Experiment #1 - Clock and Periodic Signal Generation

Student Name: Marzieh Bagheri Nia Amiri
Student ID: 810197682

First Part

1.1 Ring Oscillator

As stated in the experiment agenda, a ring oscillator is a device composed of combining an individual number of NOT gates in a loop, the output of which oscillates between two voltage levels.
In this case we use it for calculating the Inverter gate delay and generating clock cycle.

1. As we see in figure1, time period for node A is (0.52 μs = 0.96 – 0.44).

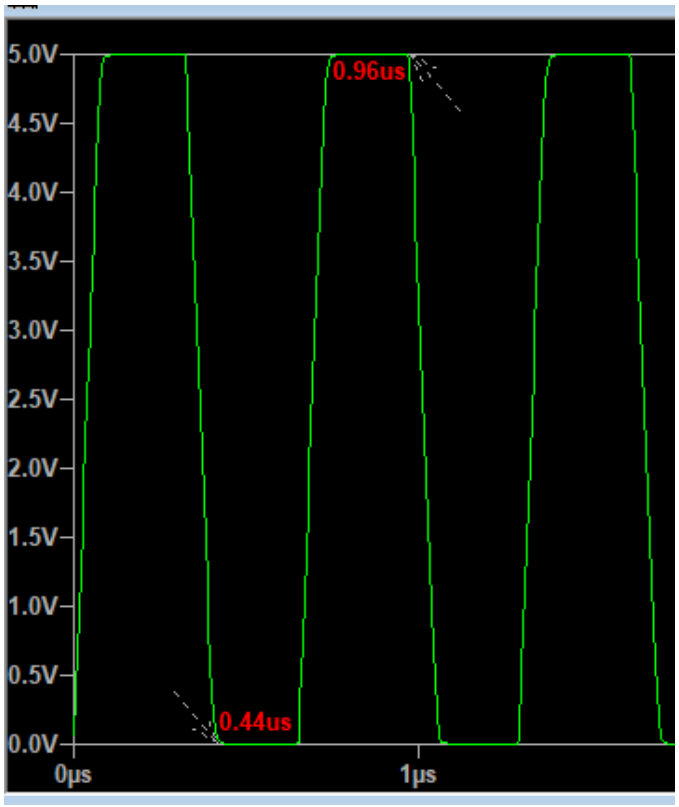

Figure 1

2. We know that the time obtained in the previous part is equal to 6D, that D is the delay of single inverter.
So the delay of one inverter is: $0.52 \div 6 = 0.086\ \mu s\ or\ 86\ ns$.
According to https://www.futurlec.com/74/IC7404.shtml , the propagation delay time Low to High level output of TTL 7404 is 22ns and the propagation delay time High to Low level output of TTL 7404 is 15ns.

The entire implementation of this section is shown in In figure2 and also the executable file of this part in LTspice is in Project/Part1/section1 folder.

1.2 LM555 timer

We implement the LM555 timer in astable mode in this part. At first, in figure3 we have the entire implementation of this section in LTspice.

1. in this part, we should calculate the clock frequency and Duty cycle.
So we do the following:

$$\text{ClockFreq} = \frac{14}{0.3 \times 10^{-3}} = 466666.6 = 0.4666666\ \text{MegaHertz}$$

$$\text{DutyCycle} = \frac{0.014}{0.022} \times 100 = 63\%$$

2. the different result by changing value of R2, prepared in following table:

Value of R2\ parameters	ClockFreq	DutyCycle	Figure
1k	47.0366 kH	66%	3
10k	6.8041 kH	52%	4
50k	1.4164 kH	50%	5
100k	0.714 kH	50%	6

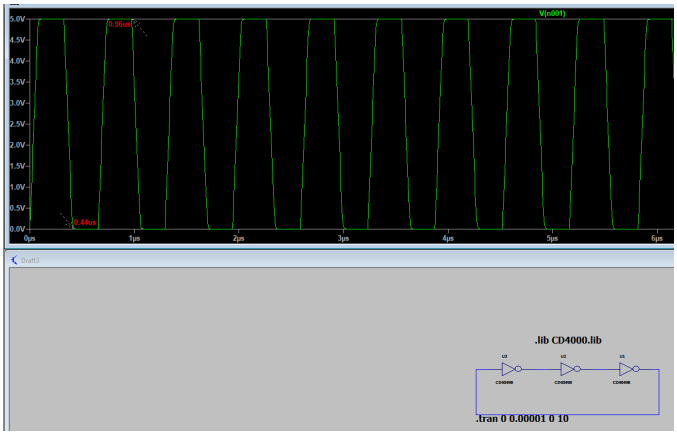

Figure 2

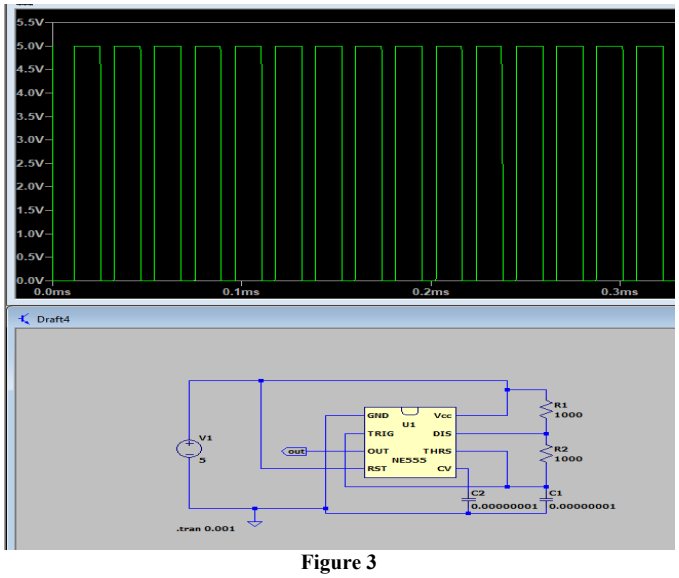

Figure 3

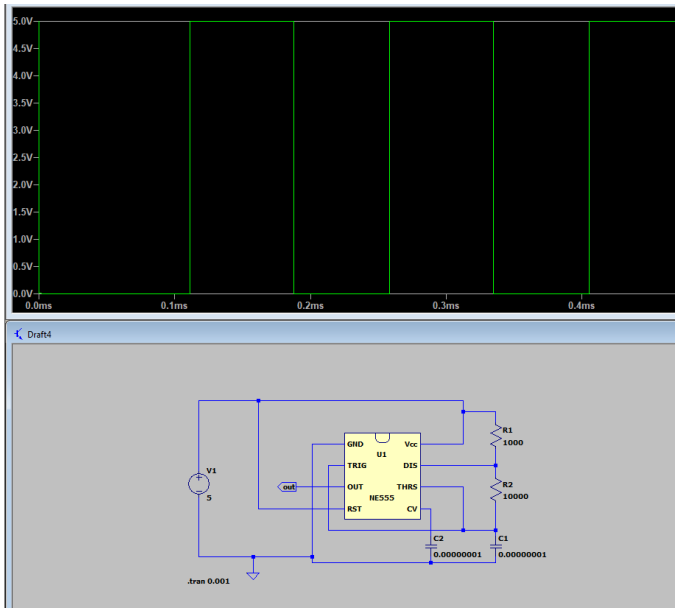

Figure 4

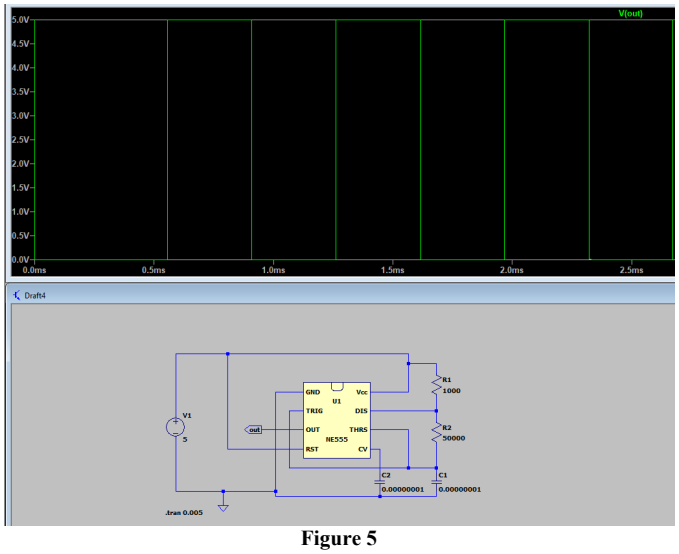

Figure 5

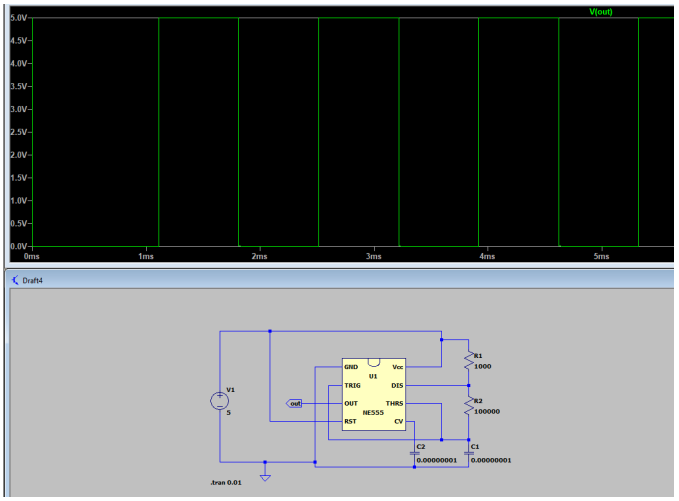

Figure 6

1.3 Scmitt Trigger Ocsillator

In this section we want to implement the Scmitt Trigger Ocsillator .
1. this part we change the value of R1 and obcerve the changing:

Value of R2	Figure
470	7
1k	8
2.2k	9

According to the results seen in the figures, we see that with increasing the amount of resistance, the height of the clock and also its shape becomes more complete and better.

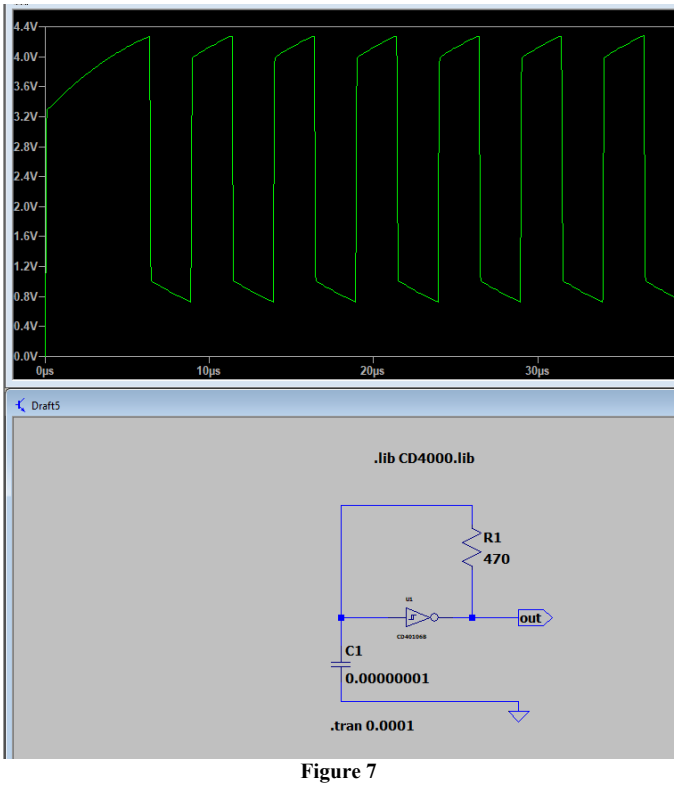

Figure 7

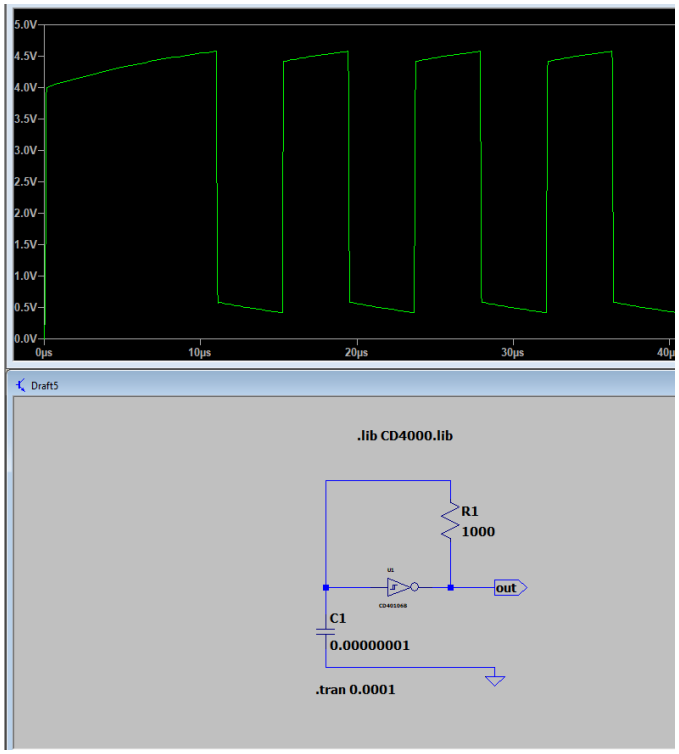

Figure 8

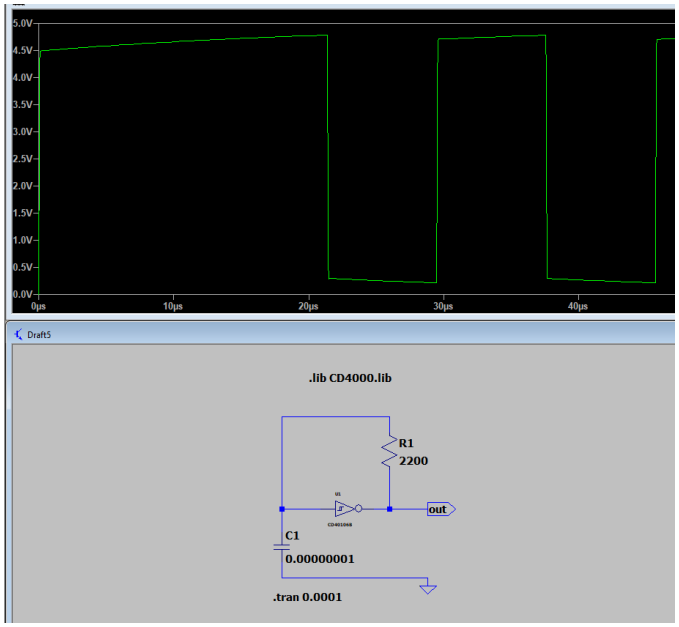

Figure 9

2. for calculating α, we select 2.2 for R2:

$$f = \frac{\alpha}{RC} \rightarrow 6.169 \times 10^4 = \frac{\alpha}{2.2 \times 10^3 \times 10^{-8}}$$

$$\alpha = 1.35718$$

Second Part

In this part we want to examine how the LM555 timer works in Verilog.
1. Done.
2. In this module, we have two counters: on_counter and off_counter.
We can set the on_duration and off_duration of pulse signal with change the amount of R1, R2, C.
At frist, we reset the circuit. According to count_off counting condition, count_on counting starts. Counting continues until the values of count_on and on_duration are the same. In this time count_off_en is enabled and count_off counting starts. Counting continues until the values of count_off and off_duration are the same. This process continues as long as the clock signal is active.

The block diagram can be seen below:

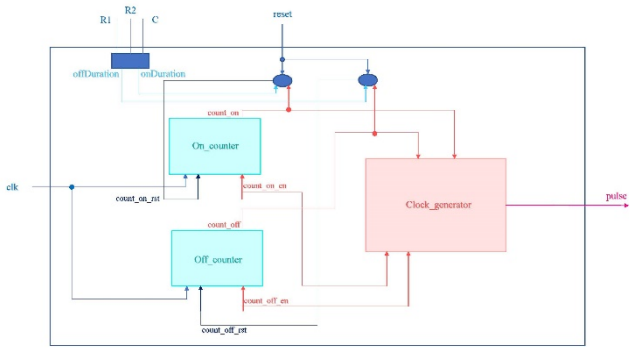

Figure 10

- A clearer picture of Figure10 exists in Project/Part2 folder.

3. I write a testbench for the testing the design. This testbench is available in Project/Part2 folder. The output image of this simulation is below:

- A clearer picture of Figure11 exists in Project/Part2 folder.

4. The only parameter that needs to be changed is the resistance, which we Set to 50. The output image of this simulation is below:

- A clearer picture of Figure12 exists in Project/Part2 folder.

The duty cycle of this circuit is 50%. By comparing the result of this part with Section 1.2, it can be seen that the two result are completely matched.

5. The alternative method written for this design. Related files can be viewed Project/Part2 folder.
The output image of this simulation is below:

- A clearer picture of Figure13 exists in Project/Part2 folder.

The duty cycle of this design is:

$$\mathrm{DC} = \frac{353}{700} \times 100 = 50.4\%$$

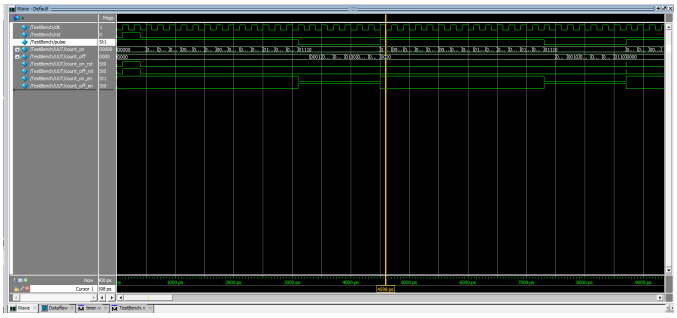

Figure 11

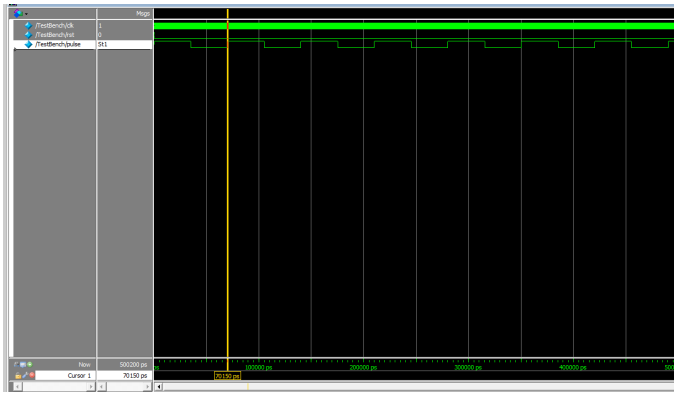

Figure 12

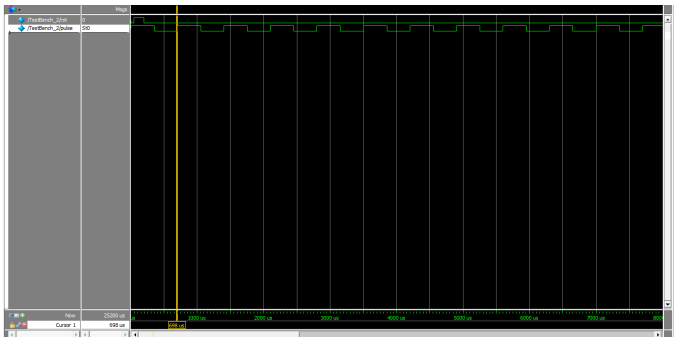

Figure 13

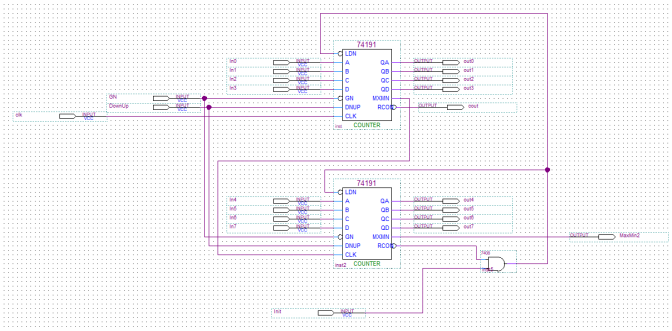

Figure 15

3. In this section, I write a Testbench for CounterDivider circuit. Related files can be viewed Project/Part3/Section2/simulation/qsim folder. After simulating it in modelsim, the waveform I received is below:

- A clearer picture of Figure16 exists in Project/Part3/Section2 folder.

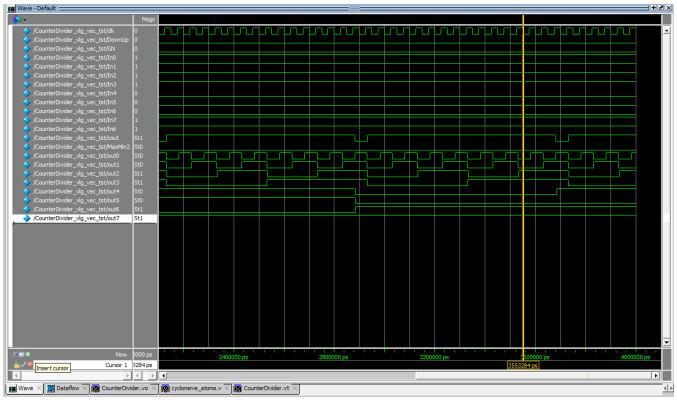

Figure 16

4. For this section I write a new testbench. Related files can be viewed Project/Part3/Section2/simulation/qsim folder. After simulating it in modelsim, the waveform I received is below:

- A clearer picture of Figure17 exists in Project/Part3/Section2 folder.

Third Part

3.1 Ring Oscillator

1. In this part, we write a Verilog description of the ring oscillator in modelsim. The file exists in Project/Part3/Section1 folder.

2. In this part, we write a testbench for the ring oscillator in modelsim. The file exists in Project/Part3/Section1 folder. Calculated Delay in part 1.

3. the frequency of Ring Oscillator is:

$$f = \frac{1}{(22-10) \times 10^{-9}} = 83.333333\ MH$$

The output image of this simulation is below:

- A clearer picture of Figure13 exists in Project/Part2 folder.

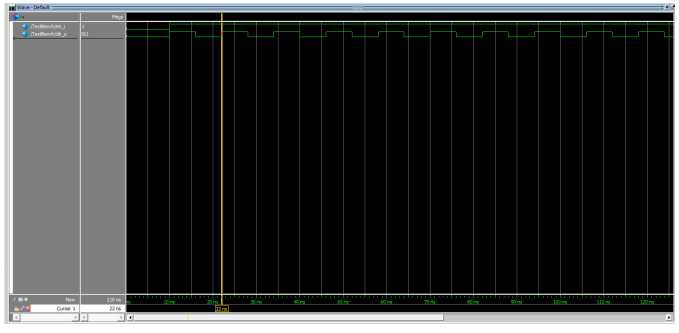

Figure 14

3.2. Synchronous Counter as a Frequency Divider

In this part, we design a CounterDivider:

1. Done.
2. Done. The output image of this simulation is below:

- A clearer picture of Figure15 exists in Project/Part3/Section2 folder.

3.3 TFF

In this section, we make a TFF from DFF and implement it in Quartus. Now it is ready for use in next section. Its block diagram Picture is below (Figure18) and also exists in Project/Part3 folder.

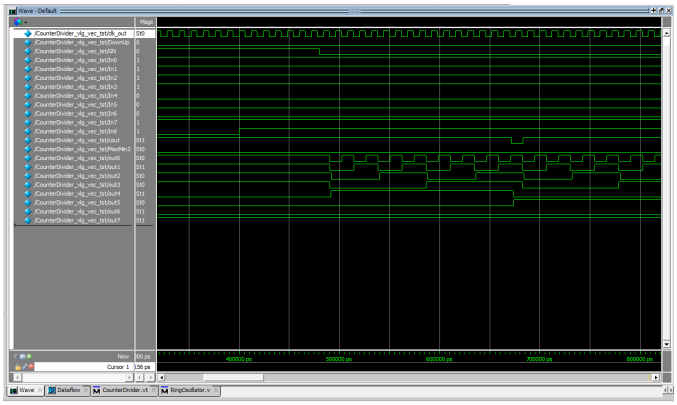

Figure 17

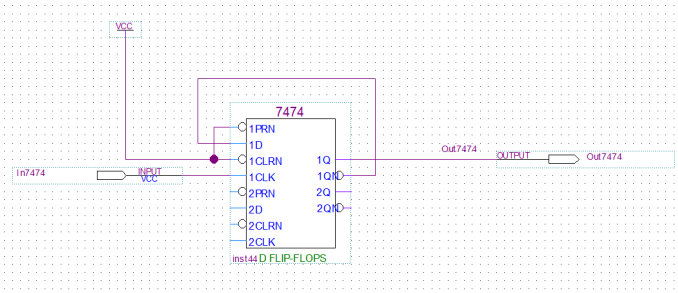

Figure 18

3.4 Display module in FPGA