Advanced Traffic Light Controller

Matthew Tomlinson

Faculty Mentor: Dr. Puteri Megat Hamari

ECET Department, Minnesota State University, Mankato

BACKGROUND

The idea behind this design is to create a more efficient traffic light controller that can control multiple intersections jointly. By controlling multiple intersections with unity, overall traffic flow can be improved. The implementation in this system uses delays between lights at different intersections to help improve efficiency. It was designed using the Moore state machine model. Seen below in 'Figures 1, 2 and 3' are the state machine, truth table, and logic equations. The inputs are as follows: DA, DB, and DC are input blocks leading to D-FlipFlops, X1-X8 are outputs going to the traffic lights.

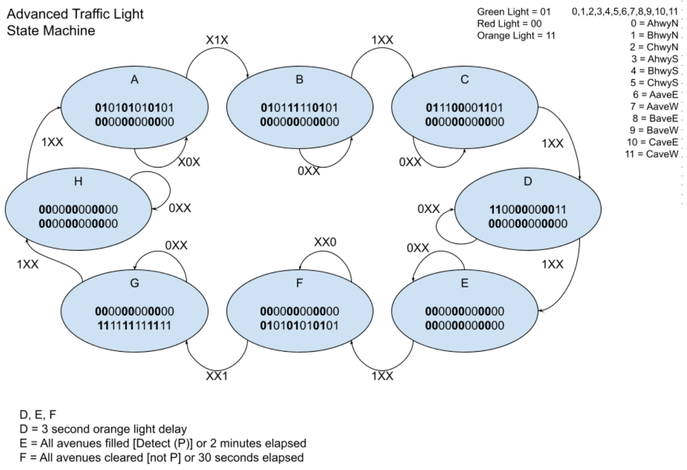

Figure 1: Moore state machine logic diagram

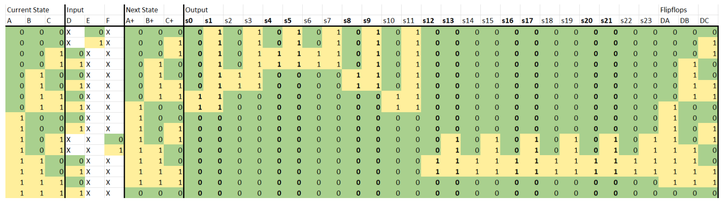

Figure 2: Moore state machine truth table

DA = AB' + AC' + AD' + A'BCD
DB = BC' + BD' + A'B'CD + AB'CF
DC = AC'D + A'CD' + BCD' + BC'D + AB'CF' + A'B'C'E
X1 = A'BC
X2 = A'
X3 = A'BC'
X4 = A'B'+A'C'
X5 = A'B'C
X6 = A'B'
X7 = ABC'
X8 = AB'C+ABC'

Figure 3: Equations from Moore state machine truth table

SYSTEM DESIGN

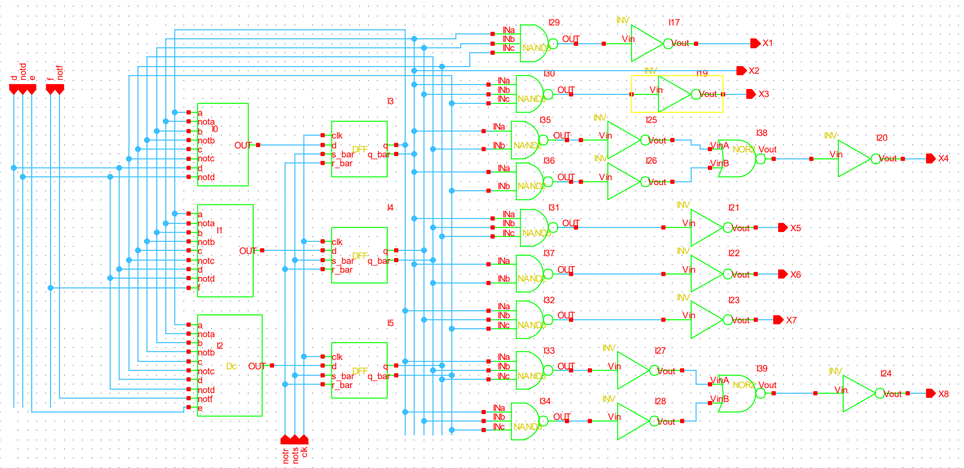

Figure 4: System level schematic diagram

As seen below, states A, B, and C are the top three waveforms. These combine to three bits to decide which state the logic is in out of a total of 8 states. The next 8 waveforms are the output waveforms, they go directly to the traffic lights to tell it which color of light needs to be on or off. Lastly, the three bottom waveforms are the system input bits, these bits tell the system when to move onto the next state in the Moore state diagram.

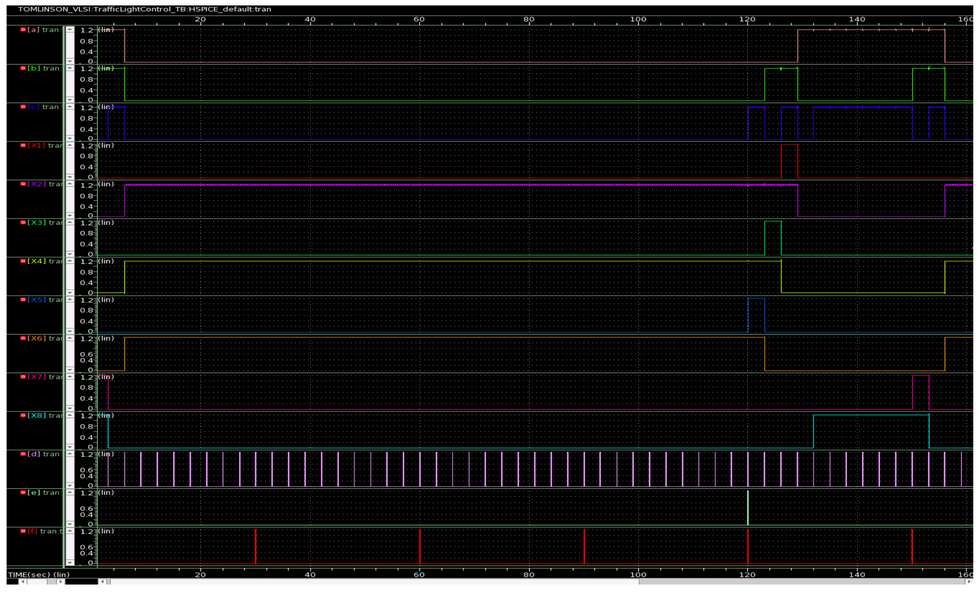

Figure 5: System level testbench

REFERENCES

Y. Wiseman, "Conceptual design of intelligent traffic light ... - CS@BIU," 2016. [Online]. Available: https://u.cs.biu.ac.il/~wisemay/ijca2016t.pdf. [Accessed: 01-Dec-2021].

S. Arneson, E. Reiter, and P. M. Hamari, rep. "Traffic Light Controller"

SYSTEM DESIGN

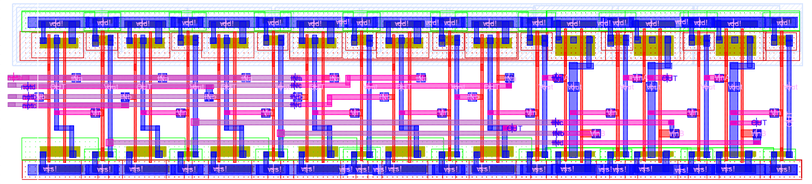

Figure 6: DA FlipFlop input layout

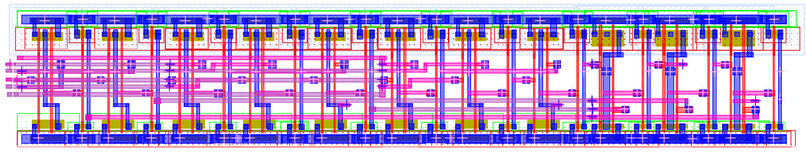

Figure 7: DB FlipFlop input layout

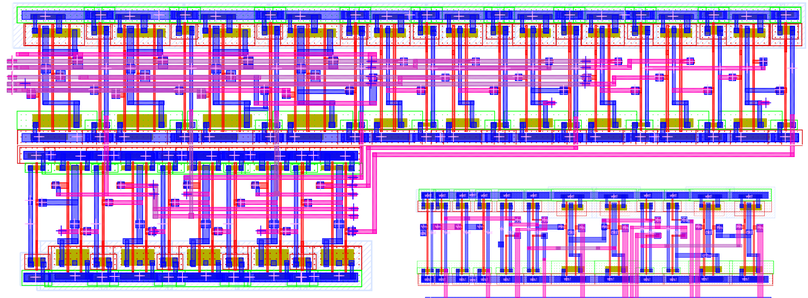

Figure 8: DC FlipFlop input layout

Figure 9: D-FlipFlop layout

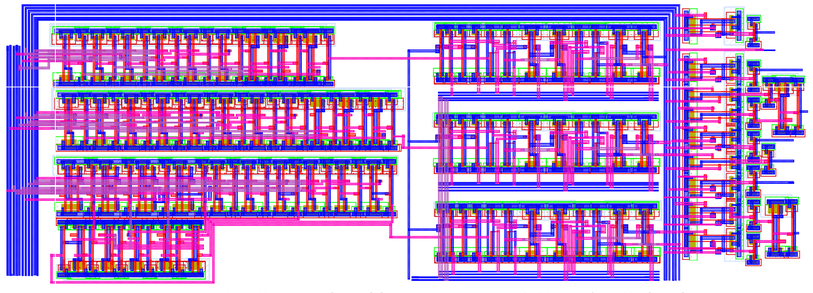

Figure 10: System level layout, passes DRC and LVS checks

ACKNOWLEDGEMENTS

I would like to lend thanks to Dr. Hamari for guiding me through the VLSI design process and providing her time for any questions I had regarding it outside of class. I'd also like to acknowledge the work of S. Arneson, and E. Reiter for their previous work done on the Traffic Light Controller concept for VLSI design.

CONTACT INFORMATION

Feel free to contact me at matthew.tomlinson-1@mnsu.edu with any questions or for more details about my project.