Computer Organiation Laboratory

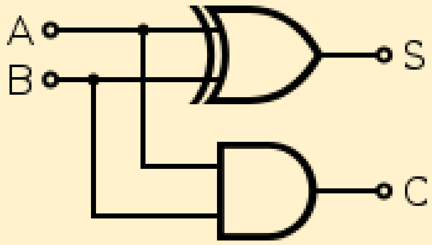
CS39001

Verilog Assignment 1

Group 15

Gaurav Malakar	20CS10029
Prakhar Singh	20CS10045

Half-Adder



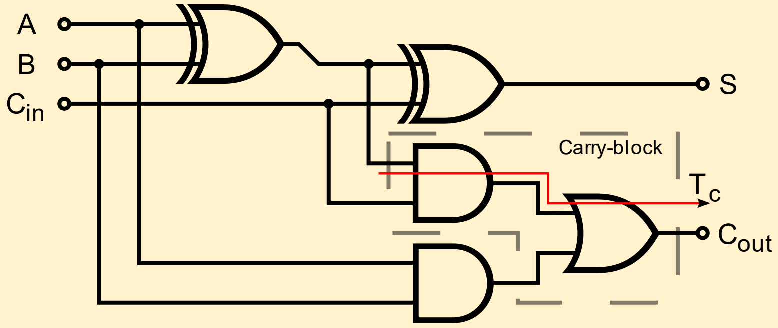
Inputs		Outputs	
a	b	s	c
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

The boolean expressions for the above table are:

$$\mathrm{s} = \mathrm{a} \oplus \mathrm{b}$$

$$c = a \,\&\, b$$

Full-Adder



Inputs			Outputs	
a	b	c_0	s	c
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
1	0	0	1	0
0	1	1	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

The boolean expressions for the above table are:

$$s = a \oplus b \oplus c_0$$

$$c = a \,.\, b + b \,.\, c_0 + c_0 . a$$

Synthesis Summary

Circuit	Delay (in ns)
8-bit RCA	3.471
16-bit RCA	6.167
32-bit RCA	11.559
64-bit RCA	22.343

Question: How can you use the above circuit, to compute the difference between two n-bit numbers?

Solution: Given two n-bit numbers, x and y, we want to calculate x - y using a Ripple Carry Adder. For this, we can use an n-bit Ripple Carry Adder.

We know,

x - y = x + (-y)

Here, (-y) is the 2's complement of y

2's complement of y = ~y + 1

So, we can calculate x - y by giving the following inputs to our n-bit RCA:

input1 = x,

input2 = ~y,

carry-in = 1

To provide ~y as an input, we can put NOT gates in all the input ports of y and a NOT gate in the input port of carry-in (as carry-in is provided as 0 generally). A more elegant way to do this will be to connect all these NOT gates via a switch such that if the switch is on, the inputs are received through NOT gates. Otherwise, we get the inputs normally. This will allow us to calculate both x-y and x+y in the same circuit, depending on whether the switch is on or off.

NOTE: Here we are using signed numbers for input and output.