

# CS 232 Lab 3 Q3

Arya Amin 200050014

1st February 2022

## **4-bit ripple carry adder-subtractor( FourbitRipCarAddSub.vhd)**

I have designed the 4-bit ripple carry adder-subtractor where 'a' and 'b' are the two 4-bit unsigned numbers that are to be added. Here 'cin' is not an input carry like in the previous question but it is a mode selection bit. If cin=0 then 'a' and 'b' should be added (a+b). If cin=1 then 'b' should be subtracted from 'a' (a-b). 'sum' is the 4-bit unsigned addition or subtraction output and 'cout' is the single-bit output carry using the 4-bit ripple carry adder designed in the previous question and basic gates.

I have taken XOR of b(4-bit logic vector) with cin(0 or 1)) and then stored the output inside d(4-bit logic vector), So when b is same as cin d will result in 0 else d will be 1.

The 4-bit ripple carry adder-subtractor Logical Expression is as follows,

$d = \text{cin XOR } b$  i.e.,

$d(0) = \text{cin XOR } b(0),$   
 $d(1) = \text{cin XOR } b(1),$   
 $d(2) = \text{cin XOR } b(2),$   
 $d(3) = \text{cin XOR } b(3)$

$\text{sum} = \text{FourbitRipCarAddSum}(a, d, \text{cin}),$   
 $\text{cout} = \text{FourbitRipCarAddcout}(a, d, \text{cin})$

Here d is stdlogicvector (3 downto 0) signal.

For implementing the XOR Gate I have designed xorgate entity inside the main entity where I have used basic OR, NOT and AND Gates to design the XOR Gate and then I used these instance of XOR to design the 4-bit ripple carry adder-subtractor

Therefore using the above shown way I created the 4-bit ripple carry adder-subtractor using only the 4-bit ripple carry adder and basic gates.

XOR = 4  
FourbitRipCarAdder = 1

Inside each XOR gate,  
OR = 2  
AND = 1  
NOT = 2