Computer Organization and Architecture Laboratory

Assignment 3 - Group Number 70

Kushaz Sehgal - 20CS30030   Jay Kumar Thakur - 20CS30024

1. Design of a Ripple Carry Adder

(a) Half Adder

A **Half Adder** is a combinational arithmetic circuit that adds two bits (a and b) and produces a sum bit (S) and a carry bit (S) both as output.

Truth Table

Input		Output	
a	b	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

Boolean Expressions

$$S = a \oplus b \qquad C = a \cdot b$$

Module File : HalfAdder.v

Test File : HalfAdderTest.v

Circuit Diagram

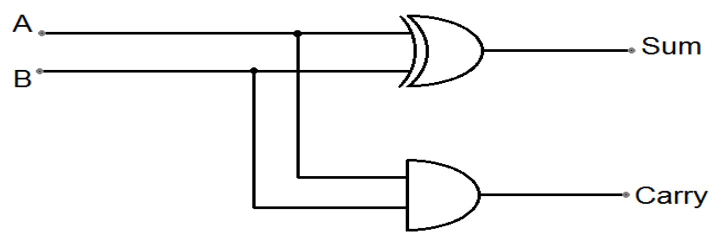

Figure 1 : Half Adder

(b) Full Adder

A **Full Adder** is a combinational arithmetic circuit that takes in 3 bits ('a' , 'b' and previous carry 'c') and produces a sum bit (S) and a carry bit (S) both as output.

Truth Table

Input			Output	
a	b	c	S	C
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	1	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

Boolean Expressions

$$S = a \oplus b \oplus c \qquad C = a \cdot b + b \cdot c + c \cdot a$$

Circuit Diagram

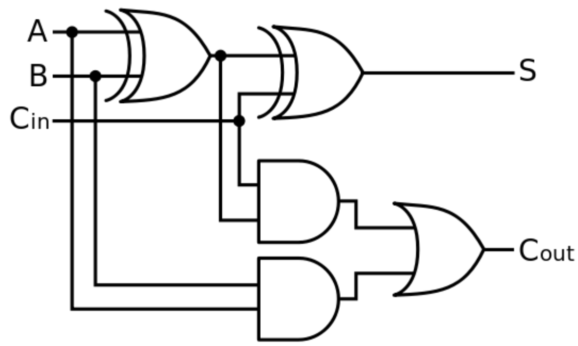

Figure 2 : Full Adder

Module File : FullAdder.v **Test File** : FullAdderTest.v

(c) Ripple Carry Adder

A **n-bit Ripple Carry Adder** is a combinational arithmetic circuit that takes in 2 n-bit numbers (a,b) and produces a sum bit (S) and a carry bit (C) both as output.

We first create a **8-bit Ripple Carry Adder** by cascading **8 Full Adders**. We then cascade two 8-bit Ripple Carry Adders to create a 16-bit Ripple Carry Adder. We similarly follow the procedure to create a **32-bit Ripple Carry Adder** and a **64 bit Ripple Carry Adder**.

Module File : RCA_8bit.v **Test File** : RCA_8bit_Test.v

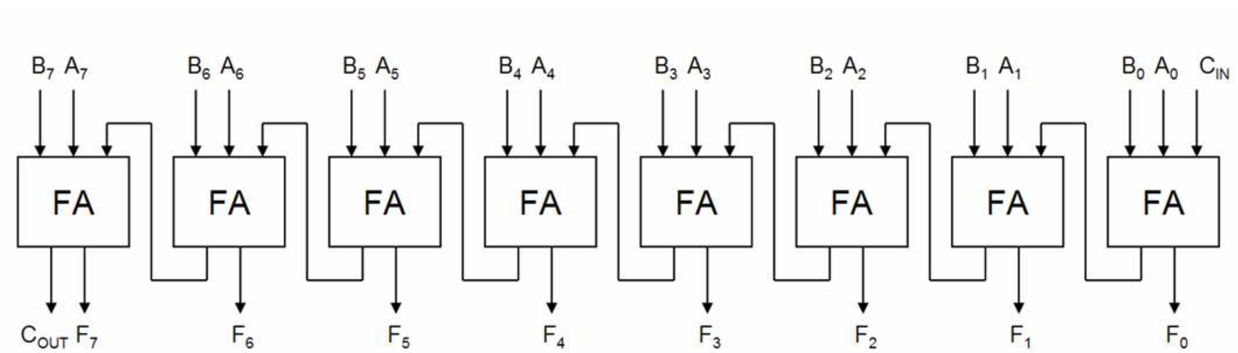

Figure 3 : 8-bit RCA

Module File : RCA_16bit.v **Test File** : RCA_16bit_Test.v

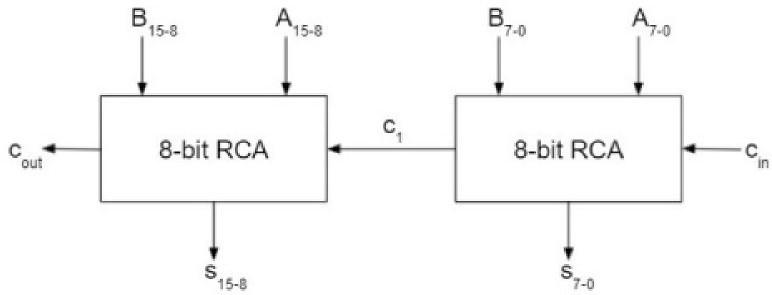

Figure 4 : 16-bit RCA

Module File : RCA_32bit.v **Test File** : RCA_32bit_Test.v

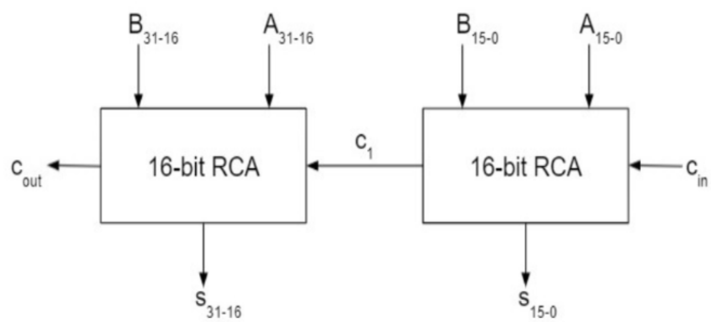

Figure 5 : 32-bit RCA

Module File : RCA_64bit.v **Test File** : RCA_64bit_Test.v

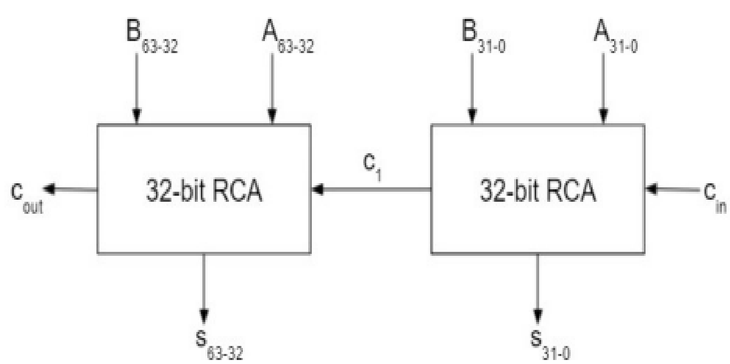

Figure 6 : 64-bit RCA

Summary of Synthesis Report

Circuit	Delay (logic, route) in ns	Logic Level	#Slice LUTS	#Bounded IOBS
8BIT_RCA	3.471 ns (0.497, 2.974)	6	12	26
16BIT_RCA	6.167 ns (0.993, 5.174)	10	24	50
32BIT_RCA	11.559 ns (1.985, 9.574)	18	48	98
64BIT_RCA	22.343 ns (3.969, 18.374)	34	96	194

(d) Ripple Carry Adder

Given two n-bit binary numbers a and b , a Ripple Carry Adder calculates $a + b$.

Also, $a - b$ can be rewritten as $a + (-b)$. ($-b$ is the 2's complement of b)

$(-b) = \sim b + 1$. ($\sim b$ is the 1's complement of b)

$Consider\ RCA(a, \sim b, 1)$ -

$RCA(a, \sim b, 1) = a + (\sim b) + 1 = a + (\sim b + 1) = a + (-b)$

So $a - b = RCA(a, \sim b, 1)$

Thus, $a - b$ can be calculated by flipping each bit of b by a XOR operation with 1 and and sending 1 as carry.

Module File : RCA_subtractor_64bit.v **Test File** : RCA_subtractor_64bit_Test.v