CA Innovative Assignment 2

Topic: Implement the circuit of basic calculator for signed magnitude.

Submitted By: -

18BCE011(Akshit Patel)

18BCE008(Krunal Agrawal)

18BCE040(Karan Chauhan)

Guided by: -

Prof. Pimal Khanpara

Assistant Professor at Institute of Technology, Nirma University.

Input Part: - For an input there are two numbers A and B which is of 4bit (magnitude part) + 1bit (sign bit) = 5 bit. From left the first bit is sign bit. For example, you want to give -6 and 3 then you have to give an input like 10110 (-6) and 00011 (3).

Output Part: - in output you will get 4 answers which will be of Addition, Subtraction, Multiplication and Division.

1). Addition Part: -

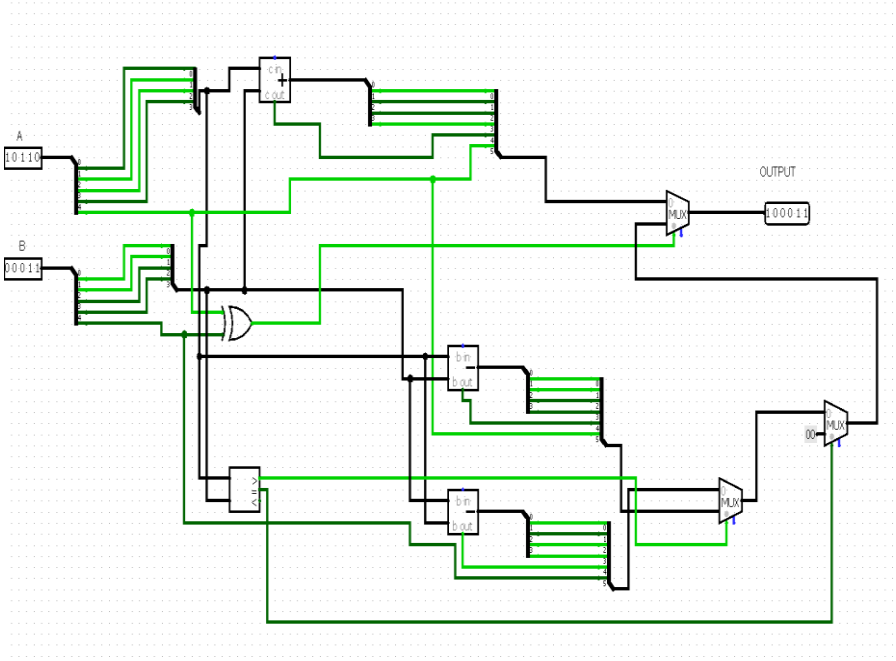
There are mainly three parts on the basis of the signs of A and B.

- A and B both have same signs: - If the signs are same then simply it adds two numbers and give the sign of any number to the output. The output is of 7 bits total, 6 bits of magnitude and 1 bit of sign.

- A and B both have different signs: - In this case first it will compute A – B and B – A and then compare the numbers A and B magnitude wise and on the basis of it like if A > B then in output multiplexer will select answer of magnitude from A – B (6 bits) and will give the sign of A (1 bit) and the opposite will happen when A<B. Here the answer 0f -6+3 will be 100011(-3)

- A and B both are zero: - In this case it will compare the numbers if the both numbers are same then mux will select the 00000 from the input of the multiplexer from both number (1 is output from 2nd part and other 000000).

- So here is the picture of circuit.

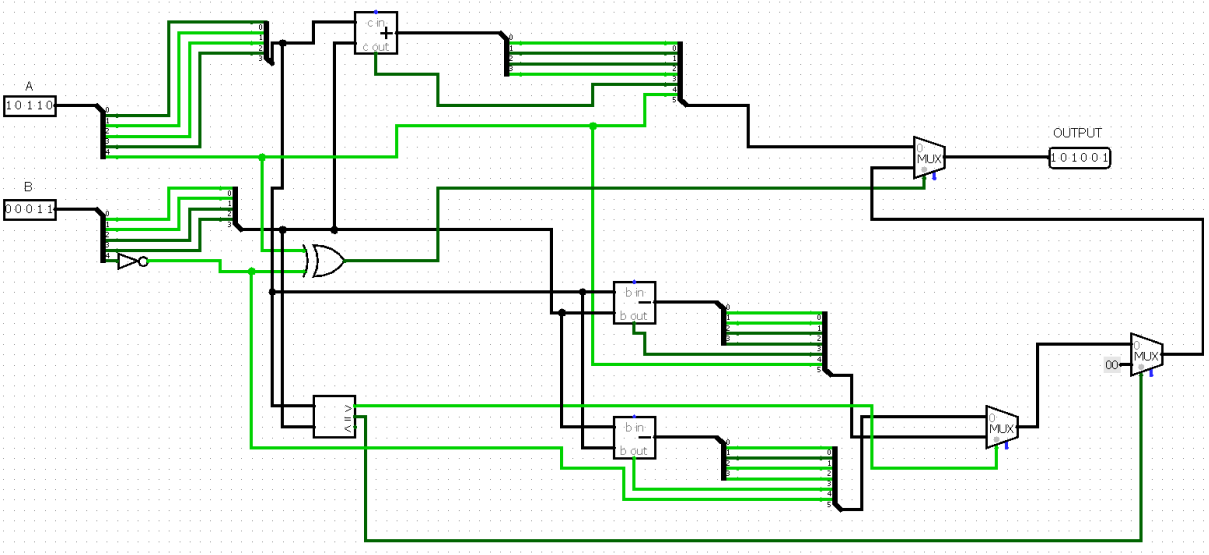


2). Subtraction Part: -

It is same as the Addition part except one thing. The one thing is that you have to change the sign of the second number B. And then you will get the answer of subtraction. Yeeeee!!!

Here answer will be -6-3 = 101001

Here is the circuit picture.

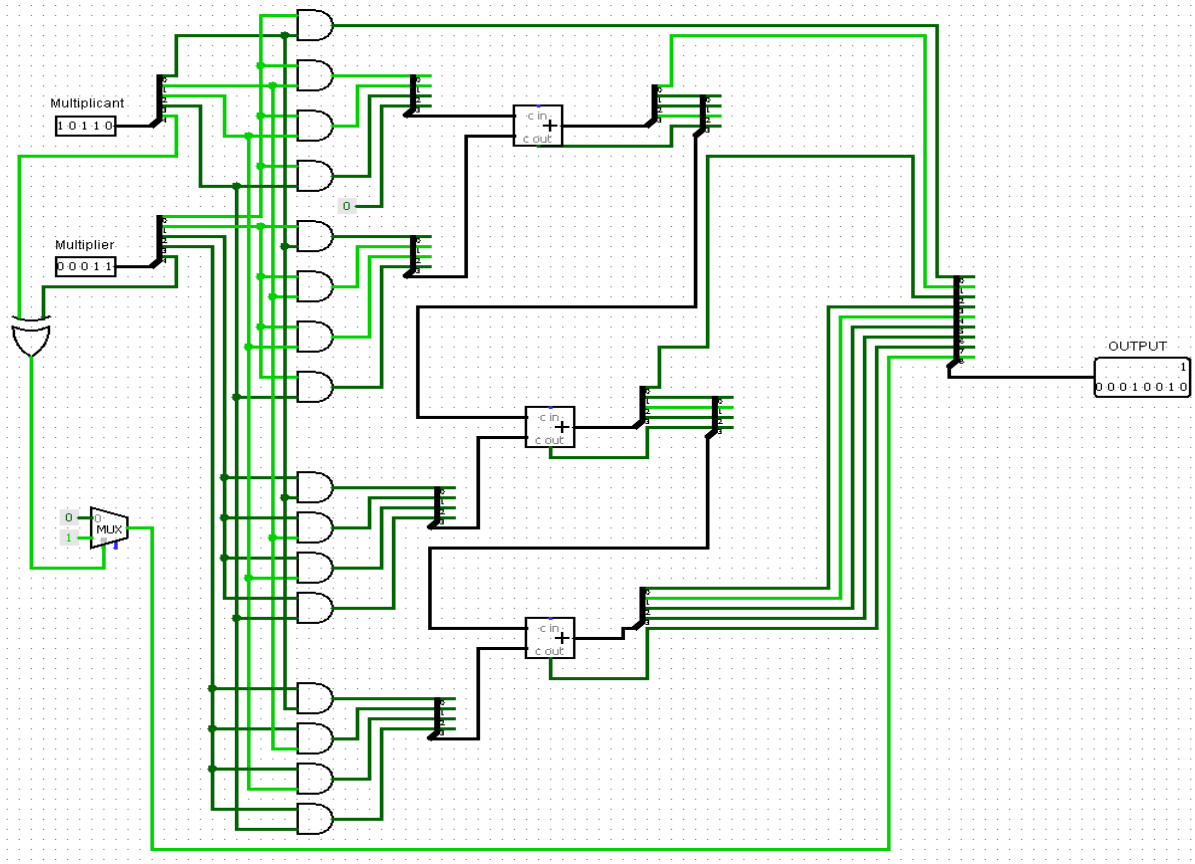


3). Multiplication Part: -

In the multiplication I have just implemented normal binary multiplication method. First, I have done the ANDing of the all bit of A (multiplicand) with the 0^{th}, 1^{st}, 2^{nd} and 3^{rd} bit of multiplier (B) means we have just multiplied each bit of multiplier with the whole multiplicand. Now the 0^{th} bit of answer will be 0^{th} bit of multiplier * 0^{th} bit of multiplicand. Now do the adding of 3 bit of remaining AND answer of 0th bit of multiplier and 0 with the 4 bit of next ANDing answer. Now 0th bit will be the 1^{st} bit of answer and the 4 bit will be the carry out of the sum and the remaining bit of answers. Now do the same for 2 times. Now for the sign of the answer do XOR of the sign bit and if both are same then the sign will be 0 (plus) otherwise 1 (negative). Now using splitter do join the answer.

Here the answer of -6*3 will be 100010010 (-18).

Here is the picture of the circuit.

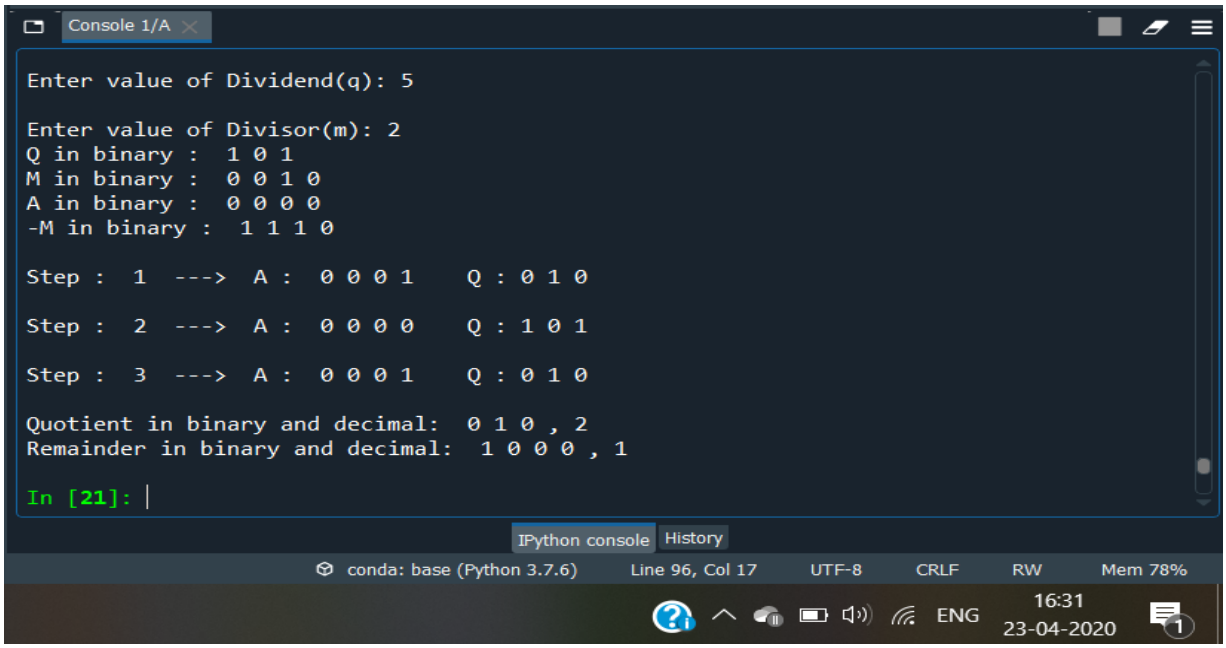


4). Division Part (we have implemented python program for restoring division algorithm): -

We have implemented python program for restoring division algorithm. In this we take an input of dividend and divisor from the user. And then in the output we have showed the steps of restoring algorithm, final answer of Quotient and Remainer.

Steps: -

1. Compute and assign the value to the Q(Dividend), M(Divisor), A=0, n=number of bits in the dividend.
2. Now call the left shift function to A and Q as a single unit.
3. Now do A=A-M (binary subtraction)
4. Then check the most significant bit of the A if 1 then put Q[0]=0 And do A=A+M and if 0 then Q[0]=1.
5. Now do the n - -;
6. If n==0 then print Q and A as Quotient and Remainder else repeat step 2 to 5 till you get n==0.



We have submitted the three self-implemented circuits (For 4 bit) of Addition, Subtraction, Multiplication and also, we have submitted the python program which simulates the restoring algorithm.

Thank you