**Lab 7: Sequential Circuit Design**

**Primary Objectives**

1. Analyze given circuit of sequence detector

2. Design and implement Moore sequence detector

3. Design and implement Mealy sequence detector

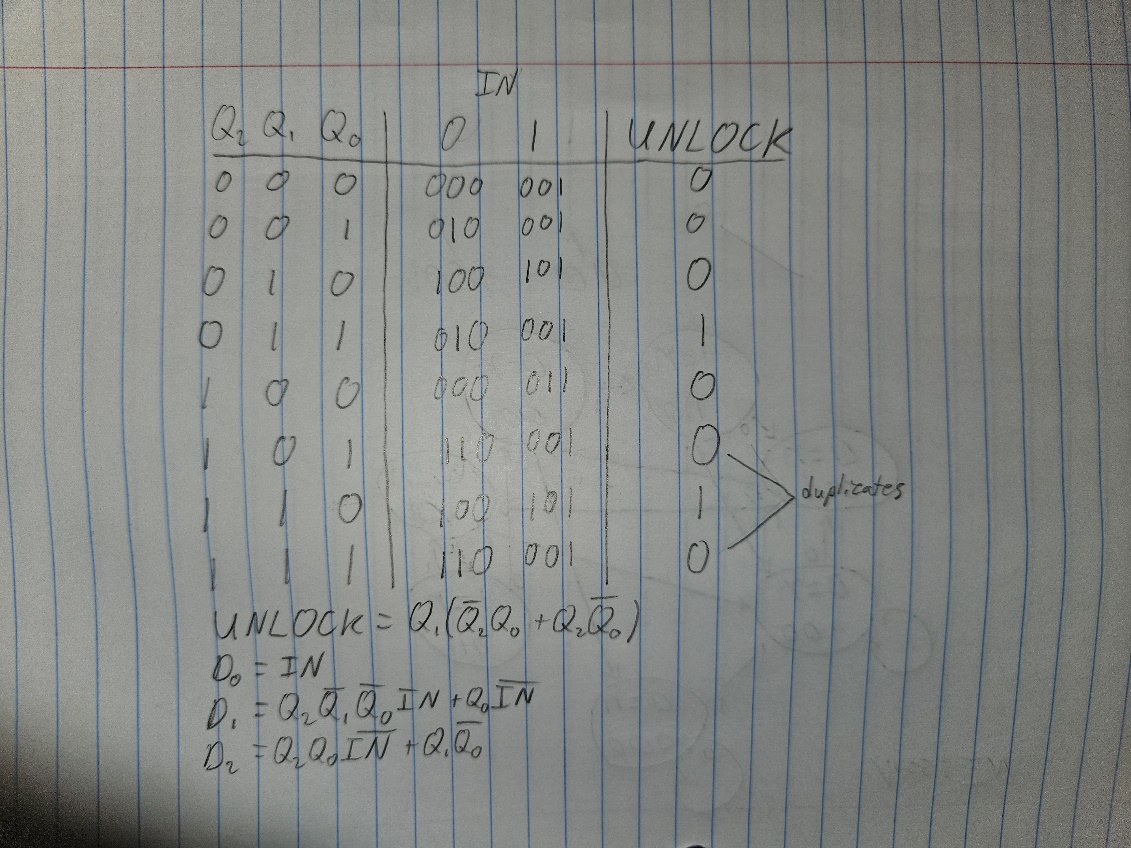
*Objective 1 Analysis*

For the first objective, I was given a picture of the below circuit to analyze. Because IN is not directly connect to the output UNLOCK, this is a Moore device.

A diagram of a circuit

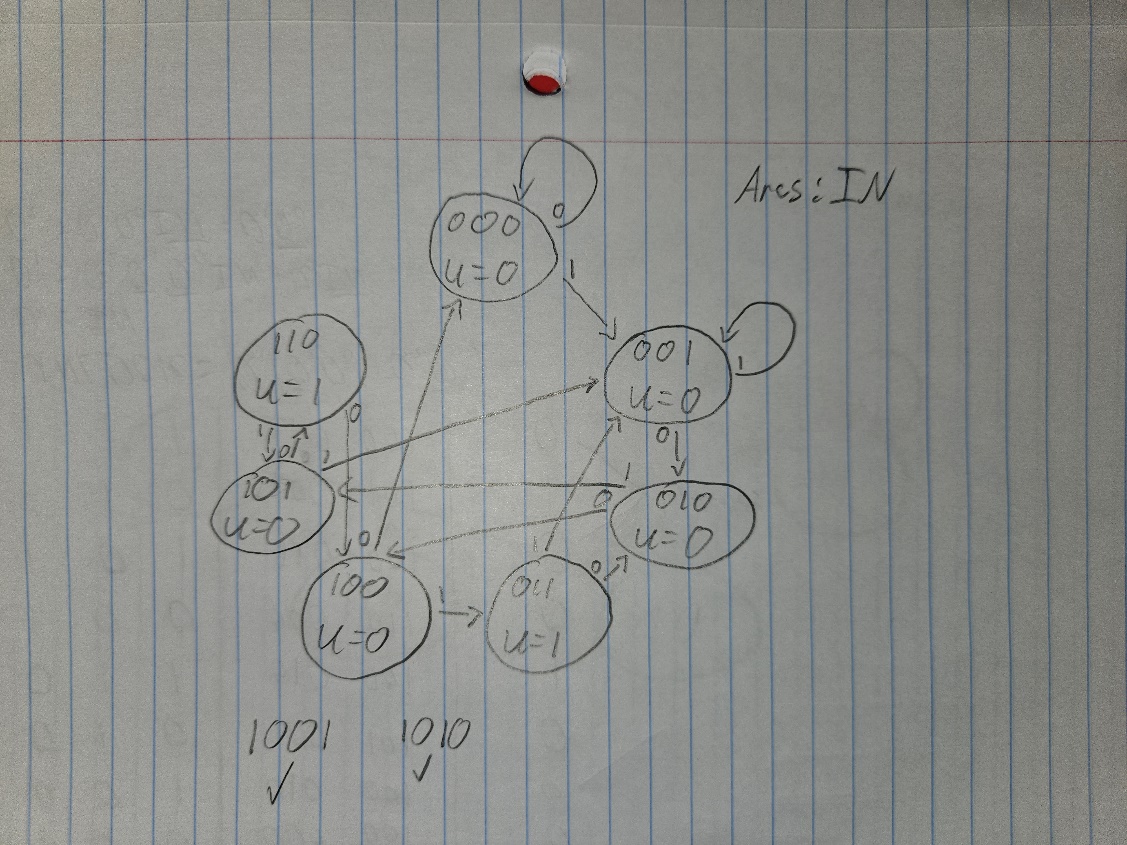
Description automatically generated Circuit Analysis

This circuit is a sequence detector that sets UNLOCK to 1 when the last four inputs from IN were either 1010 or 1001. This was determined by reversing the implementation/design process of constructing a circuit. Boolean expressions for the output and D-type flip-flops were determined by analyzing the logic gates used in the circuit. With these equations, state assignments and minimization could be done to determine that the circuit uses seven of which only two set UNLOCK to 1. State assignments and equations can be seen in the image below.



State Table Analysis

Using these state assignments, a state diagram could be constructed to track what sequential values of IN would set UNLOCK to 1 (see below).



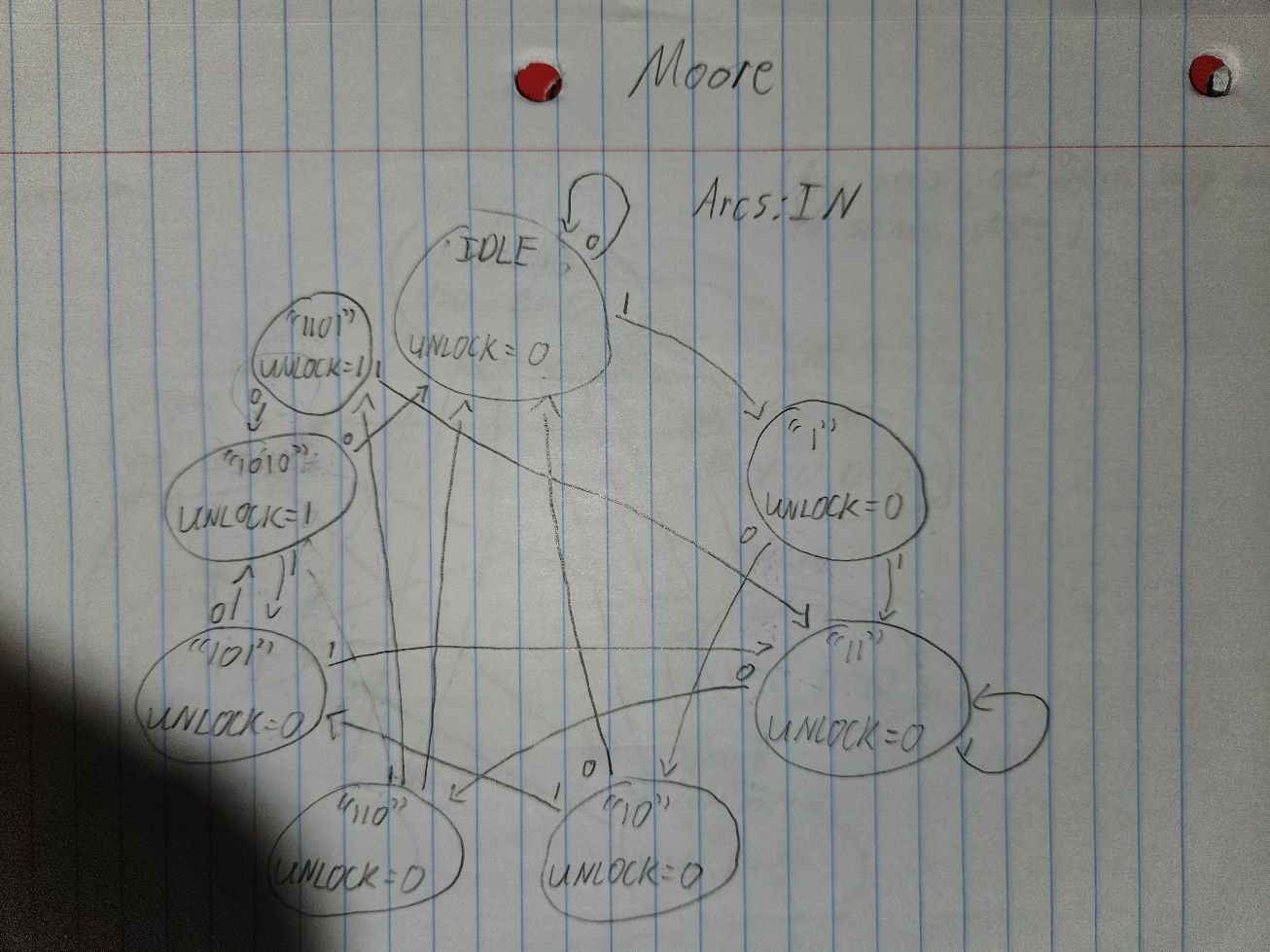
State Diagram Analysis

By tracing arcs, the sequences 1010 and 1001 were determined to be the correct numeric sequences to set UNLOCK to 1. The timing characteristics of the device were calculated to be as follows:

* Setup time, input to clock: 5
* Hold time, input to clock: 1
* Propagation delay, clock to output (min): 10
* Propagation delay, clock to output (max): 18
* Maximum clock rate of device: 1/15

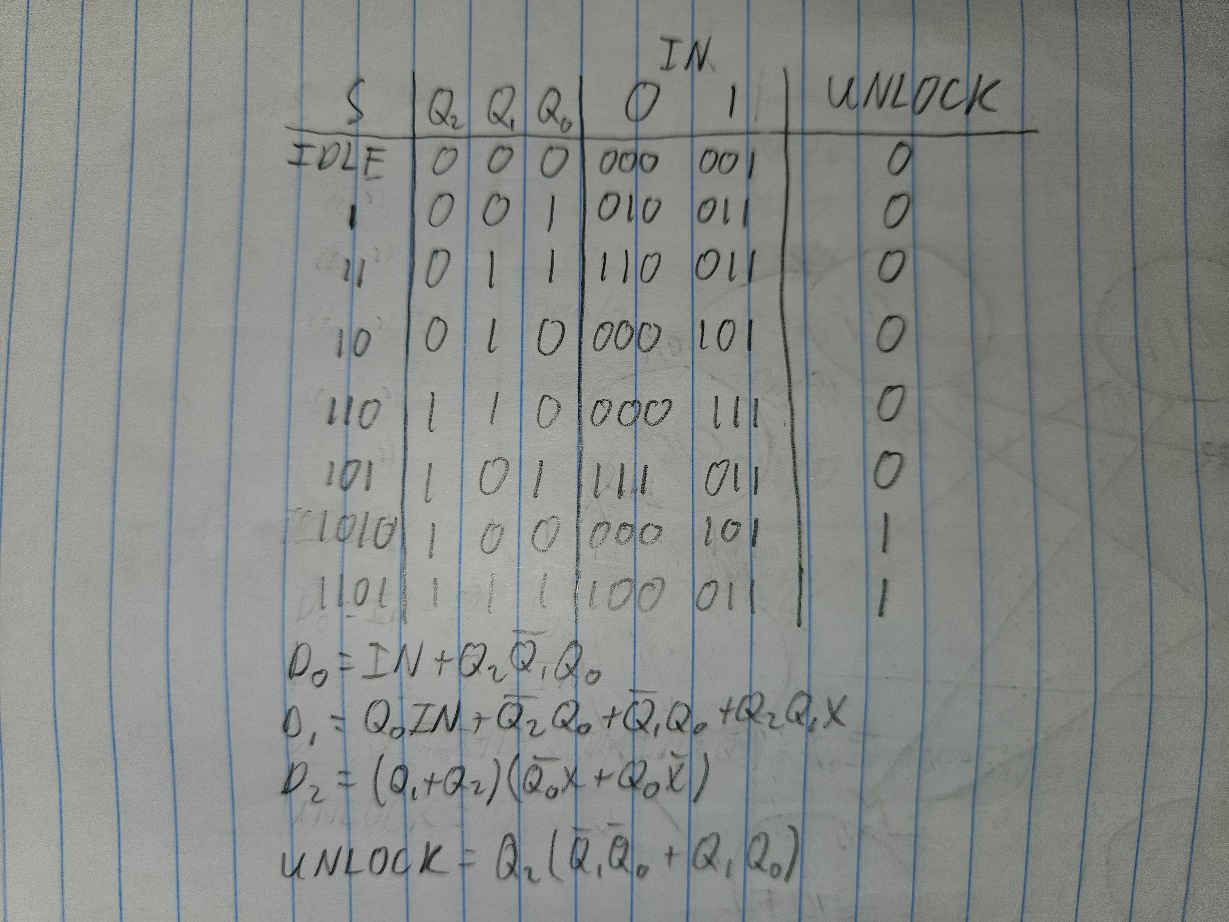
*Objective 2 Design*

For the second and third objectives, I needed to design a Moore and a Mealy device that would detect the sequences 1101 and 1010 from the input IN. For the Moore device, I drew an initial state diagram with eight states which starts in IDLE and can set UNLOCK to 1 by reaching one of two states labeled 1101 and 1010.



Moore State Diagram

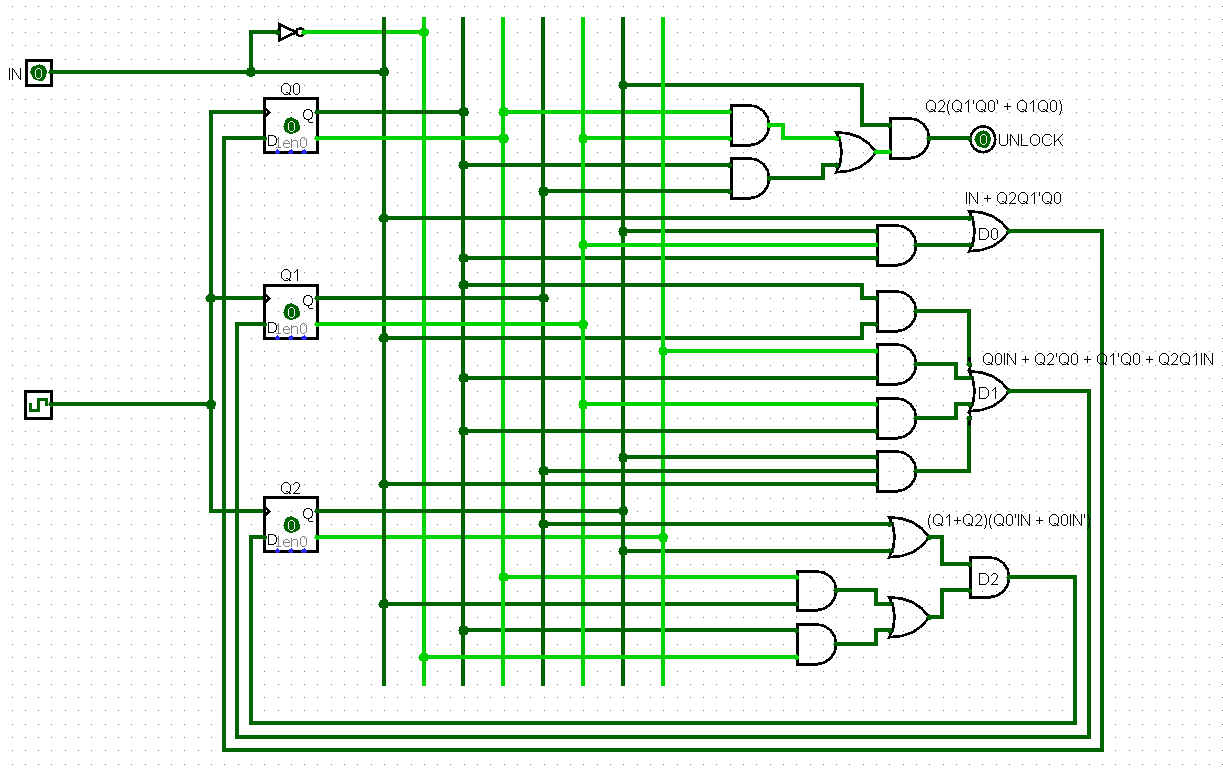
From this diagram, a state table was constructed. The minimization of the state table showed that eight states was the minimum number of states.



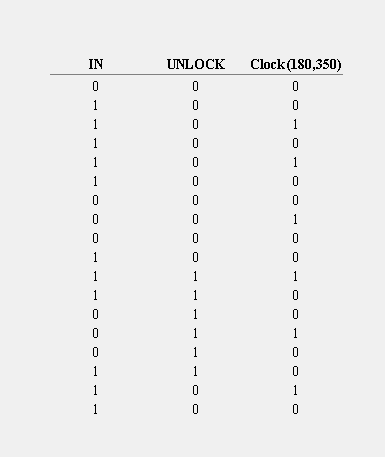
Moore State Table

From this state table, Boolean expressions for the three flip-flops and UNLOCK were determined and simplified. These are shown at the bottom of the above image. Using these expressions, the circuit was constructed in Logisim and tested to verify its functionality. Timing characteristics of the circuit are as follows:

* Setup time, input to clock: 5
* Hold time, input to clock: 1
* Propagation delay, clock to output (min): 12
* Propagation delay, clock to output (max): 22
* Maximum clock rate of device: 1/15



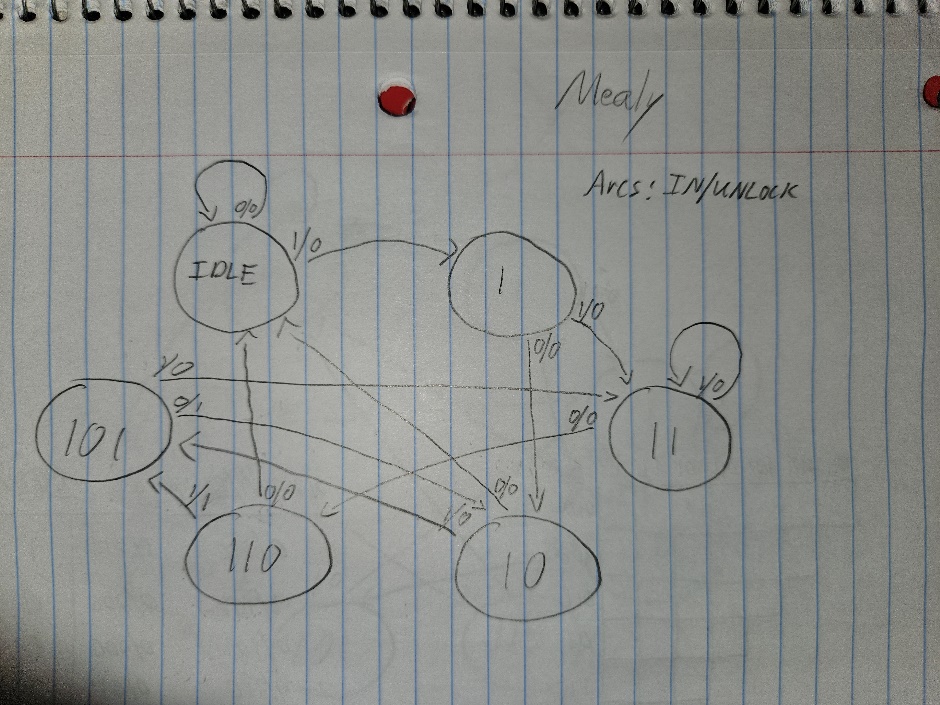
Moore Circuit



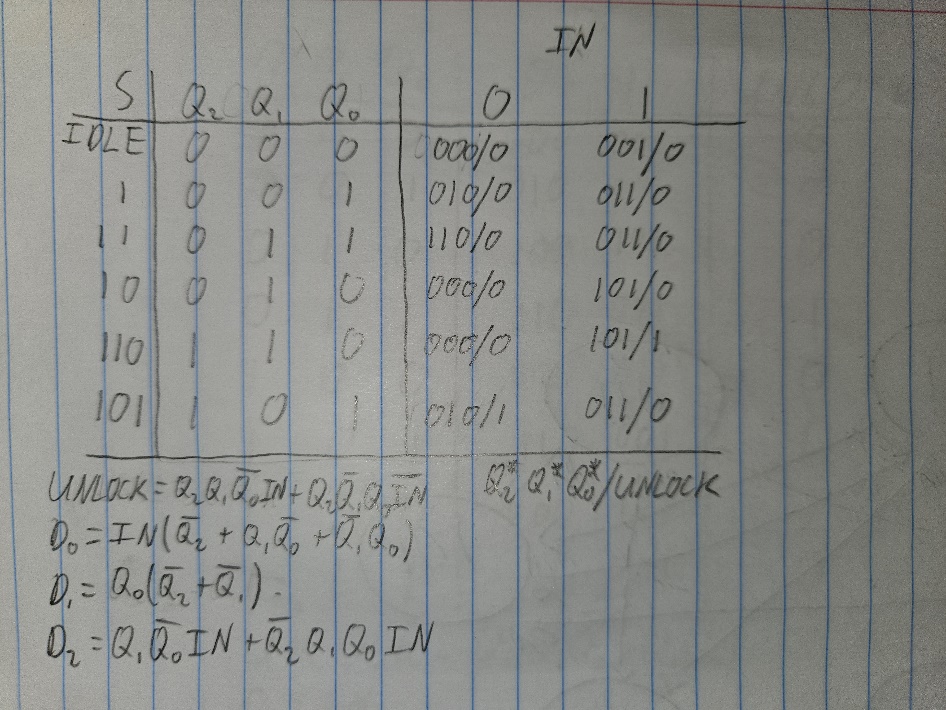
Moore Testing Log

*Objective 3 Design*

For the Mealy device, the system would register the first three bits on clock ticks and use the current value of IN (without a clock tick) for the four bits of the sequence. I began by drawing an initial state diagram with eight states which starts in IDLE, similarly to the Moore device. After state minimization, I realized that the 1010 and 1101 states were unnecessary, given that the Mealy functionality of the device allows for UNLOCK to be set to 1 without a state change. Because of this, I went back and altered the state diagram and table to only have six states.



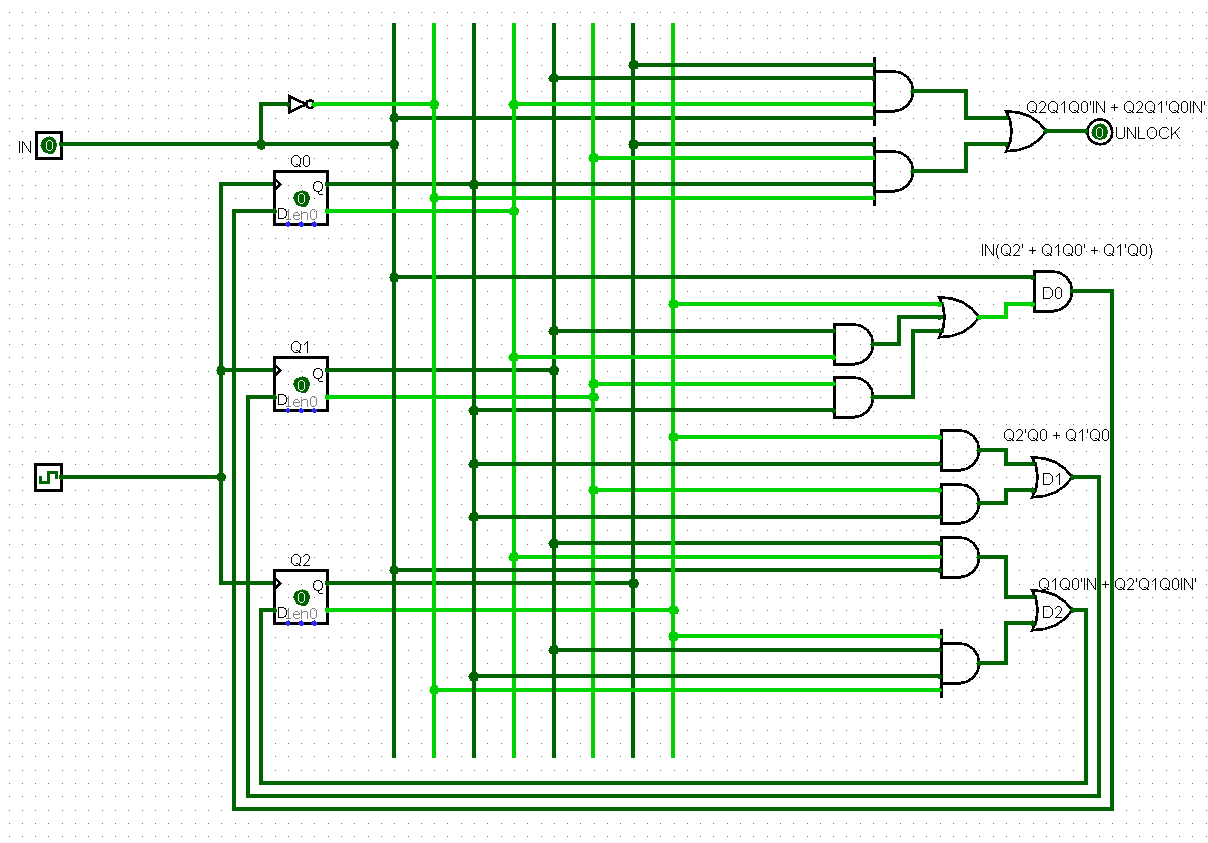
Mealy State Diagram



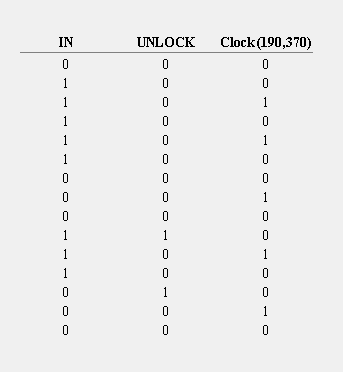
Mealy State Table

From this state table, Boolean expressions for the three flip-flops and UNLOCK were determined and simplified. These are shown at the bottom of the above image. Using these expressions, the circuit was constructed in Logisim and tested to verify its functionality. Timing characteristics of the circuit are as follows:

* Setup time, input to clock: 5
* Hold time, input to clock: 1
* Propagation delay, clock to output (min): 12
* Propagation delay, clock to output (max): 22
* Maximum clock rate of device: 1/15



Mealy Circuit



Mealy Testing Log

*Conclusion*

While the Moore design of the circuit was easier to design because of its straightforward functionality, it was more complex than the Mealy design. This was because of the additional two states that were necessary in the Moore design since the output could not be directly changed by the input. The Mealy design also contained simpler Boolean expressions and, consequently, fewer necessary logic gates in the circuit. However, the Mealy design would be slightly easier to break without knowing the code because a clock tick is not necessary to read the final bit in the four-bit sequence.