Andrew Friedman
Lab 4 Report
ECE 2031 CS
16 June 2023

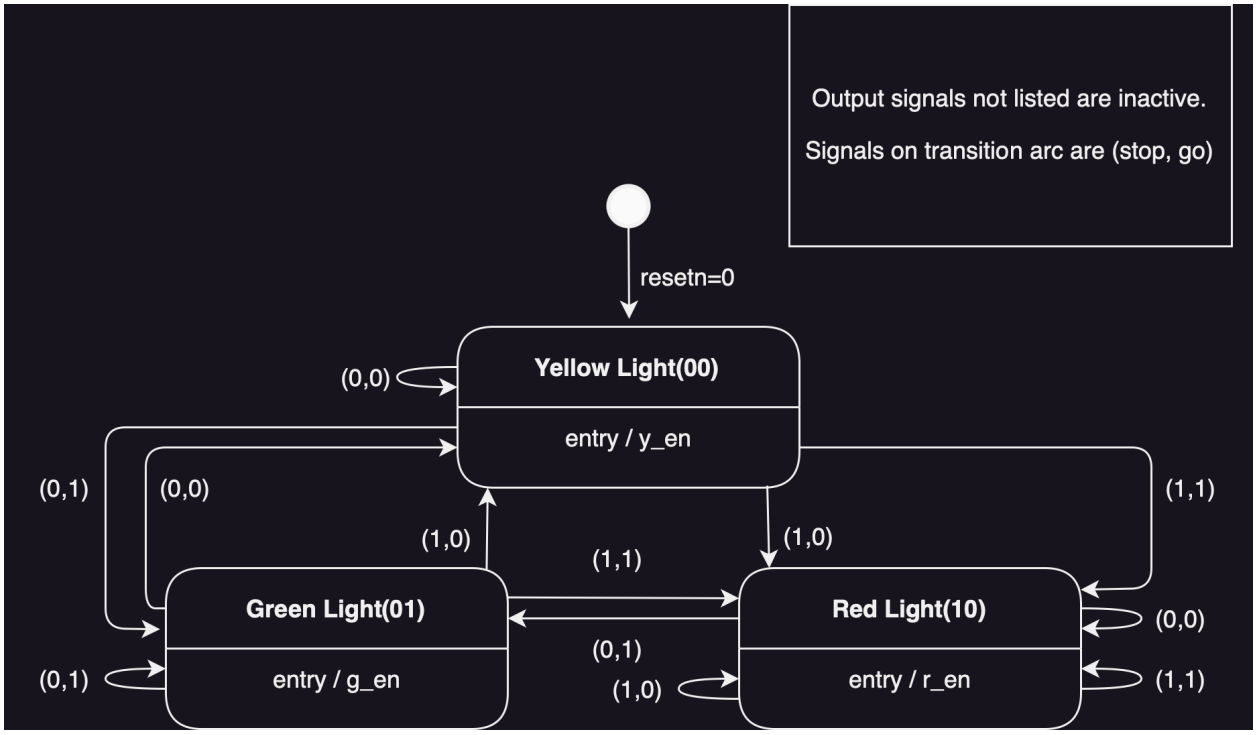

Figure 1. UML state diagram representing the logic for controlling the status lights of an industrial pick-and-place machine.

State Name	Q1	Q0		
Yellow Light	0	0	-	(your reset state, yellow light on)
Green Light	0	1	-	(the state with the green light on)
Red Light	1	0	-	(the state with the red light on)

Current State	Q1	Q0	STOP	GO	Next State	Q1+	Q0+	G	Y	R
Yellow Light	0	0	0	0	Yellow Light	0	0	0	1	0
Yellow Light	0	0	0	1	Green Light	0	1	0	1	0
Yellow Light	0	0	1	0	Red Light	1	0	0	1	0
Yellow Light	0	0	1	1	Red Light	1	0	0	1	0
Green Light	0	1	0	0	Yellow Light	0	0	1	0	0
Green Light	0	1	0	1	Green Light	0	1	1	0	0
Green Light	0	1	1	0	Yellow Light	0	0	1	0	0
Green Light	0	1	1	1	Red Light	1	0	1	0	0
Red Light	1	0	0	0	Red Light	1	0	0	0	1
Red Light	1	0	0	1	Green Light	0	1	0	0	1
Red Light	1	0	1	0	Red Light	1	0	0	0	1
Red Light	1	0	1	1	Red Light	1	0	0	0	1
unused	1	1	0	0	x	x	x	x	x	x
unused	1	1	0	1	x	x	x	x	x	x
unused	1	1	1	0	x	x	x	x	x	x
unused	1	1	1	1	x	x	x	x	x	x

Figure 2. Transition table for the state machine controlling the status lights of an industrial pick-and-place machine with the selected state encoding table.

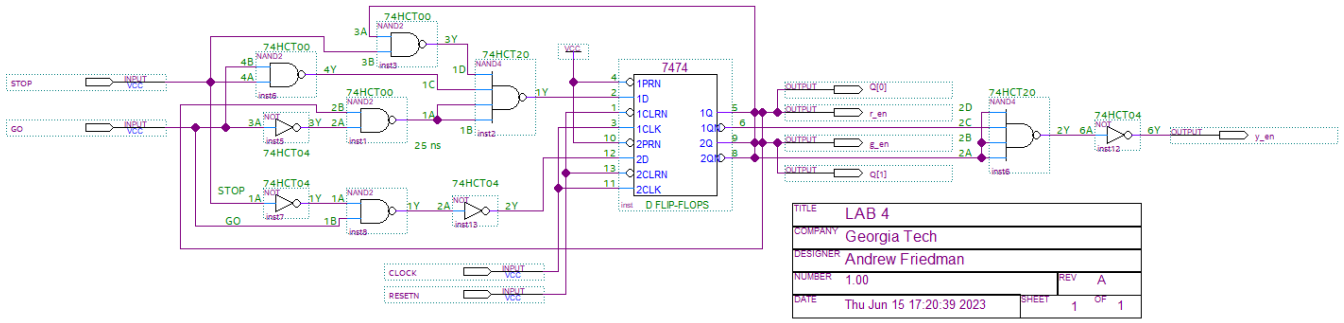

Figure 3. Gate-level schematic of the state machine implementation, including inputs and outputs connected to Quartus pin symbols and pin labels corresponding to DIP chips.

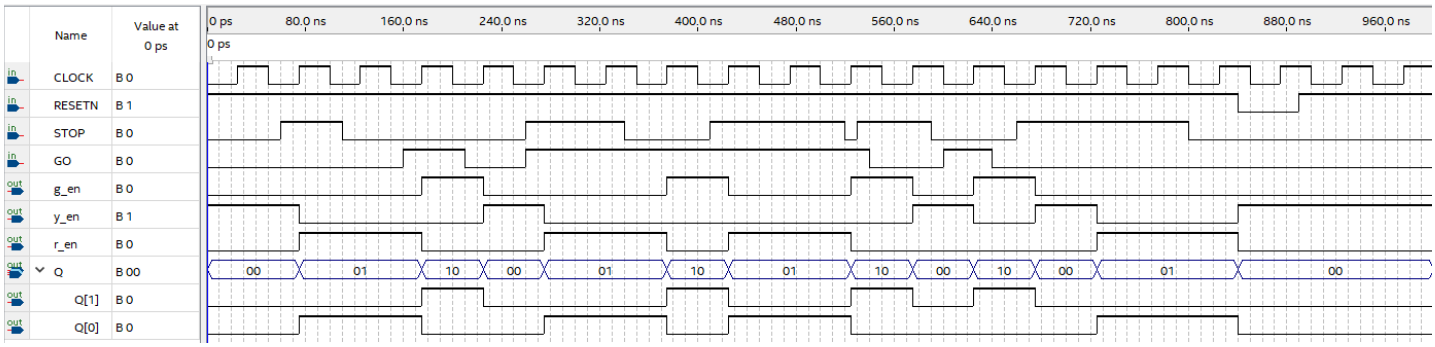

Figure 4. Circuit simulation in Quartus, showcasing input and output waveforms for the state machine in accordance with the created test scenario.

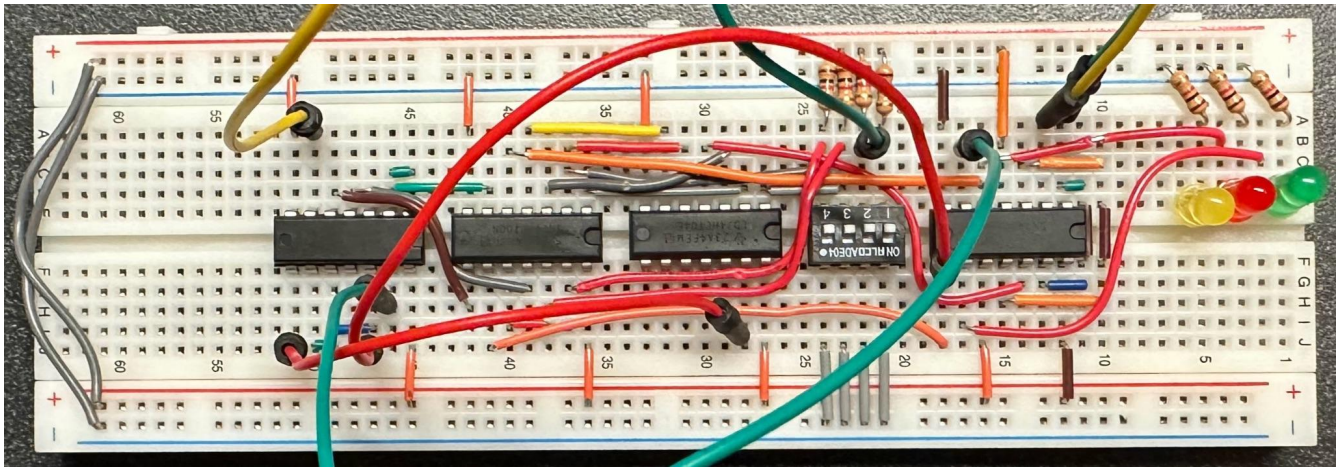

Figure 5. The finalized breadboard setup, showcasing the implemented state machine and associated wiring connections in detail.

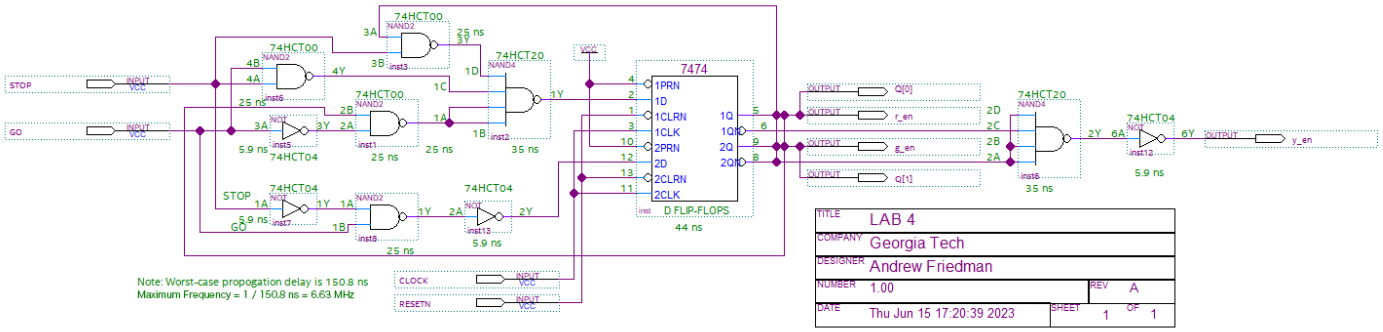

Figure 6. The fully annotated state machine schematic, including propagation delays and maximum clock frequency calculations.