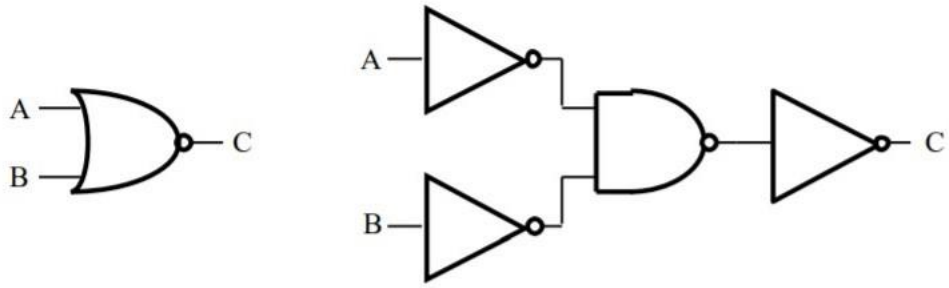
Logic Gates and Combinatorial Logic

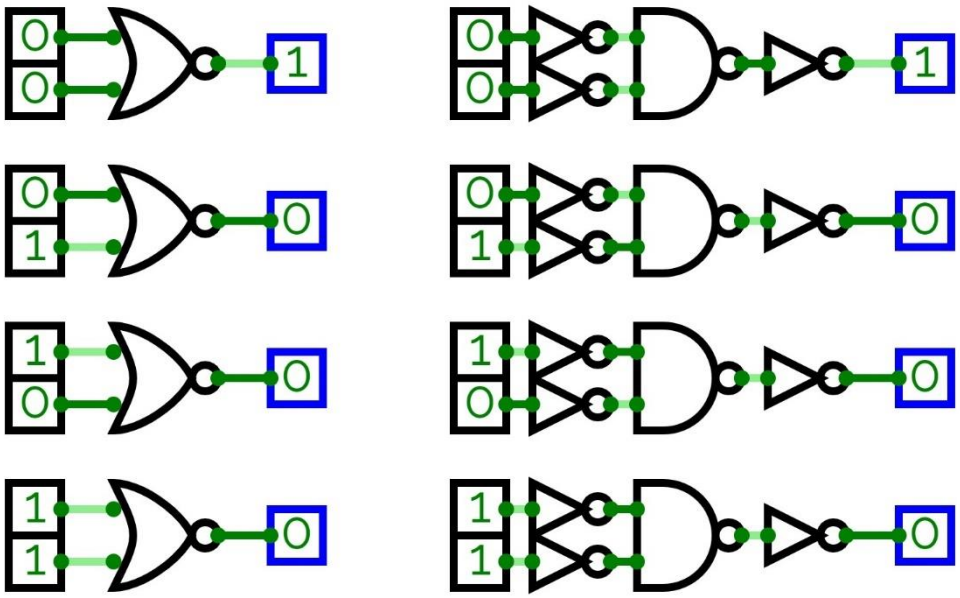
Lab Report

Circuitverse simulation link: https://circuitverse.org/users/68960/projects/eeen202-lab-1-038b9dbd-f0a1-474f-8a8e-9aa6bf355bda

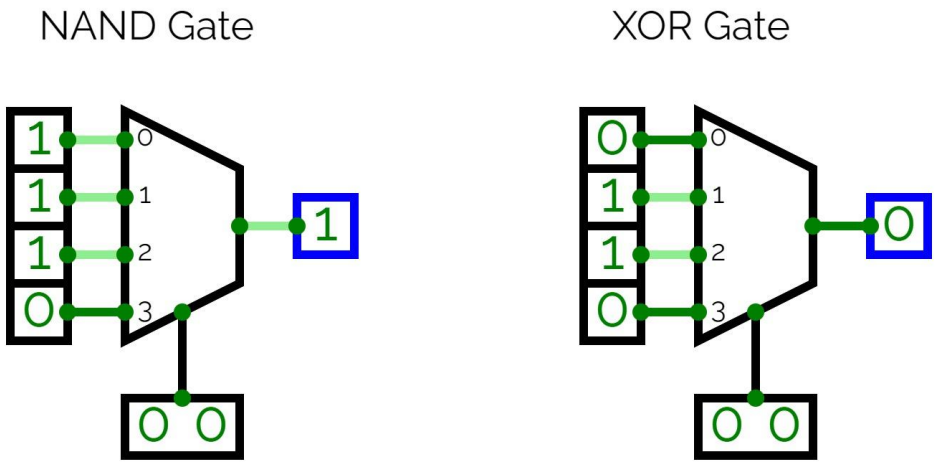
5.1. De Morgan's Rule Circuitverse simulation (Section 3.2.4)



NOR gate				Inverted NAND		
A	B	C		A	B	C
0	0	1		0	0	1
0	1	0		0	1	0
1	0	0		1	0	0
1	1	0		1	1	0

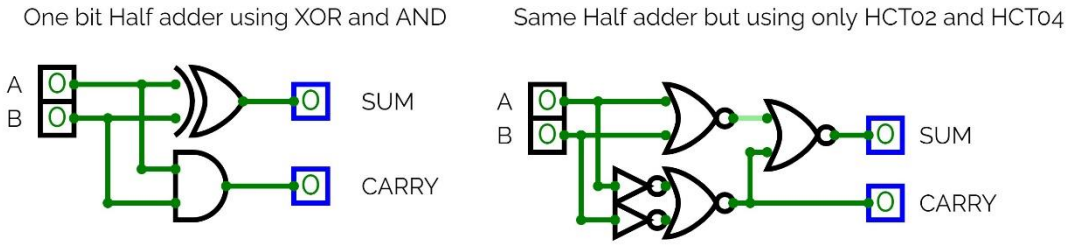


5.2. 4:1 MUX as NAND gate (Section 3.2.6)

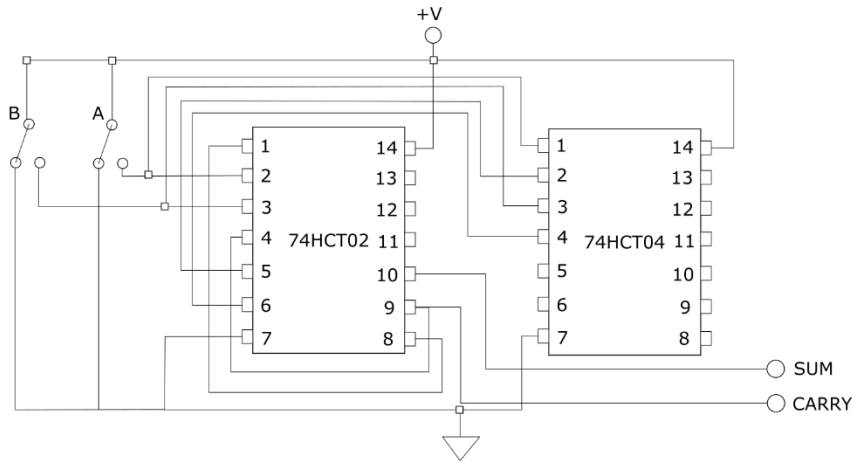


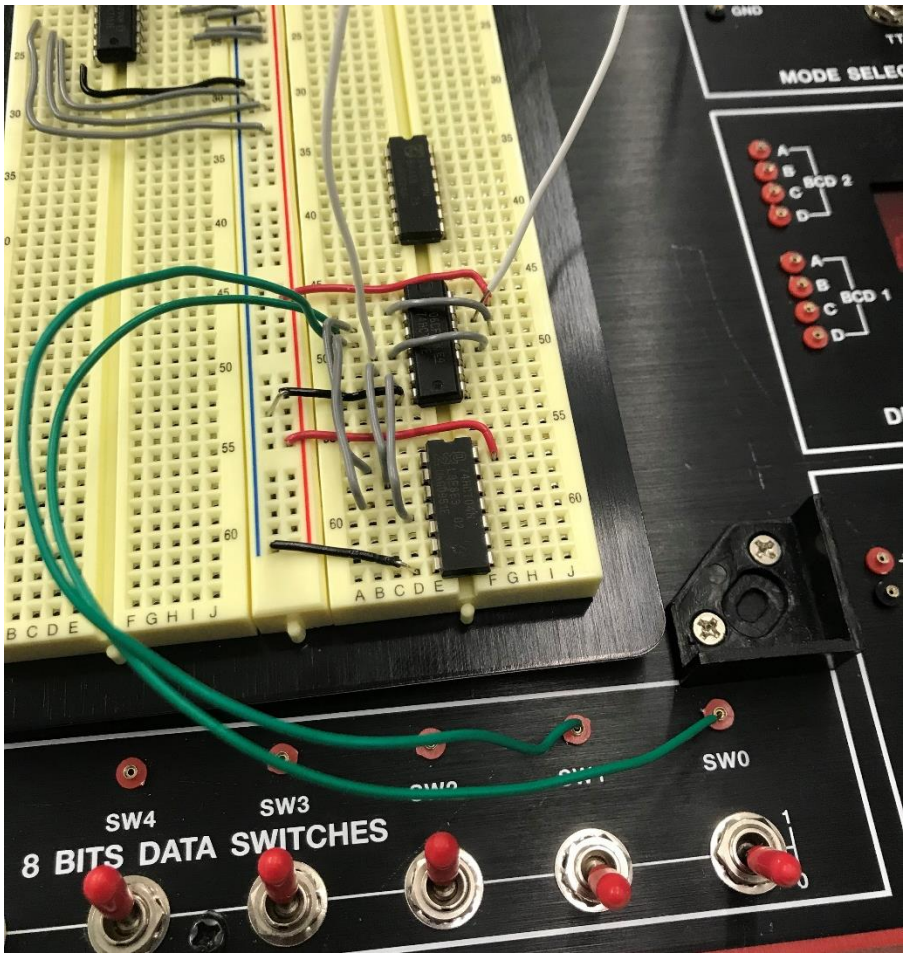
5.3 One bit half adder and Circuitverse simulation (Section 3.4)

I		O	
A	B	SUM	CARRY
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



5.4 Circuit diagram sketch (Section 4.4)





5.5. (i) (Section 2.6)

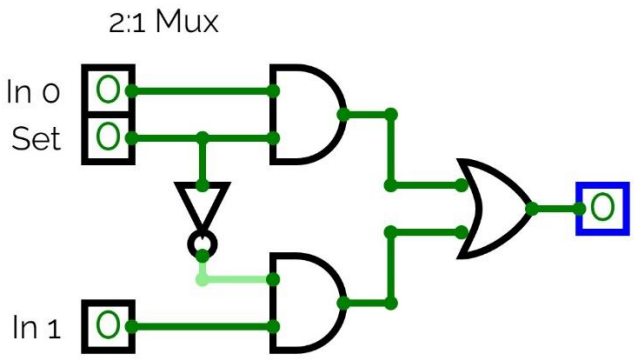
Multiplexers, or MUX's, are logic circuits that act in a similar way to a rotary switches. Each MUX has a single output and multiple inputs called channels, which are controlled by a number of additional inputs called control lines or "input select lines". The control lines act as the switch, connecting one of the channels to the output; what channel is switched depends on the state of the control lines, and what the output signal is depends on the state of the input channel selected. For a 4:1 MUX, which has 4 channels, and 2 control inputs, the truth table is as follows:

4:1 Mux Truth table

B	A	Y
0	0	I_0
0	1	I_1
1	0	I_2
1	1	I_3

Because the control signals only connect each channel to the output, the final output and truth table can be changed by giving each channel HI or LOW signals in order to make the MUX act as a logic gate. For example, to create an AND gate, I_3 (which is switched on when both A and B are HI, just like an AND gate), is given a HI signal, while the other channels will have a LOW. To create an OR gate, every channel except I_0 will have a voltage input.

5.5. (ii)



5.5. (iii)

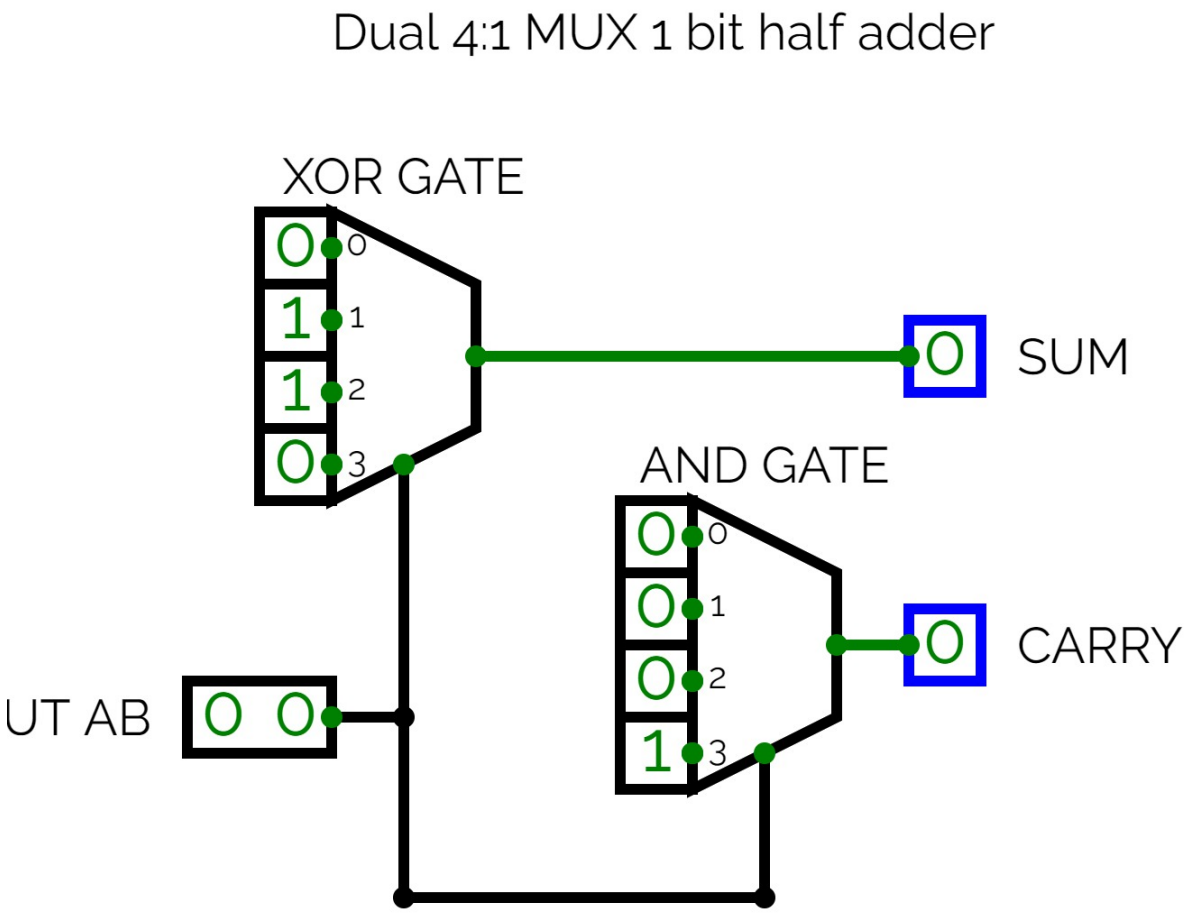
Table 3. Function table

H = HIGH voltage level; L = LOW voltage level; X = don't care.

select Inputs		data inputs				output enable	output
S0	**S1**	**nI0**	**nI1**	**nI2**	**nI3**	**nE̅**	**nY**
X	X	X	X	X	X	H	L
L	L	L	X	X	X	L	L
L	L	H	X	X	X	L	H
H	L	X	L	X	X	L	L
H	L	X	H	X	X	L	H
L	H	X	X	L	X	L	L
L	H	X	X	H	X	L	H
H	H	X	X	X	L	L	L
H	H	X	X	X	H	L	H

The enable inputs basically turn on the mux outputs; as they are active low, the ENABLE inputs need to be pulled to ground for the mux to output a signal.

5.5. (iv) (Section 4.6)



5.6. Prime number detector

5.6. (i)

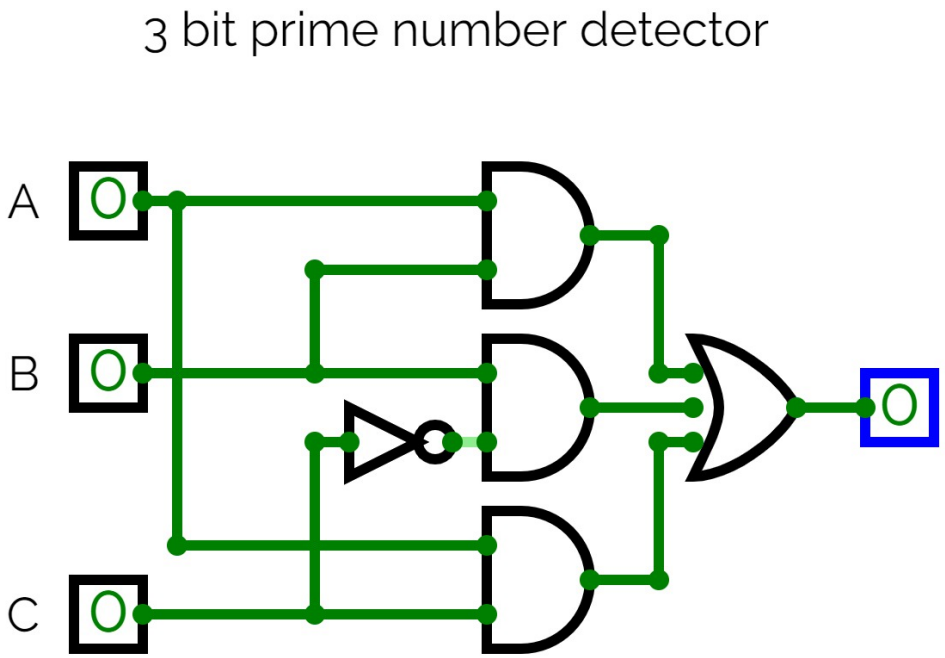
C	B	A	Y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

$$X = \bar{A}B\bar{C} + AB\bar{C} + A\bar{B}C + ABC$$

5.6. (ii)

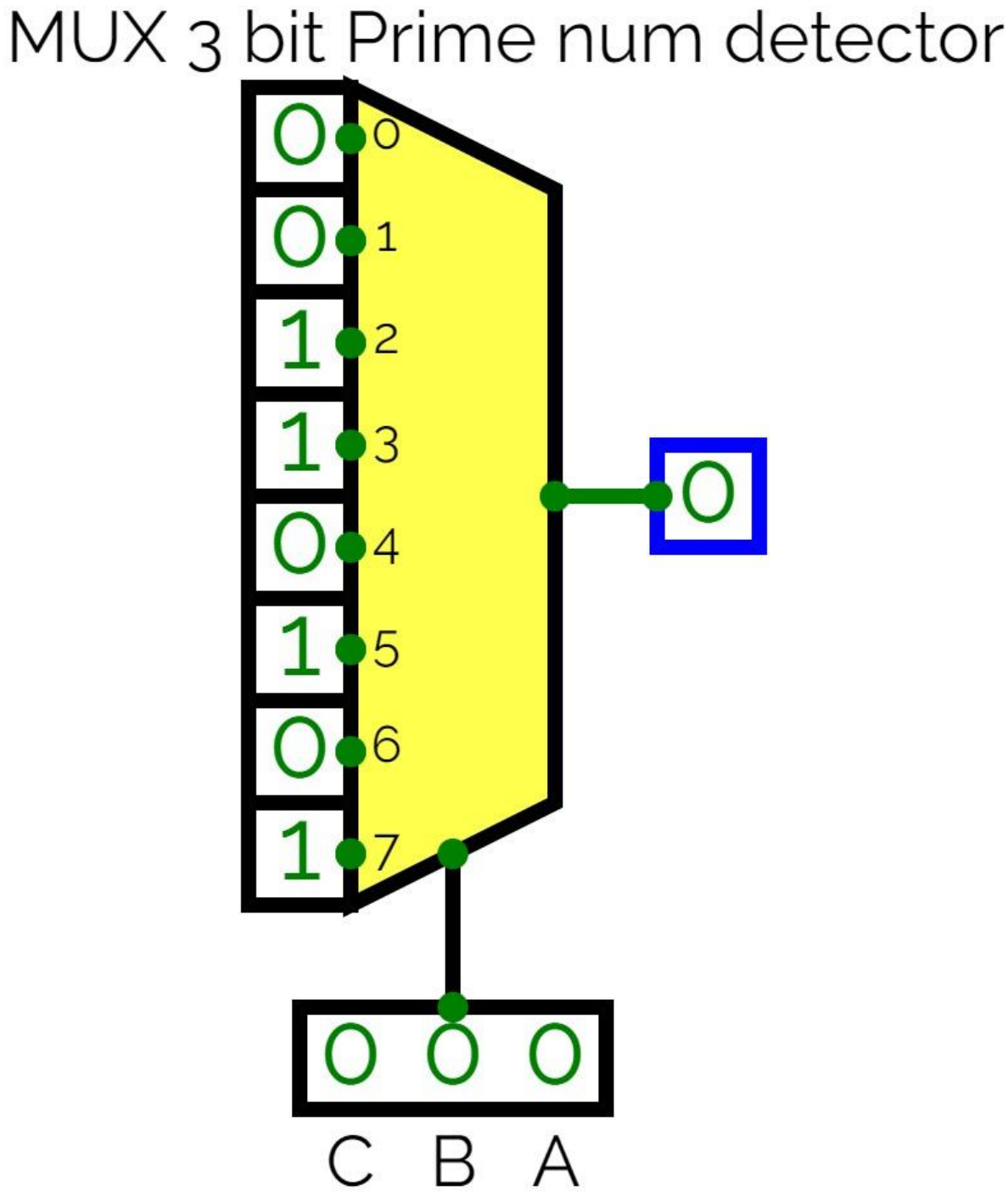
$$X = \bar{A}B\bar{C} + AB\bar{C} + A\bar{B}C + ABC + (AB\bar{C} + ABC)$$
$$X = AB(C + \bar{C}) + B\bar{C}(\bar{A} + A) + AC(\bar{B} + B)$$
$$X = AB + B\bar{C} + AC$$

5.6. (iii)



5.6. (iv)

Instead of using discrete logic gates, a single MUX can be used instead. As a MUX has an input size of 2^n, where n is number of select lines, an 8:1 MUX is needed to create the prime number detector.


MUX 3 bit Prime num detector
0
0
1
1
0
1
0
1
0
1
2
3
4
5
6
7
0
0 0 0
C B A

5.6. (v)