EGCP 1010
Digital Logic Design (DLD)
Laboratory #5

Prepared By:
Alex Laird

on

October 11th, 2007

Lab Partner:
Ryan Morehart

Objective:

The purpose of this laboratory is to introduce and familiarize the student with Multiplexers and Decoders, Medium Scale Integration (MSI) components that will be used to implement one combinational logic network and several multiplexed communication channels.

Procedure and Results:

In the Pre Lab, we created the schematics shown in Figure 1 and Figure 2.

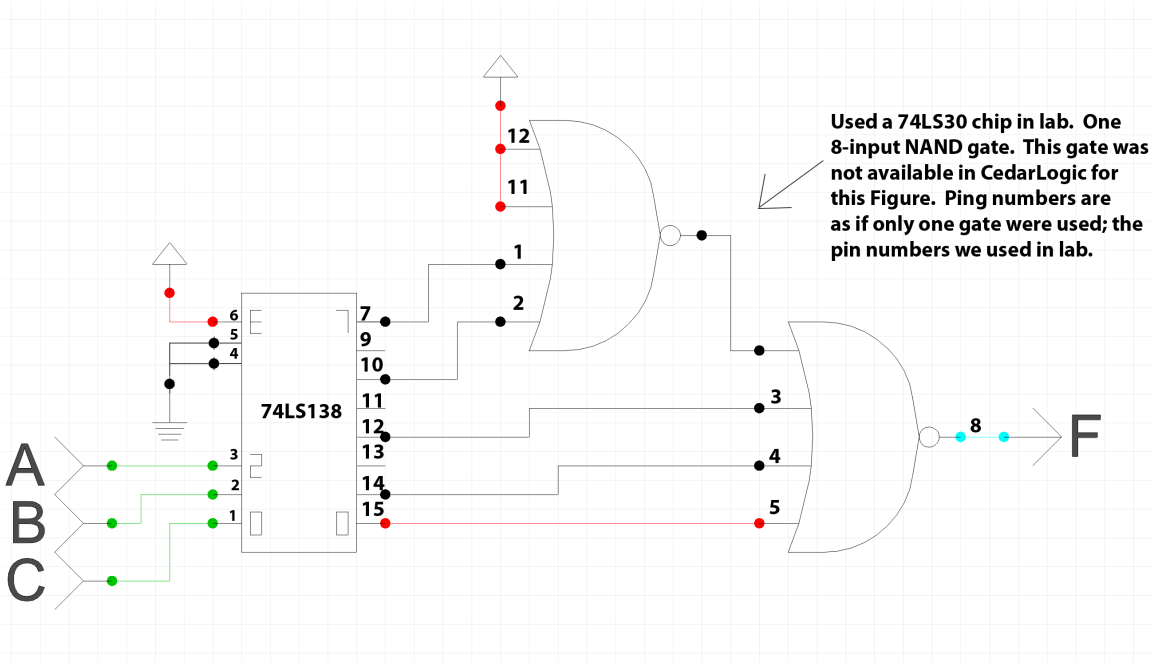

Figure 1: Schematic for Low Active Three (3) to Eight (8) Decoder

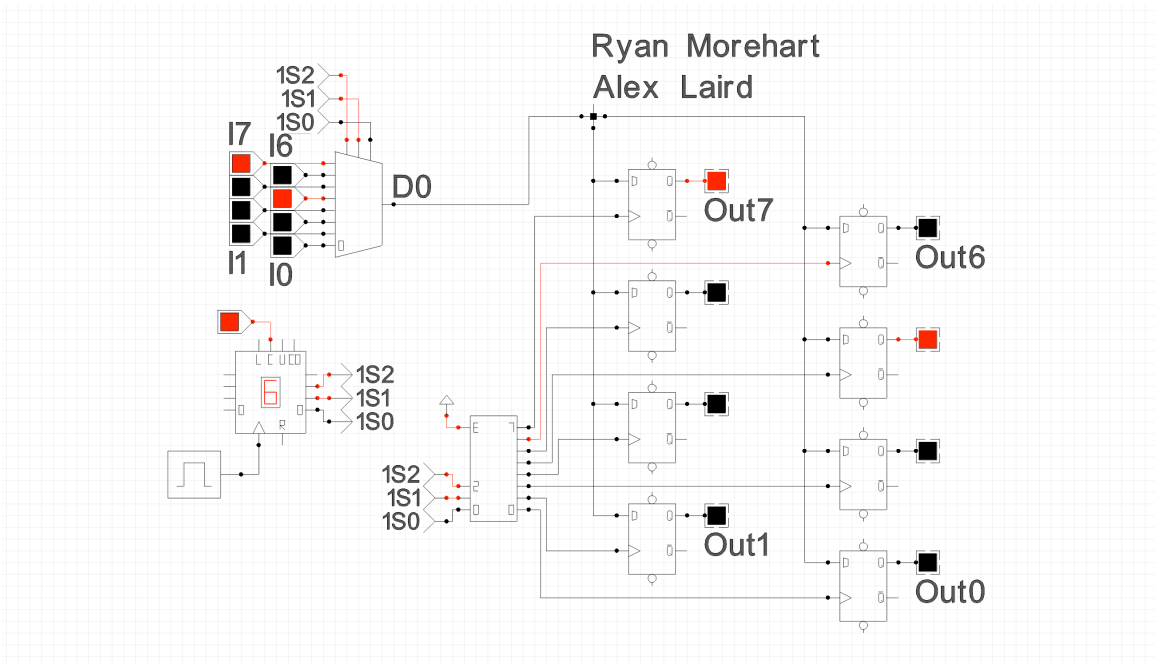

Figure 2: Schematic of Latched Time Division Multiplexed Communication Channel

In lab, we implemented Figure 1 on the CADET II Logic Trainer Boards in manner described below.

We placed a 74LS138 chip and 74LS30 chip on the board. We connected the power (+5V) from the CADET II board to pin 16 (power) of the 74LS138 chip, and the ground from the CADET II board to pin 8 (ground) of the 74LS138 chip. The power from the CADET board was also connected to pin 14 (power) of the 74LS30 chip, and the ground from the CADET board was also connected to pin 7 (ground) of the 74LS30 chip.

For the 74LS138 chip, we connected pins 3, 2, and 1 to inputs S1, S2, and S3. We also connected pins 4 and 5 to the ground from the CADET bored and pin 6 to the power from the CADET board.

We connected pin 7 of 74LS138 to pin 1 of 74LS30; pin 10 of 74LS138 to pin 2 of 74LS30; pin 12 of 74LS138 to pin 3 of 74LS30; pin 14 of 74LS138 to pin 4 of 74LS30; pin 15 of 74LS138 to pin 5 of 74LS30. We connected the remaining pins of the 74LS30 chip, pins 6, 11, and 12, to the power from the CADET board.

After verifying that our circuit was only set to active for the binary inputs 000, 001, 011, 101, and 111, and acquiring the instructors approval, we moved onto the CedarLogic software portion of the lab.

Since we had already completed the Latched Time Division Multiplex Communication Channel, shown in Figure 2, as part of the Pre Lab, we got the instructors approval for this circuit and moved onto other implementations of it.

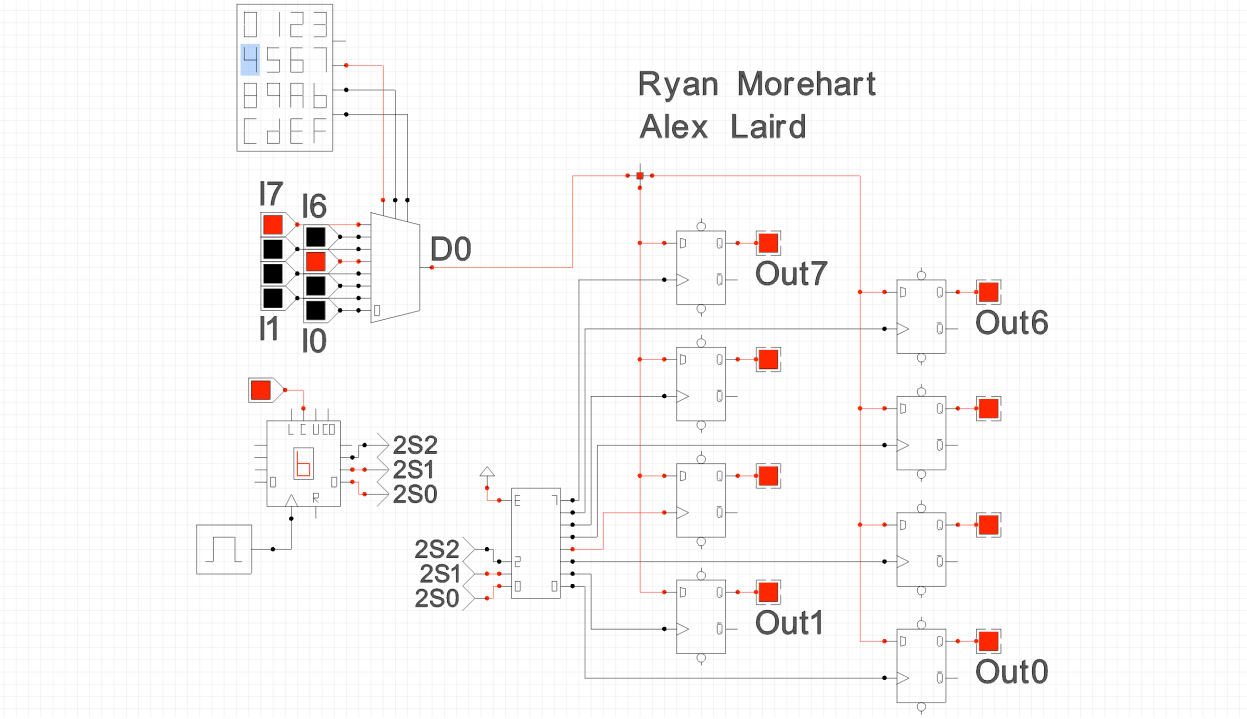

Figure 3: Latched Time Division Multiplexed Communication Channel Implemented with Keypad for Select Lines

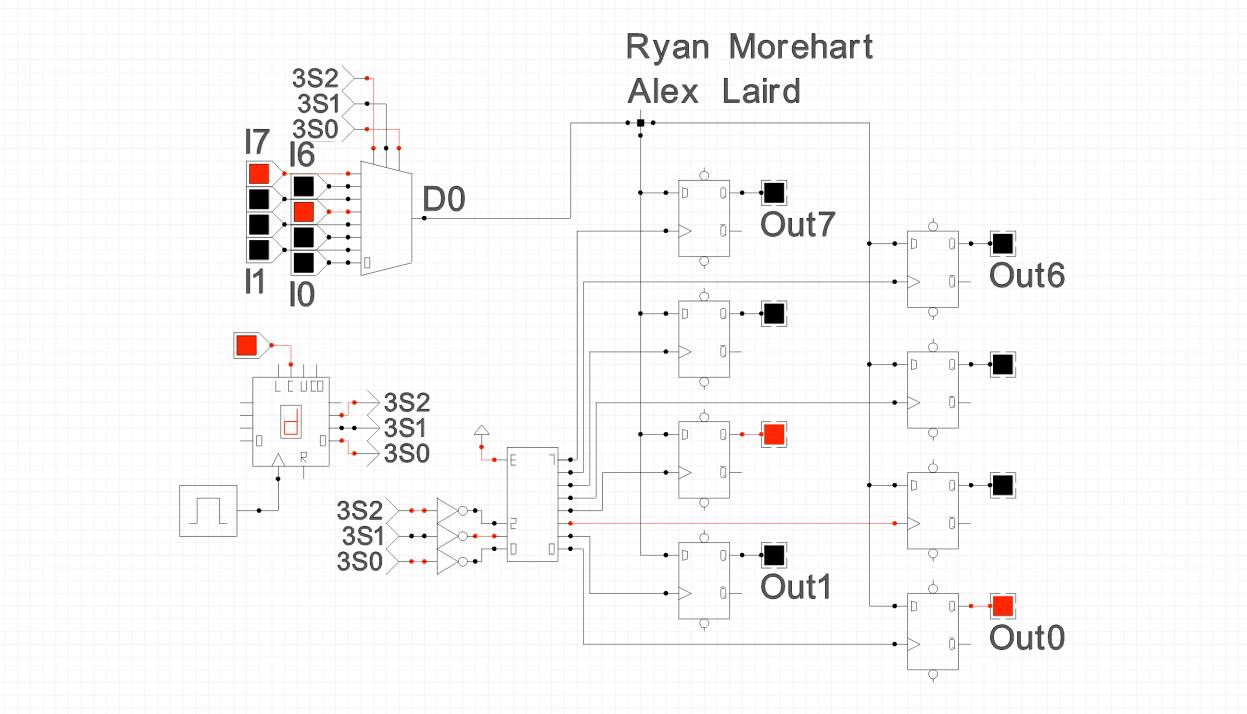

Figure 4: Latched Time Division Multiplexed Communication Channel Mapped to Opposite

Figure 2, Figure 3, and Figure 4 all have similarities in their outputs. Figure 2 and Figure 3, given the same input, will produce the same output. Figure 4 will produce the opposite output; that is, the inputs are mapped to the opposite outputs as from the previous Figures. This outcome is achieved simply by inverting the selection lines on the Decoder.

The basic principle of these three Figures is that, given some active binary bits from the Mux, those outputs will be activated as the next clock cycle is reached for that Flip Flop. Except in the Figure 4, in which the opposite outputs are activated.

The Keypad in Figure 3 doesn't do much except allow you to control which select lines are active at a given time.

The beauty of this circuitry is that, given any amount of inputs or outputs, only one data line is ever needed. If the clock could then be added to the implementation of the Decoder, only the wires coming out of the Decoder would be needed--in this case, only eight (8) wires would be needed.

Conclusion and Suggestions:

Though the concepts of this lab were especially difficult to grasp, after much mulling over them and doing practice problems, I started to understand them and can now see how useful they are, especially in minimizing wire usage. Flip Flops help save much time in the lab, and much cost by eliminating wires and gates.

Questions:

I. Draw a schematic similar to Part 1 to implement the function F(A,B,C) = ∑m (1,2,4,5). Then briefly describe how to implement any function of three (3) variables A,B,C.

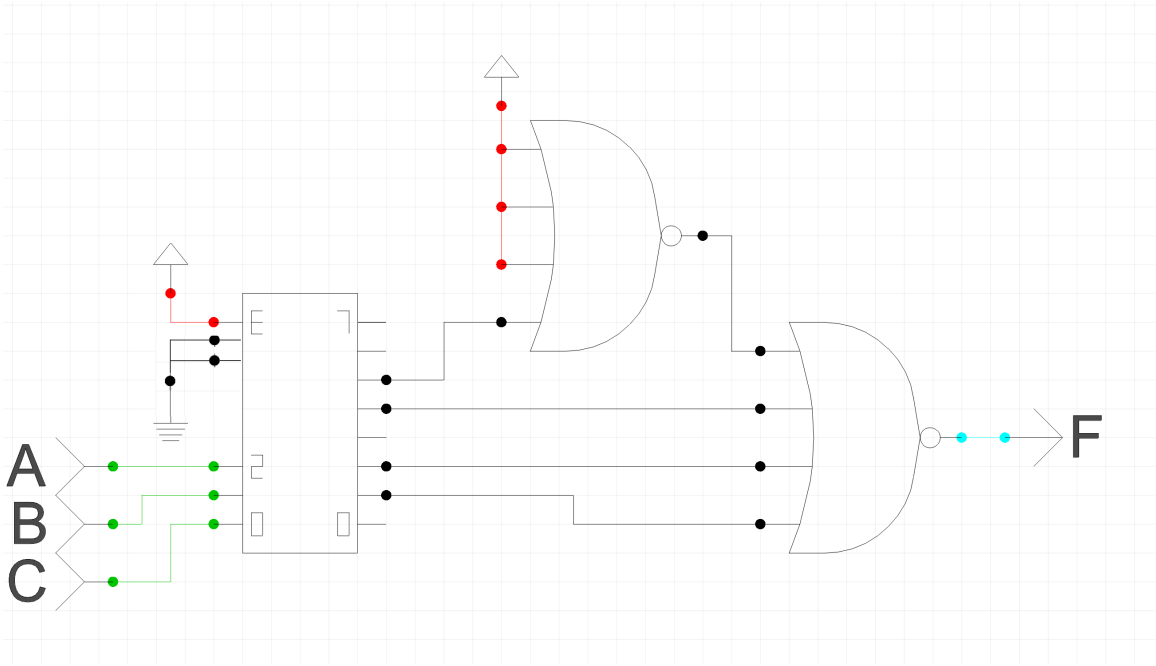

Figure 6: Schematic for F

Simply by attaching the outputs from the Decoder, whichever are specified in the solution of F, to the OR gate, you can implement any function of three variables. Remember to check whether zero (0) starts at the bottom or the top of the Decoder, as different diagrams show it differently.

II. If a communication channel similar to Figure 2 was expanded to accept 64 inputs and communicate to 64 outputs, how many data line(s) and control line(s) would be needed? _1_ data line(s); _6_ control lines

a. What about 1024 to 1024? _1_ data line(s); _10_ control lines

b. Briefly describe the main benefits of a multiplexed communication channel.

It communicates through selection, using the a minimum number of wires. Since wires cost significantly more than chips and other elements of circuitry, this is a very good thing. Multiplex Communication is always carried through one data line. If we represent the number of control (select) lines as n, 2^n will give you the number of inputs and outputs. In the case of 64 inputs and outputs, 2^6 = 64, so six (6) is the number of select lines needed.

III. In a paragraph, give a detailed description of how the circuit to Figure 2 (a latched Time Division Multiplexed Communication Channel) works. Explain in detail how the values of the Input switches I0-I7 are one at a time sent through the Mux, onto Data Lines (D0) then capture by the corresponding Flip Flop O0-O7 due to the Decoders operation at just the right time.

The activated lines from the Mux are passed through the Mux sequentially, one at a time, at each clock. Depending on what the clock is currently set to, which is deduced from the Decoder, that current Flip Flop will be active. Since a D Flip Flop is generally only good for storing, whatever value that is passed from the Mux will be stored in that Flip Flop. A value is only stored in a Flip Flop *when the clock for that Flip Flop is active!* After the clock deactivates for that Flip Flop, the current value is held until the next clock.