 & 1 & 1 & & & \text{(carry)} \\ & & 3 & C & B & & \text{(minuend)} \\ & + & D & 9 & 9 & & \text{(16's complement)} \\ \hline & 1 & (17) & (22) & (20) & & \\ & - & 16 & 16 & 16 & & \text{(subtract value of carry)} \\ \hline \text{(ignore carry)} & \cancel{1} & 1 & 6 & 4 & & \text{(difference)} \end{array}$$

Dec	Hex
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	A
11	B
12	C
13	D
14	E
15	F

Summary of Complement Method of Subtraction

1’s and 9’s complement method for subtraction are very similar

2’s, 8’s, 10’s and 16’s complement method of subtraction are very similar

Signed Numbers

In base 10, indicating positive and negative numbers is simple ...(-14...+14)

In binary (base 2), if number is declared as signed bit, then left-most bit (MSB) becomes the sign bit

0 = positive sign
1 = negative sign

In signed representation, MSB is the sign bit
In unsigned representation, MSB is part of the magnitude

$00001110_2 = 14_{10}$

$10001110_2 = -14_{10}$

where 0 or 1 is the sign bit and the rest of the bits are the magnitude

In computers, complement methods are used to represent negative numbers

Signed Number Representation

There are 3 different methods to represent signed numbers

Signed magnitude, 1's complement, 2's complement

Signed Magnitude	
Dec	**Bin**
+0	000
+1	001
+2	010
+3	011
-0	100
-1	101
-2	110
-3	111

???

One's Complement	
Dec	**Bin**
+0	000
+1	001
+2	010
+3	011
-0	111
-1	110
-2	101
-3	100

Two's Complement	
Dec	**Bin**
+0	000
+1	001
+2	010
+3	011
-1	111
-2	110
-3	101
-4	100

Assume a 3-bit signed number

2's complement is best representation of (+) and (-) numbers

Signed Number Addition

Assume the following examples use signed 5-bit 2’s complement binary numbers to represent negative numbers

Example 1

Base 10	Base 2	
+11	01011 (+11)	0 1 0 1 1 (+11)
- 05	00101 (+5)	+1 1 0 1 1 (-5...using 2’s complement)
+06	(ignore carry)	~~1~~ 0 0 1 1 0

When the positive number magnitude is larger

MSB	9SB	8SB	7SB	6SB	5SB	4SB	3SB	2SB	LSB		Fractional Base 2	
2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	.	2^{-1}	2^{-2}
512	256	128	64	32	16	8	4	2	1		0.5	0.25

Signed Number Addition

Example 2

Base 10	Base 2		
03	00011 (+3)	0 0 0 1 1	(+3)
- 05	00101 (+5)	+1 1 0 1 1	(-5...using 2's complement)
- 02	(signed bit)	1 1 1 1 0	

When the negative number magnitude is larger

Example 3

Base 10	Base 2		
- 03	00011 (+3)	1 1 1 0 1	(-3...using 2's complement)
- 05	00101 (+5)	+1 1 0 1 1	(-5...using 2's complement)
- 08	(signed bit)	1 1 1 0 0 0	

When both numbers are negative

MSB	9SB	8SB	7SB	6SB	5SB	4SB	3SB	2SB	LSB		Fractional Base 2	
2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	.	2^{-1}	2^{-2}
512	256	128	64	32	16	8	4	2	1		0.5	0.25