Simple Calculator

Interactive Calculator using Nexys DDR board and Keyboard

Ardian Djekovic, Heather Lobbezoo, Cory Bledsoe, Alexandra Gaber
Electrical and Computer Engineering Department
School of Engineering and Computer Science
Oakland University, Rochester, MI
E-mails: adjekovic@oakland.edu, lobbezooh@gmail.com, cbledsoe@oakland.edu, agaber@oakland.edu

***Abstract*— Our final project was to create an interactive signed calculator that can add, subtract, multiply, and divide using a Nexys-DDR board with VHDL programming. We found that we learned most of this programming in our previous labs, it was up to us to use our previous labs and have them work simultaneously. Implementing a keyboard to the project had seemed easy but took a long time to program. Together with teamwork, we were able to create a functional signed calculator that performs calculations in hexadecimal form.**

I. Introduction

The scope of the project is to design a basic calculator that will add, subtract, multiply, and divide. In this report, you will learn the process of creating a simple calculator using a Nexys DDR board along with VHDL programming.

The motivation of the project is to be able to create such a program that is used by people in their everyday lives. We will be implementing what we learned from our previous labs such as the full adders, shift registers, state machines and multiplexers into this project. We also researched material outside of what was covered in our class, such as the external keyboard input. Our projects application is being used in everyone's daily lives, from simple calculations to infinite numbers.

II. Methodology

A. Layout

The layout of our projects controls can be seen in Figure 1. The keyboard was plugged into the USB Host port in the Nexys board. The he CPU reset button was used to clear the board from any calculation and start from scratch. The 7-segment display was utilized to display the output the value based on the users input/calculation. Switches 0 to 1 were used for the chosen method of calculation: addition, subtraction, multiplication, and division in a binary system. Switch 15 was used as and enable for the keyboard. Lastly, DataA was configured for switches 02-08 and DataB was configured for switches 09-14.

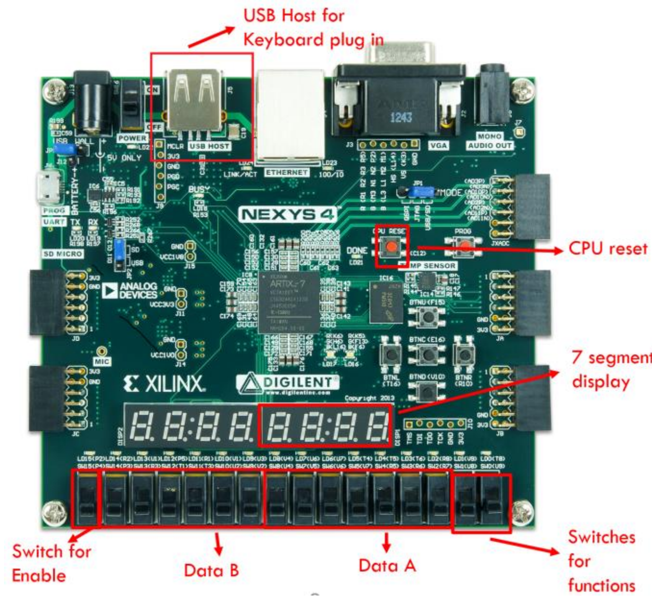

Figure 1: Board Layout

B. Inputs

The design for the input methods was a long procedure that involved a keyboard and board input. We are utilizing a keyboard as our method of input along with 5 switches on the Nexys board. Switches 0-1 were used for each function select (addition, subtraction, multiplication, and division) and switch 05 was set as an enable when plugging in the keyboard. Also, the CPU reset button was programmed to clear the calculation or any data on the board.

The first half of the calculation is set in Data A. An FSM is used to determine when Data A has been loaded, and enables the correct register. Once 8 bits are received, the first register is enabled and the 8 bits received are sent through the decoder, and output as a 4 bit signal which is then stored into the register that has been enabled. The next time a key is pressed, the data is loaded to the second register of Data A. This method is repeated for Data B. The signal for Data A is composed as 8 bits, with the MSB being the MSB from Reg A1, downto the LSB of Reg A2. There is a 2-digit maximum calculation that can be entered for Data A and for DataB. Every digit inputted from the keyboard is in hexadecimal.

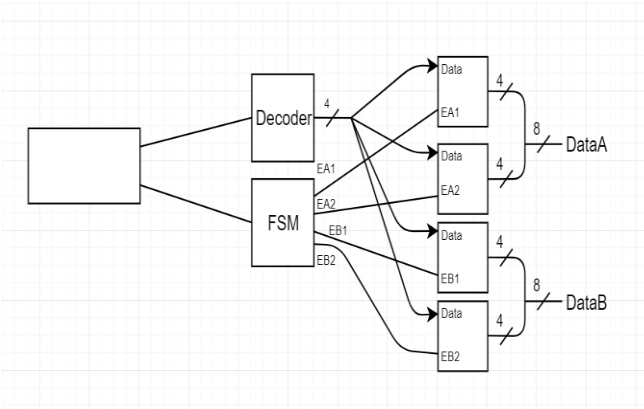

Figure 1: Keyboard Design

C. Functions

The task of the function inputs is utilized in the Nexys board. Using the four switches for each function for the preferred method of calculation. Function generators were used for the addition, subtraction, and multiplying. An FSM, registers, counter, and logic gates were used for the division portion. Most of these functions came from previous labs that were designed. They were each modified to work with one another in the top level file which can be seen in Figure 3 at the bottom of this report. Seven bits was input into Data A, and six bits into Data B, through the switches on the Nexy board. The four to one mux is connected to the two switches in order to control which function is selected. The answer of the selected function appears on the output, where the signal is split, into 2 four bit signals, which then become inputs into the next four to one mux. The other two inputs to the second mux are from the register called Neg and Rem. The register Neg stores the MSB from the subtractor, before the bits are flipped. If the value stored is a one, it means the answer is negative, from subtraction. The register Rem stored the remainder from the division function. Both of these registers are tied into the function selector, and are only enabled if the subtraction or division functions are selected. The selector for the second mux is tied into the output of an FSM. This cycles the data once every millisecond and enables the correct display. The data passes through a 7-seg decoder, before it is displayed. The second selector is also connected to the seven seg decoder, to control where the negative sign is displayed.

D. Output

One of the last steps of the project was to display the output of the operation. The results were outputted on the 7-segment display in hexadecimal. We used a serializer, described in the above section, to display the result of the calculation. The component displays an unsigned hexadecimal value of the result signal into the 7-Segment display. It also displays the remainder on the display when division is selected. One of the issues with the outputs was displaying the negative sign, without causing a negative answer.

III. Experimental Setup

The setup that was used was a Nexys4-DDR board along with VHDL coding. The main software tool used to do the coding was all written inside the Vivado program. Each component/module was designed and simulated separately from one another to ensure success between each component. It was possible to see if any mistakes were made in the code from the timing and behavioral diagrams. After completing any troubleshooting needed, we implemented the code in the overall design and then moved on to the next component. This allowed the project to be completed smoothly in increments and in a timely manner. As a team, it became more effective to work out problems and bugs with each component one at a time. After simulating all the components and verified everything worked in the calculator itself, it was time to insert the keyboard in the design. The keyboards main code was given by Dr. Llamocca, which in return saved the group a bit of time. Implementing the keyboard in the project was not as easy as planned, but after some troubleshooting, the team was able to get the job done. The projects hardware tools that were used was the Nexys board itself along with the keyboard. To test our cases, the switches on the Nexys were first used for the calculations just as in the past labs. This verified the code and components were correct and slowly paved the way to implement the keyboard as a number input.

The expected results were to have a functional calculator that operates by a keyboard and switches and displays the result on the 7-segment display of the Nexys board. It was decided early on to input numbers from the keyboard and the switches from the board to be used for the functions. Along with this, to display the result onto the 7-segment display with a signed integer and remainder.

IV. Results

After implementing all key components in the project and moving to the keyboard portion, the group ran into a problem. We were not able to implement the keyboard due to constant issues. After naming many of our modules the same name, we kept getting errors after errors. This in the long run shorted our time left to work on the keyboard. An interactive calculator was constructed with working switches used as functions and also for the input instead of the keyboard. Our components of the calculator portion functioned seamlessly as seen on the simulations attached on the bottom of this report. The desired result was displayed on the 7-segment display and the goal was completed. The procedure of inputting two numbers and a method of mathematical operation to receive a correct result was achieved.

In the figure below, a subtraction calculation was made. With 01 inputted into the function, subtraction was chosen as the preferred method of calculation. Switch 03 was flipped on for 2 to be inputted as DataA. Switch 11 was flipped on for 4 to be inputted as DataB. After calculation 2 (DataA) minus 4 (DataB) was made, negative two was outputted onto the display.

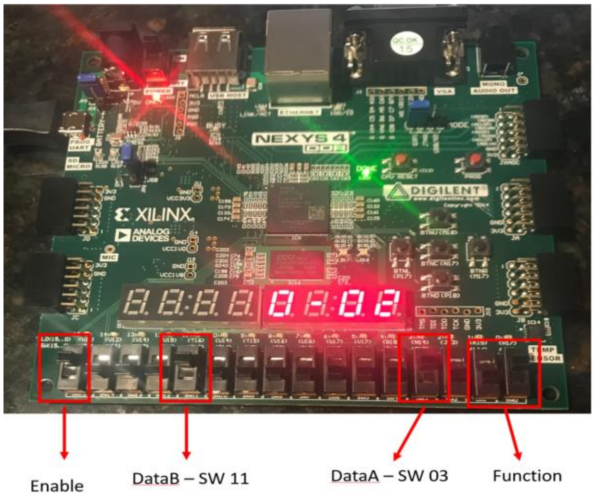

Figure 3: Subtraction Calculation

In the next example (Figure 4), a division calculation is made. With 11 inputted into the function (both switches flipped on), division was chosen as the preferred method of calculation. Switch 06 was flipped on for 16 to be inputted as DataA. Switch 09 and 10 was flipped on for 3 to be inputted as DataB. After calculation 16 (DataA) divided by 4 (DataB) was made, 5 with a remainder of 1 was outputted onto the display.

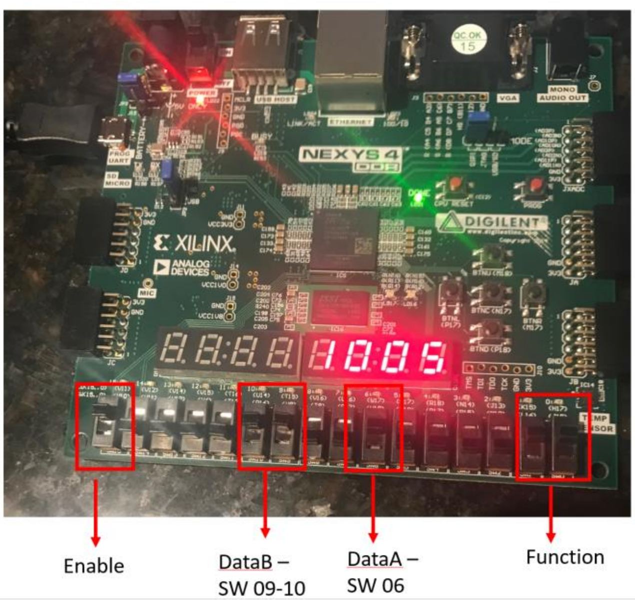

Figure 4: Division Calculation

Our project had a mix of topics taught in our class along with topics learned outside of class which created the perfect balance of combining the old with the new. The components used in the past labs that were incorporated in this project yielded just the results expected. It was a bit of a task to have them all seamlessly work together, but in the end it all functioned but just without the keyboard. One unexplainable result was constant troubleshooting of the keyboard. Error after error occurred when implementing the keyboard. If there was more time, it would have been possible.

Conclusions

The constant challenges of the project were a great learning experience along with including past labs and researching additional material that was not covered in class. Integrating multiple components such as adders, state machines, and registers together in one big design became a challenge which was overcome. Areas not covered in class such as the implementation of the keyboard to the Nexys board was further research done by the group to meet the needs of the project.

Some improvements that could've been made were larger number calculations. As of right now there are only up to 2-digit calculations that can be made on either DataA or DataB. Having larger inputs would create more of an ideal calculator that are used on a daily basis. Another improvement that could have been made is a user interface on a display. Utilizing the VGA output on the Nexys board would allow a virtual calculator to be made on a display. This in return would act like any other calculator program on a computer and become more accessible. Lastly, having a keyboard input would be great. After constant trouble shooting and countless hours, a calculator was created.

References

[1] Llamocca, Daniel. "VHDL Coding for FPGAs." *VHDL Coding for FPGAs*. Web 14 Apr. 2018.

[2] N/A."Nexys 4 DDR Keyboard Demo." *Nexys 4 DDR Keyboard Demo [Reference.Digilentinc]*, N/A. Web. 11 Apr. 2018.

Graphs and Simulations

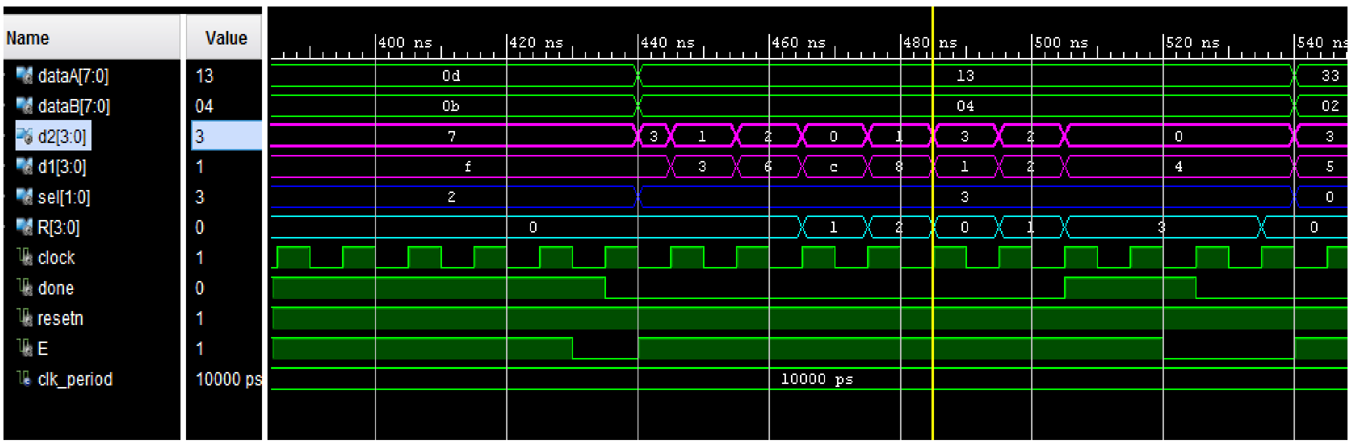

Figure 5: Simulation of the first half of the calculator

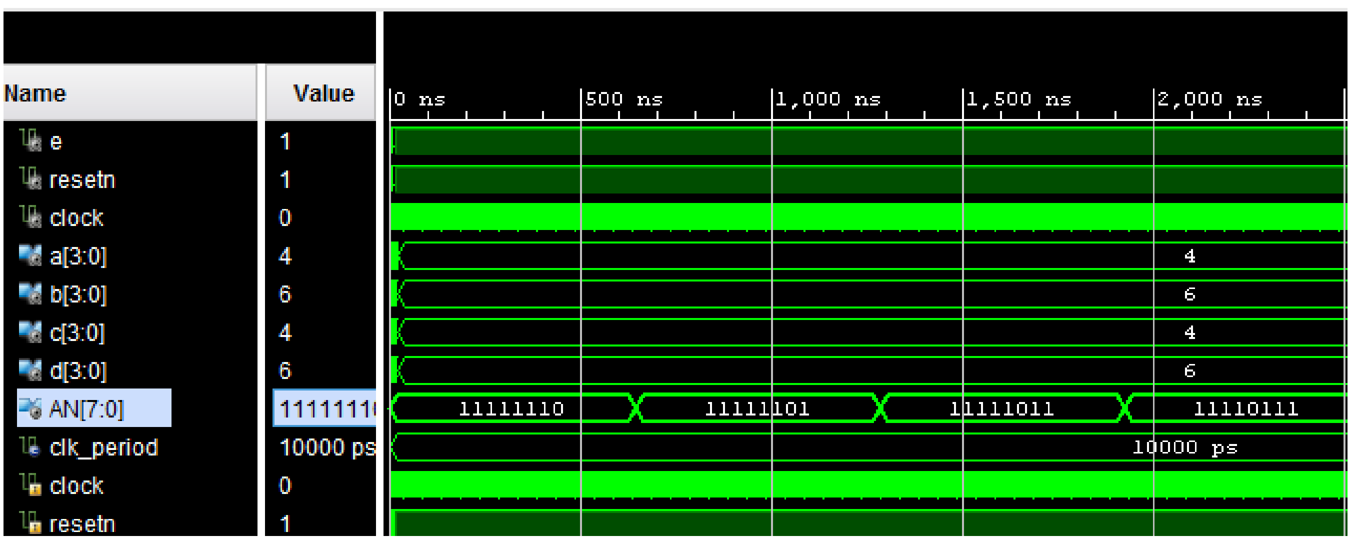

Figure 6: Simulation of the 7-segment display

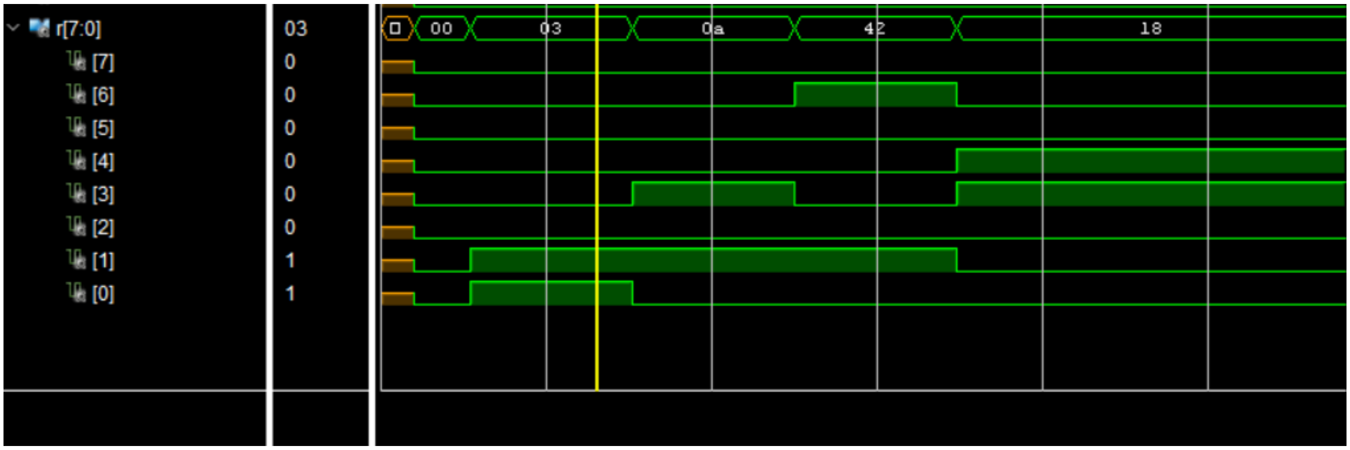

Figure 7: Addition Simulation

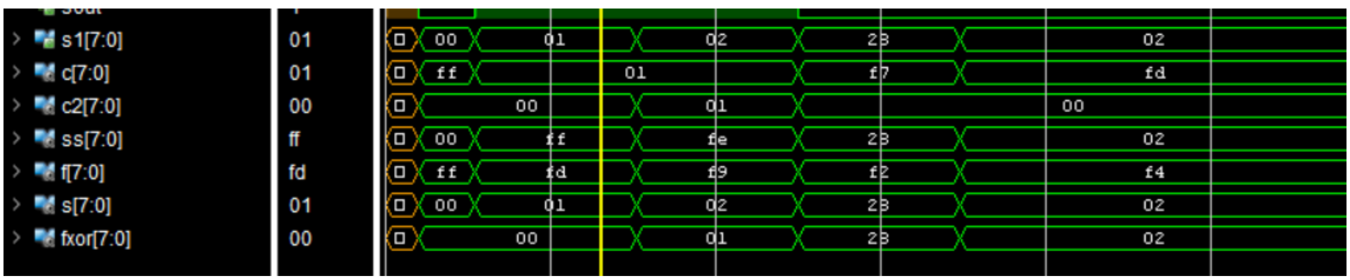

Figure 8: Subtraction Simulation

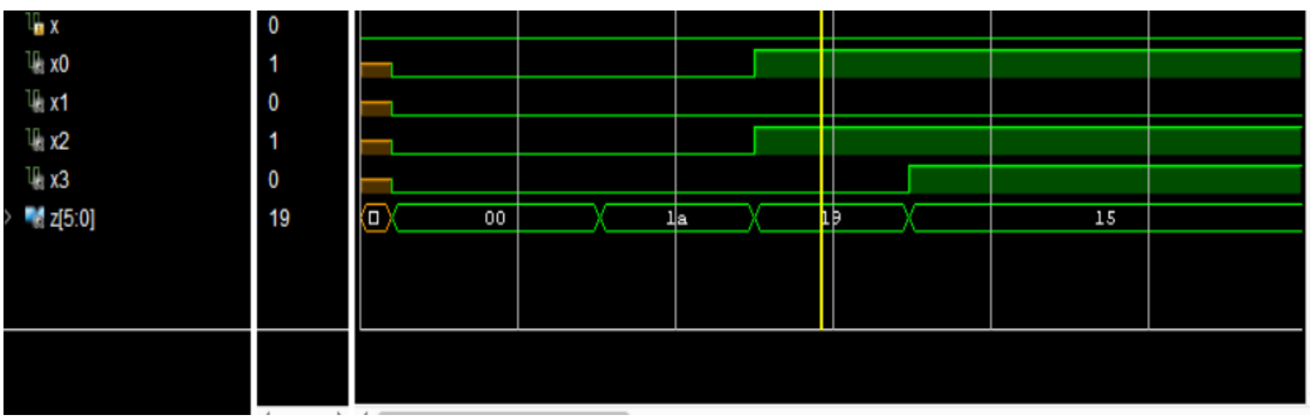

Figure 9 : Multiplication

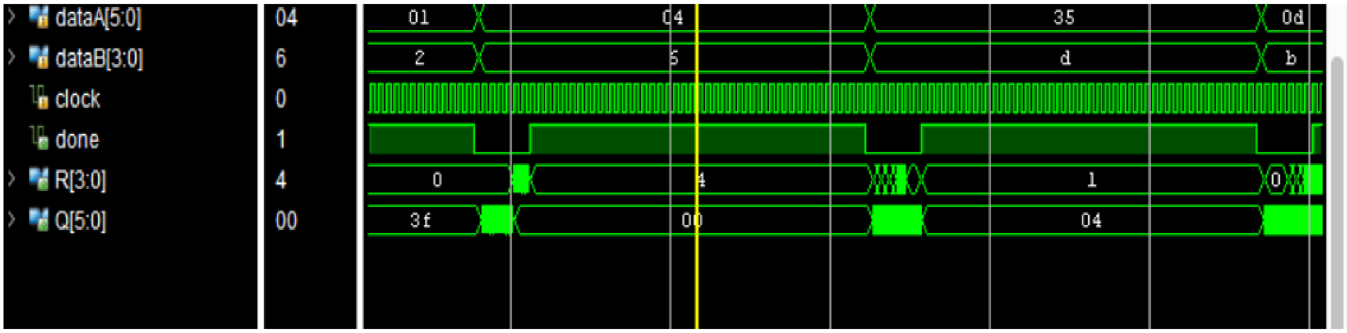

Figure 10 : Division

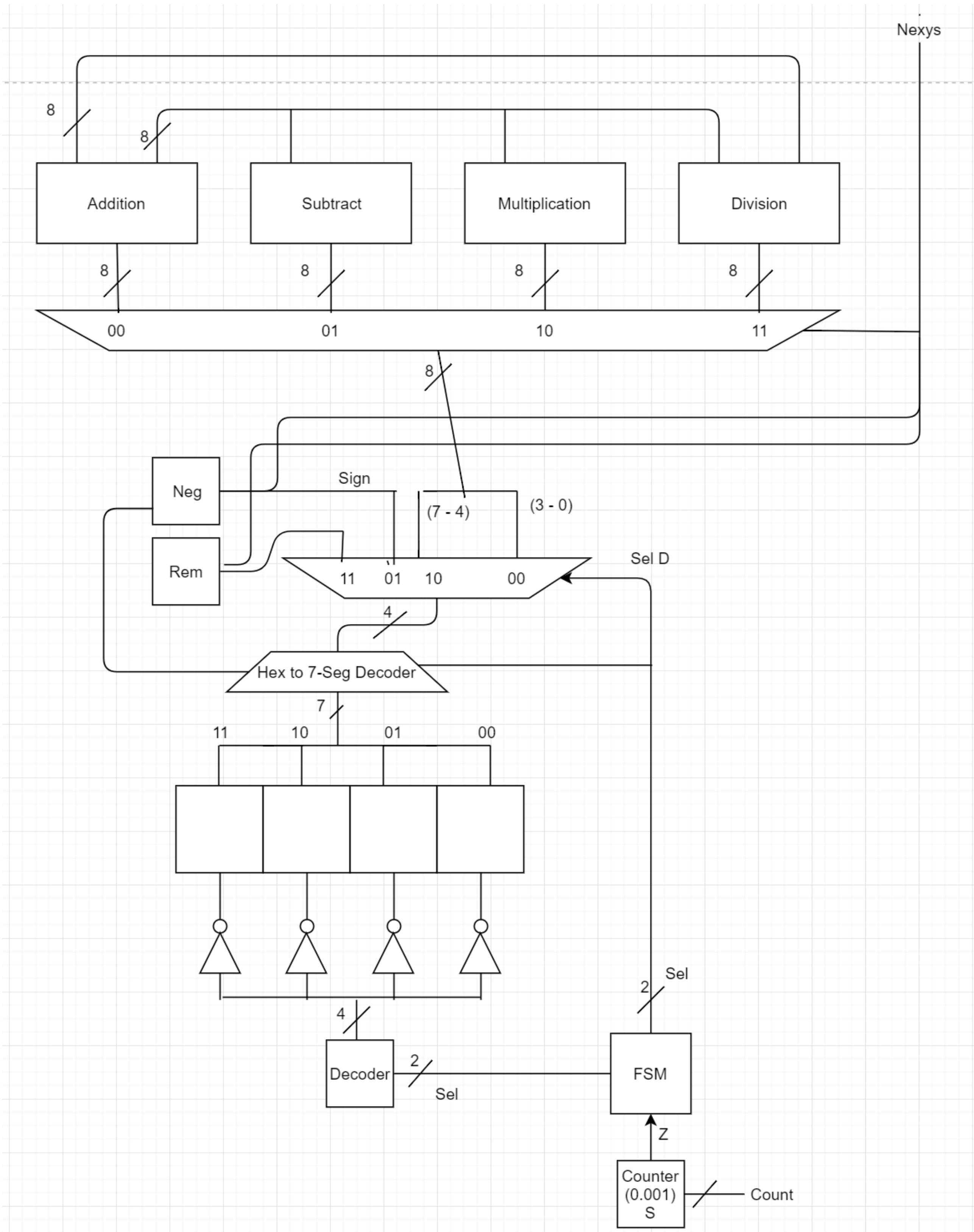

Figure 11: Design of the Calculator