Course Name: A Level (1st Sem) **Subject: CO**

Topic: Number Representation Contd. **Date: 22-04-20**

Negative Numbers

Signed 1's Complement Method: In this representation, again we use a significant size of register. Again the MSB bit position in the register works as SIGN bit. However the number is not represented normally. Instead, it's represented in its 1's complement format.

Thus all that we have to do is, compute the 1's complement of that number in n-digits in order to fit it into an n-bit register and place every bit to its respective bit position in the register from right to left. Since the MSB bit is always "0" for a positive number, it'll be converted to "1" in the 1's complement of thereby making that number a NEGATIVE number.

For Ex.: We have to represent $(-56)_{10}$ in an 8-bit register. Following is the diagram that depicts this representation:

$(+56)_{10}$ = $(00111000)_2$

1's Compliment of $(+56)_{10}$ = $(11000111)_2$

R_7	R_6	R_5	R_4	R_3	R_2	R_1	R_0
1	1	0	0	0	1	1	1

Here in this diagram, see that we have-

- Converted 56 in to binary.
- Appended two 0s to the left making it an 8-bit number.
- Calculated its 1's complement. See that the two 0s appended to the left have been transformed into 1 making it a negative number.
- Stored every bit in the 8-bit register.

We took 8-bit register because original (56) was a 6-bit data in binary. We appended two 0s in the left to make it 8-bit positive number. Then we calculated 1's for getting (-56)

Signed 2's Complement Method: In this representation, again we use a significant size of register. Again the MSB bit position in the register works as SIGN bit. However the number is neither represented normally nor in 1's complement. Instead, it's represented in its 2's complement format.

Thus all that we have to do is, compute the 2's complement of that number in n-digits in order to fit it into an n-bit register and place every bit to its respective bit position in the

 © Shashi Kant Mani Tripathi

register from right to left. Since the MSB bit is always “0” for a positive number, it’ll be converted to “1” in the 2’s complement of thereby making that number a NEGATIVE number.

For Ex.: We have to represent $(-56)_{10}$ in an 8-bit register. Following is the diagram that depicts this representation:

$(+56)_{10}$ = $(00111000)_2$
1’s Compliment of $(+56)_{10}$ = $(11000111)_2$
2’s Compliment of $(+56)_{10}$ = $(11001000)_2$

R_7	R_6	R_5	R_4	R_3	R_2	R_1	R_0
1	**1**	**0**	**0**	**1**	**0**	**0**	**0**

Again here, see that we have-

- Converted 56 in to binary.
- Appended two 0s to the left making it an 8-bit number.
- Calculated its 1’s complement. See that the two 0s appended to the left have been transformed into 1 making it a negative number.
- Then added 1 to make it 2’s complement.
- Stored every bit of this 2’s complement in the 8-bit register.

NOTE: Though, all three- Signed Magnitude, Signed 1’s and Signed 2’s are methods to represent –ve numbers but electronic industries follow only Signed 2’s representation. From hereafter, we’ll use only Signed 2’s method for representation of –ve numbers.

Assignments:

1. Represent the following in Signed 1’s and Signed 2’s Representation:

  a. $(-307)_{10}$
  b. $(-268)_{10}$