Quinn Roemer

Engineering – 303

Lab 8

3/28/2017

Introduction/Description

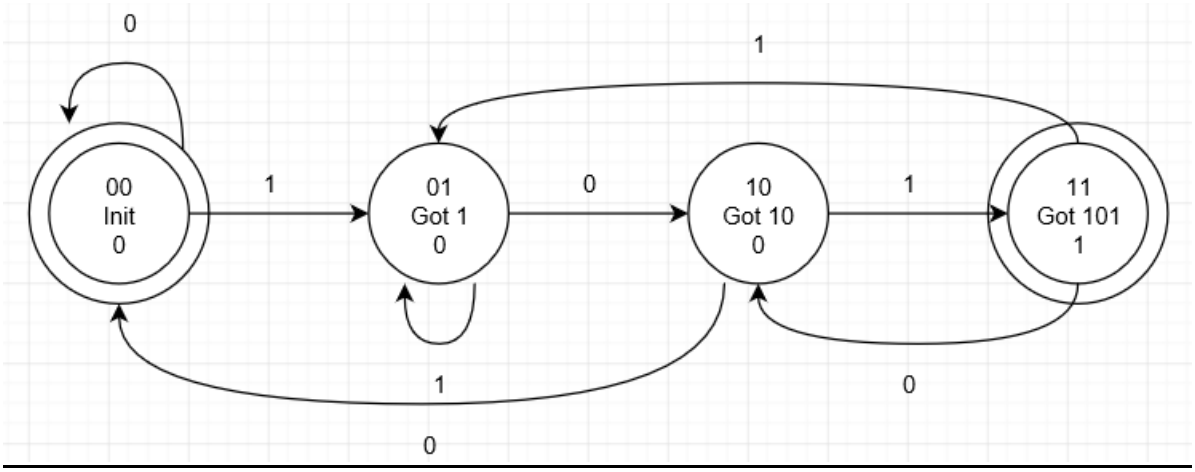
The goal of this lab was to design and implement several State Machine circuits in Quartus. This lab built upon all of the previous labs by combining combinational logic and memory circuits into one type of circuit capable of performing simple tasks. In this lab, we were supposed to make a total of six circuits. The first was a sequence detector that was capable of looking for the binary sequence of 101. The second was a state machine that would progress through a pre-set number as longs as it was receiving a binary 1. The third was to attach the previous circuit to seven segment decoders. The fourth was a circuit that detected the binary sequence of 101010 which is 42 in the decimal system. The fifth was similar to the second as it would progress through pre-set numbers as long as its input was a binary 1. The sixth was to attach the previous circuit to hex displays.

Design

Part 0 – Sequence Detector 101

In the first part of the lab, I was instructed to implement a circuit design that would be capable of detecting the binary input of 101. When the combination was detected by the circuit it would output a binary 1. In this design, I was given the truth table, Kmaps, and circuit design. My only goal was to input it all into Quartus and get it working.

Here is the State Diagram for the circuit.



Here is the truth table for the circuit.

Inputs			Outputs			MinTerms		
State		X	Next State		Z	mDa	mDb	mZ
A	B		Da	Db				
0	0	0	0	0	0			
0	0	1	0	1	0		a'b'x	
0	1	0	1	0	0	a'bx'		
0	1	1	0	1	0		a'bx	
1	0	0	0	0	0			
1	0	1	1	1	0	ab'x	ab'x	
1	1	0	1	0	1	abx'		abx'
1	1	1	0	1	1		abx	abx

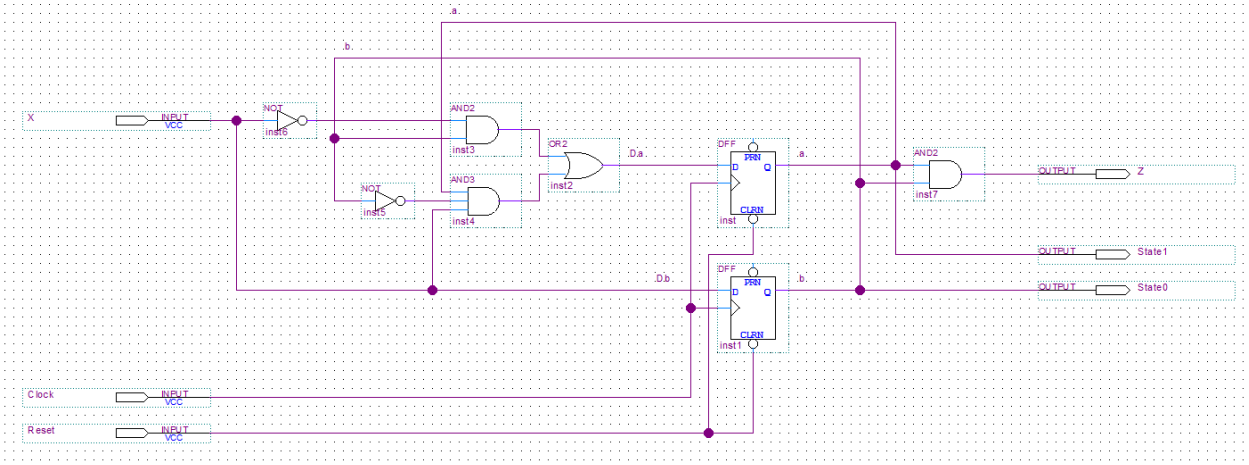
Here are all the Kmaps used to create the circuit.

Kmap Da	bx	b'x	b'x'	bx'
a		1		1
a'				1
Da = bx' + ab'x				

Kmap Db	bx	b'x	b'x'	bx'
a	1	1		
a'	1	1		
Db = x				

Kmap Z	bx	b'x	b'x'	bx'
a	1			1
a'				
Z = ab				

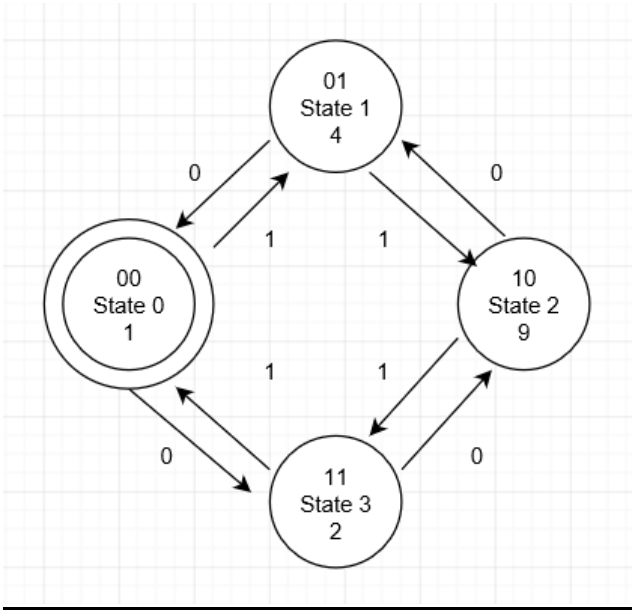
Here is the Block-Diagram for the circuit.



Part 1 – State Machine with 4-bit Moore Output 1492.

In the next part of the lab, I was told to design a circuit that would progress through 1492 in binary as long as the input it was receiving was a 1. If the input was 0 the circuit would then progress backward through 1294 in binary. Just like the circuit before this, I was given the truth table, kmaps, and circuit design.

Here is the State Diagram for the circuit.



Here is the Truth Table for the circuit.

Inputs			Outputs						MinTerms					
State		X	State		Z3	Z2	Z1	Z0	mDa	mDb	mZ3	mZ2	mZ 1	mZ0
A	B		Da	Db										
0	0	0	1	1	0	0	0	1	a'b'x'	a'b'x'				a'b'x'
0	0	1	0	1	0	0	0	1		a'b'x				a'b'x
0	1	0	0	0	0	1	0	0				a'bx'		
0	1	1	1	0	0	1	0	0	a'bx			a'bx		
1	0	0	0	1	1	0	0	1		ab'x'	ab'x'			ab'x'
1	0	1	1	1	1	0	0	1	ab'x	ab'x	ab'x			ab'x
1	1	0	1	0	0	0	1	0	abx'				abx'	
1	1	1	0	0	0	0	1	0					abx	

Here are the Kmaps used in the circuit.

Kmap Da	bx	b'x	b'x'	bx'
a		1		1
a'	1		1	
Da = a'b'x' + a'bx + abx' +ab'x				

Kmap Db	bx	b'x	b'x'	bx'
a		1	1	
a'		1	1	
Db = b'				

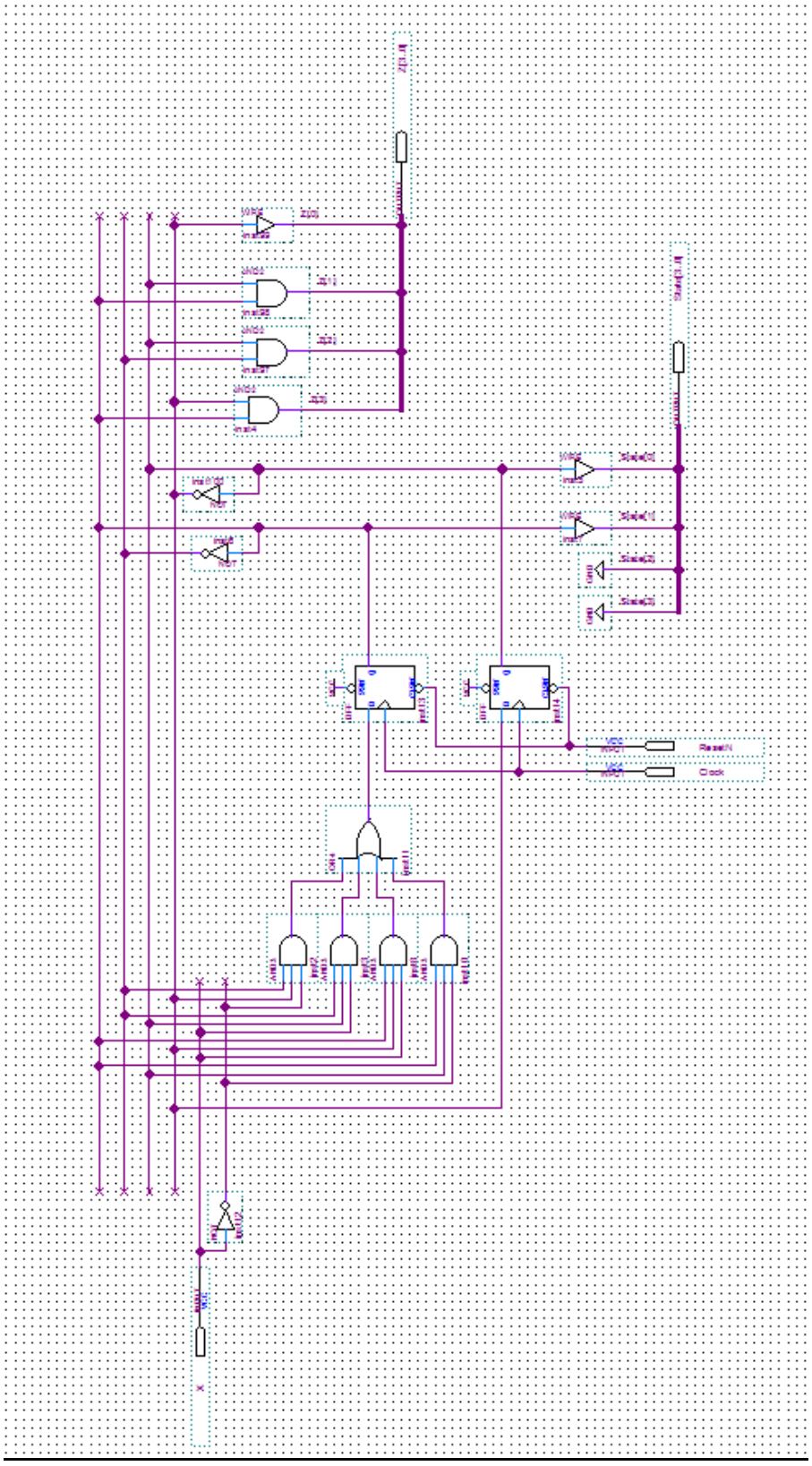
Kmap Z0	bx	b'x	b'x'	bx'
a		1	1	
a'		1	1	
Z0 = b'				

Kmap Z1	bx	b'x	b'x'	bx'
a	1			1
a'				
Z1 = ab				

Kmap Z2	bx	b'x	b'x'	bx'
a				
a'	1			1
Z2 = a'b				

Kmap Z3	bx	b'x	b'x'	bx'
a		1	1	
a'				
Z3 = ab'				

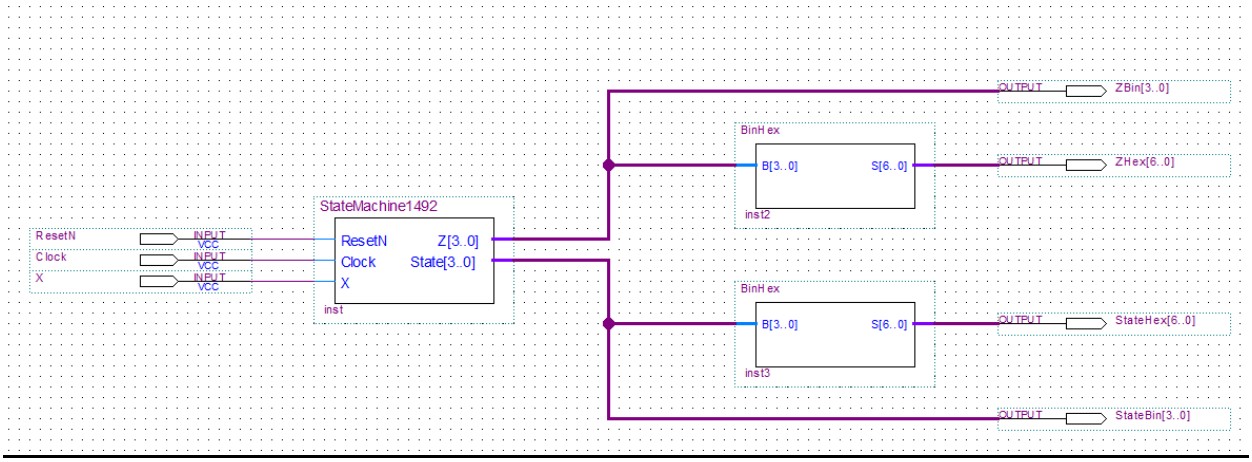
Here is the Block-Diagram for the circuit.



Part 2 – 1492 with Seven Segment Decoders

In this section of the lab, I was supposed to take the circuit that I just finished creating and attach a couple seven segment decoders to it. This required no extra truth tables or diagrams of any type.

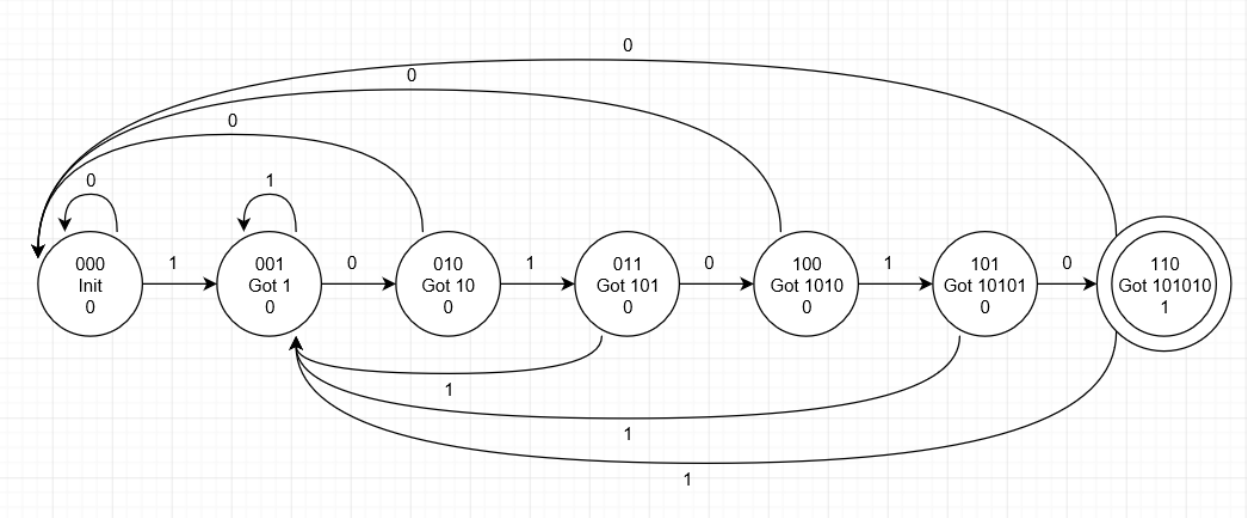
Here is the Block-Diagram for the circuit.



Part 3 – Sequence Detector 42

In this part of the lab, I was instructed to build a circuit that would be capable of detecting the binary sequence of 101010 which is 42 in the decimal system. Unlike the previous circuit, I was not given a filled out truth table or Kmaps with the simplified equation already figured out. Please note, the "DC" in truth tables and Kmaps simply means don't care. This allows it to either represent a binary 1 or 0, whatever is more beneficial at the time.

Here is the State Diagram for the circuit.



Here is the Truth Table for the circuit.

Inputs				Outputs				MinTerms			
State			X	Next State			Z	mDa	mDb	mDc	mZ
A	B	C		Da	Db	Dc					
0	0	0	0	0	0	0	0				
0	0	0	1	0	0	1	0			a'b'c'x	
0	0	1	0	0	1	0	0		a'b'cx'		
0	0	1	1	0	0	1	0			a'b'cx	
0	1	0	0	0	0	0	0				
0	1	0	1	0	1	1	0		a'bc'x	a'bc'x	
0	1	1	0	1	0	0	0	a'bcx'			
0	1	1	1	0	0	1	0			a'bcx	
1	0	0	0	0	0	0	0				
1	0	0	1	1	0	1	0	ab'c'x		ab'c'x	
1	0	1	0	1	1	1	0	ab'cx'	ab'cx'	ab'cx'	
1	0	1	1	0	0	1	0			ab'cx	
1	1	0	0	0	0	0	1				abc'x'
1	1	0	1	0	0	1	1			abc'x	abc'x
1	1	1	0	0	0	0	0	dc	dc	dc	dc
1	1	1	1	0	0	0	0	dc	dc	dc	dc

Here are all the Kmaps used to create the circuit.

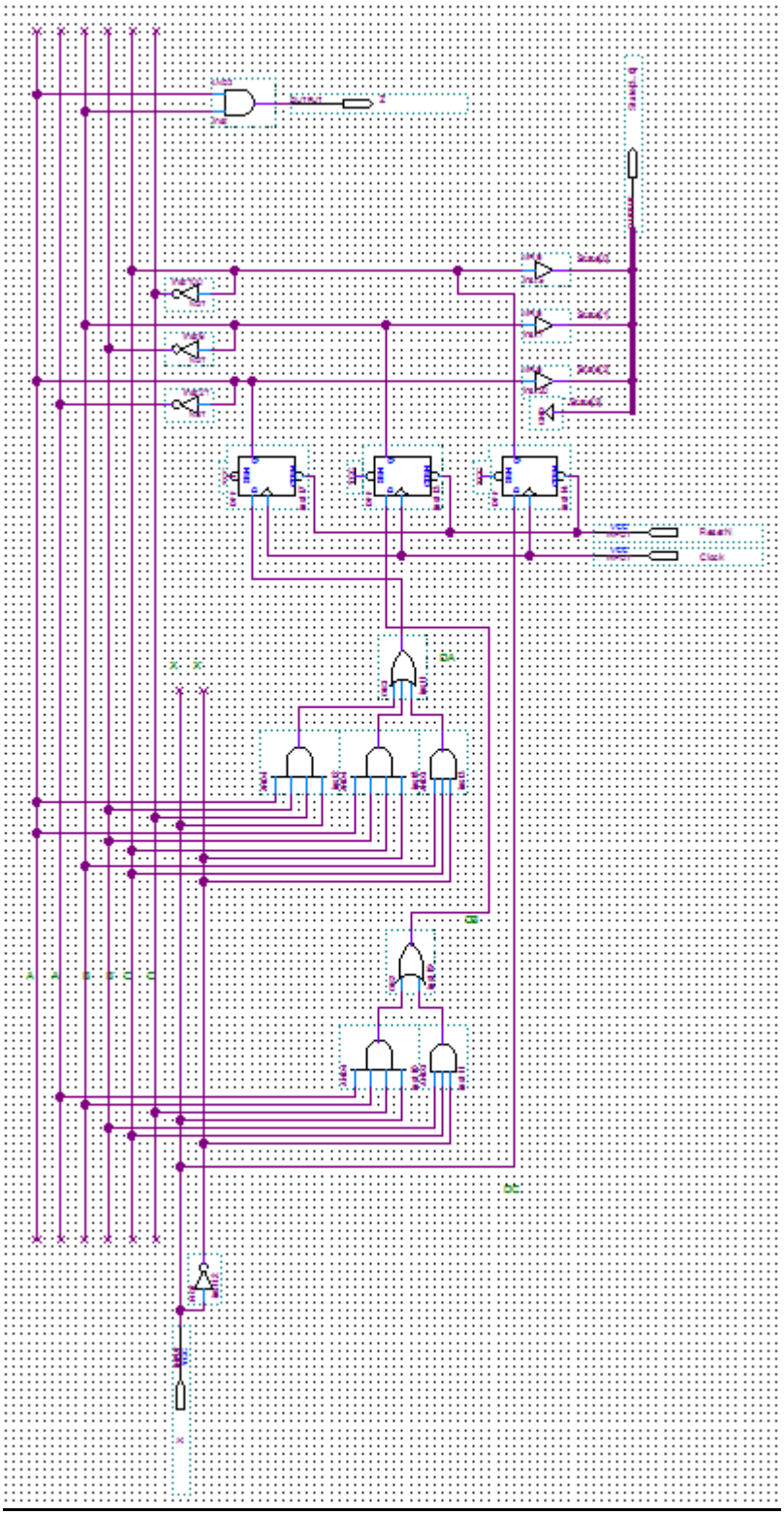
Kmap Da	CX	C'X	C'X'	CX'
AB	DC			DC
A'B				1
A'B'				
AB'		1		1
Da = AB'C'X + AB'CX' + BCX'				

Kmap Db	CX	C'X	C'X'	CX'
AB	DC			DC
A'B		1		
A'B'				1
AB'				1
Db = A'BC'X + B'CX'				

Kmap Dc	CX	C'X	C'X'	CX'
AB	DC	1		DC
A'B	1	1		
A'B'	1	1		
AB'	1	1		
Dc = X				

Kmap Z	CX	C'X	C'X'	CX'
AB	DC	1	1	DC
A'B				
A'B'				
AB'				
Z = AB				

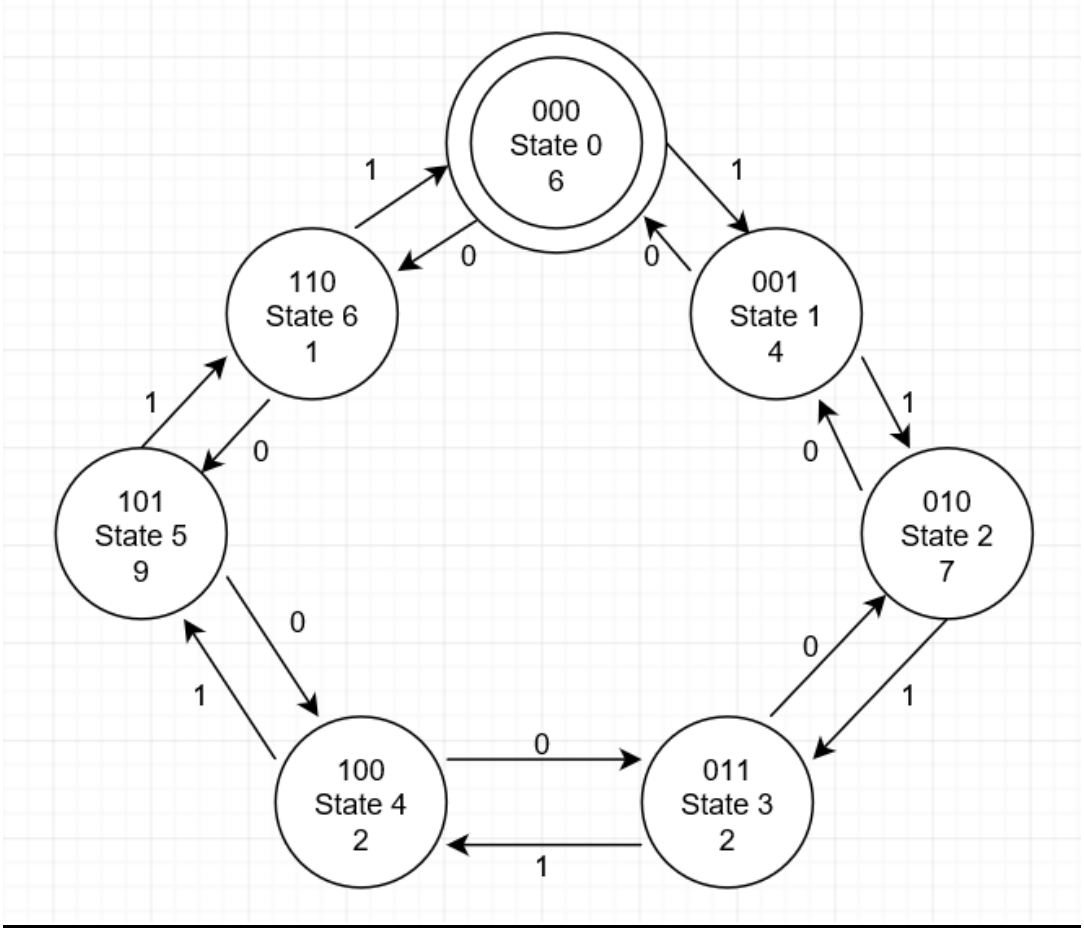
Here is the Block-Diagram for the circuit.



Part 4 – Phone Number Machine

In this part of the lab I was instructed to build a machine very similar to the 1492 circuit. Just like that circuit, this circuit progresses through a series of pre-programed numbers as long as the input is high. If the input is low it will progress in the opposite direction. I was responsible for creating the truth tables, kmaps, and circuit designs for this circuit. Please note, as in the previous circuit, the "DC" in truth tables and Kmaps simply means don't care.

Here is the State-Diagram for the circuit.



Here is the Truth Table for the circuit.

Inputs				Outputs							MinTerms						
State			X	Next State			Moore Outs				mDa	mDb	mDc	mZ3	mZ2	mZ1	mZ0
A	B	C		Da	Db	Dc	Z3	Z2	Z1	Z0							
0	0	0	0	1	1	0	0	1	1	0	A'B'C'X'	A'B'C'X'			A'B'C'X'	A'B'C'X'	
0	0	0	1	0	0	1	0	1	1	0			A'B'C'X		A'B'C'X	A'B'C'X	
0	0	1	0	0	0	0	0	1	0	0					A'B'CX'		
0	0	1	1	0	1	0	0	1	0	0		A'B'CX			A'B'CX		
0	1	0	0	0	0	1	0	1	1	1			A'BC'X'		A'BC'X'	A'BC'X'	A'BC'X'
0	1	0	1	0	1	1	0	1	1	1		A'BC'X	A'BC'X		A'BC'X	A'BC'X	A'BC'X
0	1	1	0	0	1	0	0	0	1	0		A'BCX'				A'BCX'	
0	1	1	1	1	0	0	0	0	1	0	A'BCX					A'BCX	
1	0	0	0	0	1	1	0	0	1	0		AB'C'X'	AB'C'X'			AB'C'X'	
1	0	0	1	1	0	1	0	0	1	0	AB'C'X		AB'C'X			AB'C'X	
1	0	1	0	1	0	0	1	0	0	1	AB'CX'			AB'CX'			AB'CX'
1	0	1	1	1	1	0	1	0	0	1	AB'CX	AB'CX		AB'CX			AB'CX
1	1	0	0	1	0	1	0	0	0	1	ABC'X'		ABC'X'				ABC'X'
1	1	0	1	0	0	0	0	0	0	1							ABC'X
1	1	1	0	0	0	0	0	0	0	0	DC	DC	DC	DC	DC	DC	DC
1	1	1	1	0	0	0	0	0	0	0	DC	DC	DC	DC	DC	DC	DC

Here are the Kmaps used to create the circuit.

Kmap Da	CX	C'X	C'X'	CX'
AB	DC		1	DC
A'B	1			
A'B'			1	
AB'	1	1		1

Da = BCX + AB'X + ABX' + ACX' + A'B'C'X'

Kmap Db	CX	C'X	C'X'	CX'
AB	DC			DC
A'B		1		1
A'B'	1		1	
AB'	1		1	

Db = B'CX + B'C'X' + BCX' + A'BC'X

Kmap Dc	CX	C'X	C'X'	CX'
AB	DC		1	DC
A'B		1	1	
A'B'		1		
AB'		1	1	

Dc = AB'C' + A'C'X + BC'X'

Kmap Z0	CX	C'X	C'X'	CX'
AB	DC	1	1	DC
A'B		1	1	
A'B'				
AB'	1			1

Z0 = ACX + ACX' + BC'

Kmap Z1	CX	C'X	C'X'	CX'
AB	DC			DC
A'B	1	1	1	1
A'B'		1	1	
AB'		1	1	

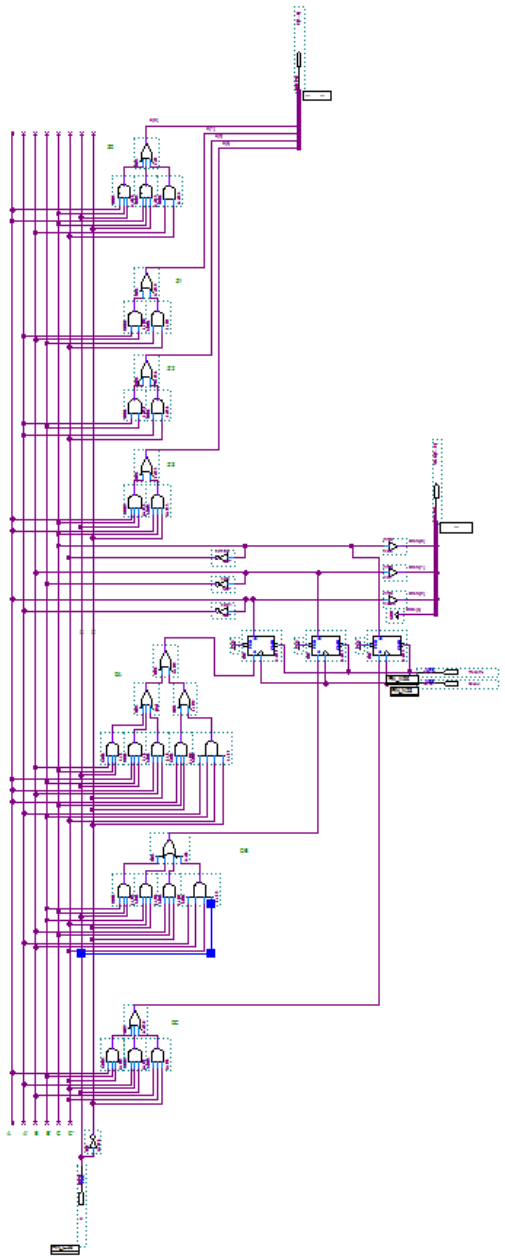
Z1 = A'B + B'C'

Kmap Z2	CX	C'X	C'X'	CX'
AB	DC			DC
A'B		1	1	
A'B'	1	1	1	1
AB'				

Z2 = A'B' + A'C'

Kmap Z3	CX	C'X	C'X'	CX'
AB	DC			DC
A'B				
A'B'				
AB'	1			1
Z3 = ACX + ACX'				

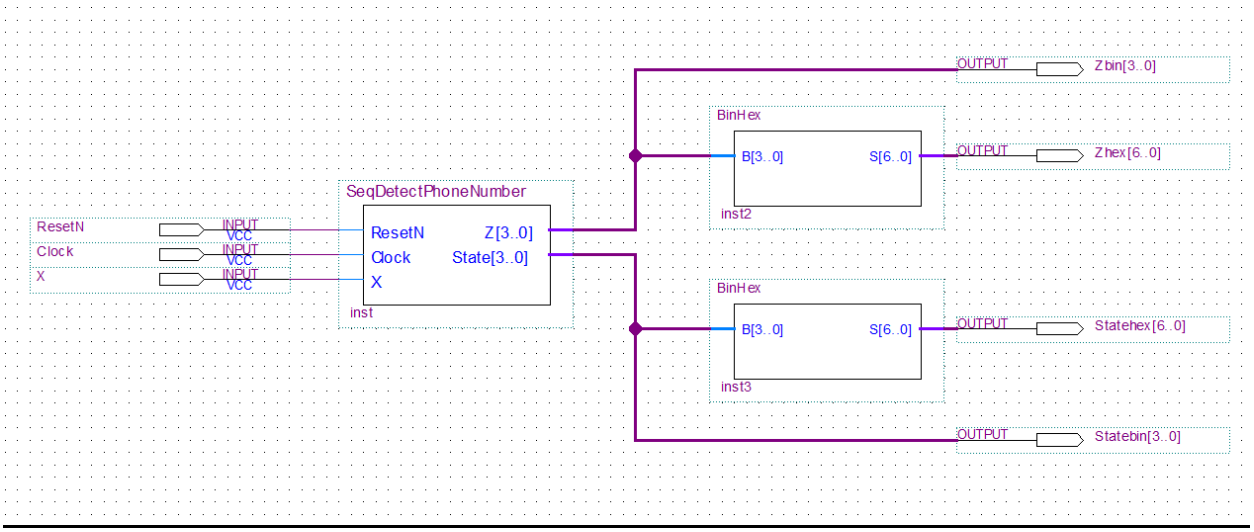
Here is the Block-Diagram for the circuit.



Part 5 – Phone Number Machine with Hex Display

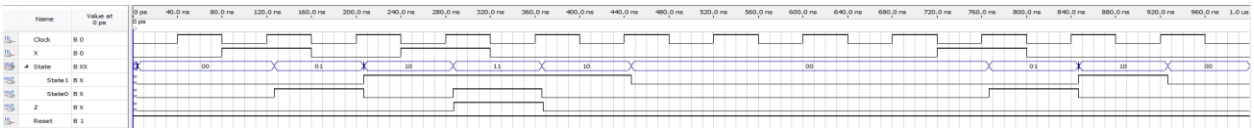
In this section of the lab, I was supposed to take my recently finished Phone Number machine and attach a couple Hex display modules to it. This will make the outputs more readable and easier to test.

Here is the Block-Diagram for the circuit.

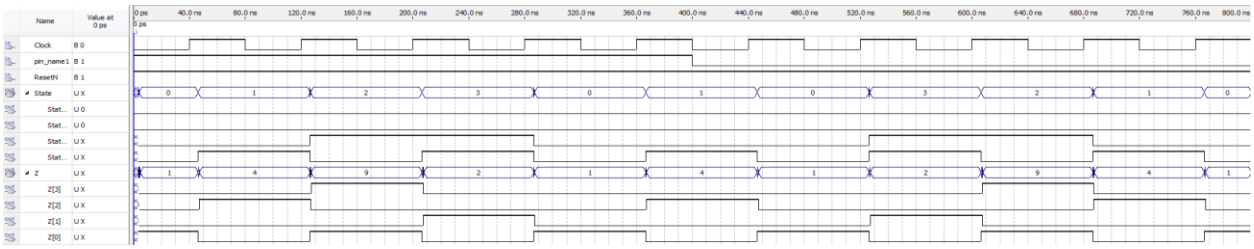


Testing

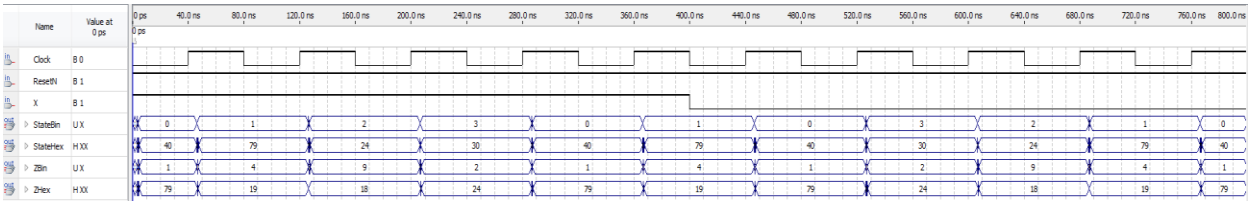
When testing the Sequence Detector 101 I encountered no problems and it performed as expected for every single input combination.



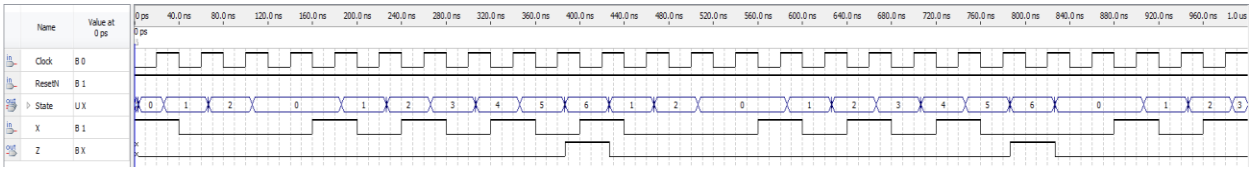
When testing the State Machine 1492 circuit I encountered no problems and it performed as expected for every single input combination.



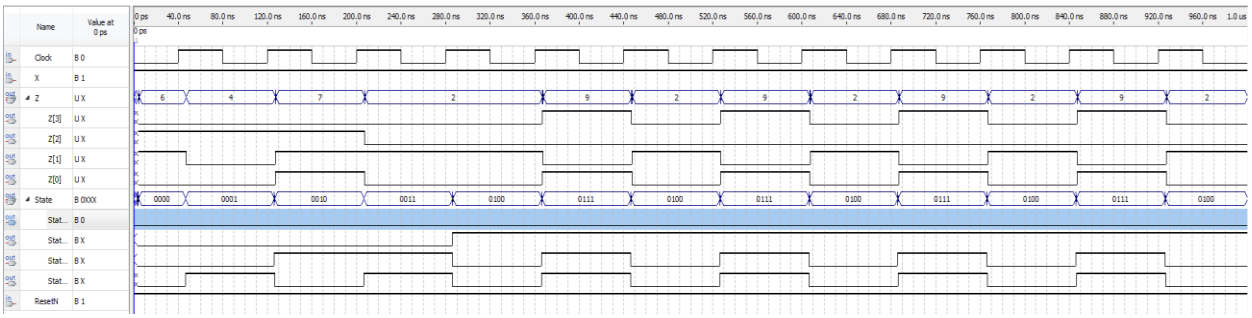
When testing the State Machine 1492 with Hex display I encountered no problems and it performed as expected for every single input combination.



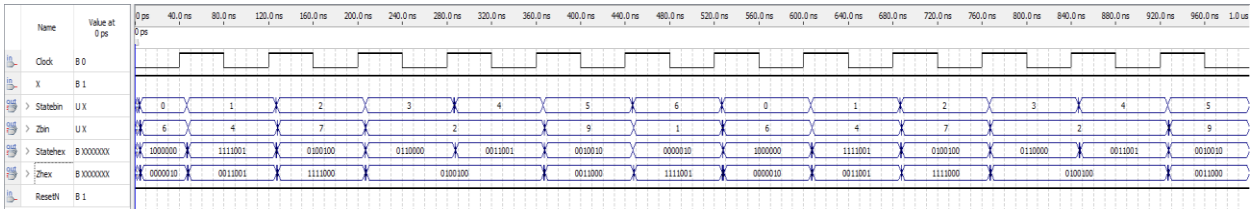
When testing the Sequence Detector 42 I encountered no problems and it performed as expected for every single input combination.



When testing the Phone Number Machine I encountered one crippling problem. One of my circuit equations was wrong. I did not catch this until very late in the design of the circuit. However, once this problem was resolved the circuit performed as expected for every single input combination.



When testing the Phone Number Machine with Hex display I encountered no problems and it performed as expected for every single input combination.



Conclusion

In this lab, I learned how to combine combinational logic machines and memory circuits into state machines. I also learned how to convert my equations from Kmaps into Block Diagrams for the circuits. Previously I was only capable of doing that with Verilog. In addition, I learned how to create a new type of diagram. These State Diagrams allow you to see how a circuit should perform when it is in certain states depending upon the input it receives next. If I was to perform this lab again I would double check my circuit equations before building the actual circuit. As that would've saved me a couple hours of my time. Despite this, I still enjoyed this lab a great deal and was amazed to see my Truth Tables, Kmaps, and State Diagrams actually turn into working circuits.