

Lab 2: Design and Simulation of Sequential Logic Circuits - Synchronous Counters

By

Sailendhar Vippu (**300299480**)

Nisarg Patel (**300277261**)

CEG2136 Section B03 (B4O2)

Group 30

Date of Experiment: October, 3rd 2023
Date of Submission: October, 7th 2023

Department of Computer Engineering

University of Ottawa

Preface:

- Similar to the first lab of the course, this lab (Lab II) serves as an introduction to using virtual logic softwares like Quartus II and giving more practice on operating the Altera board.
- In this lab we want to design sequential circuits based on Altera's Quartus environment, and implement and test them with an FPGA.
- In the pre-lab we will need to make 6 K-Maps The K-Maps will need to be simplified to get the next state.

Lab Objectives:

- To initiate the students who are not familiar with the Altera Quartus II Design Software and the Altera FPGA based DE2-115 platform.
- To understand the basics of the Altera environment.
- To compile, simulate, debug, and test their design.
- To familiarize on using a Oscilloscope Waveform

Required Equipment:

- Quartus II Software
- Altera DE2-115 Board
- USB Blaster Cable
- Oscilloscope Keysight/Agilent MSOX2012A (2 analog + 8 digital channels)

(Pre-Lab) - Logic Design of Counters:

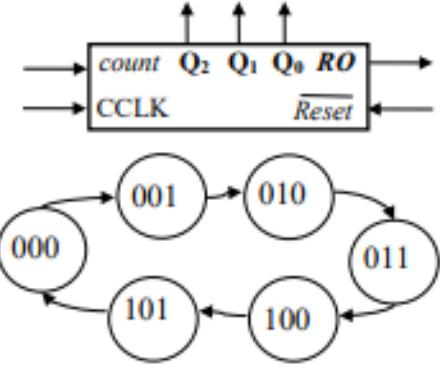


Figure 1: (a) Block diagram and
(b) State Diagram of a Modulo 6 counter

In	Present State			Next State			Out	Synchronous Inputs					
	msb	lsb		msb	lsb			J ₂	K ₂	J ₁	K ₁	J ₀	K ₀
count	Q ₂	Q ₁	Q ₀	Q ₂ ⁺	Q ₁ ⁺	Q ₀ ⁺	RO						
0	x	x	x	Q ₂	Q ₁	Q ₀	0	0	0	0	0	0	0
1	0	0	0	0	0	1	0	0	x	0	x	1	x
1	0	0	1	0	1	0	0	0	x	1	x	x	1
1	0	1	0	0	1	1	0	0	x	x	0	1	x
1	0	1	1	1	0	0	0	1	x	x	1	x	1
1	1	0	0	1	0	1	0	x	0	0	x	1	x
1	1	0	1	0	0	0	1	x	1	0	x	x	1
1	1	1	0	x	x	x	x	x	x	x	x	x	x
1	1	1	1	x	x	x	x	x	x	x	x	x	x

Table 1: The Excitation Table for the JK flip-flops Modulo 6 counter

Figure 1

3-Bit (KMaps)

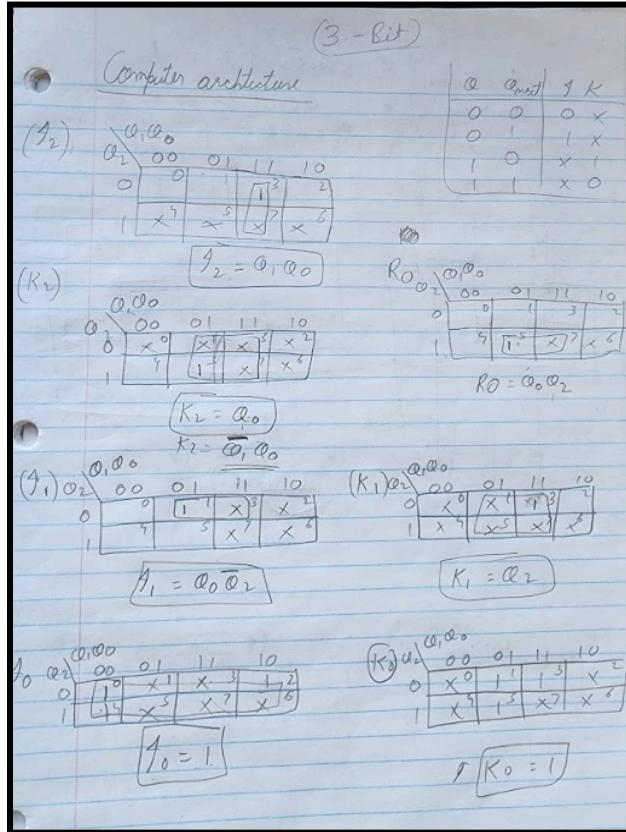


Figure 2

BCD 4-Bit (KMaps)

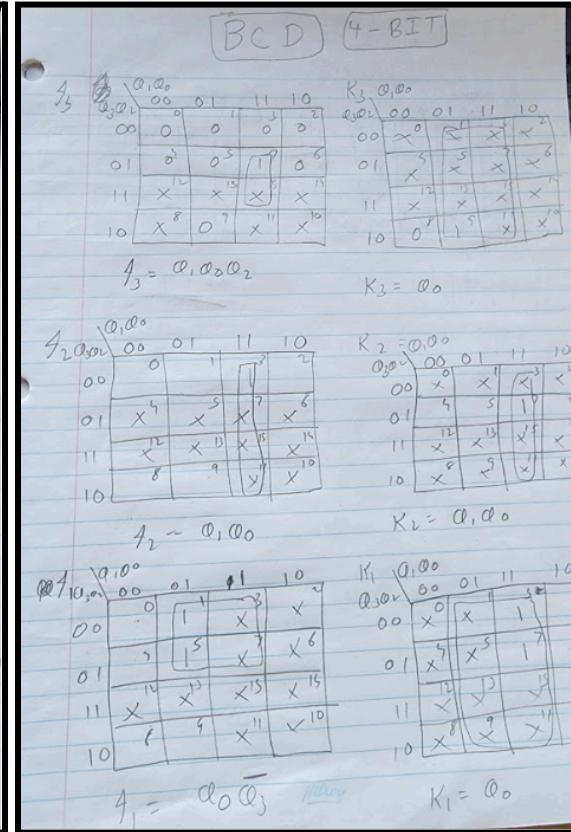


Figure 3A

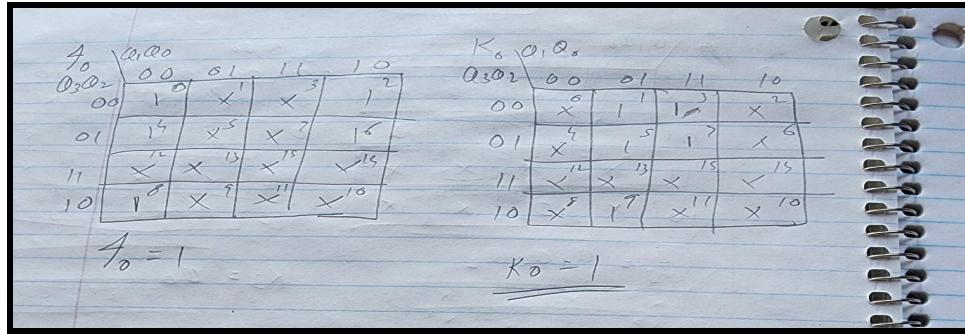


Figure 3B

Part I *A* (3-Bit counter):

Using the K-Maps from Figure 2, we could design a 3 bit sync mod 6 counter. With the help of the K-maps we were able to progress on creating a circuit in the Quartus Env.

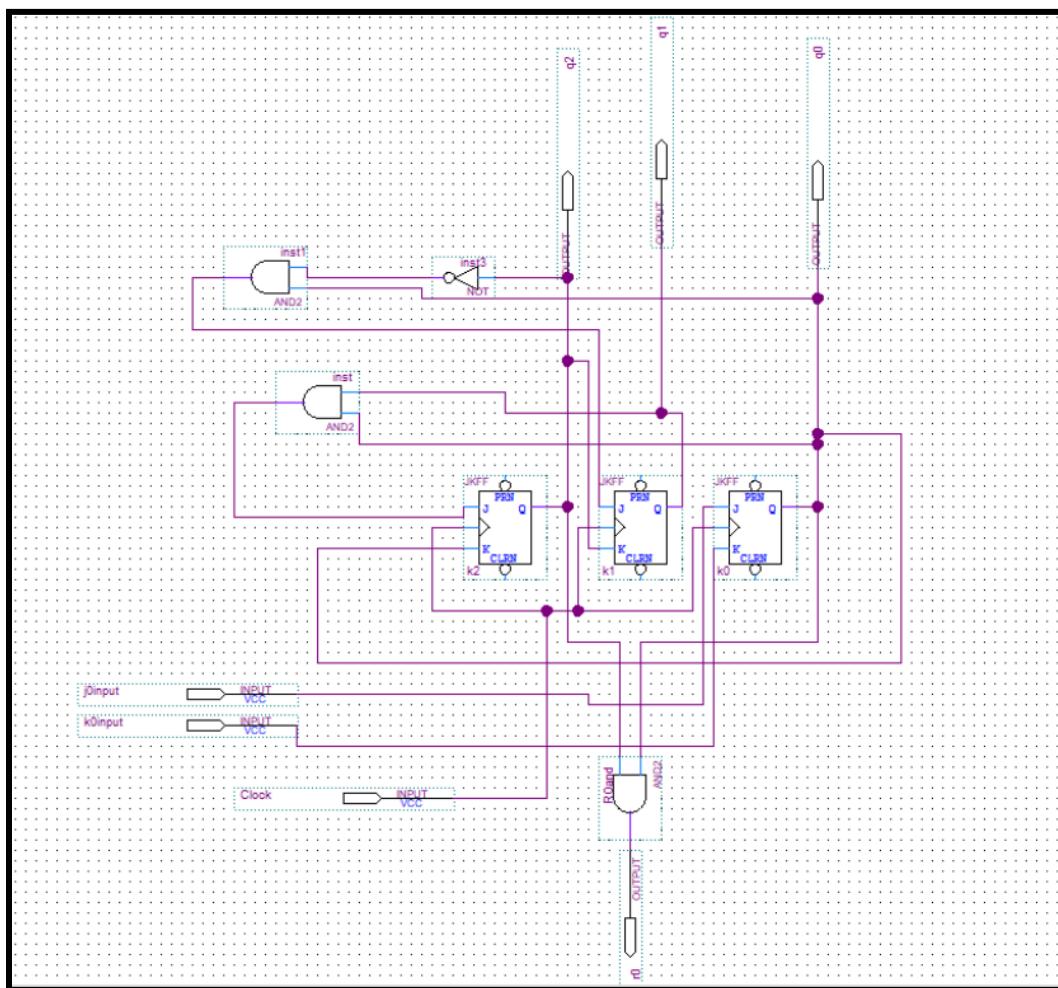


Figure 4

Using figure 4, we assigned EP4CE115F29C7 to the project. It was simulated for a waveform.

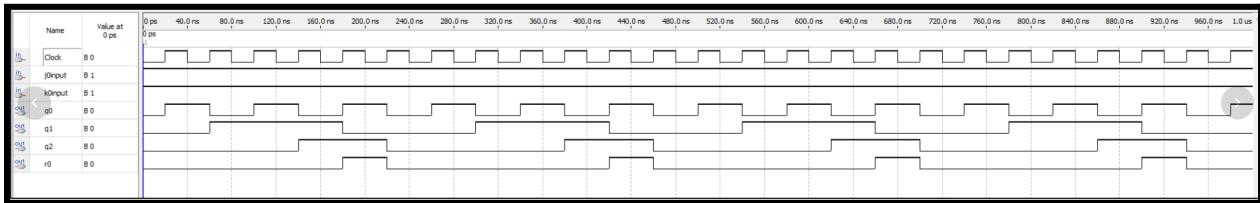


Figure 5

Part I *B*(4-Bit counter):

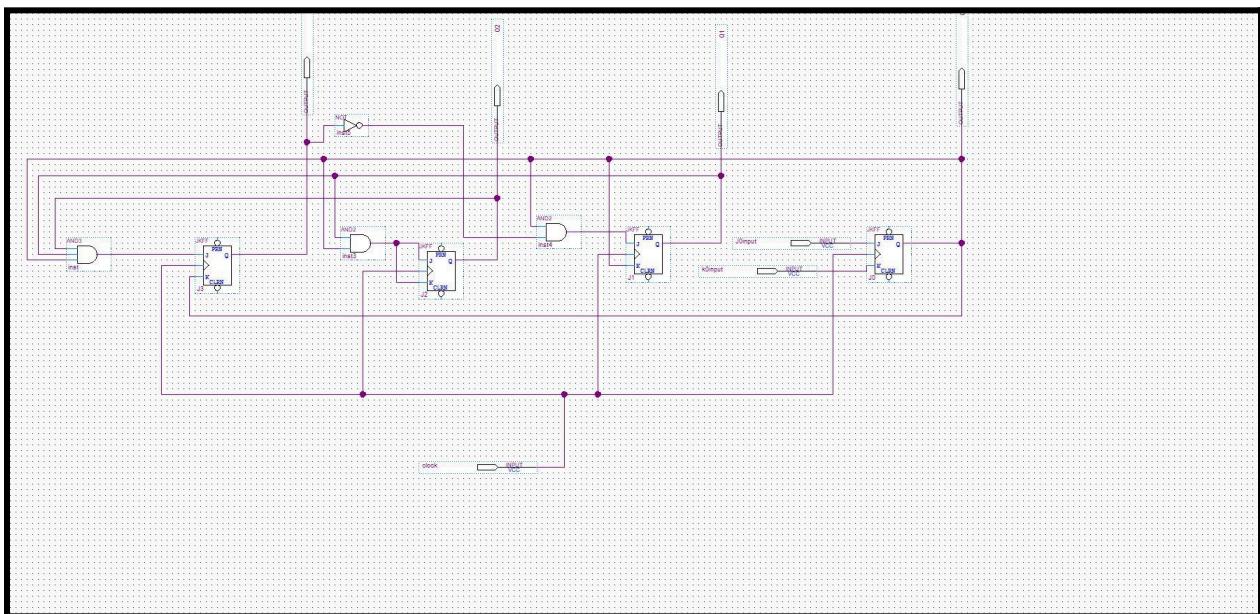


Figure 6

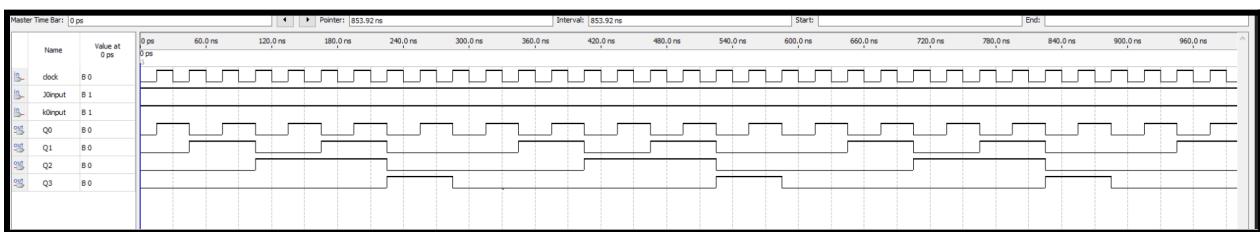


Figure 7

Modulo60 Circuit:

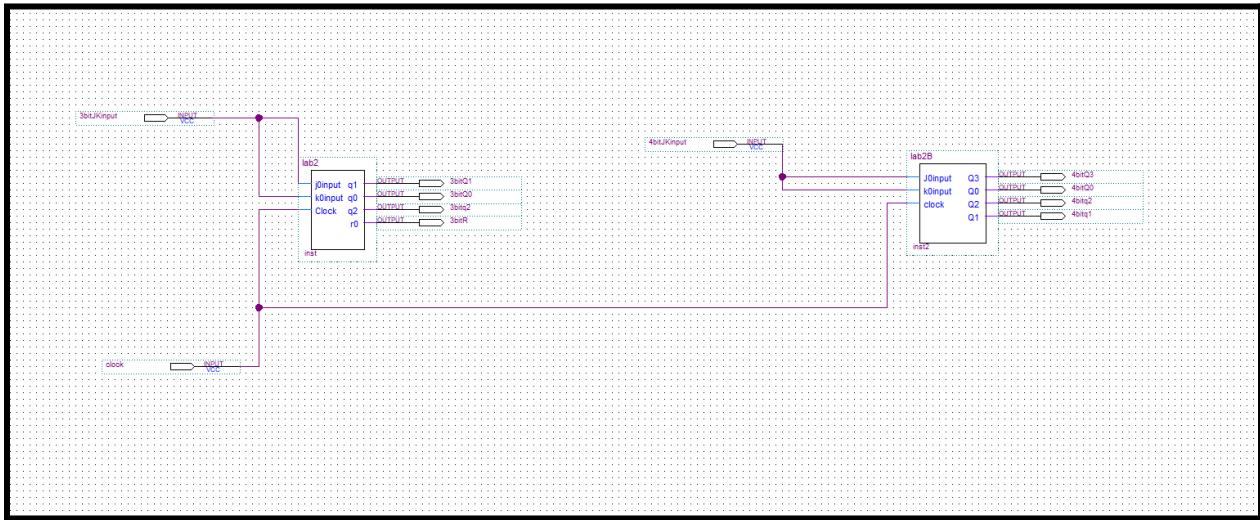


Figure 8

Above is the Modulo60 Circuit, this was designed by putting both counters together.

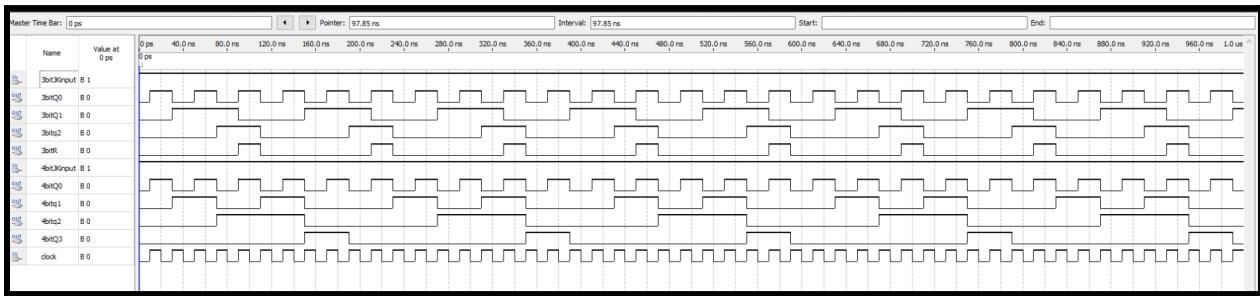


Figure 9

Part1-B BCD State Table (4-bit) -

Q3	Q2	Q1	Q0	Q3'	Q2'	Q1'	Q0'	J3	K3	J2	K2	J1	K1	J0	K0
0	0	0	0	0	0	0	1	0	X	0	X	0	X	1	X
0	0	0	1	0	0	1	0	0	X	0	X	1	X	X	1
0	0	1	0	0	0	1	1	0	X	0	X	X	0	1	X
0	0	1	1	0	1	0	0	0	X	1	X	X	1	X	1
0	1	0	0	0	1	0	1	0	X	X	0	0	X	1	X
0	1	0	1	0	1	1	0	0	X	X	0	1	X	X	1
0	1	1	0	0	1	1	1	0	X	X	0	X	0	1	X
0	1	1	1	1	0	0	0	1	X	X	1	X	1	X	1
1	0	0	0	1	0	0	1	X	0	0	X	0	X	1	X
1	0	0	1	0	0	0	0	0	1	0	X	0	X	X	1
1	0	1	0	X	X	X	X	X	X	X	X	X	X	X	X
1	0	1	1	X	X	X	X	X	X	X	X	X	X	X	X
1	1	0	0	X	X	X	X	X	X	X	X	X	X	X	X
1	1	0	1	X	X	X	X	X	X	X	X	X	X	X	X
1	1	1	0	X	X	X	X	X	X	X	X	X	X	X	X
1	1	1	1	X	X	X	X	X	X	X	X	X	X	X	X

Table 1

Part 1 Discussion:

The 3-bit circuit was used to counter until 5 and return to 0. In the same way the 4-bit circuit was designed to count until 9 and return to 0.

We constructed a 4-bit counter based on the logic equation derived from the K-maps. We designed the circuit in Quartus and simulated it by comparing its waveform to excitation table 1 given above. This confirmed that the logic circuit is performing the exact task we anticipated.

The waveform is matching with the excitation table that is provided in the Lab Manual. Thus, this proves all the boolean expressions that were formed from the k-maps are true. Given this, part one of this experiment was successful.

There were no issues encountered while dealing with this part.

Part 2 *A*(Simulating 4-bit with a digital BCD display):

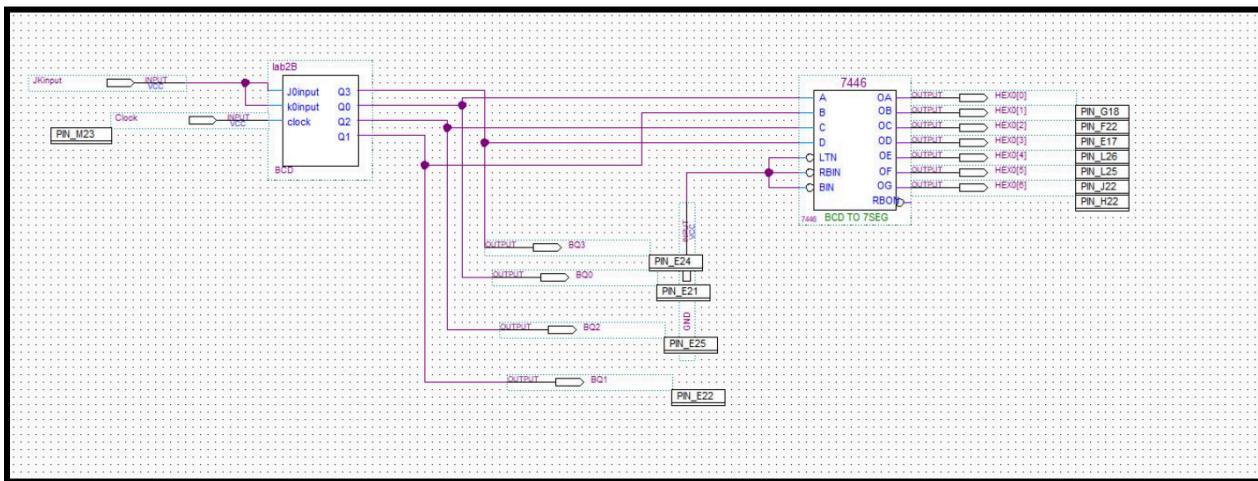
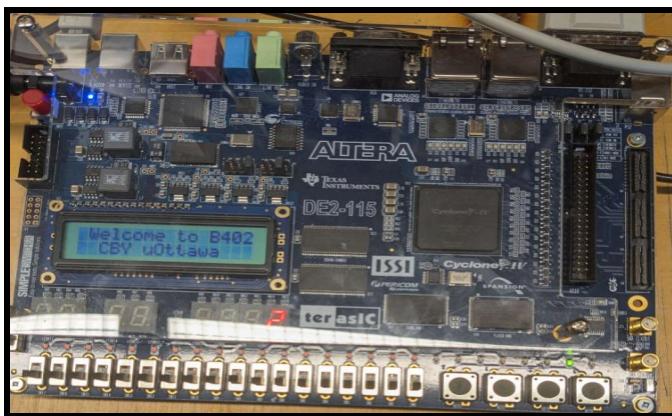


Figure 10

Part-2 I. Discussion:

Above we had made the BCD - 7 Segment Decoder inside Quartus. This was later on assigned to EP4CE115F29C7 and then was given FPGA pins for each logic signal. We used BCD to 7-segment decoder (7446) to connect the input of the 4-bit counter to 7446 decoder and the output of decoder is connected to the led of the 7 segment display. This was then compiled and ran through the USB Blaster to Demonstrate the circuit on the Altera board. Below is a picture of the Altera board after pushing KEY 0 in order to get a full counting sequence. The results matched with the wwf file.



Part 2 *B* (Using Oscilloscope):

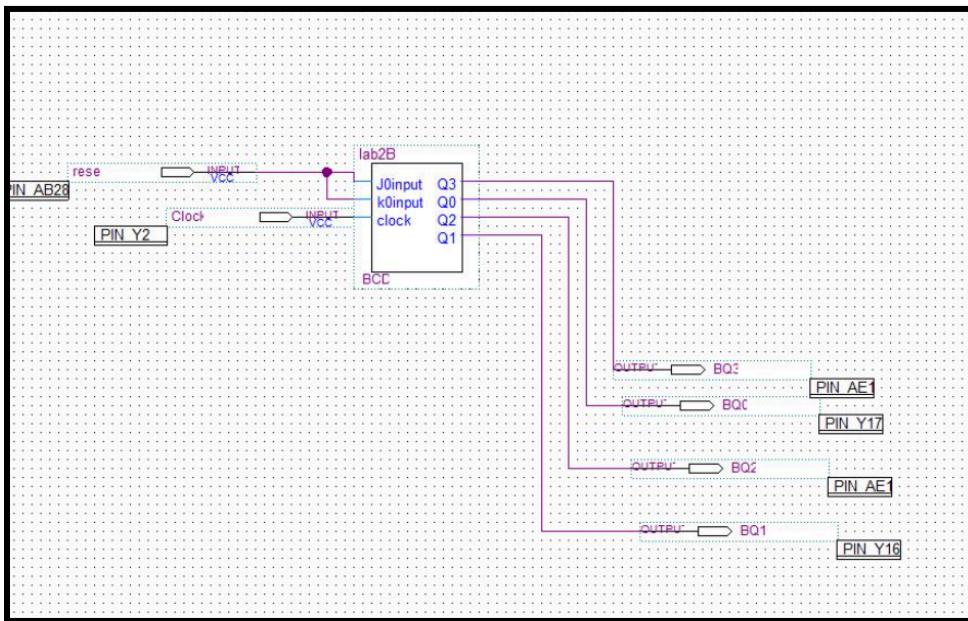


Figure 12

Table 4: Free-running BCD Pin Assignment

Signal	Pin assignment	Component
CCLK	PIN_Y2	50MHz oscillator
CLKOUT*	PIN_AE23	SMA_CLKOUT (BNC J15 - coax)
BQ3	PIN_AE15**	GPIO(9) as output to Oscilloscope ***
BQ2	PIN_AE16**	GPIO(7) as output to Oscilloscope***
BQ1	PIN_Y16**	GPIO(5) as output to Oscilloscope***
BQ0	PIN_Y17**	GPIO(3) as output to Oscilloscope***
GND3	PIN_AH26**	Ground for probe 3
GND2	PIN_AG23**	Ground for probe 2
GND1	PIN_AF26**	Ground for probe 1
GND0	PIN_AE24**	Ground for probe 0
RESET	PIN_AB28	SW0

Figure 13

P2II. Discussion:

In this part of the lab, we had to utilize the oscilloscope. This device will emulate the same or similar signals to what we will be getting within the waveform obtained in Part 1. All the signals were assigned with a custom pin and then it was sent into the USB Blaster. Probes were connected from the Oscilloscope to the Altera Board based on the pin assignments.

During this process, we found difficulty in getting the oscilloscope to work, there was no output even though all the pins were assigned properly. Below is an image of the pin placements.

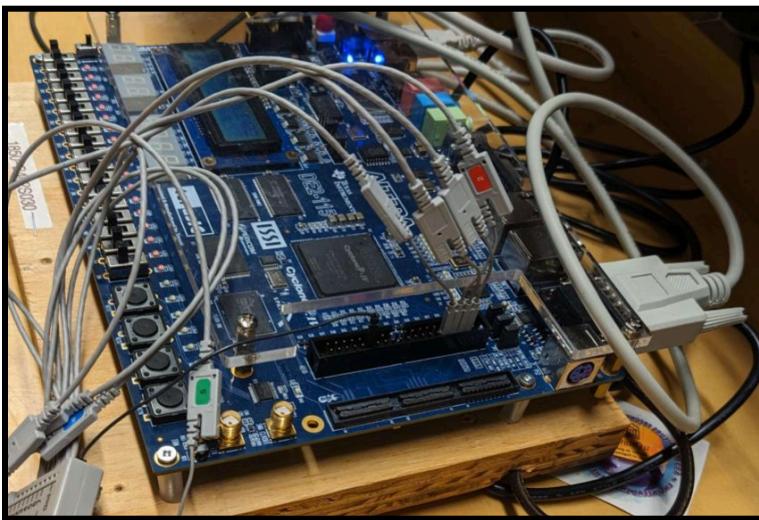


Figure 14

Conclusion:

To complete this lab experiment, we learned how to create a minimized logic equation from a given state diagram and construct a 3-bit and 4-bit counter. We also learned how to simulate these circuits and download them to an FPGA board to test their real-world functionality. Additionally, we familiarize ourselves with the oscilloscope, understand how to properly connect it to an FPGA board and obtain a waveform that matches the simulation results in the software (Quartus II).

JSR 0,X Print wha terminal LDX \$EE84 Load the vector for getchar routine JSR 0,X Get a new character in B LDAA #3 Initialize loop counter PSHA Save the counter on stack PSHB Save contents of B on stack LDD #\$20 Load B with a space LDX \$EE86 Load the vector for putchar routine JSR 0,X Print it on terminal PULB Get original character LDX \$EE86 Load the vector for putchar routine JSR 0,X Print it on terminal PULA Retrieve the counter DECA Decrement loop counter BNE \$200F If counter \neq 0, repeat SWI Return to the monitor