

ELC 2137 Lab 3: Adders

Alexander Noll, Jonah Trahan

February 2, 2020

Summary

The lab was designed to put into practice logic gate in order to perform basic mathematical functions. Using only AND and XOR gates a half, full, and 2-bit full adder were created and tested. The construction was planned using foreknowledge of Boolean arithmetic and truth tables.

Q&A

Which gates could be used for combining carry bits?

OR or XOR gates can be used to add carry bits. We should use XOR gates as they minimize the number of different gates needed and work functionally the same as an OR gate. See Table 1 for exhaustive proof in truth table.

Results

ELC 2137

Lab 3. Adders

Circuit Demonstration Page

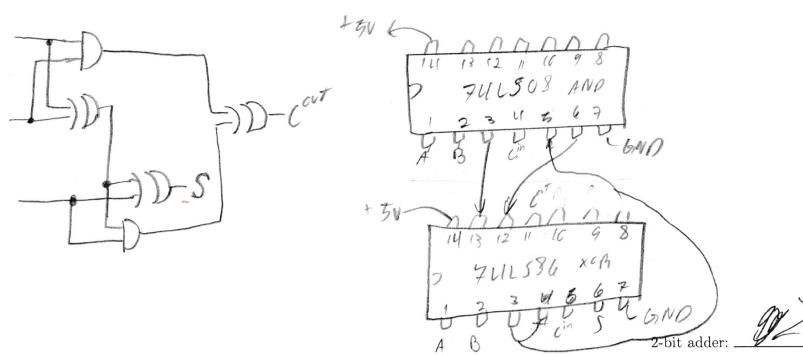
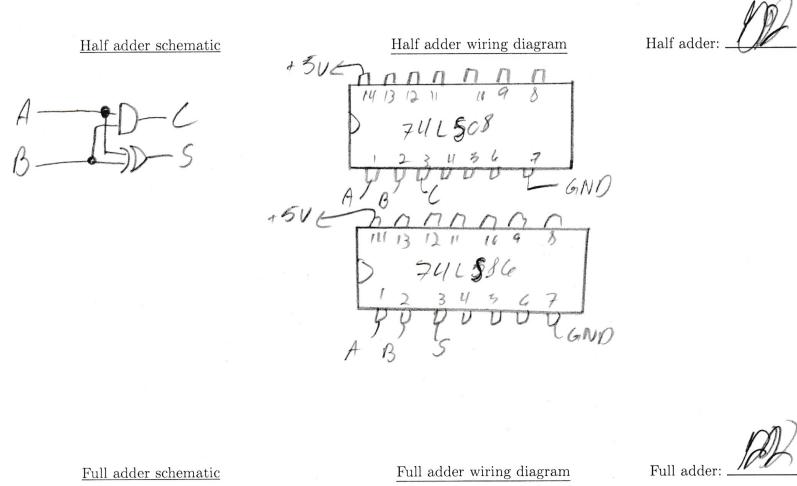


Figure 1: Circuit demonstration page with Half and full adder schematics and wiring diagrams. The document also includes instructor initials for the completion of each adder type

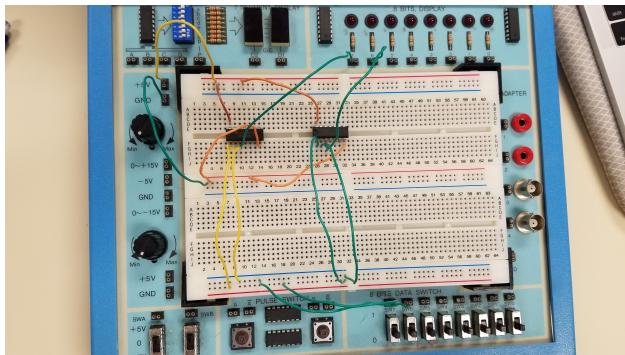


Figure 2: Image of constructed Half Adder.

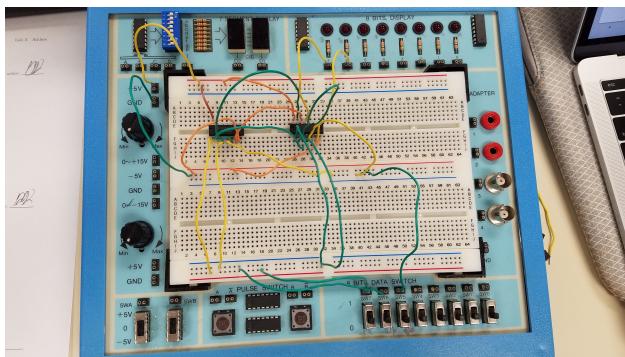


Figure 3: Image of constructed full adder.

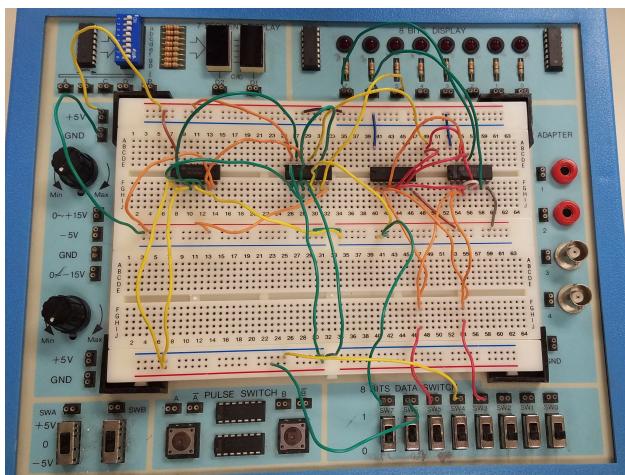


Figure 4: Image of constructed 2 bit full adder

Table 1: Carry Bit Addition Truth Table

| Cin | Half Adder Sum (S) | S*Cin (X) | C | Cout(S XOR C) |
|-----|--------------------|-----------|---|---------------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 |

Table 2: The Truth Table demonstrates through exhaustive proof that no matter the inputs both carry digits X and C will never both be 1 and the XOR gate can always add them as an OR gate would.*

* Two repeated lines in the truth table were omitted as they only exist as a consequence of the intermediate carry bit, C, being unable to be 1 when S=1 and visa versa. This is due to S being the result of an AND gate of only two one-bit inputs and thus the largest binary out is 10. The rightmost bit being the carry bit C, and the leftmost bit being S.