Eli Sobylak
A00317144

9-Oct-2014
Digital Logic: Lab 2 - Adders

Part 1- One Bit Half Adder

1)

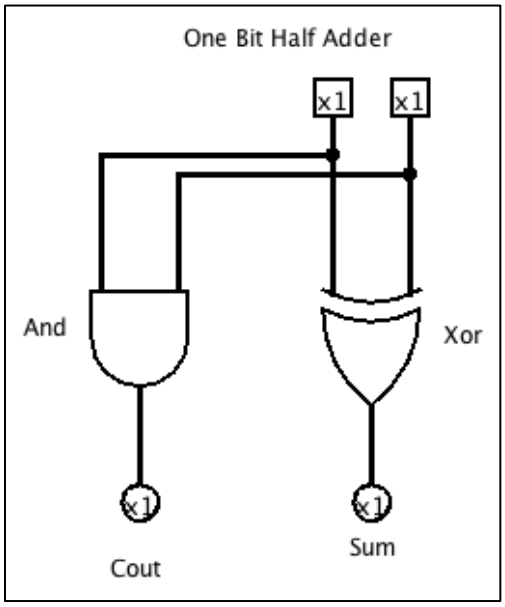
A	B	Sum	Carry out (Cout)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

2)

Sum= A Xor B

Cout= A•B

3)



Eli Sobylak
A00317144

Part 2- One Bit Full Adder

1)

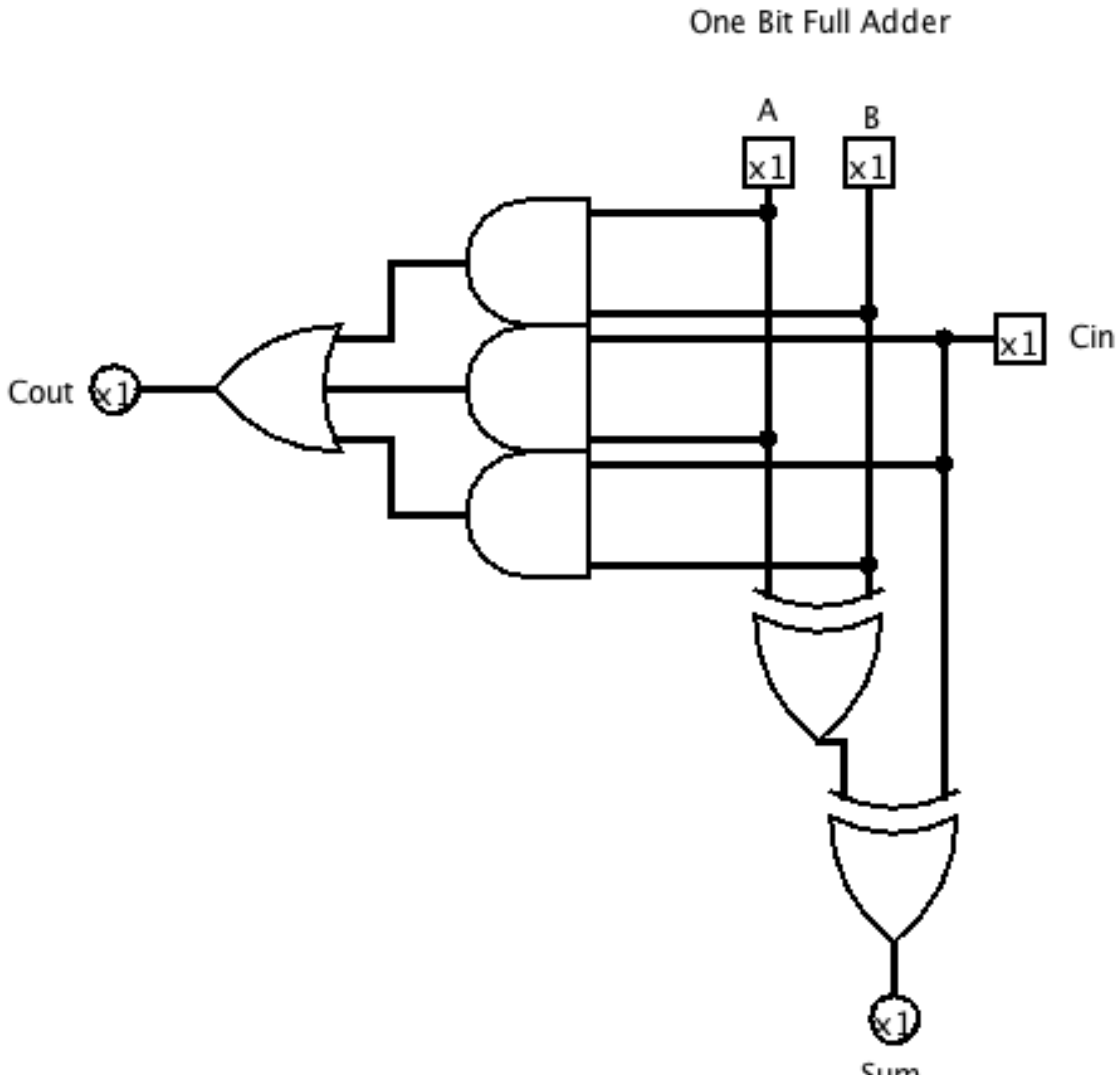
A	B	Carry in (Cin)	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

2)

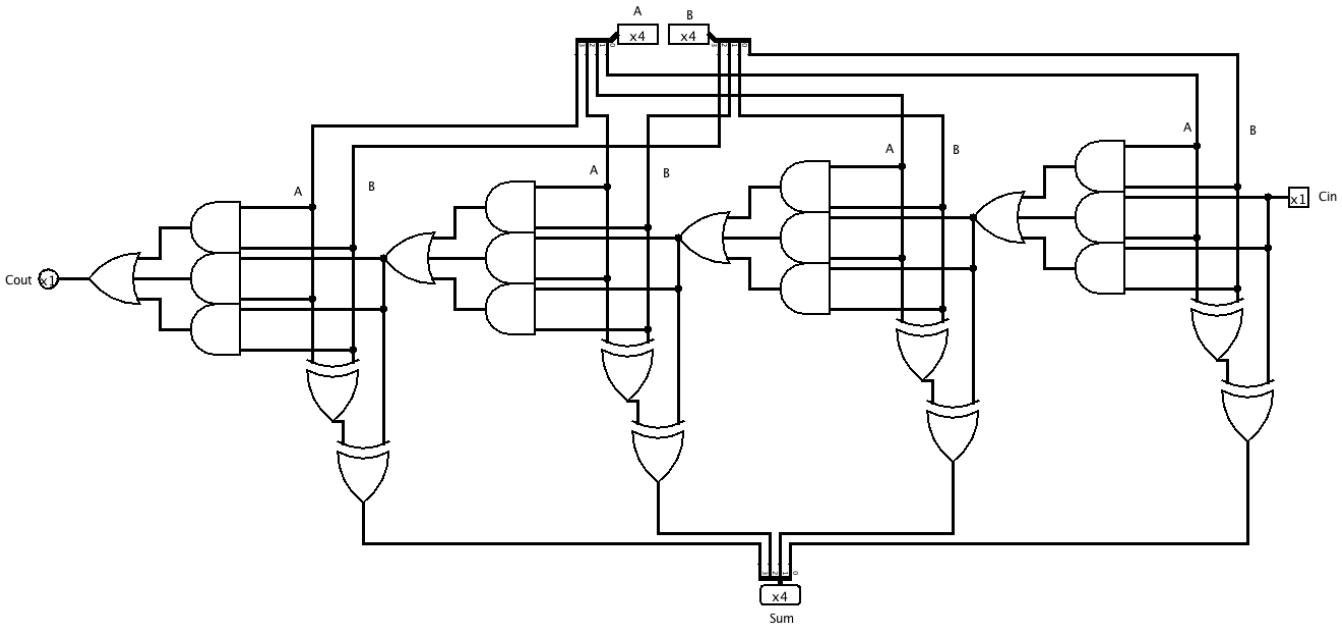
Sum= !A•!B•C + !A•B•!C + A•!B•!C + A•B•C

Cout= A•B + A•C + B•C

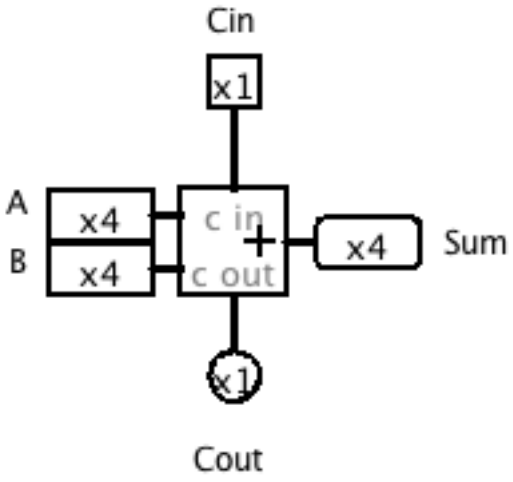
3)



Part 3- 4-bit Adder



Part 4- Logisim 4-bit Adder



Questions about a 4-bit Adder

Answer the following question about a 4-bit adder without consulting your circuit. Unless otherwise indicated, give your numerical answers in decimal. After you have filled out the chart, use your 4-bit adder to check your answers. Make sure you understand and can calculate all the answers correctly.

1. What is the range of unsigned numbers that you can represent in 4 bits?

We can represent 0 through 15.

2. Fill out the following table of sums, carry, and borrow that your 4-bit adder circuit will give. Assume unsigned representation of numbers in 4 bits.

Binary A input	Binary B input	Binary sum	Decimal A input	Decimal B input	Decimal Sum	Carry (0 or 1)
0000	0111	0111	0	7	7	0
1100	0101	0001	12	5	17	1
0101	0101	1010	5	5	10	0
1111	1111	1110	15	15	30	1
0010	0110	1000	2	6	8	0

3. Assuming unsigned 4-bit representation of numbers, under what conditions does adding produce a result that is not meaningful with respect to normal addition and the constraint of only 4 bits to hold the sum?

A 4-bit adder will result in non meaningful answers when you add two numbers that yield an answer greater than 15. This is because a 4-bit adder can only output up to the number 15. After 15, we start to get a Carry out.

4. What does the carry out pin signify?

The Carry out pin signifies the next bit place after what can be displayed. EX| the result 17 will be displayed as 0001 with a Cout of 1, since $17_{10} = 10001_2$

5. Assuming unsigned 4-bit representation of numbers, what does your 4-bit adder produce if you try to add two numbers whose sum exceeds the 4-bit range of values? Give an arithmetic expression for the unsigned value of the sum bits in terms of x and y input values (use mod; if you don't know what mod is, then hold this question until later).

If you try to add two numbers whose sum exceeds the 4-bit range it starts to roll over again from the beginning and carrying out a remainder.

Sum			
Binary	Decimal	What's Displayed	Carry out(Cout)
1110	14	1110	0
1111	15	1111	0
0000	16	0	1
0001	17	1	1
0011	18	10	1

Can use the expression $f(x, y) = x + y\ (mod 16)$