Lecture Four: Complements

**Complements :**

 Complements are used in digital computer for simplifying the subtraction operation and for logical manipulation . There are two types of complement for each base (R) system :

1-The R´s complement

2-the (R-1)´s complement

For binary number 1´s and 2´s complement

For decimal number 9´s and 10´s complement

For octal number 7's and 8's complement

For hexadecimal number 15's and 16's complement

**The 1´s and 2´s complement :**

 The 1´s complement of a binary number is the no. we get when we change each (0) to (1) and each (1) to (0) (or subtracting each binary no. from 1 )

***EX***: 1´s comp. of 1001 0110

 1´s comp. of 110010 001101

 2´s comp. = 1´s comp. + 1

 2´s comp. of 1011 is 0100 + 1 = 0101

 2´s comp. of 1110 is 0001 + 1= 0010

**Using 2´s complement in subtraction :**

 Instead of subtraction a number , we can add it’s$ $2's comp, and disregard the last carry.

***EX***: decimal

 7 111 111

 -5 -101 1´s 010 2´s 011

 2 1+ 1 010 + ve. No.

 X carry 011

***EX***: 13 1101 1101

 -10 1010 1´s 0101 2´s 0110

 3 1+ 1 0011 +ve. No.

 0110 X carry

***EX***: 4 100 100

 -7 -111 1´s 000 2´s 001+

 -3 1+ 101

 001 No carry -ve. No.

 So 101 010 011

**Using 1´s complement in subtraction :**

 Instead of subtracting a number we add the 1´s complement of the number , the last carry is then added to the number to get the final answer .

***EX***: 7 111 111

010 +

 -5 - 101 1´s

 2 carry 1 001

 + ve. No. 1

010

**EX:** 3 011 011

 -5 101 1’ s 010 +

 101

 No carry - ve. No.

 101 010

**Signed numbers:**

There are three ways in which signed binary numbers may be expressed:

* + Signed magnitude (SM)
	+ One’s complement and
	+ Two’s complement.

In an 8-bit word, signed magnitude representation places the absolute value of the number in the 7 bits to the right of the sign bit.

Ex: in **8-bit signed magnitude(SM),**  positive 3 is: 00000011

 Negative 3 is: 10000011

Ex: in 8-bit **one’s complement**, positive 3 is: 00000011

 Negative 3 is: 11111100

Ex: Adding 1 gives us -3 in **two’s complement** form: 11111101

Ex:convert using SM method**(01011001)2** = +(1 \* 26 + 0 \* 25 + 1 \* 24 + 1 \* 23 + 0 \* 22 + 0 \* 21 + 1 \* 20)

 = + (64 + 0 + 16 + 8 + 0 + 0 + 1)

 = **(+89)10**

Ex: convert using SM method **(10011100)2** = - (0 \* 26 + 0 \* 25 + 1 \* 24 + 1 \* 23 + 1 \* 22 + 0 \* 21 + 0 \* 20)

 = - (0 + 0 + 16 + 8 + 4 + 0 + 0)

 = **(-28)10**

**7´s and 8´s complements in octal :**

 7´s = 7 – each digit

 8´s = 7´s + 1

***EX:*** 7777

 - 2415

 is 7´s comp. 5362

 +1

 5363 is 8´s comp.

**15´s and 16´s complements in hexadecimal :**

 ***EX***: Find 15´s and 16´s comp. of ( 1 F A D )16

 15 15 15 15

 - 1 F A D

 E 0 5 2 15´s comp.

 1 +

 E 0 5 3 16´s comp.