

# ELC 2137 Lab 03: Adders

Kyra Rose

October 11, 2020

## Summary

In this lab we used AND and XOR gate to build circuits that performed the functions of a half adder, full adder, and a two-bit adder.

## Q&A

1) Which gates could we use for combining the carry bits?

We could use the XOR or the NAND gate.

2) Which one should we use and why?

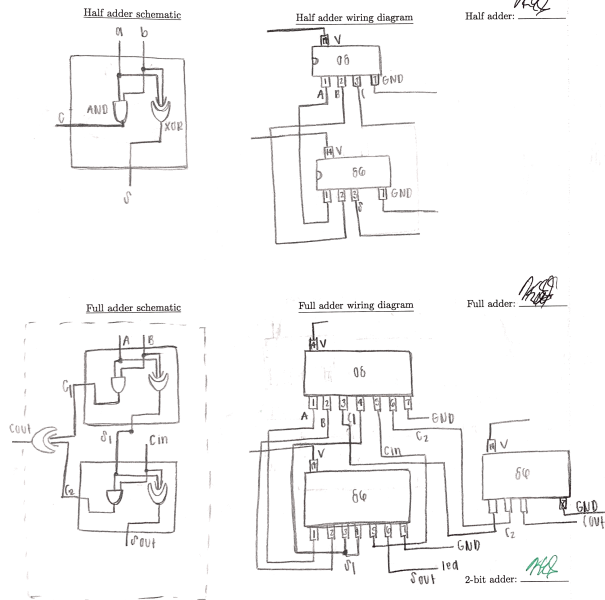
We should use the XOR gate, because we don't want our Cout LED to be one when both carry bits are off.

## Results

Table 1: FA Expanded Truth Table

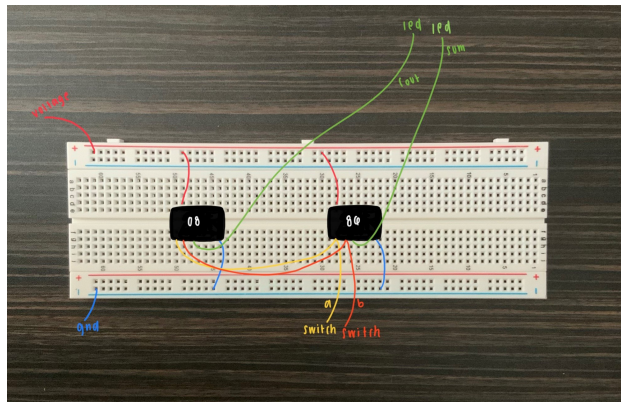
Cin	A	B	C1	S1	C2	S2	Cout	S
0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0	1
0	1	0	0	1	0	1	0	1
0	1	1	1	0	0	0	1	0
1	0	0	0	0	0	1	0	1
1	0	1	0	1	1	0	1	0
1	1	0	0	1	1	1	1	0
1	1	1	1	0	0	1	1	1

## Circuit Demonstration Page

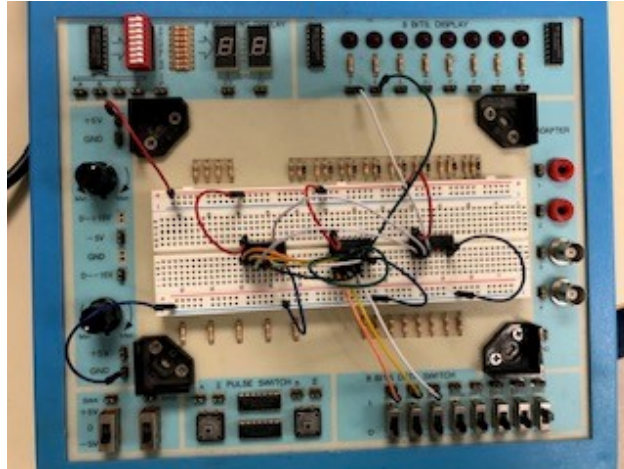


4

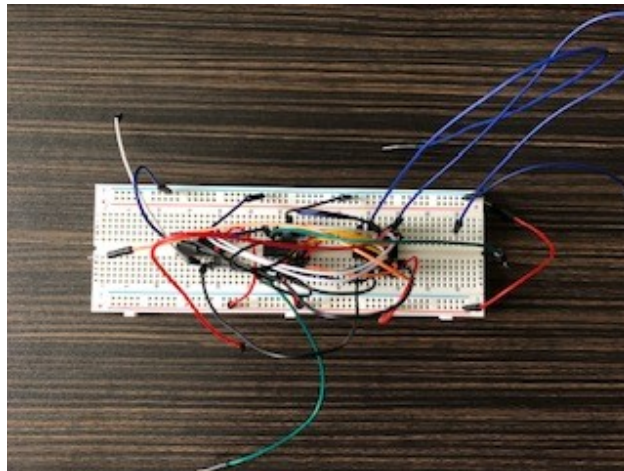
## Circuit Demonstration Page



Half Adder



## Full Adder



## 2 Bit Adder

## Code

Listing 1: Direct Verilog code example

```
\begin{center}
\caption{Circuit Demonstration Page}

\includegraphics[width=0.5\textwidth]{circuit_demo_page}

\caption{Half Adder}

\includegraphics[width=0.5\textwidth]{half_adder}

\caption{Full Adder}

\includegraphics[width=0.5\textwidth]{full_adder}

\caption{2 Bit Adder}

\includegraphics[width=0.5\textwidth]{2_bit_adder}
\end{center}
```

```

\begin{table}[ht]\centering
\caption{FA Expanded Truth Table}
\label{tbl:example_table}
\begin{tabular}{ccc|cccc|cc}
\toprule
Cin & A & B & C1 & S1 & C2 & S2 & Cout & S & \\
\midrule
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & \\
0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & \\
0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & \\
0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & \\
1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & \\
1 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & \\
1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & \\
1 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & \\
\bottomrule
\end{tabular}
\end{table}

```

---