ECEN 429: Introduction to Digital Systems Design Laboratory

North Carolina Agricultural and Technical State University

Department of Electrical and Computer Engineering

Freddie Boadu (Reporter)

Mike Umelo (Partner)

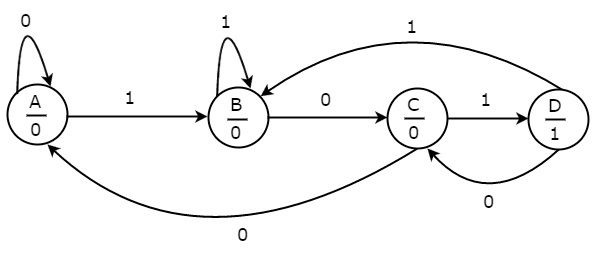
March 11, 2020

Pre Lab #6

**Introduction**

The prelab for lab 6 is focused around using state machines in VHDL. A finite state machine is a machine that allows for input and output for each state. There are two types of Fsm’s that will be used, moore and mealy. The fsm uses a clock and reset to move from state to state. In VHDL, it typically consists of 2 process blocks. One for the actual states and one for if a reset is hit.

**Background, Design Solution and Results**

The prelab provided us with some code for an fsm to learn how they work. The questions revolve around the state machine given: 

Problem A: The inputs on the moore machine are 2. The inputs can be 0 or 1 to represent off and on.

Problem B: The outputs on the moore machine are 2. The outputs would be 0 or 1 to represent the output at each state.

Problem C: The moore machine has 4 states.

Problem D: The process blocks need for coding this machine would be two. A process control block is needed for a reset button and another for when the model is moving state to state.

Problem E:

Entity moore is

Port (clk,input,reset: in STD\_LOGIC;

Output: out STD\_LOGIC);

End moore;

Architecture bev of moore is

Process(clk,reset)

Begin (reset=’1’) then

Current\_state<=”00”;

Elsif (clk’ event and clock= ‘1’) then

Current\_state<=Next\_state;

End if;

End process;

Process(current\_state)

Next\_state<=state0;

Begin

Case Current\_state is

When”00”=>

If (input= ‘1’) then

Next\_state<=”01”;

Else

Next\_state<=”00”;

When”01”=>

If (input= ‘1’) then

Next\_state<=”01”;

Else

Next\_state<=”10”;

When”10”=>

If (input= ‘1’) then

Next\_state<=”11”;

Else

Next\_state<=”00”;

When”11”=>

If (input= ‘1’) then

Next\_state<=”01”;

Else

Next\_state<=”10”;

When others=> next\_state<= “00”;

End case;

End process;

**Conclusion**

After doing the prelab, I understand FSM a lot better. I am able to understand the inputs, outputs, process blocks, and states it will take to make a moore fsm in particular.