# REQUIREMENTS NOT MET

N/A

# VIDEO FILE LINK

https://youtu.be/L6-vcZsUQaY?si=u0EfzrGOJMk5BzOp

# PROBLEMS ENCOUNTERED

I had trouble connecting my DE10, I tried changing ports, getting an extender, and using the extender ports, as well as using a different DE10, but I still have the same problem. Additionally, when filming the demo video there were some bugs, despite simulation working. However, when looping through, it does show all the correct states

# FUTURE WORK/APPLICATIONS

VDHL helps describe the structure and behavior of circuits, this lab was a good foundation for VDHL to apply to more complicated logic later on eventually. Additionally, ASM is needed to design the path of digital systems and visually show logic and control decisions, this lab was a good basic application of it to understand before implementing it in a more complicated way in the future.

# PRE-LAB QUESTIONS OR EXERCISES

N/A

# PRE-LAB REQUIREMENTS (Design, Schematic, ASM Chart, VHDL, etc.)

Each section of the pre-lab requirements should be completed separately, and in order. Include each of the following items in order. Note that some of these items will not apply to every lab. Anything scanned or copied ***must be clear and legible***.

- Logic equations. (Note that logic equations do not contain activation levels.)
- **All** tables and figures should have captions with Figure/Table numbers and a description of for which part of the lab it references, e.g., ***Table 3****: Truth Table for Part B*.
- Truth tables and voltage tables (and/or next-state truth tables).
- When applicable, include Karnaugh Maps (i.e., K-Maps).
- Include hand-drawn circuits (when required). Label all input and output activation-levels and intermediate equations in the circuits.
- Include screenshots of the BDF designs of circuits.
  - Label all input and output activation-levels, i.e., use _L suffix for active-low signals and no suffix for active-high signals. **Add chip and pin numbers to any schematic that will be constructed.**
  - Images should be large enough so that inputs, outputs, labels, and parts are clearly visible and distinguishable to any reader.
  - Each BDF should have the following info on the top left corner (similar to the top right of this page):

    *Last Name, First Name*
    *Lab #, Part #*
    *Class #*
    *PI Name:*
    *Description:* (short description of what is to be accomplished in the design; perhaps an equation)

  - In Windows, I use the ***Snipping Tool***, which is now built into Windows. Just type "snip" in the Windows search box and then select ***Snipping Tool***.
- When necessary, include ASM Charts. These can be hand-drawn, but clear and legible. We recommend that you use resources like https://www.draw.io/ to create computer-generated ASMs.
- Truth tables or next-state truth tables should have the following characteristics.
  - Can be either typed or hand-written and scanned (must be clear and legible)
  - Must be in **counting order** (i.e., inputs of 000, 001, 010, 011, …, 111)
  - Clearly distinguish inputs from outputs (see the example below that uses a thick line)
  - If you are designing a state machine or a controller, clearly indicate and separate signal values both before the clock and after the clock (i.e., Q1 and $Q1^+$, respectively)
  - Tip: divide rows into consecutive groups of 4 (or 2 or 8) to make it easier for both you and your PI to read.
  - **Example**

| A | B | C | Y |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 |

Table: Caption for above table. It should reference the part of the lab, e.g., "Table 3: Truth Table for Part B."

- Voltage tables should have the following characteristics.
  - Must be in counting order (i.e., inputs of LLL, LLH, LHL, … , HHH)
  - Use similar formatting to truth tables (described above)

- **Example**

| A(H) | B(H) | C(L) | Y(L) |
|---|---|---|---|
| L | L | L | H |
| L | L | H | H |
| L | H | L | L |
| L | H | H | H |
| H | L | L | H |
| H | L | H | H |
| H | H | L | H |
| H | H | H | L |

- Include **meaningfully annotated** functional simulations.
  - Using the grouping tool, ***group as many signals together as possible*** (when it makes sense to group them)! If you are simulating a basic logic equation, group all the inputs together. If you are simulating a circuit that includes MSI elements, group signals of the form $X_{N-0}$. The most-significant bit should appear first, ending with the least-significant bit. If you are simulating an ALU, group the buses together as just described.
  - Not every row of your voltage table must be annotated in the waveform simulation, but your choices of rows that you annotate must be ***encompassing***
  - If you are designing a state machine or a controller, your CLK signal should appear ***at the top*** of your inputs and outputs. The general order is CLK -> Reset -> state bits -> inputs -> outputs.
  - ***Tip***: Use Microsoft Paint to annotate your waveforms. An alternative is to print out the waveforms, annotate them by hand, then scan and upload
  - ***Hint***: Consider a truth table where the output is true in significantly less cases than it is false (or vice versa). If the output signal is active-high, it would be wise to annotate only the cases where the output voltage is HIGH.
- Include every line of VHDL programs, including both ***architecture*** and ***behavior*** sections.
- Include every line of any **MIF** files. If these are associated with assembly language programs, you can either put the assembly code as comments or separately include assembly language programs

1. State Machine Design

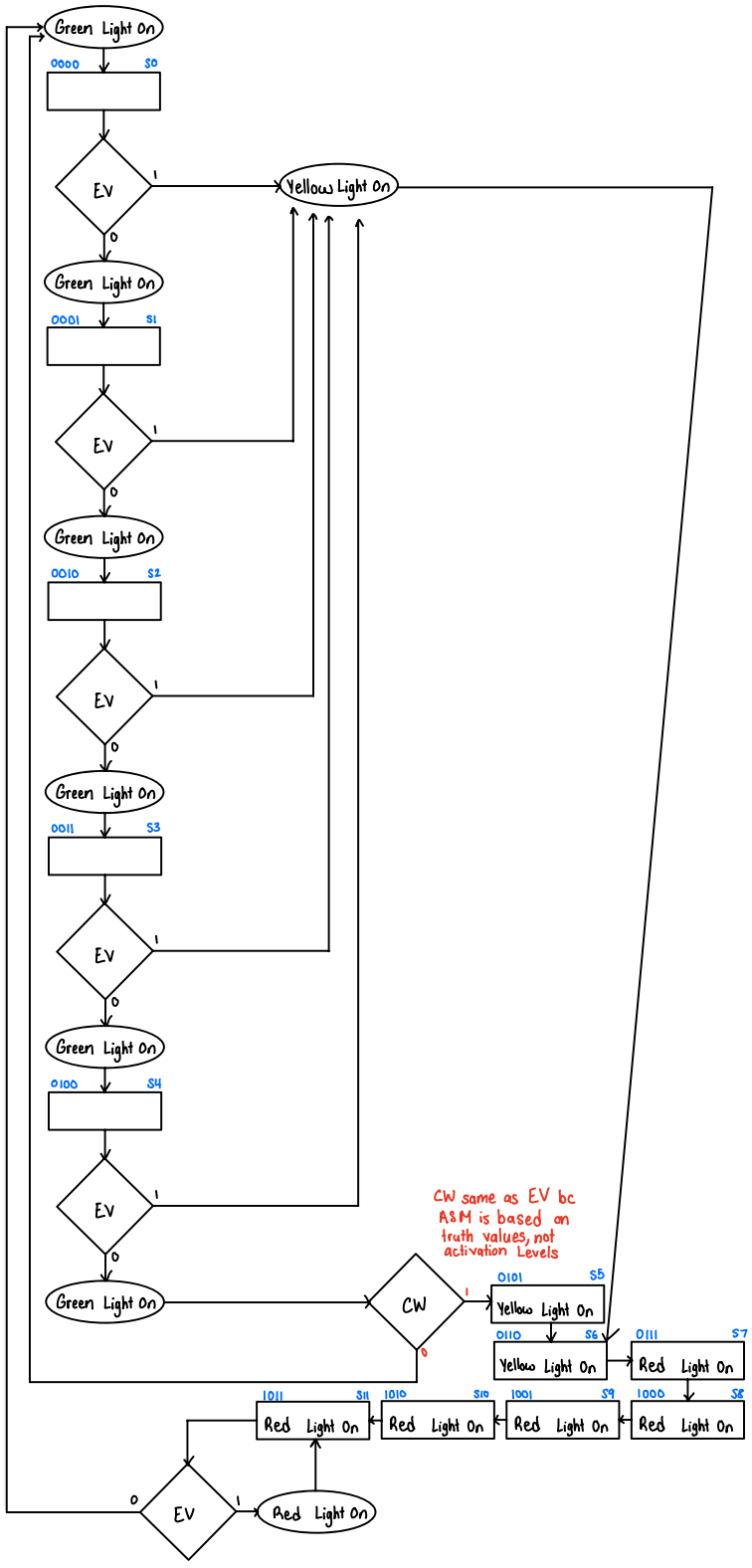


*Figure 1: Lab 5 Part 1.1 - ASM Chart for Traffic Controller*

| $Q_3$ | $Q_2$ | $Q_1$ | $Q_0$ | EV | CW_L | $Q_3^+$ | $Q_2^+$ | $Q_1^+$ | $Q_0^+$ | Gr | Ye | Re |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | - | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 0 | - | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 0 | 0 | 0 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 0 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 0 | - | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 | - | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 | - | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 0 | - | 0 | 0 | 1 | 1 | 1 | 0 | 0 |
| 0 | 0 | 1 | 0 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 0 | - | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 | - | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | - | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | - | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | - | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 1 | - | - | 0 | 1 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | - | - | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | - | - | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | - | - | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | - | - | 0 | 1 | 1 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 1 | - | - | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | - | - | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | - | - | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | - | - | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | - | - | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | - | - | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | - | - | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 | - | - | 1 | 0 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | - | - | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | - | - | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | - | - | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 1 | - | - | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | - | - | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | - | - | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | - | - | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | - | - | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | - | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 | - | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | - | 1 | 0 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | - | 1 | 0 | 1 | 1 | 0 | 0 | 1 |

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 0 | 0 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 0 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 0 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 0 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 1 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 1 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 1 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 0 | 1 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 0 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 0 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 0 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 0 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 1 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 1 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 1 | - | - | X | X | X | X | X | X | X |
| 1 | 1 | 1 | 1 | - | - | X | X | X | X | X | X | X |

*Figure 2: Lab 5 Part 1.2 - Truth Table for Traffic Controller*

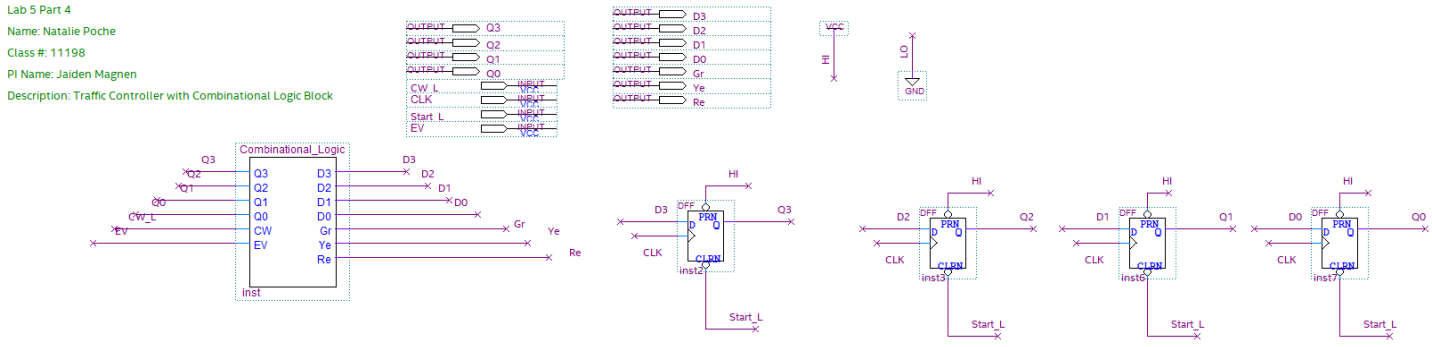


*Figure 3: Lab 5 Part 1.4 - BDF with Combinational Logic Block and DFF outputs for Traffic Controller*

```vhdl
-- Lab 5
-- Name: Natalie Poche
-- Class#: 11198
-- PI Name: Jaiden Magnan
-- Due: March 27, 2025

library ieee;
use ieee.std_logic_1164.all;

entity Combinational_Logic is port ( -- Make sure name is same as file name.
    -- State Inputs
    Q3:   in std_logic;
    Q2:   in std_logic;
    Q1:   in std_logic;
    Q0:   in std_logic;

    -- Input Checks
    CW_L:    in std_logic;
    EV:      in std_logic;

    -- Next State Outputs
    D3:   out std_logic;
    D2:   out std_logic;
    D1:   out std_logic;
    D0:   out std_logic;

    -- Light Outputs
    Gr:   out std_logic;
    Ye:   out std_logic;
    Re:   out std_logic -- Last output doesn't need a semi-colon, the end parenthesis with semi-colon takes care of it
    );
    end Combinational_Logic;

architecture behavior of Combinational_Logic is
signal CW: std_logic;
    begin
    CW <= not CW_L;
    -- ((not Q3) and (not Q2) and (not Q1) and (not Q0) and (not EV) and (not CW)) or

    D3 <= ((not Q3) and (    Q2) and (    Q1) and (    Q0) and (not EV) and (not CW)) or
          ((not Q3) and (    Q2) and (    Q1) and (    Q0) and (not EV) and (    CW)) or
          ((not Q3) and (    Q2) and (    Q1) and (    Q0) and (    EV) and (not CW)) or
          ((not Q3) and (    Q2) and (    Q1) and (    Q0) and (    EV) and (    CW)) or -- 1

          ((    Q3) and (not Q2) and (not Q1) and (not Q0) and (not EV) and (not CW)) or
          ((    Q3) and (not Q2) and (not Q1) and (not Q0) and (not EV) and (    CW)) or
          ((    Q3) and (not Q2) and (not Q1) and (not Q0) and (    EV) and (not CW)) or
          ((    Q3) and (not Q2) and (not Q1) and (not Q0) and (    EV) and (    CW)) or -- 1
```

*Figure 4: Lab 5 Part 1.3 - VHDL Code Snippet for Traffic Controller Logic*

**EEL 3701C: Digital Logic & Computer Systems**
Revision **0**
Lab **5** Report: **State Machine Example: Traffic Light Controller**

Poche, Natalie
Class #: 11198
PI Name: Jaiden Magnan
March 27, 2025

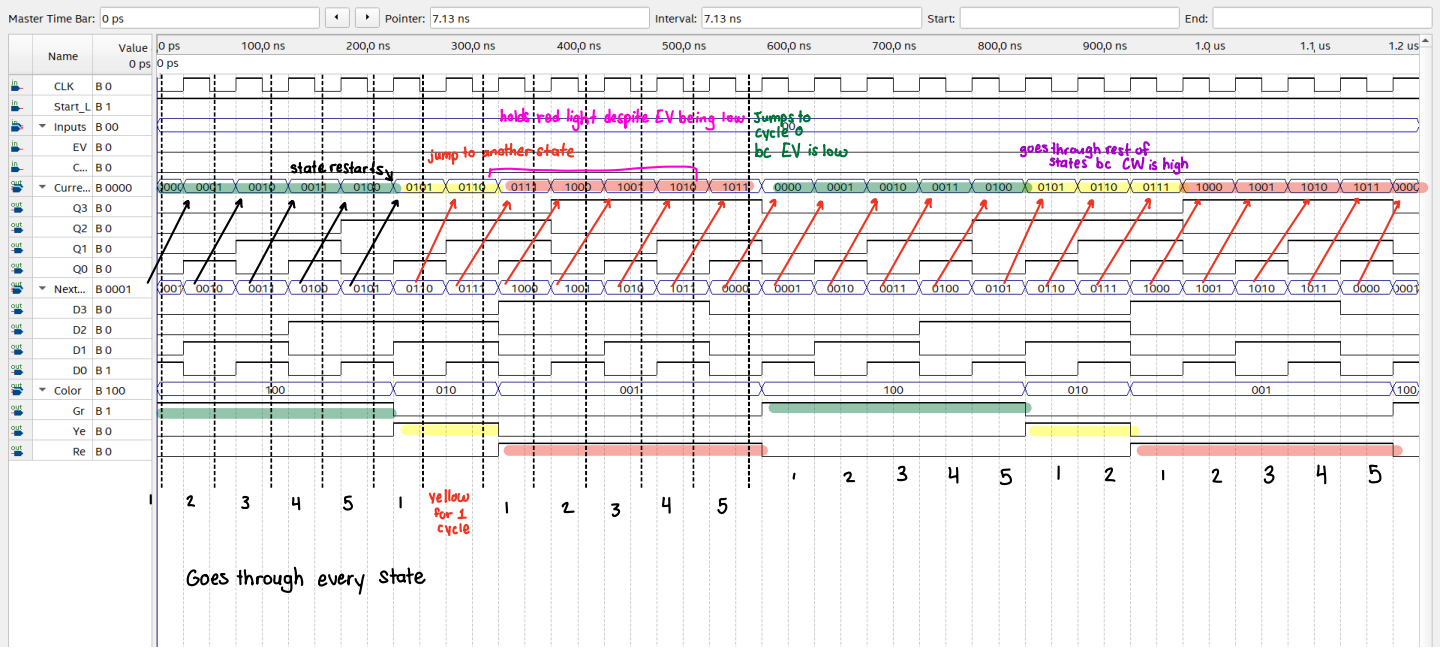


*Figure 5: Lab 5 Part 1.5 - Functional Simulation of Traffic Controller*

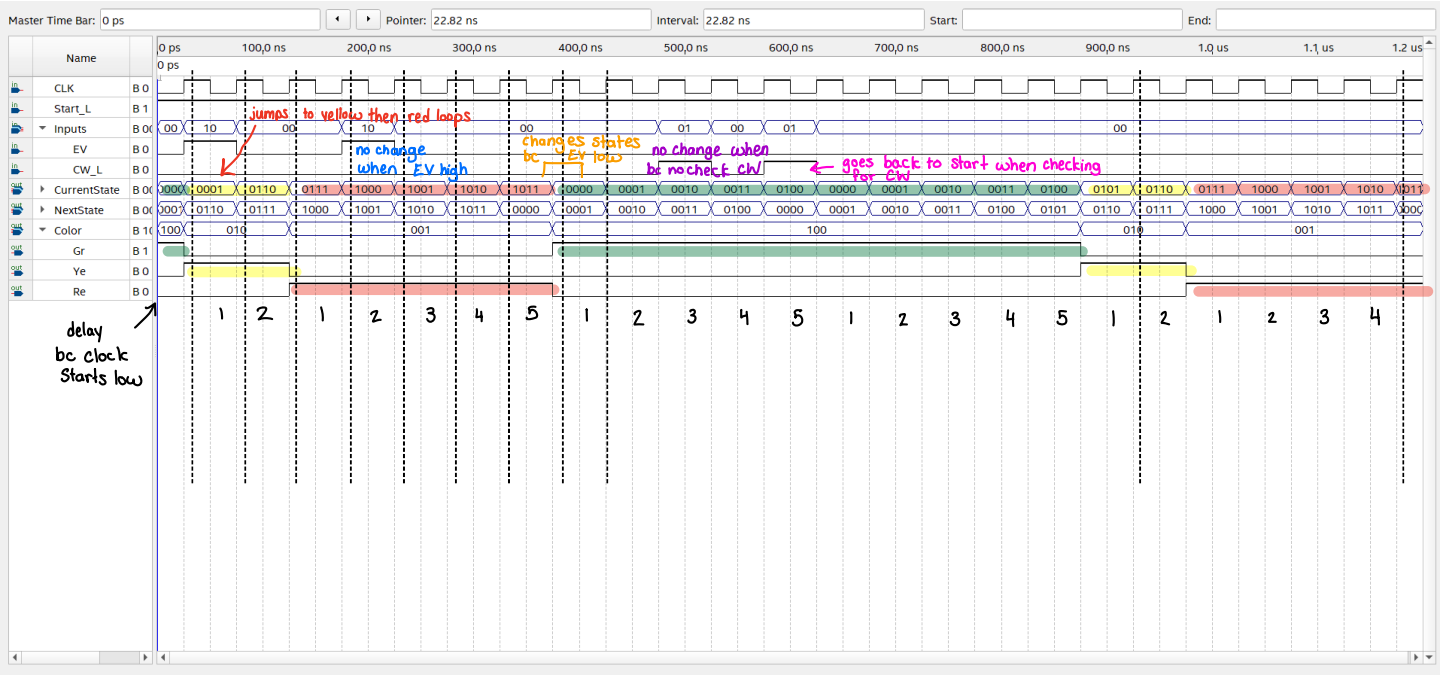


*Figure 6: Lab 5 Part 1.5 - Functional Simulation of Traffic Controller*