SCHOOL OF INFORMATION, COMPUTER AND COMMUNICATION TECHNOLOGY
SIRINDHORN INTERNATIONAL INSTITUTE OF TECHNOLOGY
THAMMASAT UNIVERSITY

LAB REPORT

EES 370 DIGITAL CIRCUIT LABORATORY

Lab 05 Medium-Scale Integration Circuits

By

Mr. Makhapoom Euaumpon **ID: 6222780833**

Mr. Paphana Yiwsiw **ID: 6222780379**

Group No. 12 Section 2

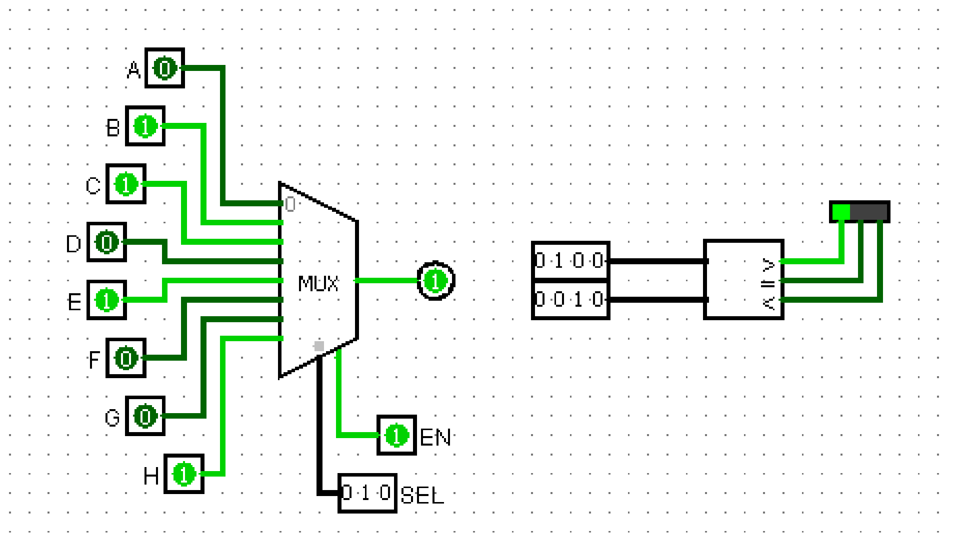
Date: 22 Feb 2021, Time: 13:00-16:00

Objectives

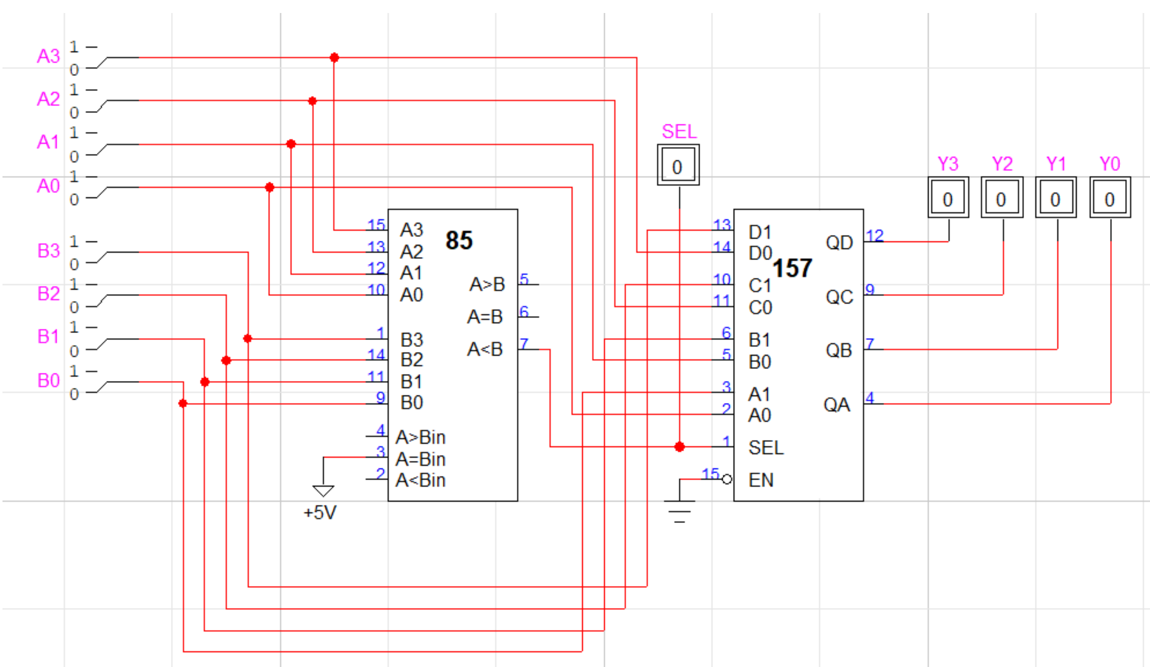
1. To know how to use and be familiar with 8-bit multiplexer and 4-bit comparator in Logisim
2. To know how to use IC 7485 and IC 74157 to create circuit that find larger input from two 4-bit unsigned binary inputs in LogicWorks 5.
3. To know how to create a circuit of 2-to-1 Multiplexer in Tinkercad.

Lab Result

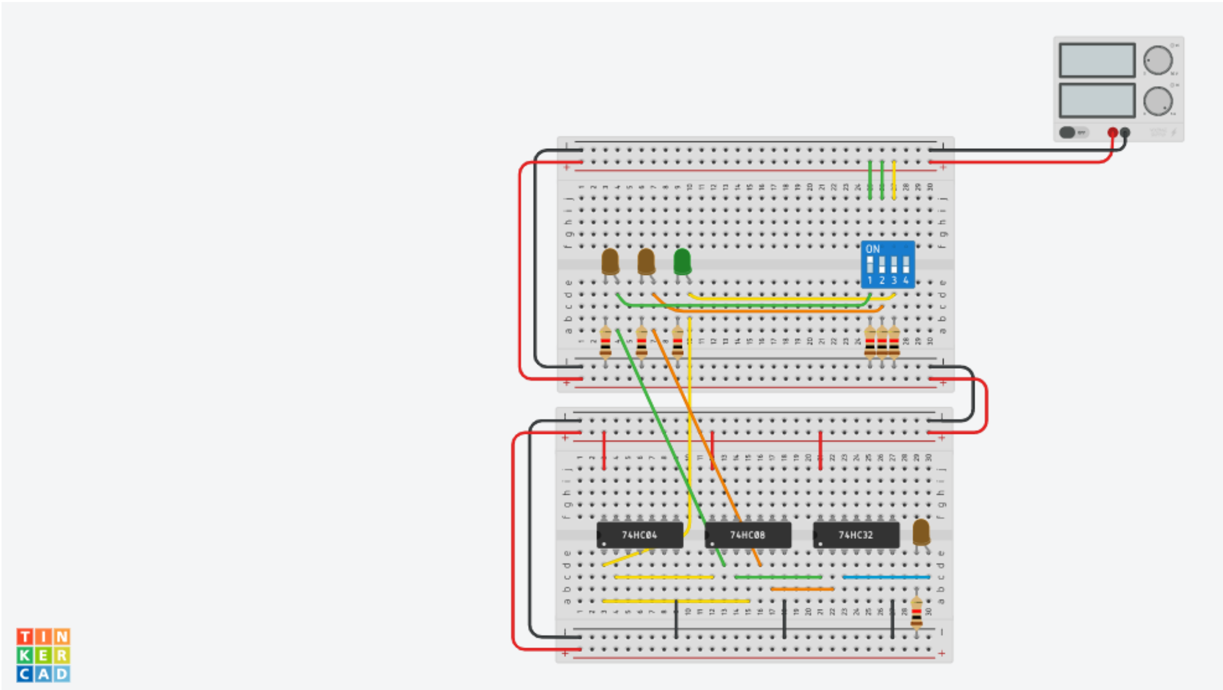
Part 1: 8-bit multiplexer / 4-bit comparator (Logisim)



Part 2: Select Larger Circuit (LogicWorks 5).



Part 3: Develop 2-to-1 Multiplexer (Tinkercad)



Discussion

From Part 1, we implement circuit by using 8-bit multiplexer and 4-bit comparator. First, we will discuss how multiplexer and comparator work. Multiplexer or Mux is a signal selector which will take an n-bit input to select one signal from all of 2^n input as its output. In this case, 8-bit multiplexer require 8 signal inputs and 3-bit selector input to select 1 of that 8 input signals corresponding to the position of input signal which is referred by binary number of selector input as its output. For example, if you have the selector signal as 101, the multiplexer will select the input from pin number 5 as its output. If that pin is HIGH (1), the output of multiplexer is also HIGH (1). Multiplexer also has the enable signal input to decide whether this multiplexer will activate or not. If enable signal is HIGH (1), the multiplexer will be enabled. If enable signal is LOW (0), the multiplexer will be disabled and thus produce no input at all. In Logisim program, we can implement the circuit as we mentioned by selecting from plexers -> multiplexer and then placed on the workspace. After that, we can adjust the multiplexer to accept 8 input by change the ‘select bits’ from 1 to 3. Then, you can connect each of 8 input to the side of multiplexer and 1 3-bit input to the select pin. Also, you need to connect 1 input as HIGH (1) to the enable pin in order to enable the multiplexer and produce the output. The circuit of 8-bit multiplexer is as follows (also shown in lab result part).

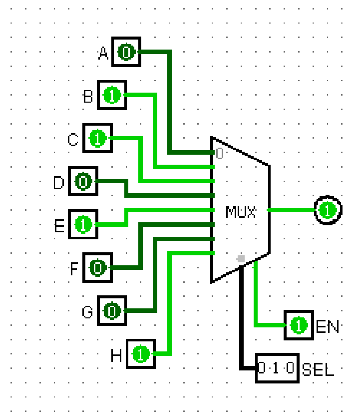


Figure 1: 8-bit multiplexer circuit.

The second half of part 1 is about implementing comparator in the circuit. The n-bit comparator takes 2 n-bit input, referred as A and B, and produce 3 output, A<B, A=B and A>B. Each output will be HIGH (1) if the condition of that pin is true and will be LOW if the condition of that pin is false. For example, if A is equal to 1001 (9) and B is equal to 0110 (6), the output from pin A>B is HIGH (1), output from pin A=B and A<B is LOW (0). This is because 9 is greater than 6 which makes the condition A>B is true. In Logisim program, we can implement this 4-bit comparator circuit by choosing arithmetic -> comparator and placed on the workspace. Then, connect 2 4-bit input to the comparator and connect the output to observe the result. The circuit is as follows (also shown in lab result part).

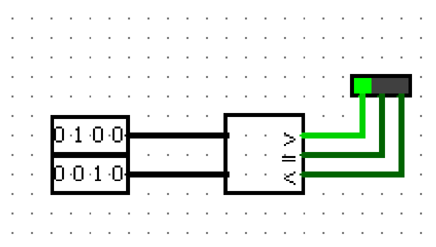


Figure 2: 4-bit comparator circuit

In part 2, we created the circuit that accept 2 4-bit unsigned binary inputs, referred as A and B, and select larger input as the output of the circuit in LogicWorks 5. This circuit can be implemented by using previous part knowledge. The multiplexer IC is IC 74157 which will accept 1-bit selector signal (SEL) to select 1 of 2 4-bit input signal (D0C0B0A0 and D1C1B1A1) as the output. But the enable signal (EN) of IC 74157 is an active-low signal, in order to always enable the IC 74157, we connected EN pin to the ground (GND). The IC 74157 will produce output as follows: If SEL is LOW (0), output is D0C0B0A0. If SEL is HIGH (1), output is D1C1B1A1. The comparator IC is IC 7485 which will accept 2 4-bit binary number and produce 3 outputs, A<B, A=B, A>B. Each output is HIGH (1) when the condition of that output pin is satisfied, similar to the comparator in Logisim. IC 7485 is also accepted cascaded input A>B, A=B, A<B in the case that you compare more than 4-bit binary number. But in this part of the lab, we only focused on comparing 4-bit binary number. Thus, we can connect +5V (HIGH or 1) to the pin number 3 of IC7485 (A=B in) in order to ensure that will not be any effect to the output. After knowing how IC 74157 and IC 7485 work, we can implement circuit by connecting the pin number 7 of IC 7485 (A<B) to the SEL pin of IC 74157. The reason is we want to select the larger number and we connected A to the position 0 and B to the position 1 in the multiplexer. If A is larger than B, the SEL should be LOW (0) and select A as its output. If B is larger than A, the SEL should be HIGH (1) and select B as its output. By defining SEL input according to desire output, we can see that output from pin A<B from IC 7485 satisfy our condition because A<B (A is smaller than B) will produce LOW (0) when A is larger than B and produce HIGH (1) when A is smaller than B (In other word, B is larger than A). For the case of A is equal to B, A will be selected as the output number because output from pin A<B is LOW (0) which SEL is connected to thus, IC 74157 select A as output. After all of this we can connect 4-bit unsigned binary A and B to the IC 7485 and to the IC 74157. A is connected to the A3A2A1A0 in IC 7485 and D0C0B0A0 in IC 74157. B is connected to the B3B2B1B0 in IC 7485 and D1C1B1A1 in IC 74157. The answer to the given table in the lab manual is as follows:

	Input A				**Input B**				**SEL**	**Output**			
(1)	0	1	0	1	1	0	1	0	1	1	0	1	0
(2)	1	0	1	0	0	1	0	1	0	1	0	1	0
(3)	1	1	0	0	0	0	1	1	0	1	1	0	0
(4)	0	0	1	1	1	1	0	0	1	1	1	0	0

Figure 3: Answer for lab part 2 table from lab manual

The circuit from lab part 2 as mentioned from above is shown as follows (also in the lab result part).

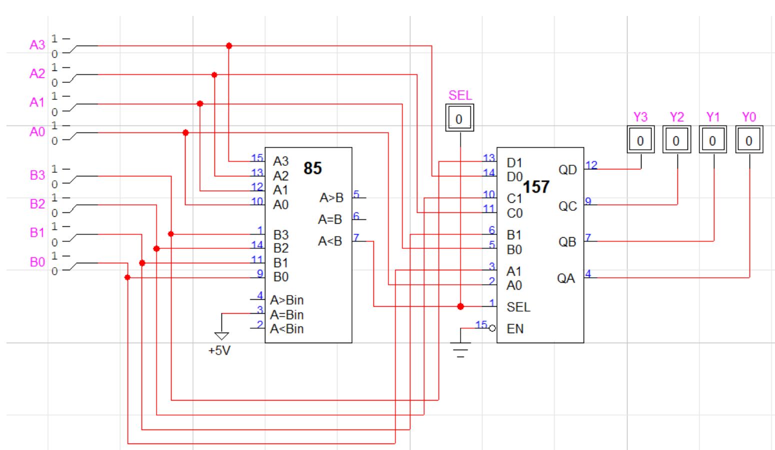


Figure 4: Select larger input circuit, implemented using IC 7485 and IC 74157 in LogicWorks 5

For part 3 of this lab, we have to implement and connect the 2-to-1 multiplexer circuit on Tinkercad. First of all, from the knowledge we have about multiplexer so far is that multiplexer will produce output from 1 of 2^n input which selected by n-bit input. In this case, this multiplexer will produce 1-bit output (referred as Y) that selected from 1 of 2 input (referred as A and B), selector input is 1-bit (referred as SEL). If SEL is 0, output Y is A. If SEL is 1, output Y is B. After that, we design the truth table and derived Boolean expression by using K-map. The truth table and K-map of 2-to-1 multiplexer are as follows:

A	B	SEL	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

Figure 5: Truth table of 2-to-1 multiplexer

SEL **AB**	**0**	**1**
00	0	0
01	0	1
11	1	1
10	1	0

Figure 6: K-map of 2-to-1 multiplexer

From K-map above, we can derive the Boolean expression as Y = A SEL' + B SEL. Thus, the circuit diagram of this 2-to-1 multiplexer are as follows:

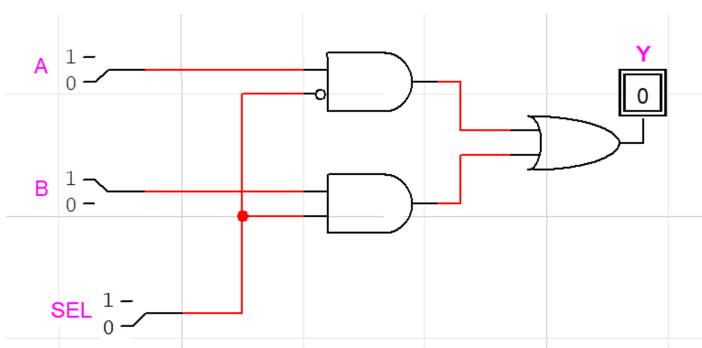


Figure 7: Circuit diagram of 2-to-1 multiplexer

After that, we can implement this circuit on Tinkercad using similar procedure as lab 1. First, we placed power supply and breadboard and connect breadboard to the power supply. Then, place DIP switch to make it as input of this circuit. In this circuit, we use switch number 1 as A, number 2 as B, number 3 as SEL. To use DIP switch in Tinkercad, you connect the power to each switch you want to use and connect resistor where that switch is to ground. After that, connect LED to observe the input more easily. After that, implement the circuit by using Inverter (74HC04 in Tinkercad) as NOT gate, 74HC08 as AND gate and 74HC32 as OR gate. Connect the power and ground to all IC. Then, connect each input to each gate following the circuit diagram we designed above. Lastly, connect output to the LED to observe the output and check the result with the truth table above. The implemented circuit from Tinkercad is as follows (also in the lab result part).

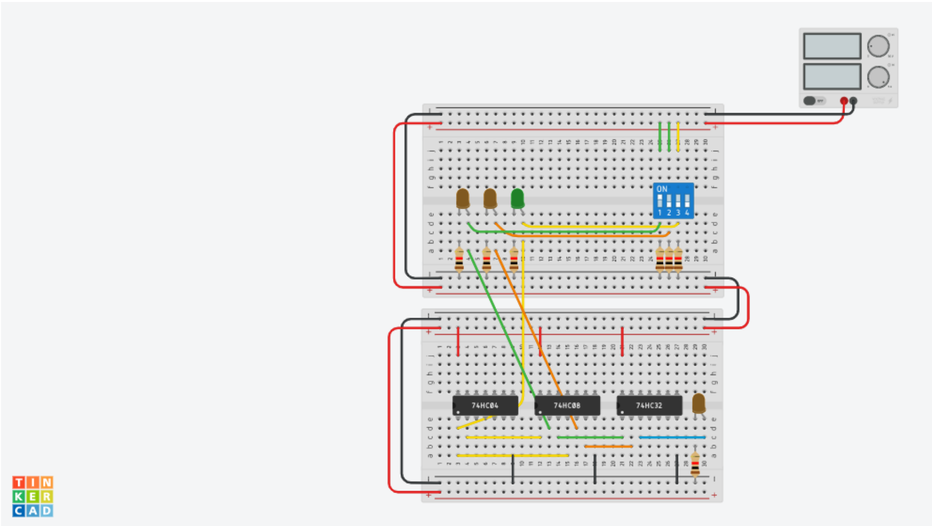


Figure 8: 2-to-1 multiplexer, implemented in Tinkercad

Conclusion

In this lab, we have learned how to implement 8-bit multiplexer and 4-bit comparator in Logisim. Also, know how to use IC 7485 and IC 74157, comparator and multiplexer respectively, in order to create a circuit that find larger input from 2 4-bit unsigned binary input and select it as its output of the circuit in LogicWorks 5. Lastly, we use knowledge from all lab sessions to create the truth table, find Boolean expression from K-map, draw circuit diagram of 2-to-1 multiplexer and implement the circuit in Tinkercad successfully.