Signed Calculator

Calculator for performing Addition, Subtraction, Multiplication, and Division

Joseph Hanna, Christopher Czarnecki, Alex Cross, Irvin Watson
Electrical and Computer Engineering Department
School of Engineering and Computer Science
Oakland University, Rochester, MI
e-mails: josephhanna@oakland.edu, cjczarnecki@oakland.edu, across@oakland.edu, irvinwatson@oakland.edu

Abstract— For this project, our group created a calculator by utilizing our joint and individual VHDL knowledge accumulated throughout the Digital Logic Design course. Our project uses the Nexys A7 to act as an interface between the VHDL code and circuit development, which will perform the arithmetic functions of addition, subtraction, multiplication, and division. The program uses selection switches to choose the operation desired and will translate the binary code from the switches on the board to hexadecimal and then use a hex to seven segment display then display the decimal on the LCD display screen on the board.

I. INTRODUCTION

During the process of completing this project the team will be constructing a schematic for a calculator via the Vivado software with the intent of having the calculator perform signed addition, subtraction, multiplication, and division operations. In order to configure and execute the project, a multitude of troubleshooting, debugging, redesign and implementation will be used to achieve the final result. The purpose of the project is to produce a fully functional signed calculator for a user to use in the same fashion as a retail device. From this project the team will be able to expand their knowledge of the interaction between seven segment displays and basis board switches. More specifically, the team will be furthering the use of the decoder, mux, full adder as well as an addition/subtraction adder from previous labs and references provided. The team has also compiled their knowledge to make the calculator show the correct signage of the positive or negative outcome. Communication and collaboration will play a vital role in the development of the project. As a team, we will develop a digital logic system that can be used by the average consumer to be of use in average everyday tasks involving mathematical calculations.

II. METHODOLOGY

The architecture of the signed calculator shown below.

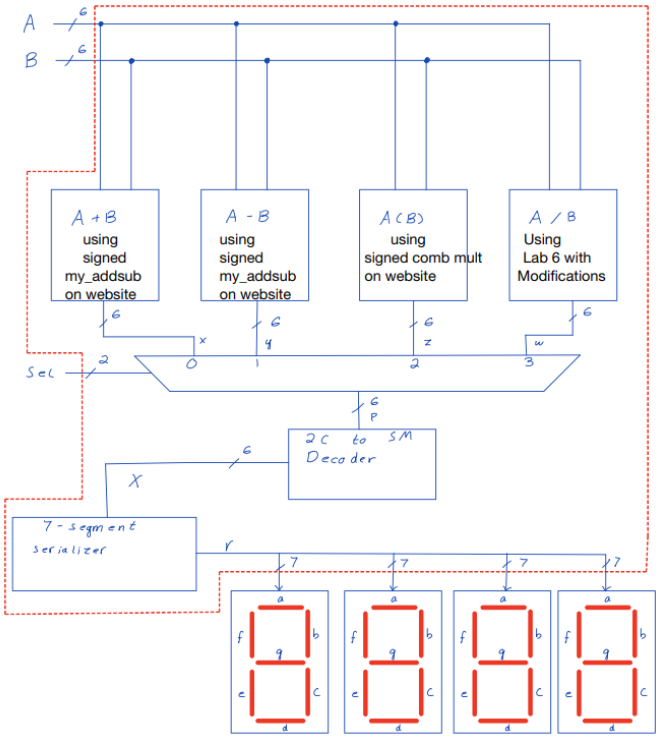


Figure 1: Full Circuit Schematic of the Signed Calculator

A. 2's Complement Adder/Subtractor - Chris Czarnecki/ Joseph Hanna

We used lab four as a foundation for the circuit design of the calculator. We used Dr. Llamocca's "Generic 2's complement Adder/Subtractor Unit" for the signed addition and subtraction component for the project. The unit is assembled by inputting two six bit values as signed 2's complement numbers and outputs a signed hexadecimal number. The component uses a full adder to compute addition. The component xor's the sign for the input and uses that to determine if it is positive or negative.

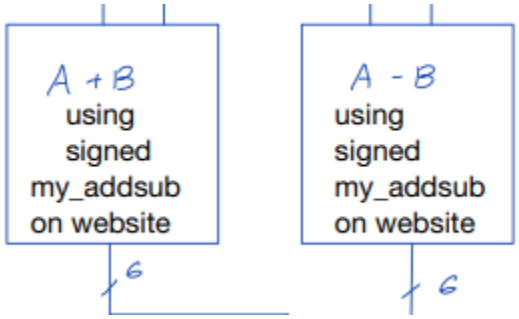


Figure 2: Adder and Subtractor Schematic

B. MUX/Seven Segment Decoder - Christopher Czarnecki

The project used Dr. Lllamoca's 1-to-8 demux from his website and modified it into a 4-to-1 mux. The mux remained the same as it was in Lab Four, using the same select options (switches) for each operation. The mux was used to select the desired operation for the calculator. The mux has 4 possible operations to be used via switches 13 and 14. Addition was represented as 00, subtraction as 01, multiplication as 10, and division as 11. For the 7 segment serializer, the project used Dr. Llamoccas "7-segment serializer (four displays)" and modified it to suit the project's needs. The 7 segment serializer was used to display the signed number on the LCD screen. Only the right 4 displays were utilized. The far left most display was used to display the correct sign of the final data result. The remaining displays were used to display the data inputs for the mathematical operation selected. The output of the calculator was in signed hexadecimal.

Figure 3: MUX Schematic

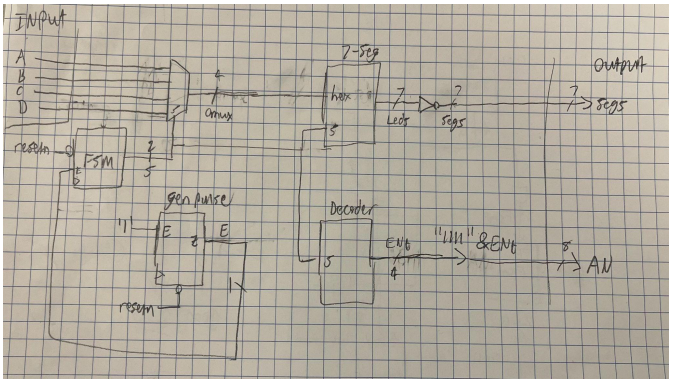


Figure 4: Seven Segment Serializer Schematic

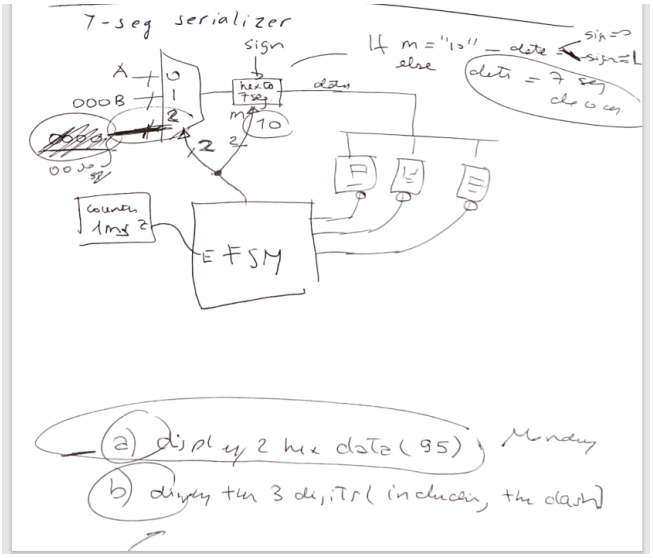


Figure 5: Detailed Description of 7 Segment Decoder

C. Multiplier -ChrisCzarnecki/Alex Cross

The project team used Dr. Llamoca's "N-bit signed/unsigned combinational multiplier" and made the modification to make the component a six by six bit multiplier to suit the project specifications. The calculator used a signed multiplier that computed the proper value. The multiplier would output 12 bits but the LCD screen is limited on the amount of bits it can display. Therefore the output of the component was altered to display the six least significant bits. For example, if the multiplication adder was computing -10*5, the answer would be -50. In 2's complement this was shown as 100110 (seven bits). The calculator would thus output 001110 (six least significant bits) which is 14 in binary, so in order to ensure the correct answer, the program must convert the unsigned number to 2's complement and take the 6 least significant bits. The LCD screen would display "E" which is 14 in hexadecimal.

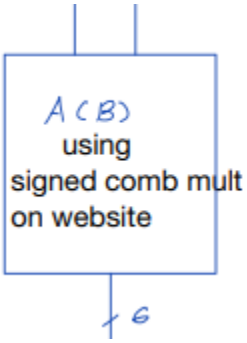


Figure 6: Multiplier Schematic

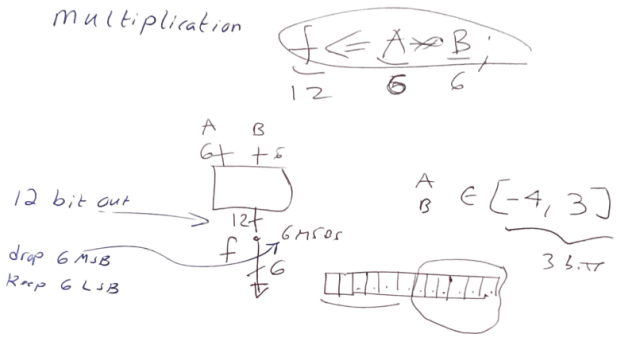


Figure 7: Detailed Description of Multiplier

D. Divider -Chris Czarnecki/ Irvin Watson

The divider used professor Llamoccas Generic Divider VHDL IPs in his digital library. This component is similar to Lab Six, but has two N-bit inputs that were modified to 6-bit inputs and a 6-bit output. The absolute value of the inputs are taken, the first bit of each input is xor'd to determine the sign of the result. The mux is then used to put the xor'd bit (1 or 0) in front of the resultant to achieve the signed answer.

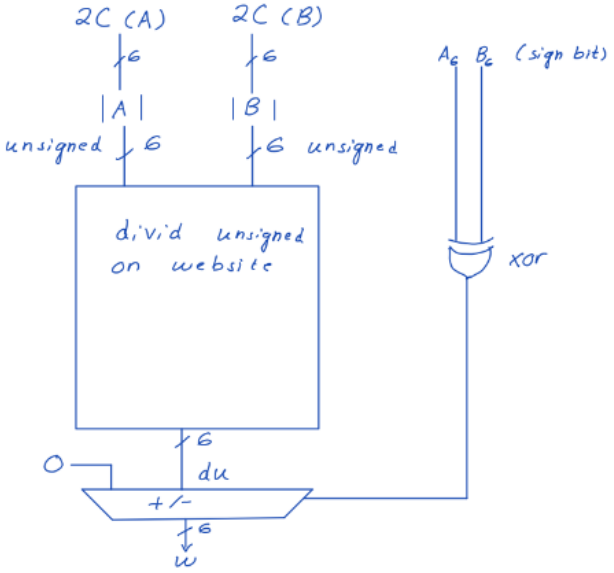


Figure 8: Divider Schematic

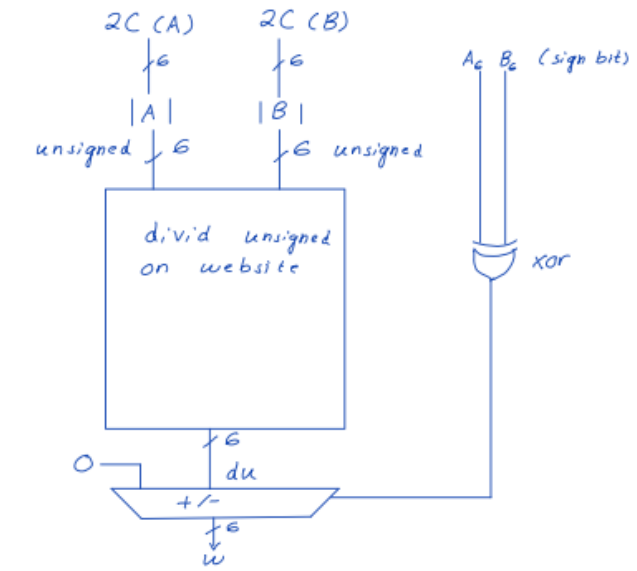


Figure 9: Additional Design of Divider

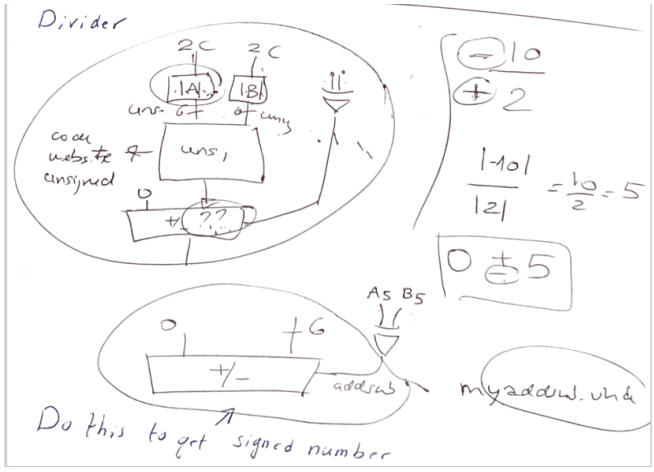


Figure 10: Detailed Description of Divider

E. 2C to SM Decoder- Chris Czarnecki

This component takes the output data from the mux and converts it from 2C to the correct signed magnitude form. The component took the 2C six bit input and converted the output to signed sign and magnitude. This was done so by using the component from lab two in figure 11, all six bits of the input were xor'd with the leading bit of the input (p(5)) and a then a mux added "000000" to the output to make sure the converted number is six bits if it was not already. The seven segment display uses a hex to seven segment decoder, the number was converted from 2C to sign and magnitude since you can easily convert from sign and magnitude to hexadecimal.

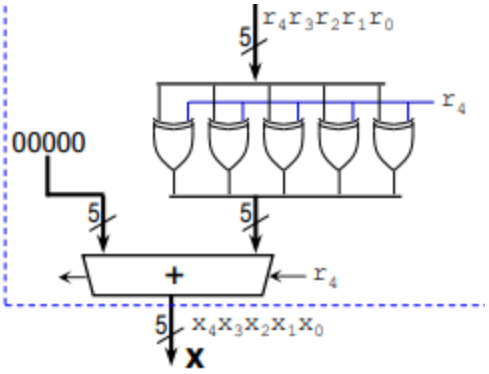


Figure 11: 2C to SM Decoder Schematic

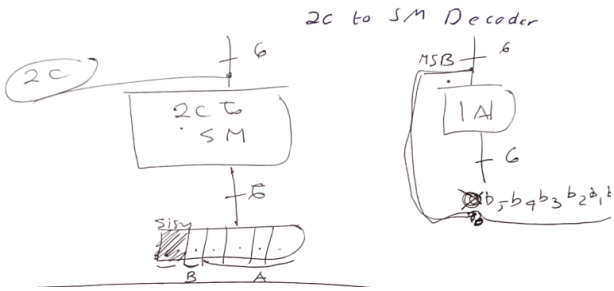


Figure 12: Detailed Description of 2C to SM Decoder

III. Experimental Setup

The leading tool used throughout the development of the project was the simulation tool provided by the Vivado software. This tool's functionality would allow for the user to run a mock simulation of the system in order to show the values of what would otherwise be displayed on the basis board. This would then allow for the team to be able to see if any values being used are receiving the correct data in order to ensure the proper functionality of the system. Each time the project would undergo a modification, a new simulation would be run in order to keep track of the progress of the project to properly track the development. For instance a consistent error the team encountered was the addition and subtraction adder were not receiving the correct values. In turn the values in the simulation were undefined and did not represent the correct information being written in the software. This error led to hours of modifications and investigations into correcting the system to properly receive the correct values for the two adders for the simulation to no longer show the undefined values. In addition, with the designing of the divider, the team found that it was not able to accept signed input values. Due to time restraints the team decided to modify the divider to accept unsigned input values in order to have the component operational.

IV. Results

According to the figures all results are shown in the image of the simulation. Each figure shows the values input as well as the correct manual output for the result. The figures show the correct simulated output matched alongside the manual calculation to show the system is properly calculating the data outputs. Each figure shows the simulation for each individual mathematical operation. As a result of the completion of the project, the team found multiple implications from the lab previously completed that assisted in the completion of the project. For instance, the 7 segment decoder became a vital component in the system to show the values being input as well as the correct output data values. In addition, the binary mathematical operations acquired assisted in being able to manually verify the correct data outputs from the input data in order to verify the simulation simulated the operations correctly. The division operation only does unsigned division which could be improved and the seven segment display has the g segment flipped for positive and negative.

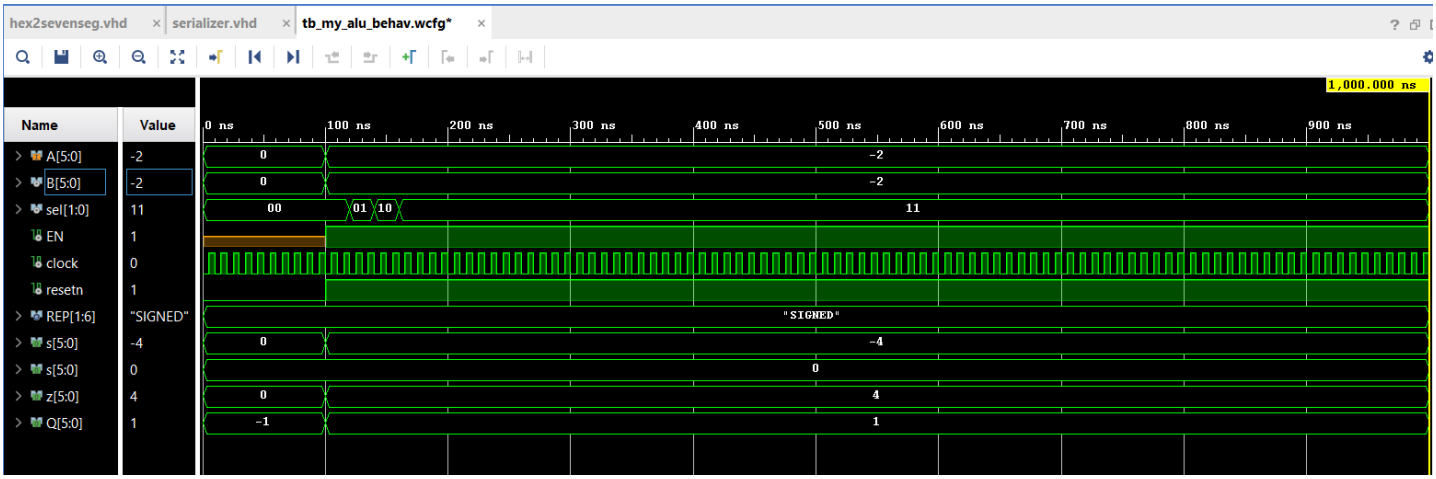
Conclusions

In conclusion, the project became a vital part of furthering future career skills. The project offered the opportunity to collaborate with others in a team environment that required communication through planning meeting times and roles being assigned. These are important skills that are required in all team based collaborations amongst all companies and this project allowed for more practice and experience to be acquired at no extra cost. In addition, the project allowed for the team to become more experienced with the Digital Logic software which can become a vital skill on a resume for companies that may require background knowledge on the use of the Vivado software. More specifically, the skills of problem solving, designing, and troubleshooting were used the majority of the time. This has led to further development of the skillset to help the team with their future endeavors in the field.

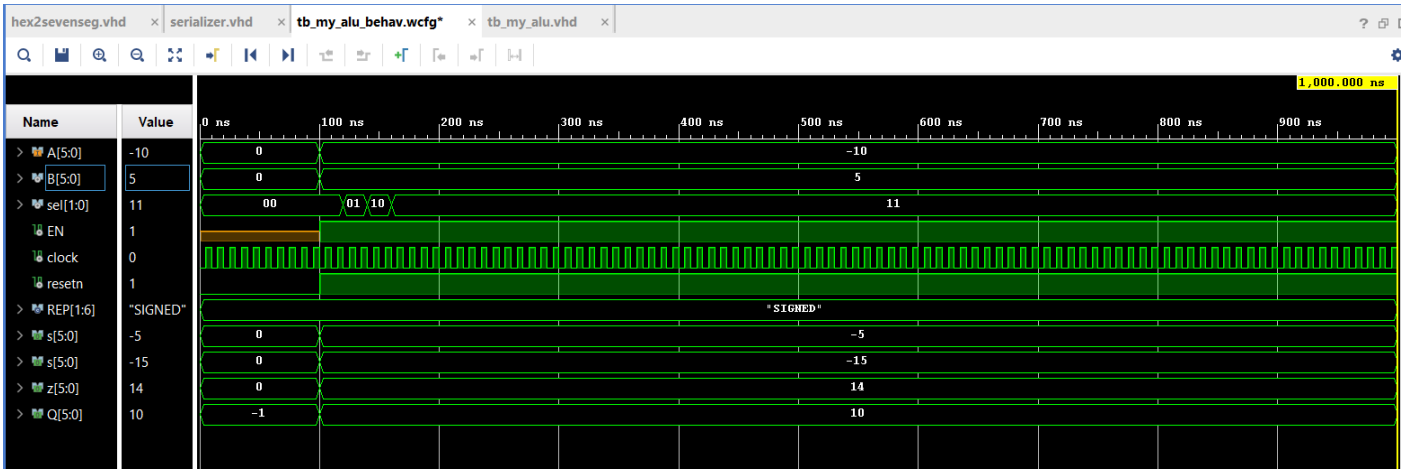
References

[1] Dr. Llamocca, VHDL Coding for FPGAs. [Online]. Available: http://www.secs.oakland.edu/~llamocca/VHDLforFPGAs.html

[2] Dr. Llamocca, Digital Library - Arithmetic Cores. [Online]. Available: http://www.secs.oakland.edu/~llamocca/arithcores.html

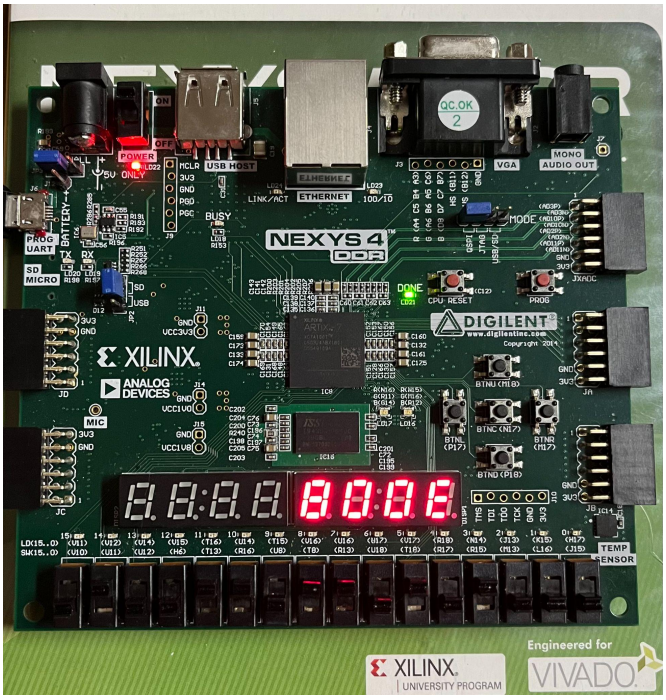


A = -2
B = -2
sel = operation selection
00 = add, 01 = subtract, 10 = multiply, 11 = divide
first s = add
second s = subtract
z = multiply
Q = divide

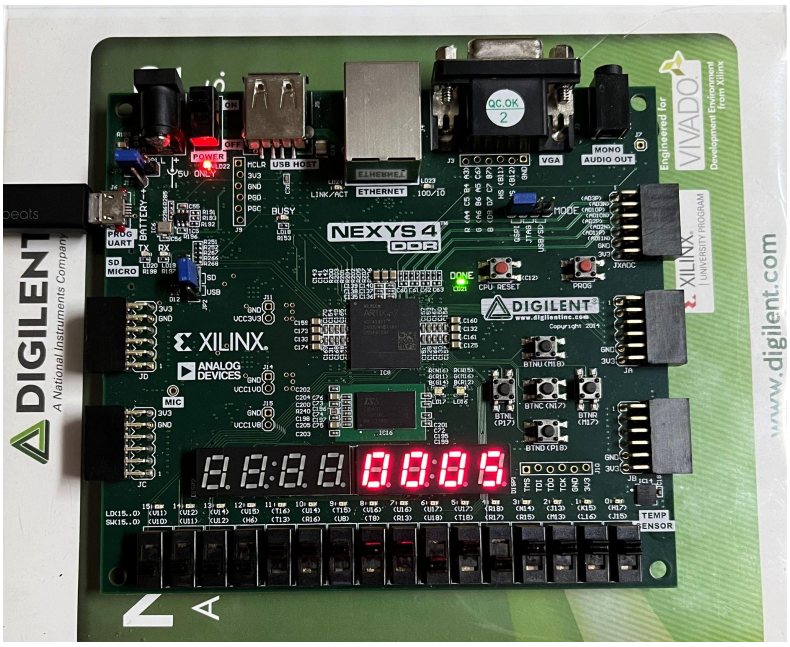


A = -10
B = 5
sel = operation selection
00 = add, 01 = subtract, 10 = multiply, 11 = divide
first s = add
second s = subtract
z = multiply
Q = divide

-10(5) = -50 which is 1001110 (7 bits) so the multiplication component will take the 6 least significant bits which is 001110 which in binary is 14
Division is unsigned, so -10 as an unsigned integer is 54 and 54 divided by 5 is 10 and the remainder is not shown



The following example A = -10 and B = 5
The calculator performs multiplication and -50 is 7 bits so the 6 least significant bits are 001110 which is 14
In hexadecimal 14 = E and the g segment of the left lcd is on which means it is positive.



The following example A = -2 and B = -3
The calculator performs addition and -2 + -3 is -5
-5 in hexadecimal is -5, the g segment on the left lcd screen is off meaning it is negative