

How to convert decimal numbers to 8 bit 2's complement

If you are not sure how to convert decimal numbers to 8 bit 2's complement form then have a look at this short video tutorial by PCC videos:

<http://www.youtube.com/watch?v=WrsMzaEkrO4>

Here's an example: Convert (– 47) into 8 bit 2's complement

Step 1) change the decimal number to binary

101111

Step 2) ensure the number is 8 bit by adding zero's to the left hand side

00101111

Step 3) invert the number by changing all the 1's to zero and all the zeros to 1

11010000

Step 4) add 1 to the right hand side

11010001

So the final answer is – 47 = 11010001

Now try these:-

- a) Convert the decimal number (– 5) into 8 bit 2's complement
- b) Change (– 21) into 8 bit 2's complement
- c) Express (– 67) as an 8 bit 2's complement number

<u>ANSWERS</u>			
a)	11111011		
b)	11101011		
c)	10111101		

Binary subtraction using 8 bit 2's complement

Computers do not manage direct subtraction very well. We can get round this problem by **adding negative** numbers when they are in 8 bit 2's complement form.

Here's an example: 150 - 47

Step 1) Convert the positive number into binary making sure it is 8 bits long, if it is too short just add zeros to the left hand side.

$$150 = 10010110$$

Step 2) Convert the negative number into 8 bits 2's complement form. DO NOT convert the positive number.

$$-47 = 11010001$$

Step 3) **Add** the numbers together using binary addition.

$$\begin{array}{r} 10010110 \\ + 11010001 \\ \hline 101100111 \end{array}$$

Step 4) Ignore any overflow: If you have a 9 bit number as your final answer, ignore the biggest place value on the left hand side so that your answer is 8 bits long.

$$\cancel{1}01100111 \quad \text{So the final answer is} \quad 01100111$$

Now try these:-

1) 15 - 5

2) 77 - 21

3) 100 - 67

<u>ANSWERS</u>			
1)	00001010		
2)	00111000		
3)	00100001		