**Lab 8: Flip Flops**

Terence Henriod

CPE 201

April 8, 2013

**Problem:** Consider the flight attendant call button discussed in the lecture. The idea is to turn on a light when the call button is pressed and turn it off when the cancel button is pressed.

1. **Truth Table**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Call  C | Reset  R | Current Value  Q | Output Data  D  [Q(t+1)] | mi |
| Store/maintain the value of D | 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| Reset the call signal  (Turn off light) | 0 | 1 | 0 | 0 | 2 |
| 0 | 1 | 1 | 0 | 3 |
| Initiate a service call  (Turn on light) | 1 | 0 | 0 | 1 | 4 |
| 1 | 0 | 1 | 1 | 5 |
| To prevent forbidden condition, prioritize Call | 1 | 1 | 0 | 1 | 6 |
| 1 | 1 | 1 | 1 | 7 |

Table 1: The truth table of the call button system, using the assumed output of a D-flip-flop (the flip flop is not implemented here), using a clock to produce discrete time intervals.

1. **“Sum of minterms” logical expression. Karnaugh maps**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C**\** RQ | 00 | 01 | 11 | 10 |
| 0 |  | 1 |  |  |
| 1 | 1 | 1 | 1 | 1 |

Table 2: The K-map of the call button logical function.

1. **Simplified logical expression**

1. **In Logisim, high-level design using the D flip-flop block diagram and additional logic gates**

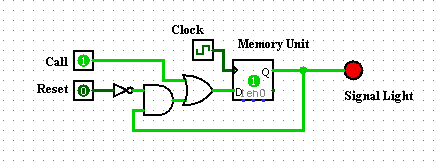


Figure 1: The Logisim diagram of the circuit using a falling edge D-flip-flop.

1. **Implement on breadboard using 7474 (D flip-flop), 7404 (INVERTER), 7408 (AND), and 7432 (OR) gates**