Borja Rojo

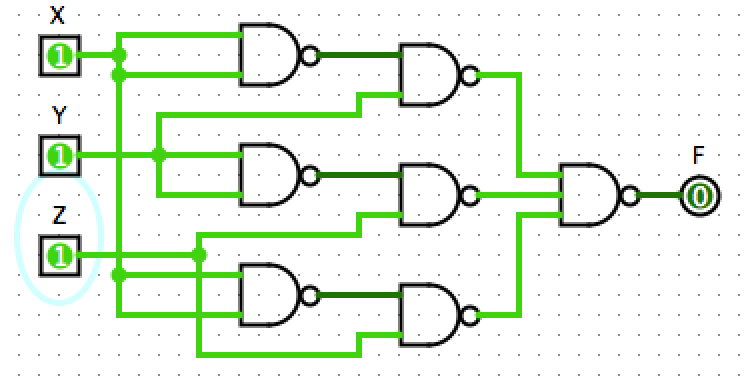
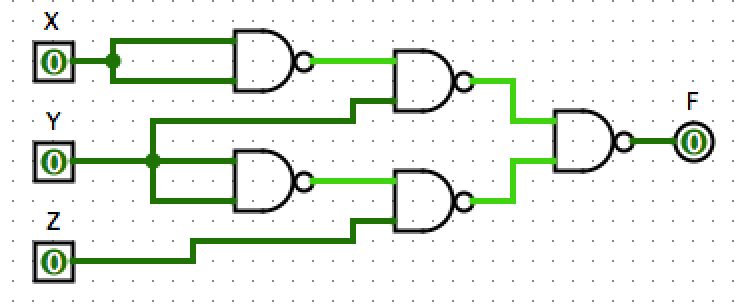
3/27/15

Logic Circuits – Investigating Three Circuits

Lab Project 5

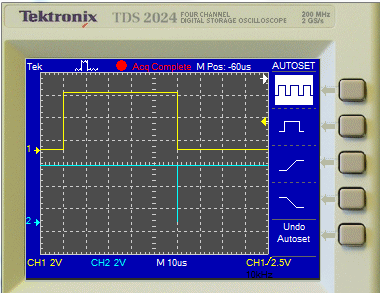
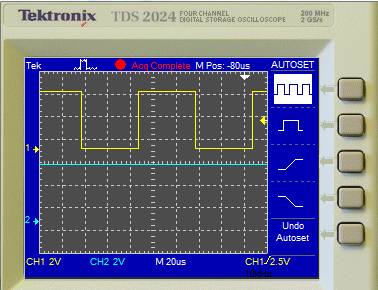
Part 1

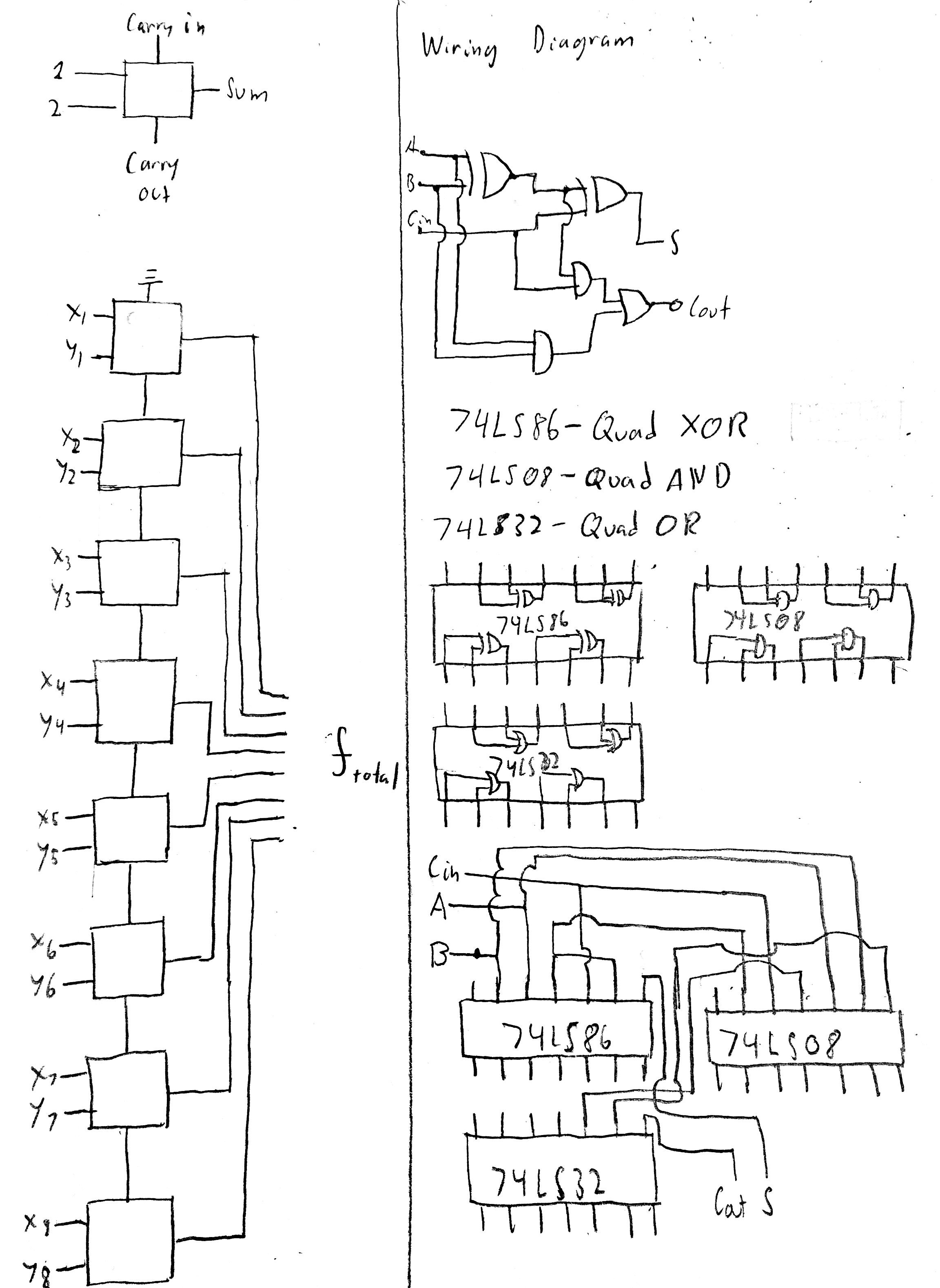
1) F(A,B,C) = (A’B) + (B’C) F(A,B,C) = (A’B) + (B’C) + (A’C)



|  |  |  |  |
| --- | --- | --- | --- |
| A | B | C | Ƒ |
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 |

2) The hazard was eliminated by considering the circuit’s Karnaugh map configuration. The issue lay where the variables X and Y were not linked in the circuit, causing propagation error and therefor a hazard. This was fixed by creating a segment in the circuit that held an (X’Z) logic subcircuit.

3) The cost of the original circuit, realized with NAND gates, is 15. The cost of the new circuit, realized with NAND gates, is 22. The difference in cost is 7.

Part 2

1 & 2) -------->

3) An estimated time for an adder to add two input bits is 100ns.

This can be calculated by adding the propagation delays of each gate. Looking at the spec sheet, the 74LS86 Quad XOR gate has a propagation delay of 15 ns when an input goes HIGH and changes the state of the output. This is the important value to consider because adding requires inputs to go high, not low. The 74LS08 Quad AND has a propagation delay of 18 ns for any configuration of switching, and the 74LS32 Quad OR has a propagation delay of 15 ns. In order for the Sum output of the adder represent the logic of the inputs, there are two XOR gates for it to go through at its longest branch. The Carry out output requires signal to pass through an XOR gate, an AND gate, and an OR gate. Therefor, it would take 30ns to get a correct Sum output and 48 ns to get a Carry out output pin.

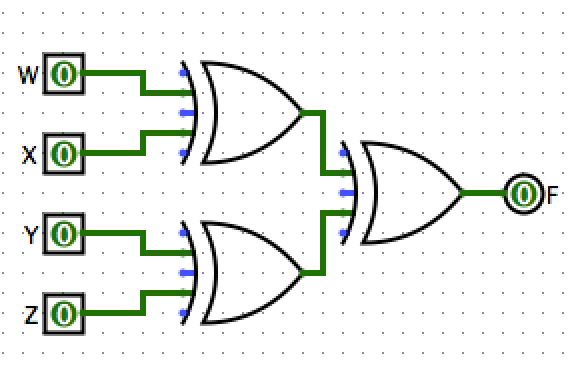
Assuming the Carry in is held constant, this time can be measured by hooking up both variable inputs and outputs to an oscilloscope and flipping one of the inputs high. With a single image being measured and the trigger set to the rising edge of the flipped input, a corresponding propagation delay trace with by created, with the rise of the flipped input at t = 0 and the outputs changing accordingly later on.

4) Each adder must consider the Carry out in this situation. The Carry in propagates through an AND gate and an OR gate in it’s longest branch, causing a total recalculation time of about 33ns for a total of 81ns. Therefore, each subsequent adder after the first will have, at most, 33ns of time added to it. This situation would only happen if a Carry out bit is changed at every adder, which is unlikely though possible. Without any Carry out bits, there is minimum calculation time of 48ns, as stated before. At best, an 8 bit adder can add 20,833,333 sums in one minute. At worst, an 8 bit adder can add 123,456,789 sums in one minute.

5) The cost for 1 full adder is 15. With 8 full adders, the cost of the circuit is 120.

6) The cost can be improved by using a half adder at the beginning of the circuit, removing 3 of the gates and reducing the cost by 15 to 105.

Part 3

1 & 2) The configuration of light switches in a house can be represented by cascaded XOR gates.

The Karnaugh Map of this circuit is characterized by alternating 1’s and 0’s in checkerboard fashion. This is due to the nature of every variable switch from High to Low or Low to High causing a change of state in the output, just like every switch flip causing a light to turn off from on, or on from off.