3-BIT ADDER

Project 1

CS 200

Spring 2022

Date Submitted: 2/2/2022

Sam Hemmelgarn

sph227@nau.edu

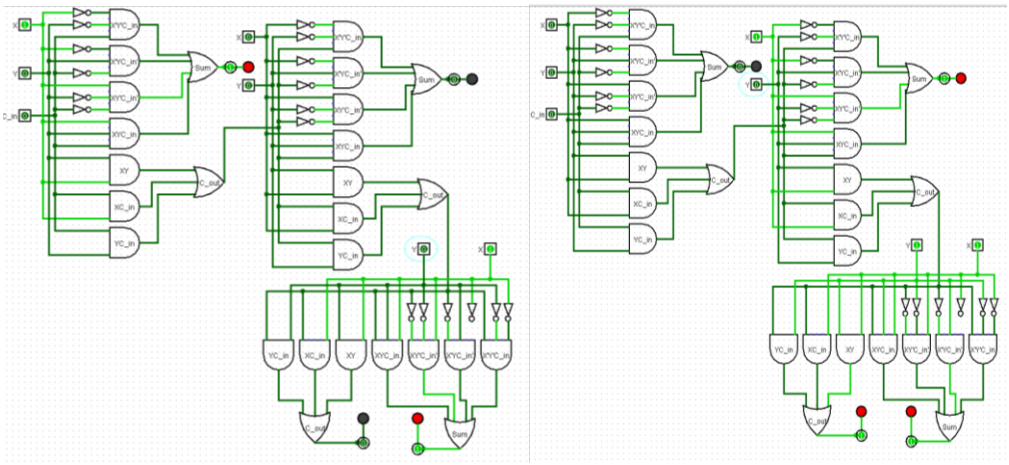
6016903

Project Overview

The purpose of this project is to create a 3-bit Adder by connecting together 3 full adders. I started this project by creating a truth table with the 3 inputs: x, y, and a carry in bit. The outputs of this truth table were a sum of the 2 bits along with a carry bit, utilized if the x and y bit were both set to 'true'. I then created 2 Boolean functions from the truth table, one for the sum bit and one for the carry out bit. A Karnaugh Map was then utilized to reduce the two Boolean functions for the sum bit and carry out bit. From the Karnaugh Map, I designed a circuit in Logisim that resembled the 2 functions using only AND, OR, or NOT gates, and connected 3 of these circuits together to make a 3-bit adder.

Results

Below are active screenshots of the circuit created showing the minimized functions translated into a circuit, along with truth tables, Karnaugh maps and functions:



Truth Table

X	Y	C_in	Sum	C_out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Sum(x, y, C_in) = x'y'C_in + x'yC_in' + xy'C_in' + xyC_in
C_out(x, y, C_in) = x'yC_in + xy'C_in + xyC_in' + xyC_in

Fig. 1

Karnaugh Maps

Sum

C_in \ XY	00	01	11	10
0	0	1	0	1
1	1	0	1	0

Reduced function for Sum(x, y, C_in) = x'yC_in' + xy'C_in' + x'y'C_in + xyC_in

Fig. 2

C_out

C_in \ XY	00	01	11	10
0	0	0	1	0
1	0	1	1	1

Reduced function for C_out(x, y, C_in) = xy + yC_in + xC_in

Fig. 3

Conclusions

With this being the first time I've studied digital logic, it was very interesting me to learn the basis of how computers work. I enjoyed learning how to make the truth tables and seeing how they can be translated into a circuit. In the first homework, I didn't overlap the groups on my Karnaugh maps, so I ended up with the wrong function. After receiving feedback from the grader on that homework, I was able to see what I had done wrong and apply it in this project. Logisim was a great way to visually see how the truth tables we created at the beginning, translate from the Karnaugh maps into the actual circuit.

After class on Monday, January 31st, Mr. Kelley drew a full adder for us showing the use of two separate inputs coming from each input pin (i.e. X pin had an X input and an X' input coming out of it). I attempted to rework my circuit in that manner, however I decided not to because I would have had to redo the entire circuit. I do see how the way that it was done in class would result in less NOT gates used overall and would make the entire circuit easier to follow. He also addressed using a ground for the first 'C_in' input, which I did change in my circuit file but is not shown in the screenshots in the report.