# INDIAN INSTITUTE OF TECHNOLOGY KHARAGPUR

# COMPUTER ORGANIZATION LABORATORY

## ASSIGNMENT 1

INTRODUCTION TO VERILOG

# GROUP MEMBERS

## GROUP 54:

1) Gaurav Madkaikar 19CS30018

2) Girish Kumar 19CS30019

# DESIGN OF RIPPLE CARRY ADDER

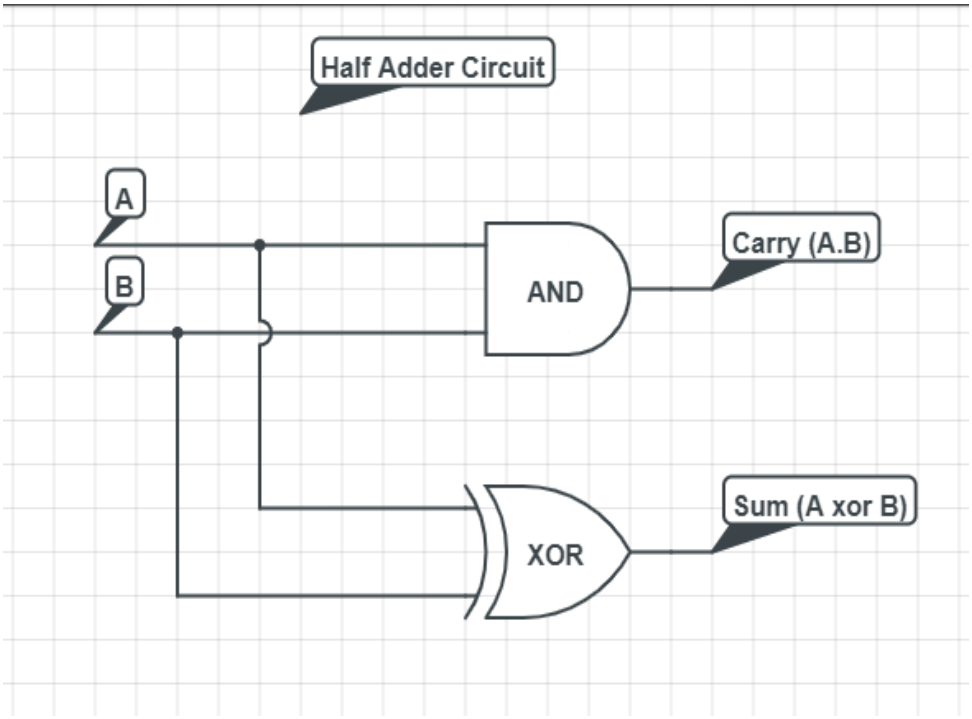
## A) Half Adder

It is a combinational circuit, which takes in two single bit inputs- A and B, and produces the sum bit (S) and the output carry bit ($C_{out}$).

## Truth Table

| Inputs | | Outputs | |
|---|---|---|---|
| **A** | **B** | **S (sum)** | **$C_{out}$ (carry)** |
| 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 0 | 1 |

## Circuit Diagram



## B) Full Adder

It is a combinational circuit, which takes in three single-bit inputs- A, B and $C_{in}$, and produces the corresponding sum bit (S) and the output carry bit ($C_{out}$).

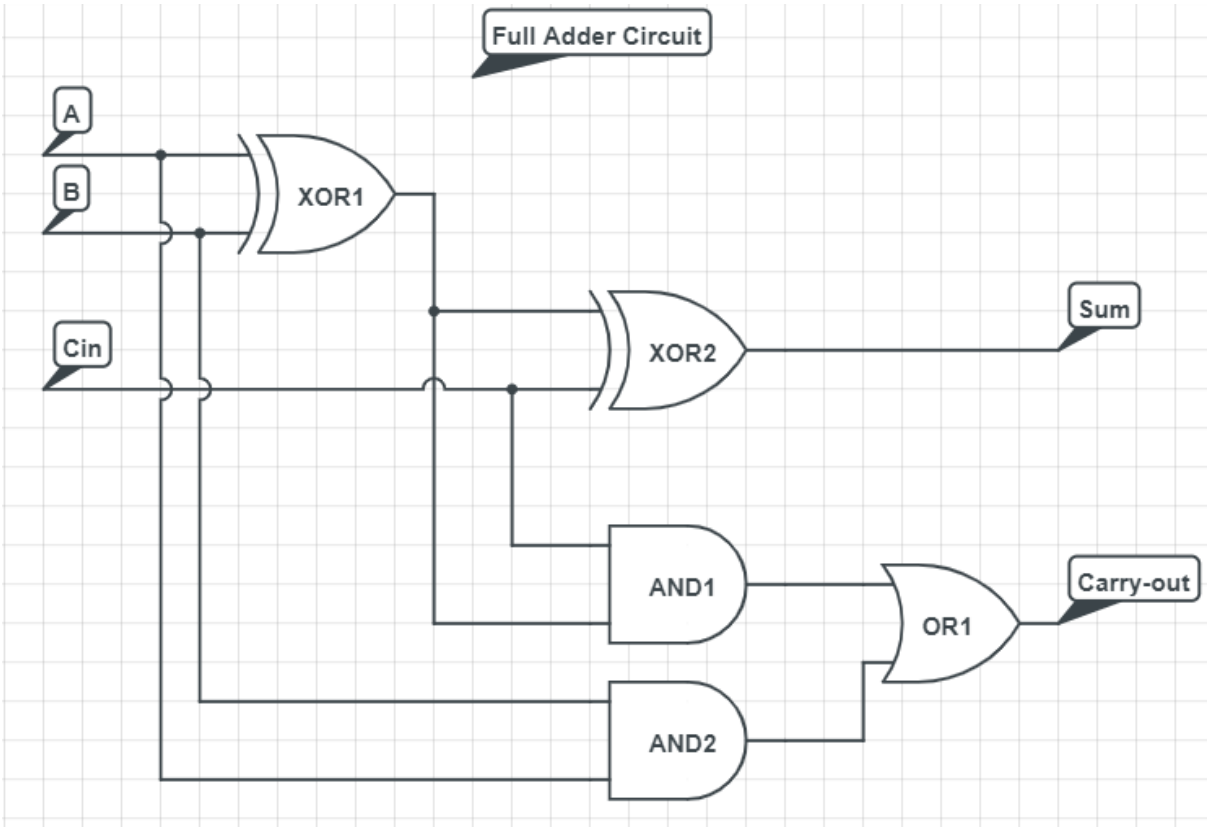
The corresponding outputs can be computed as follows:

$S = (A \wedge B) \wedge C_{in}$
$C_{out} = AB + BC_{in} + AC_{in}$

## Truth Table

| Inputs | | | Outputs | |
|---|---|---|---|---|
| **A** | **B** | **$C_{in}$ (carry-in)** | **S(sum)** | **$C_{out}$(carry-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 |

## Circuit Diagram



## C) Ripple Carry Adder

**Ripple Carry Adder** is a combinational circuit used to add two n-bit numbers. A n-bit RCA consists of *n* 1-bit full adders arranged sequentially. The carry-out produced by a full adder serves as the carry-in for its adjacent most significant full adder.

The delay associated with ripple carry adder is typically higher than other parallel adders as each sum bit is dependent on the previous carry-out bit. As a result, the total delay associated with the circuit is proportional to the number of input bits.

The total delay associated with each *ripple carry adder* are stated as follows: (Corresponding RCA's have Verilog codes)

8-bit RCA → 5.908 ns (18 logic levels)

16-bit RCA → 11.236 ns (36 logic levels)

32-bit RCA → 21.892 ns (72 logic levels)

64-bit RCA → 43.204 ns (144 logic levels).

As expected, the total delay corresponding to higher bit RCAs is much higher as compared to the delay of low bit RCAs due to the rippling effect seen at each full adder. Thus, the more the full adders used, the greater will be the rippling effect and thus greater will be the total delay.

## D) Adder/Subtractor Circuit using RCA

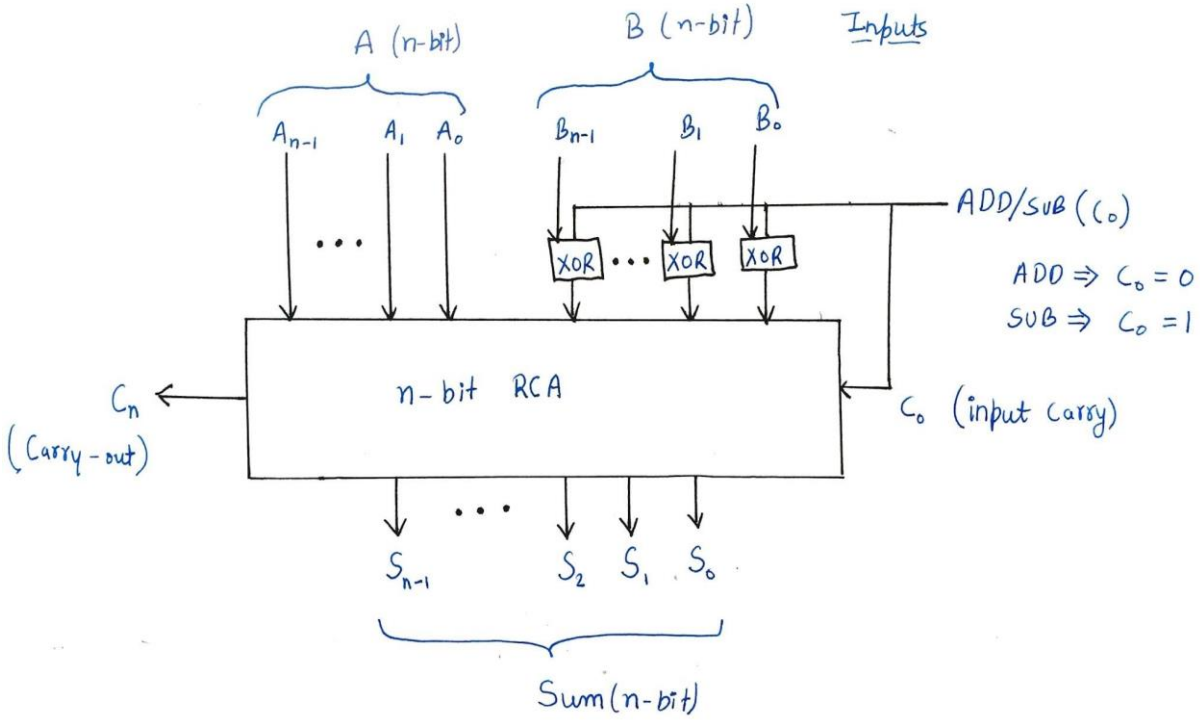
A n-bit ripple carry adder can be used to subtract bits by making a slight modification in the design as shown in the circuit diagram below. Depending on the value of carry-in, the circuit can be used to either add or subtract the given *n-bit* numbers.

Manipulate B such that it is XORed with the carry-in ($C_{in}$). For $C_{in}$ = 1, the XOR operation results in 1's complement of B.

If $C_{in}$ = 0, sum = A + B

$C_{in}$ = 1, sum = A + (1's complement of B + $C_{in}$) = A - B

## Circuit Diagram



# DESIGN OF CARRY LOOK AHEAD ADDER

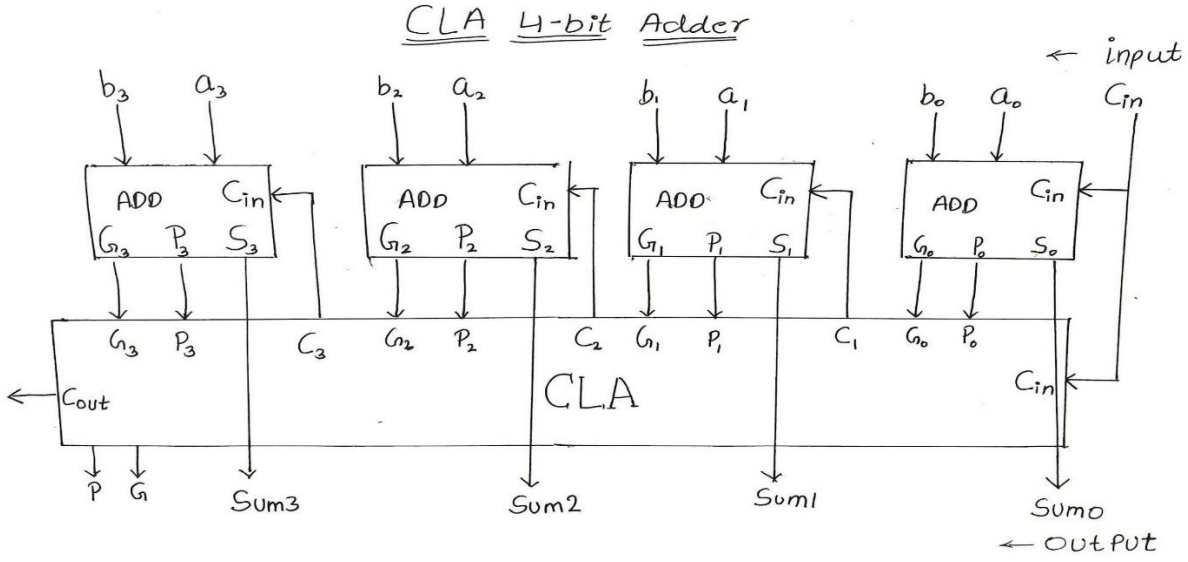
## A) 4-bit Carry Look Ahead (CLA) Adder

Boolean-equations for generate, propagate and output carry bits of a 4-bit ***Carry Look Ahead*** adder:

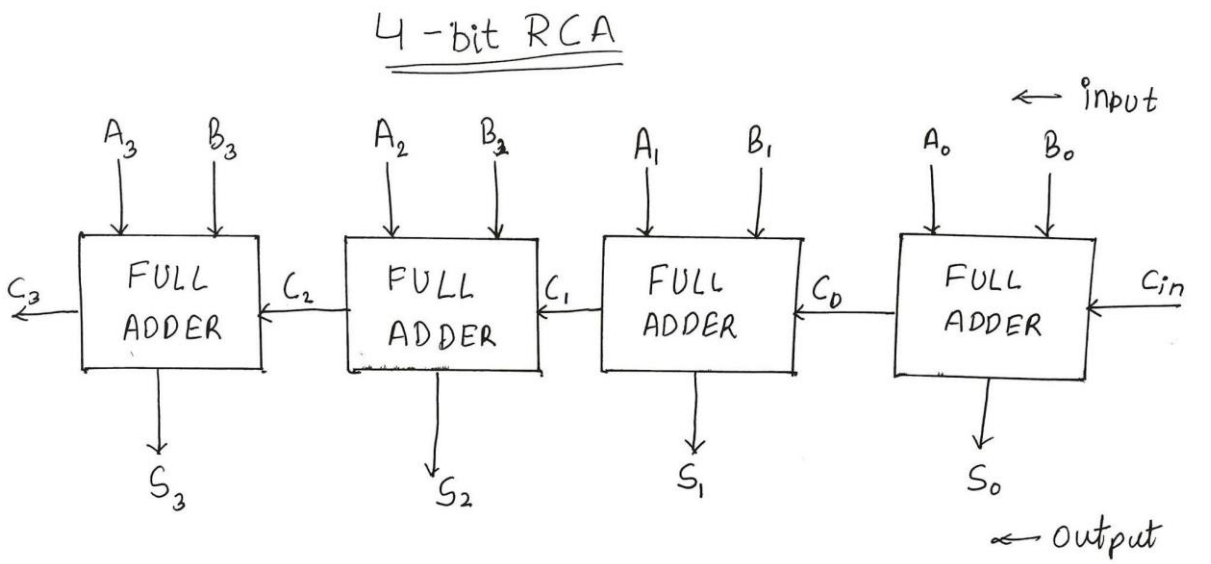
$P_i = A_i \text{ ^ } B_i$ for, i = {0, 1, 2, 3}
$G_i = A_i \& B_i$ for, i = {0, 1, 2, 3}

$C_1 = G_0 + C_0P_0$
$C_2 = G_1 + G_0P_1 + C_0P_0P_1$
$C_3 = G_2 + G_1P_2 + G_0P_1P_2 + C_0P_0P_1P_2$
$C_4 = G_3 + G_2P_3 + G_1P_2P_3 + G_0P_1P_2P_3 + C_0P_0P_1P_2P_3$

## Circuit Diagram (4-bit CLA)



## Circuit Diagram (4-bit RCA)




## B) Comparison of 4-bit RCA and 4-bit LCA

As can be seen from the circuit diagrams, the carry-output associated with the 4-bit CLA can be computed parallelly using the generate and propagate bits in constant time. As a result, all the sum bits can be generated sequentially without additional delay associated with the carry bits.

Whereas, in the 4-bit RCA we see that computation of each of the sum bits is dependent on the previous carry bit. As a result, the delay for calculating sum bits increases at a rate proportional to the number of adders in the circuit.

Observed total delays:

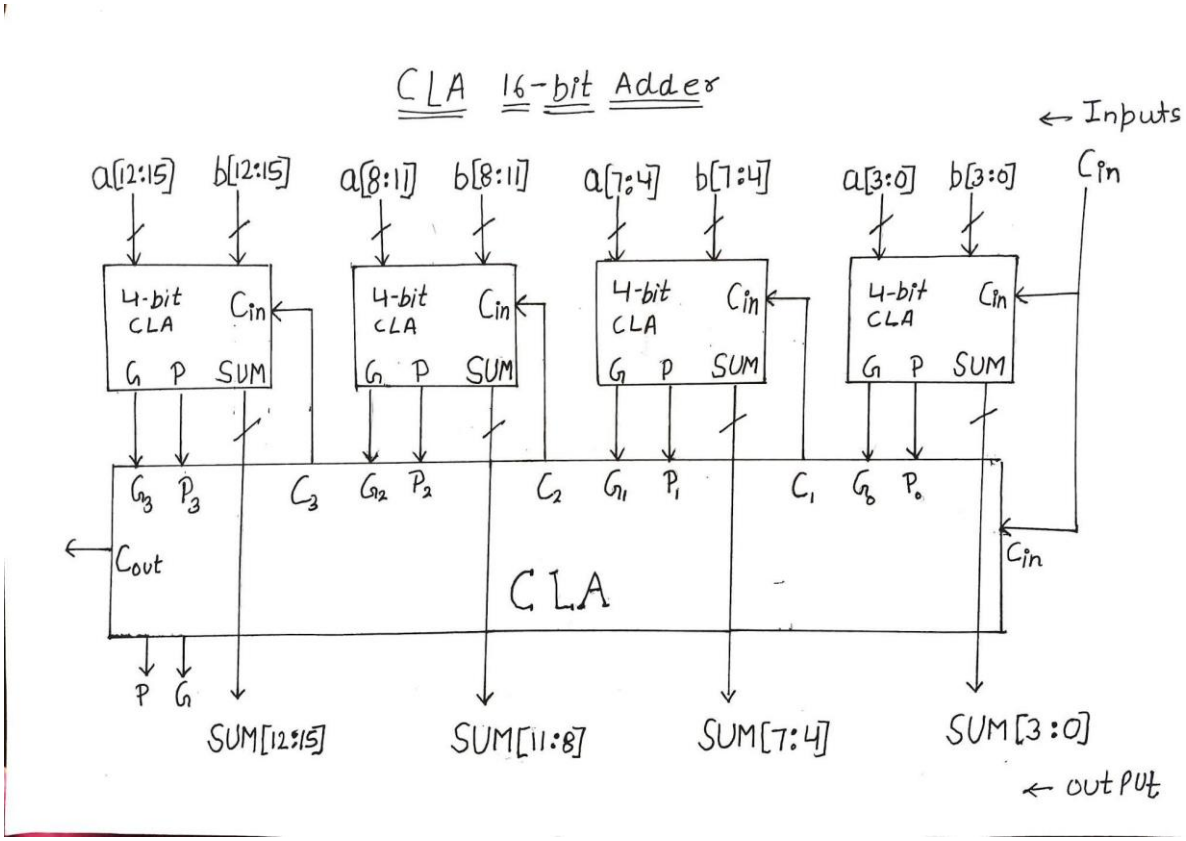
4-bit RCA → 5.908 ns (18 logic levels)
4-bit CLA → 2.586 ns (5 logic levels)

As expected, we see that the 4-bit CLA easily outperforms the 4-bit RCA in terms of the total delay. As the number of bits increase, even though the CLA is highly efficient, the hardware becomes much more complex.

## C) 16-bit Carry Look Ahead (CLA) Adder

**16-bit CLA** consists of two major components- 1) four 4-bit CLA units for generating the propagate and generate bits, 2) combinational block known as the **Look Ahead Unit** that computes all the carry bits parallelly with the help of propagate and generate bits. These carry bits are passed sequentially to process the final output sum.

## Circuit Diagram (16-bit CLA)



The delays associated with the sum and carry-bits generated in the 2 16-bit CLAs are stated as follows:

16-bit CLA (with the **Look Ahead Unit**) → 5.627 ns (10 logic levels)
16-bit CLA (with rippling carry bits) → 8.604 ns (18 logic levels)

As expected, we see that the **Look Ahead Unit** helps in computing the carry bits in constant time as compared to the O(n) rippling operation involved in the CLA without the **Look Ahead Unit**.

## D) Comparison of 16-bit RCA and 16-bit LCA

**16-bit CLA (Synthesis Details):**

*Total Delay (Maximum combinational path delay):*

5.627ns (Levels of Logic = 10)
(0.745ns logic, 4.882ns route)

- Look-Up Table (16-bit CLA)

```
Primitive and Black Box Usage:
------------------------------
# BELS                          : 43
#      LUT2                    : 1
#      LUT3                    : 10
#      LUT4                    : 9
#      LUT5                    : 22
#      LUT6                    : 1
# IO Buffers                   : 52
#      IBUF                    : 33
#      OBUF                    : 19

Device utilization summary:
---------------------------

Selected Device : 7a100tcsg324-1


Slice Logic Utilization:
 Number of Slice LUTs:                   43  out of  63400     0%
    Number used as Logic:                43  out of  63400     0%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:     43
   Number with an unused Flip Flop:      43  out of     43   100%
   Number with an unused LUT:             0  out of     43     0%
   Number of fully used LUT-FF pairs:     0  out of     43     0%
   Number of unique control sets:         0

IO Utilization:
 Number of IOs:                          52
 Number of bonded IOBs:                  52  out of    210    24%

Specific Feature Utilization:
```

**16-bit RCA (Synthesis Details):**

*Total Delay (Maximum combinational path delay):*

11.236ns (Levels of Logic = 36)
(1.985ns logic, 9.251ns route)

- Look-Up Table (16-bit RCA)

```
Primitive and Black Box Usage:
------------------------------
# BELS                          : 32
#      LUT3                    : 32
# IO Buffers                    : 50
#      IBUF                    : 33
#      OBUF                    : 17

Device utilization summary:
---------------------------

Selected Device : 7a100tcsg324-1


Slice Logic Utilization:
 Number of Slice LUTs:                 32  out of  63400     0%
    Number used as Logic:              32  out of  63400     0%

Slice Logic Distribution:
 Number of LUT Flip Flop pairs used:   32
   Number with an unused Flip Flop:    32  out of     32   100%
   Number with an unused LUT:           0  out of     32     0%
   Number of fully used LUT-FF pairs:   0  out of     32     0%
   Number of unique control sets:       0

IO Utilization:
 Number of IOs:                        50
 Number of bonded IOBs:                50  out of    210    23%

Specific Feature Utilization:
```

As can be seen from the comparison, 16-bit CLA uses a higher number of slice LUTs (43 v 32) and IO buffers (52 v 50) in comparison to the 16-bit RCA. The complex hardware used in the 16-bit CLA results in a considerably lesser time delay in obtaining the final outputs when compared to the 16-bit RCA.