KAUNAS UNIVERSITY OF TECHNOLOGY

THE FIRST PRINCIPLES OF DIGITAL LOGIC

Laboratory work 3: Registers

Student: Zahi El Helou

Supervisor: KAZIMERAS BAGDONAS

Group: IFU-1

KAUNAS, 2022

Table of Contents

1. Assignment: 6- LR1, CR1, AL2 0 one's complement.

2. Objective:

To draw a truth table, design the schematic and run the simulation of a 6-bit universal register.
To draw a truth table, design the schematic and run the simulation of a 6-bit specialized register.
To design a modified schematic for both universal and specialized registers with the implementation of FPGA.

3.Pocedure:

Universal Register:

A0	A1	Q5	Q4	Q3	Q2	Q1	Q0	Function
0	0	Q5	Q4	Q3	Q2	Q1	Q0	Save
1	0	Q4	Q3	Q2	Q1	Q0	0	LL1, 0
0	1	1	Q5	Q4	Q3	Q2	Q1	LR1, 1
1	1	X5	X4	X3	X2	X1	X0	Load

Figure 1. Truth table for universal register

We did the truth table for a 6-bit universal register, and afterwards we designed its schematic that perform the following operations, hold state, logic shift right (LR1), logic shift left (LL1) and parallel data load. We are using for vacant bits additional DR (1) for LR1, and DL for LL1.

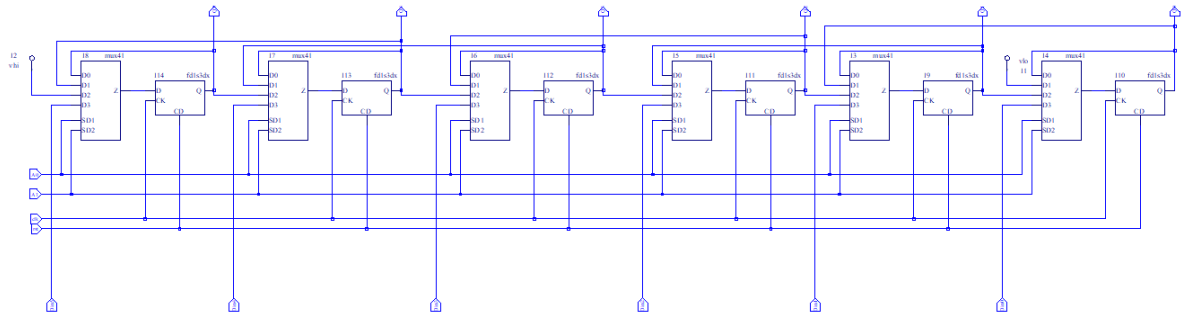

Figure 2. Schematic of a Universal Register

Afterwards, we change the properties of the project to VHDL, then we run the simulation.
We let A0 and A1 be negligible, so the data will be saved.
Then, we put A0 equals to 1 and A1 equals to 0 the register performs a logical shift to the left by 1.
When input A0 equals to 0 and A1 equals to 1 the register performs a logical shift to the right by 1.
Last, we implement 1 for signal A0 and signal A1 the register loads the data.

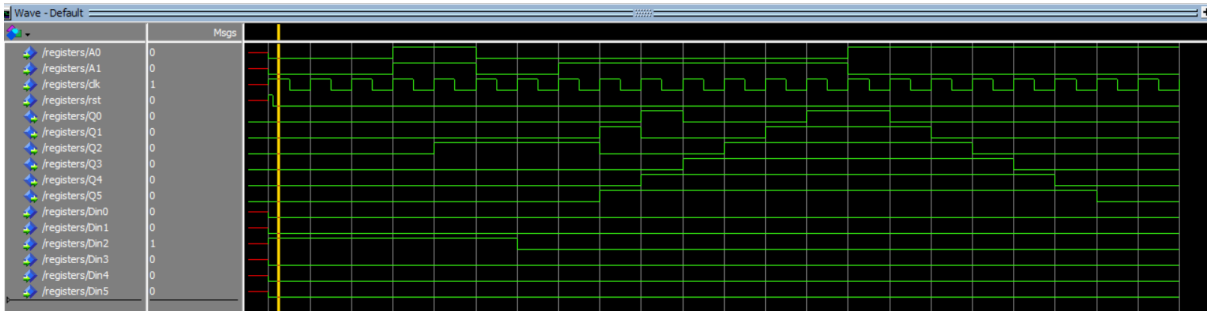

Figure 3. Timing Diagram of the Universal register

Specialized Register:

Now, we are designing a 6-bit specialized register which performs the following shift operations LR1 with input data “0”, CR1 and AL2 “one’s complement”. So in this case we will use 6 flipflops, which means 6 logic gates of fd1s3dx and 6 logic gates Mux 41, controlled by signals A0 and A1. We can now do its truth table.

A0	A1	Q5	Q4	Q3	Q2	Q1	Q0	Function
0	0	X5	X4	X3	X2	X1	X0	Load
1	0	0	Q5	Q4	Q3	Q2	Q1	LR1”0”
0	1	Q0	Q5	Q4	Q3	Q2	Q1	CR1
1	1	Q5	Q2	Q1	Q0	Q5	Q5	AL2 one’s complement

Figure 4. Truth table for the specialized register

Now we can design the schematic of a 6-bit specialized register.

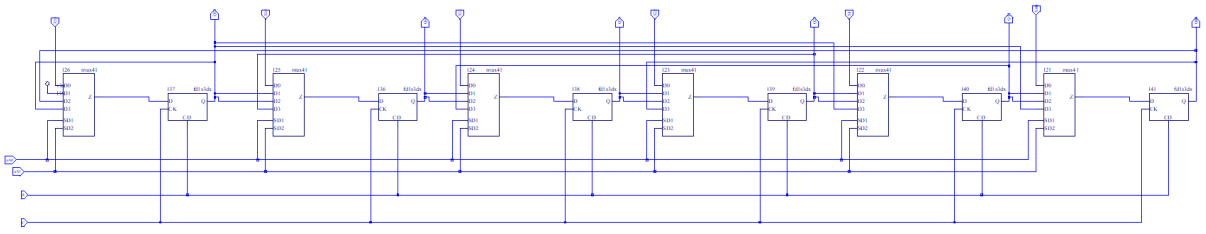

Figure 5. Schematic of a 6-bit specialized register

Afterwards we can run the simulation.

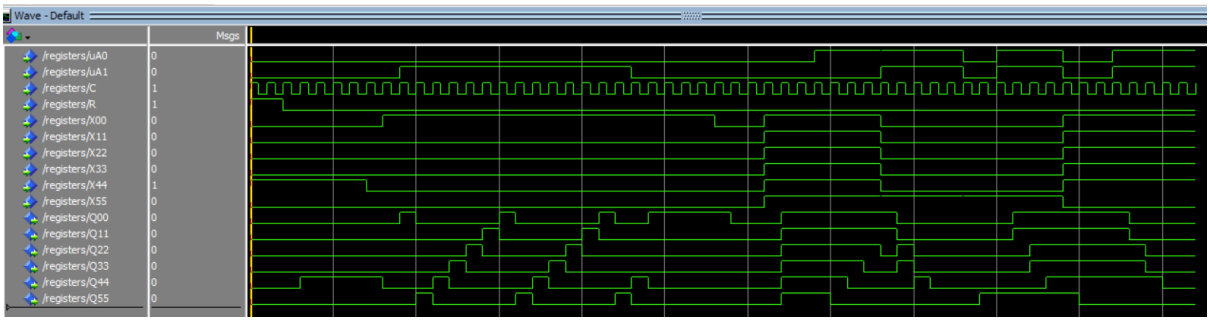

Figure 6. Timing diagram of a specialized register

We let A0 and A1 be negligible, the register loads the data.
Then, we put A0 equals to 1 and A1 equals to 0 the register performs a logical shift to the right by 1.
When input A0 equals to 0 and A1 equals to 1 the register performs a circular shift to the right by 1.
Last, we implement 1 for signal A0 and signal A1 the register performs arithmetical shift to the left by 2. We implemented for input X5 the 2 possible cases, positive and negative cases.
Last, the register performed all the required operations.

FPGA Implementation:
For universal register:

We must do some minor modification for our circuits to function on FPGA board. The list of modifications is provided below. All the outputs must be inverted, Asynchronous reset signal must be inverted since we are working with asynchronous and finally all parallel data inputs must be exchange into Vhi or Vlo components.

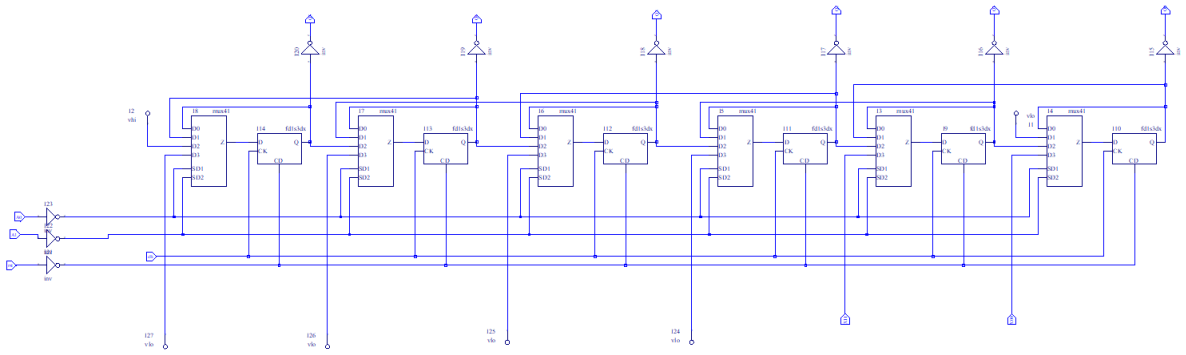

Figure 7. Modified schematic of a 6-bit universal register

Now we need to generate a JEDEC file. It is done in Process tab, by double-clicking the JEDEC label.
Once we have a design adapted to FPGA, we need to assign IO in your circuit to physical IO ports on the FPGA boards. We assigned for A0 “54” and for A1 “53”, for clock “50”, for reset “52”, for X11 ”56”, X00 “55”. As for the outputs, we assigned them bellow, Q5 “46” ,Q4 “45” ,Q3 “44” , Q2 ”42”
Q1 “40” , Q0 “39” . And now we can use the programmer to run it.

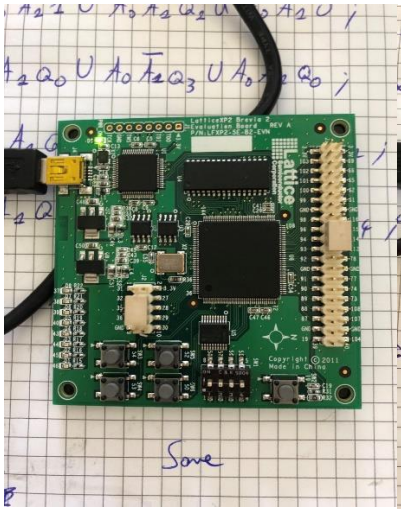

Figure 8. FPGA when it's saving

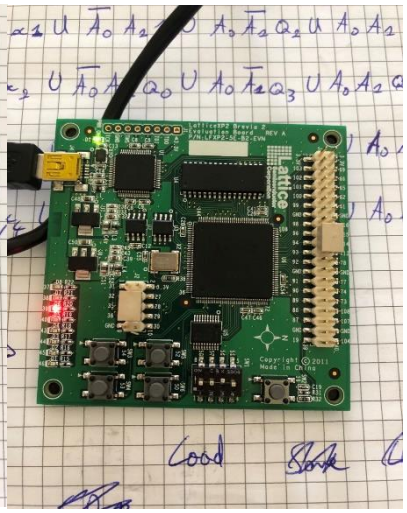

Figure 9. FPGA when it's loading

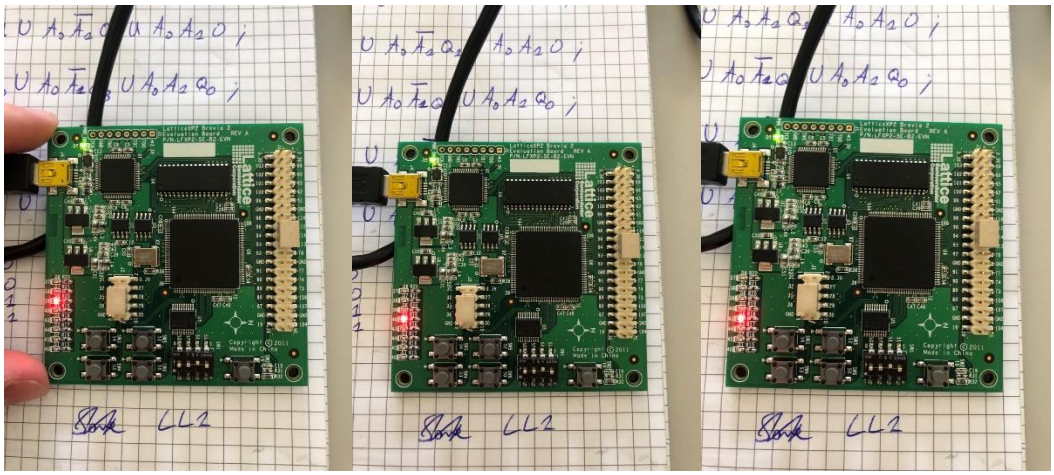

Figure 10. FPGA when it's logical shift to the left.

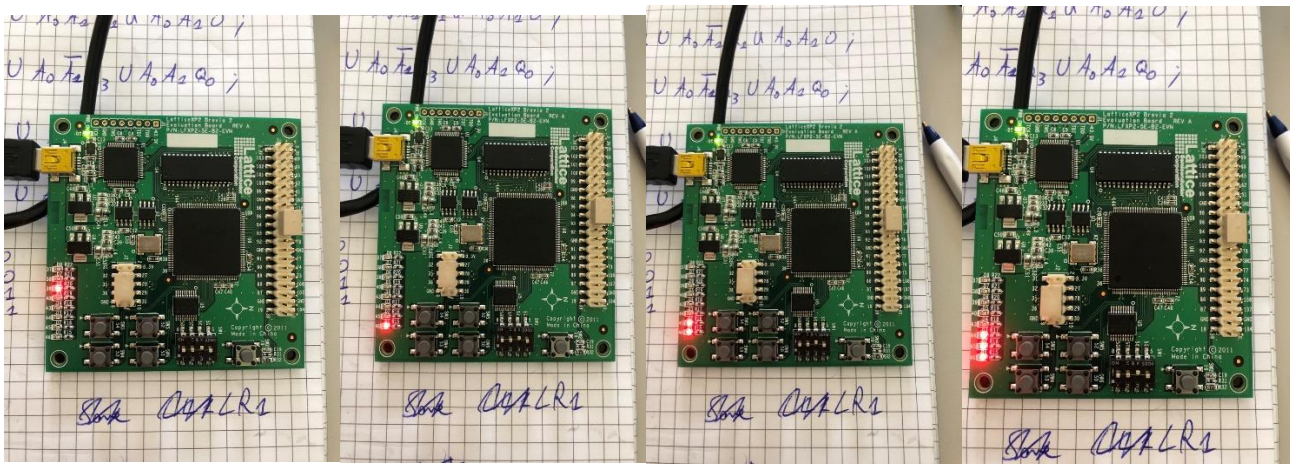

Figure 11. FPGA when it's logical shift to the right.

For specialized register:

Again, same process but now for the specialized register.
Assignment: 6- LR1, CR1, AL2 “0” one’s complement.

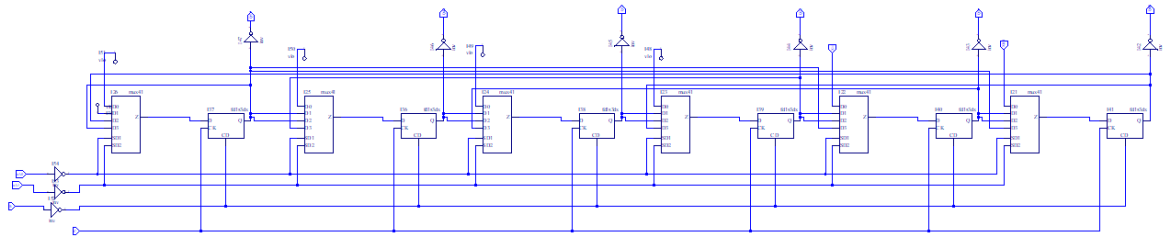

Figure 12. Modified schematic of a 6-bit specialized register

Figure 13. FPGA when it’s loading

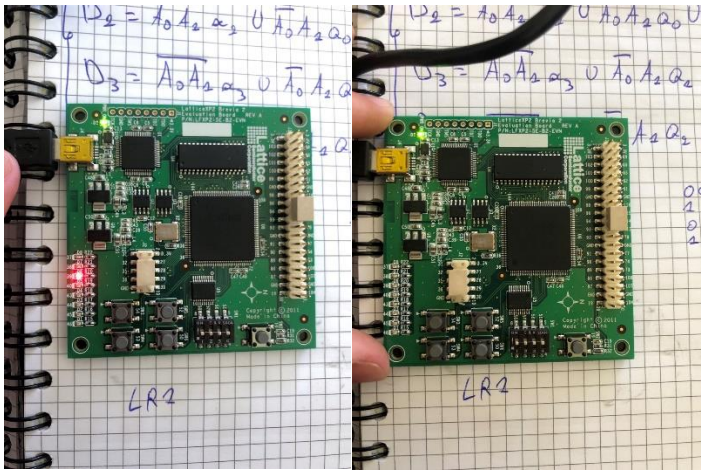

Figure 14. FPGA when it’s logical shift to the right by 1-bit.

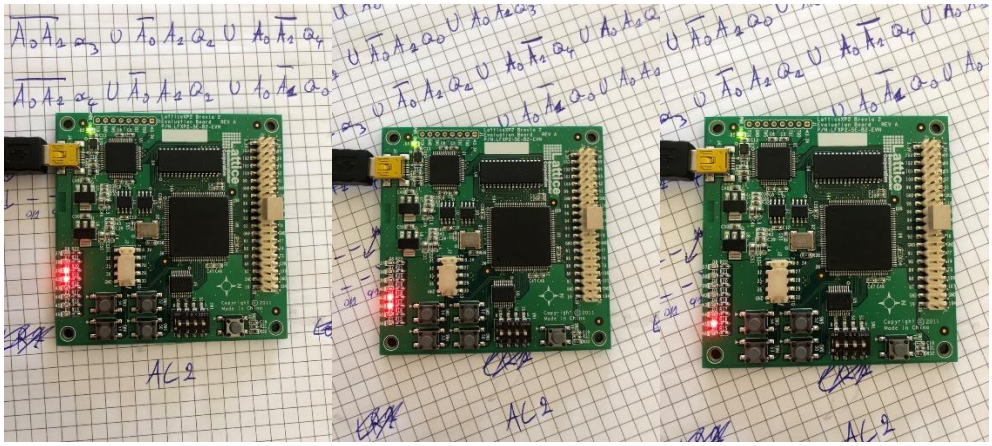

Figure 15. FPGA when it's arithmetic shift to the left by 2 bits.

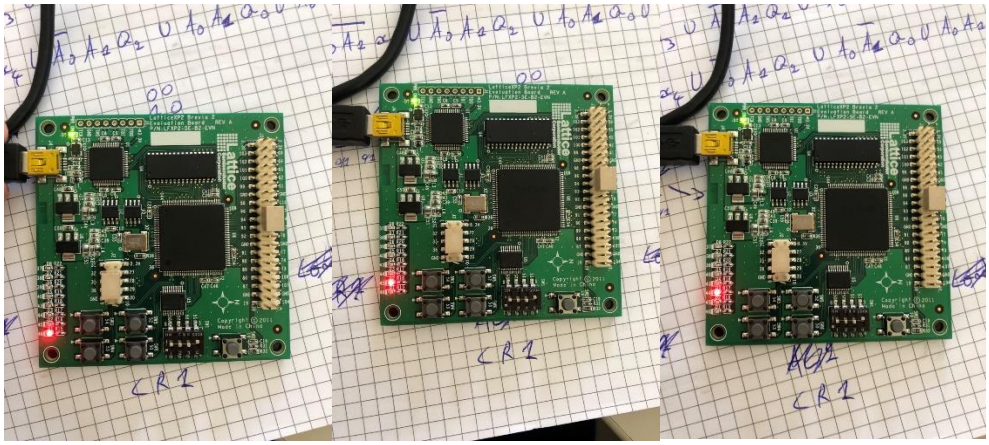

Figure 16. FPGA when it's circular shift to the right by 1-bit.

4.Conclusion:

In this Laboratory work, we have designed a 6-bit universal register and a 6-bit specialized register, and we implemented different signals to very different functions. So, we can conclude that the register is used to store such information, and the information stored within these registers can be transferred with the help of shift registers. In addition, we were able to manually edit the functions of each and every button of a Lattice XP2, which is designed with LatticeXP2 FPGAs, therefore, to be able to do multiple operations using the evaluation board.