Computer Organization and Architecture Laboratory

Assignment 3 (Verilog)

Group 63:
Pranav Nyati - 20CS30037
Shreyas Jena - 20CS30049

Part 1:
Design of Ripple Carry Adders for 8, 16, 32, and 64 bits:

(a) Half Adder:

- **Inputs: in1** and **in2 (both are 1 bit inputs)**
- **Outputs: sum** bit and **c_out** bit **(carry out)**
- **Truth Table:**

Inputs		Outputs	
in1	in2	sum	c_out
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Sum = in1 XOR in2; c_out = in1 AND in2

- **Figure:**

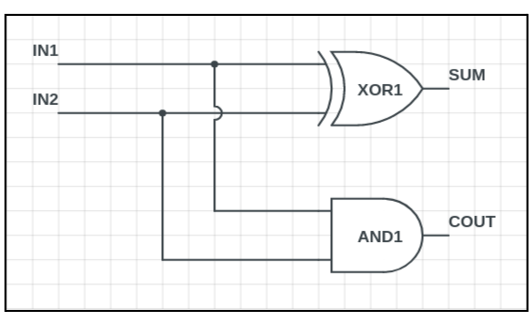

Fig 1. Logic Diagram of Half Adder

(b) Full Adder:

- **Inputs: in1, in2, c_in (all are 1 bit)**
- **Outputs: sum** bit and **c_out** bit
- **Truth Table:**

Inputs			Outputs	
in1	**in2**	**c_in**	**sum**	**c_out**
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

Sum = in1 XOR in2 XOR c_in;
c_out = (in1 AND in2) OR (in1 AND c_in) OR (in2 AND c_in)

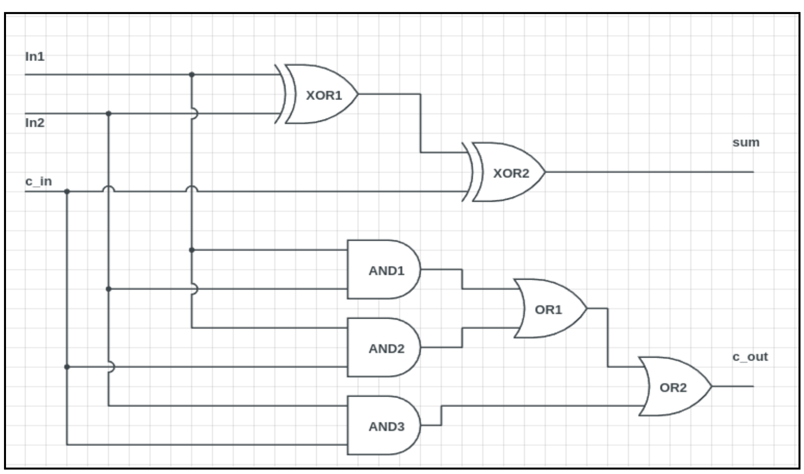

Fig 2. Logic Diagram of Full Adder

(c) Ripple Carry Adder (RCA):

We can create a 8-bit RCA by cascading 8 full adder circuits into a single circuit. Similarly, we can create a 16-bit RCA by cascading two 8-bit RCA modules, a 32-bit RCA by cascading two 16-bit RCA modules, and a 64-bit RCA by cascading two 32-bit RCA modules.

- **8-bit RCA:**

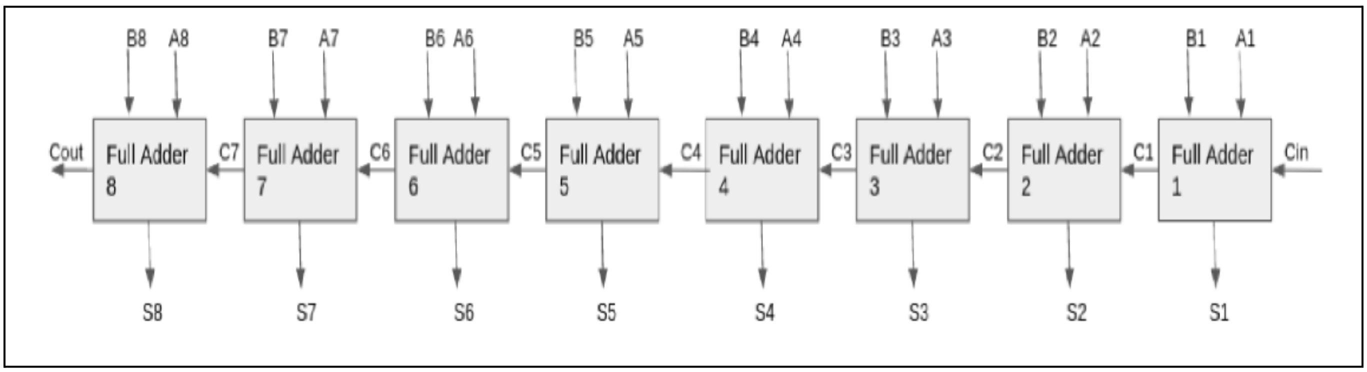

Fig 3. Logic Diagram of 8-bit RCA using 8 full adders in a cascading manner

- **16-bit RCA:**

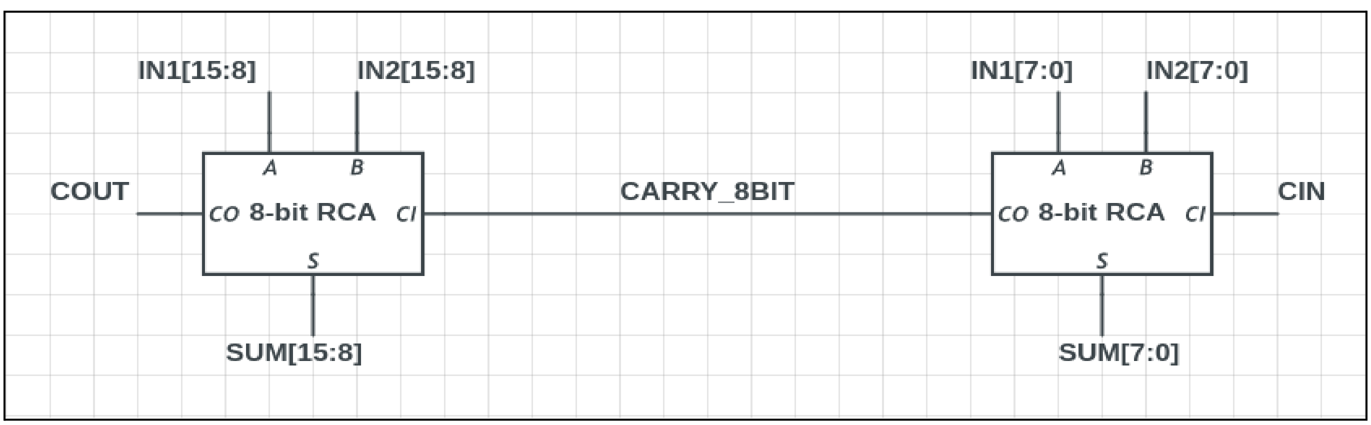

Fig 4. Logic Diagram of 16-bit RCA using two 8-bit RCA in succession

- **32-bit RCA:**

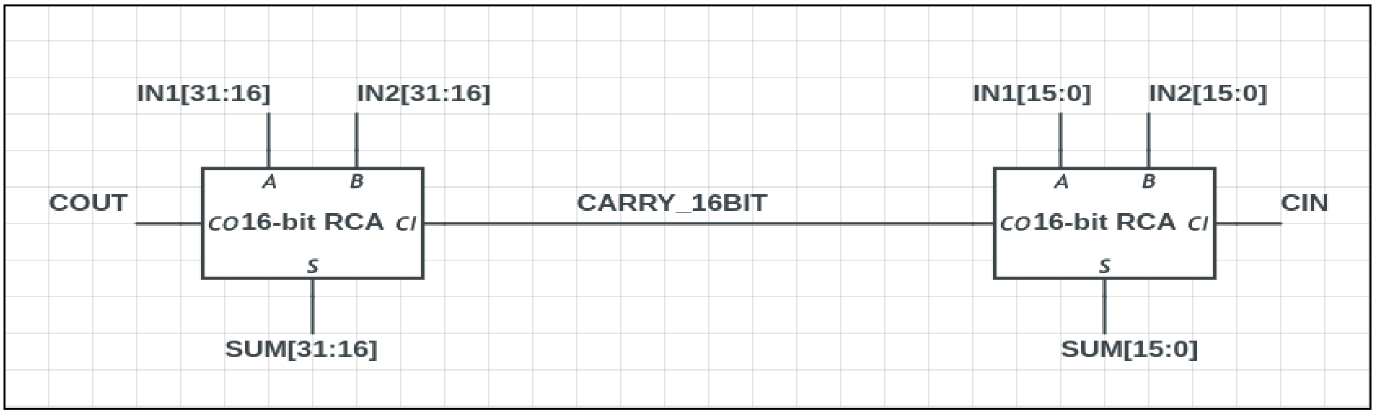

Fig 5. Logic Diagram of 32-bit RCA using two 16-bit RCA in succession

- **64-bit RCA:**

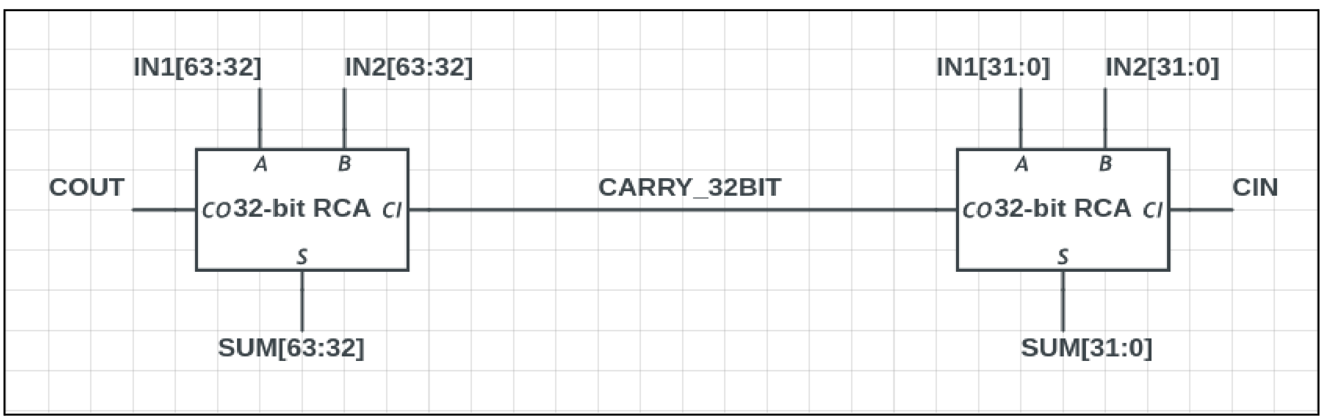

Fig 6. Logic Diagram of 64-bit RCA using two 32-bit RCA in succession

TIME DELAYS:

Observed longest timing delays:

- **8-bit RCA:** **3.471ns** (0.497ns logic, 2.974ns route)
- **16-bit RCA:** **6.167ns** (0.993ns logic, 5.174ns route)
- **32-bit RCA:** **11.559ns** (1.985ns logic, 9.574ns route)
- **64-bit RCA**: **22.343ns** (3.969ns logic, 18.374ns route)

(d) How can you use the above circuit, to compute the difference between two n-bit numbers?

Soln:-

An n-bit RCA in a general case takes two inputs in1 and in2 with the carry-in bit = 0, and calculates the sum in1 + in2. In order to calculate the difference of two n-bit numbers in1 - in2 (let us assume that in1 > in2) , we can pass the first n-bit number as in1, the 2nd n-bit number as 1's complement of in_2 (i.e, do bit wise complement of each bit in in2, and pass this 1's compliment as in_2), and the initial carry_in bit as 1.

The result will be:

Res = (in1 + 1's complement of in2 + 1)
= (in1 + (1's complement of in2 + 1))
= (in1 + 2's complement of in2)
= (in1 + (-in2))
= (in1 - in2)
