Convert the following number in base10 to its negative binary form using:

4 bits signed numbers (each number should only have 4 binary bits)
Ex: Given 7 to -7
7 is 0111 -> 1000 + 0001 = 1001

1. 1 to -1 : 0001 -> 1110+0001=1111
2. 2 to -2 : 0010 -> 1101+0001 = 1110
3. 3 to -3 : 0011 -> 1100+0001 = 1101

8 bits signed numbers (each number should only have 8 binary bits)
Ex: Given 99 to -99
99 is 0110_0011 -> 1001_1100 + 0000_0001 = 1001_1101 (underscore added for readability)

1. 127 to -127 :

// 1. 127 to -127 :

// 127%2 = 1 | 127-1/2 = 63

// 63%2 =1 | 31

//31%2 =1 |15

//15%2 = 1 | 15-1/2 =7

// 7%2 =1 | 3

//3%2 =1 |1

//1 : so 127 binary is 111 1111 so add 0 to make 8bit : 0111 1111

-127 : 0111 1111 => flip => 1000 0000 + 0000 0001 =1000 0001

// 2. 128 to -128

//128%2 =0=64

//64%2 =0 =32

//32%2=0 = 16

//16%2=0=8

//8%2=0=4

//4%2=0=2

//2%2=0=1

//1 : so 128 binary is 1000 0000

//-128 : 1000 0000 => flip => 0111 1111+0000 0001 => 1000 0000

// 3. 52 to -52

// 52%2=0 =26

//26%2 =0=13

//13%2 =1=6

//6%2=0=3

//3%2=1=1

//1: so binary for 52 is 110100:adding 00 for 8 bit: 0011 0100

//-52 : 0011 0100 =>flip=> 1100 1011+0000 0001 => 1100 1100

// 4. 36 to -36

// 36%2=0=18

//18%2=0=9

//9%2=1=4

//4%2=0=2

//2%2=0=1

//1: so binary for 36 is 100100 adding 00 to make 8 bit: 00100100

//-36 : 00100100: 1101 1011+0000 0001 = 1101 1100

// 5. 85 to -85

// 85%2 =1=42

//42%2=0=21

//21%2=1=10

//10%2=0=5

//5%2=1=2

//2%2=0=1

//1: so binary for 85 1010101

//-85 : 01010101 =>flip=> 1010 1010 +0000 0001 => 1010 1011

6. 115 to -115:
//115%2 =1 |115-1/2 = 57
//57%2 = 1 |577-1/2 =28
//28%2 = 0 | 28/2 =14
//14%2= 0 | (14-0)/2 =7
// 7%2 =1 | 7-1/2 =3
//3%2 =1 | 3-1/2 =1
// 1=>1110011
To negative -115 we add a leading 0 to make 8 bit : 0111 0011
: 01110011 => 10001100+00000001 => 1000 1101