



Advanced VLSI Design: 2021-22

Lecture 9

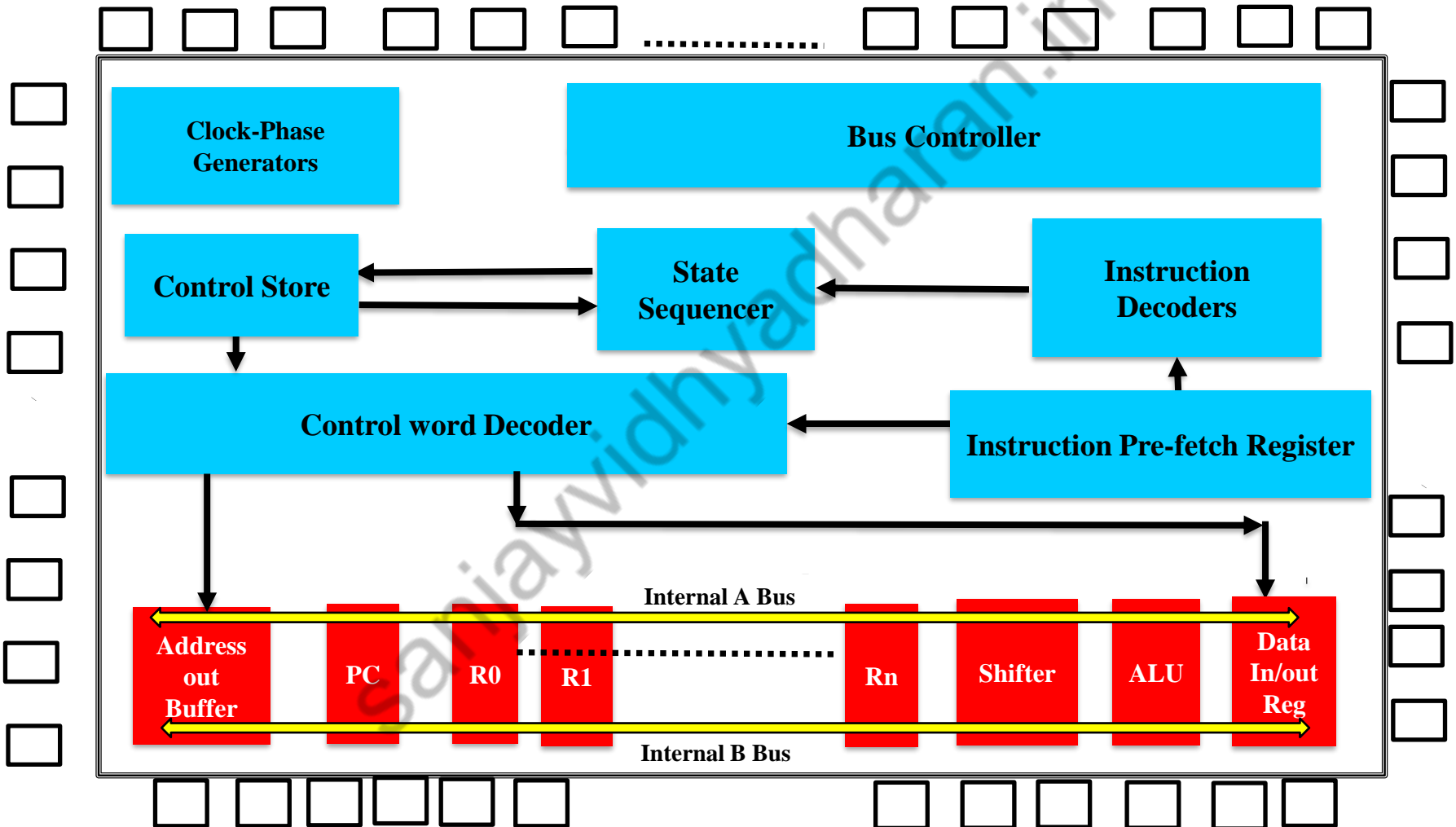
Arithmetic Circuits: Part-1

By Dr. Sanjay Vidhyadharan

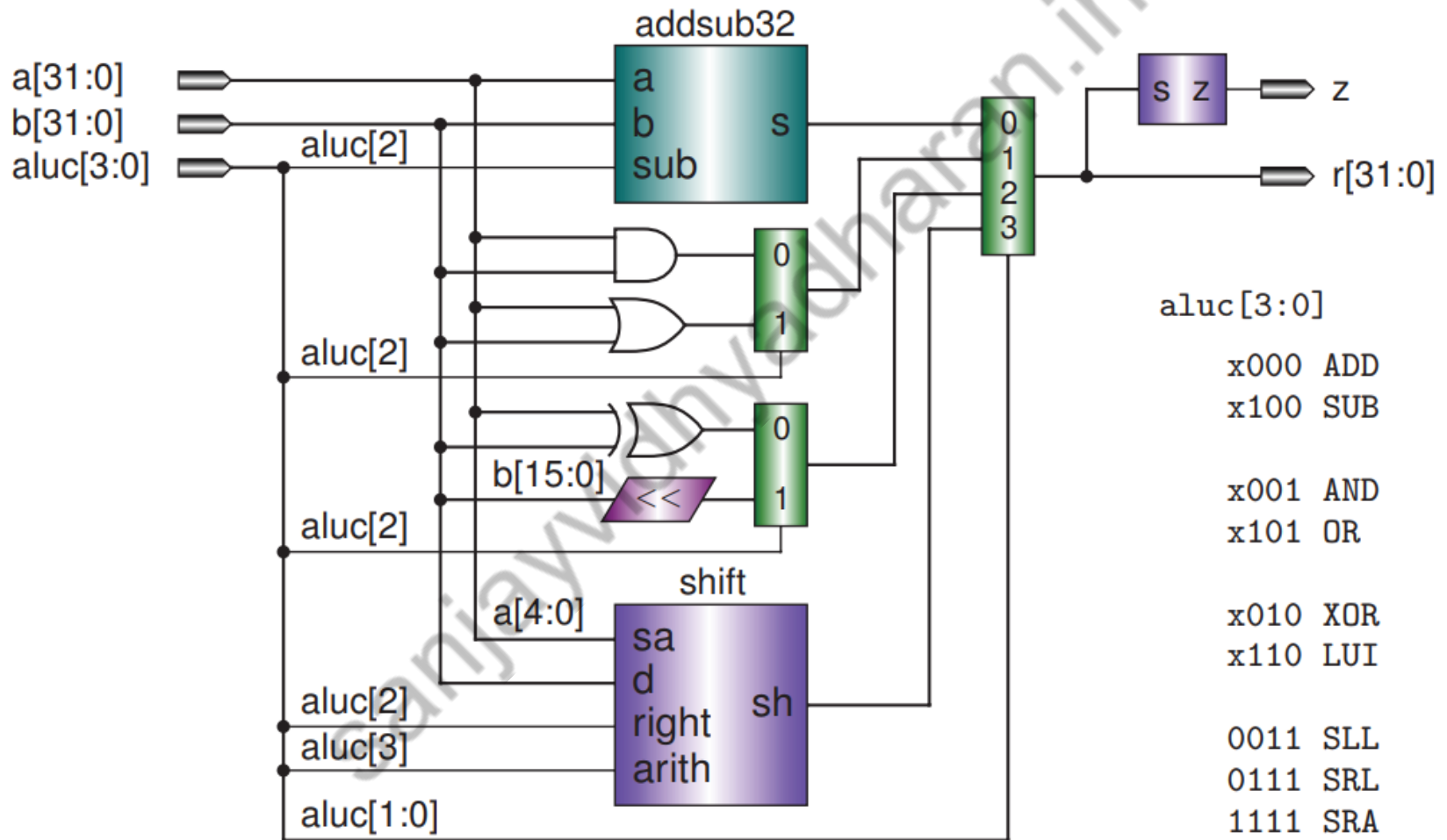
sanjayvidhyadharan.in

Microprocessor Design

Pads for Bus Control, Clock, Reset, Interrupts, Testing and Power Supply



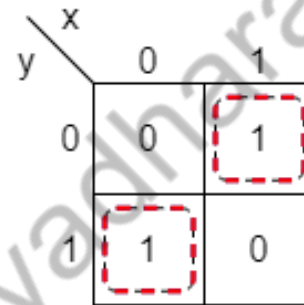
ALU Design



Half Adder

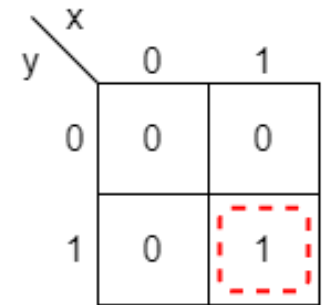
Inputs		Output	
A	B	Carry	Sum
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

K-map for Sum

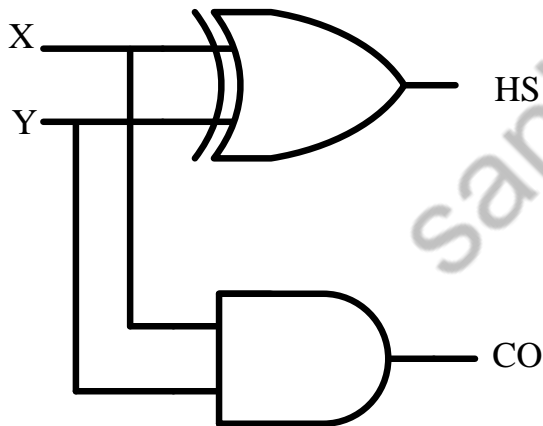


$$S = xy' + x'y$$

K-map for Carry



$$C = xy$$



$$\text{Sum} = X \cdot Y' + X' \cdot Y = X \oplus Y$$

$$\text{Carry} = X \cdot Y$$

Time Delay for the Half Adder?

1 gate delay

A gate delay of an xor for the half sum

A gate delay of an AND gate for the carry out

Full Adder

Full-adder can also implemented with two half adders and one OR gate.

$$S = A \oplus B \oplus C_{in}$$

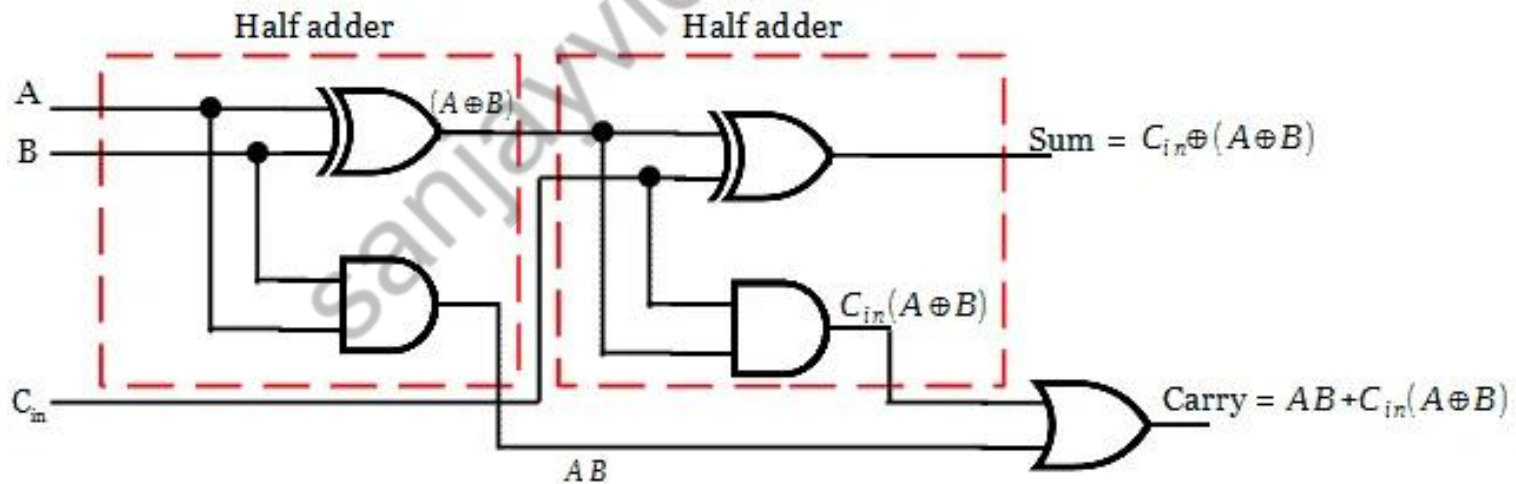
$$C_{out} = AB + BC_{in} + AC_{in}$$

$$= AB + C_{in}(A+B)$$

$$= AB + C_{in}(A \oplus B + AB)$$

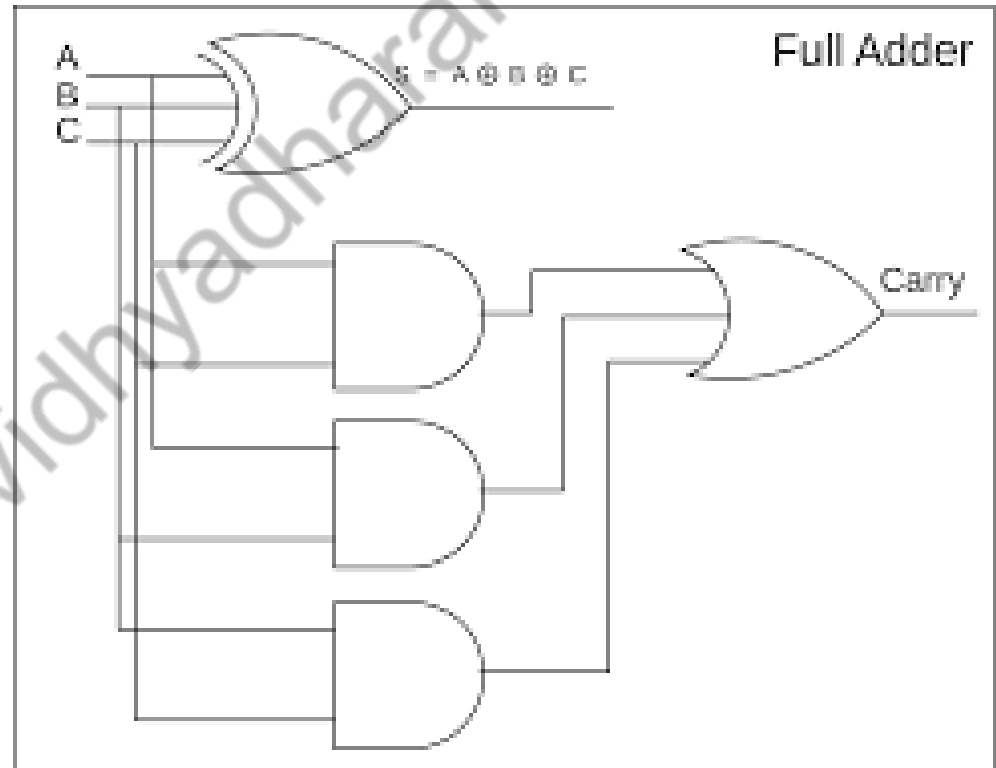
$$= AB + C_{in}(A \oplus B) + C_{in}AB = AB(1 + C_{in}) + C_{in}(A \oplus B)$$

$$= AB + C_{in}(A \oplus B)$$

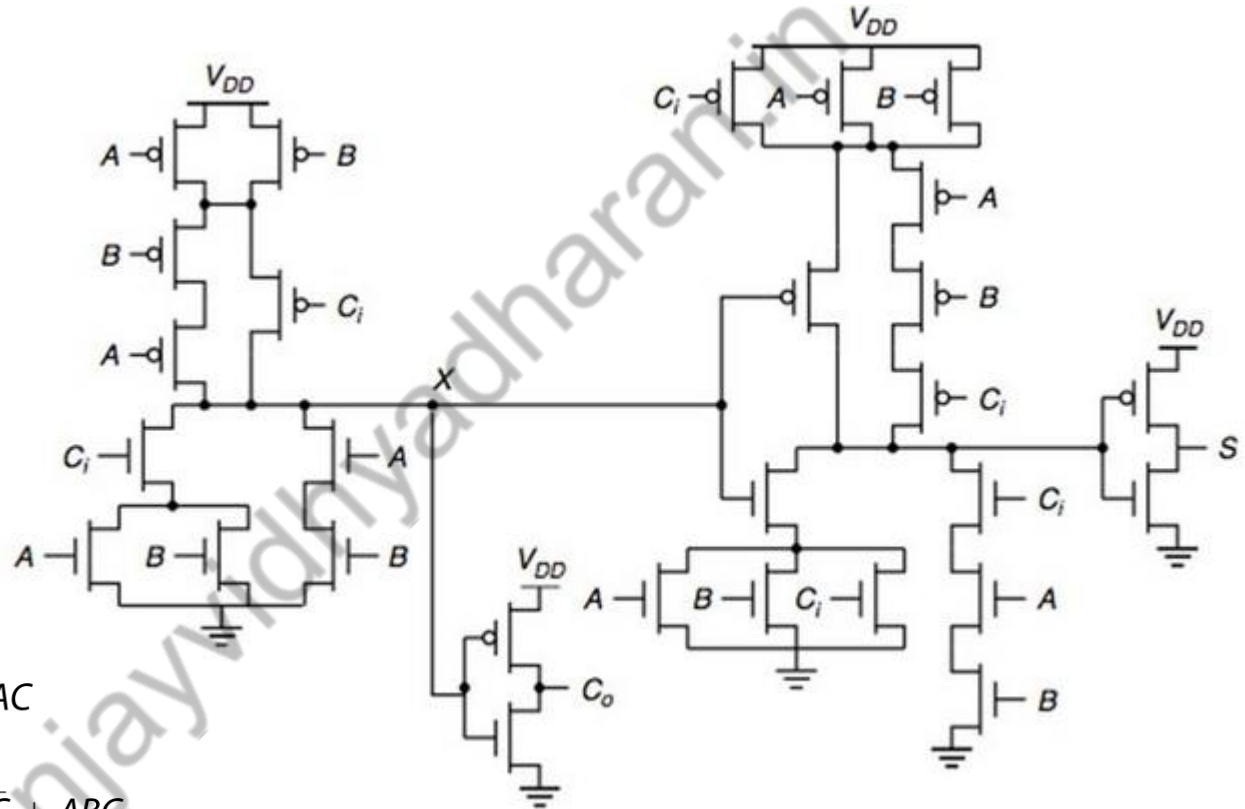


Full Adder

A	B	Carry In	Sum	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



CMOS 28T Adder



$$\text{Carry}(CY) = AB + BC + AC$$

$$\text{Sum}(S) = \bar{A}\bar{B}C\bar{C} + A\bar{B}\bar{C} + \bar{A}B\bar{C} + ABC$$

$$\left. \begin{aligned} CY &= AB + C(A + B) \\ S &= \bar{C}Y(A + B + C) + ABC \end{aligned} \right\} \begin{array}{l} \text{Simplified} \\ \text{Expressions} \end{array}$$

CMOS 28T Mirror Adder

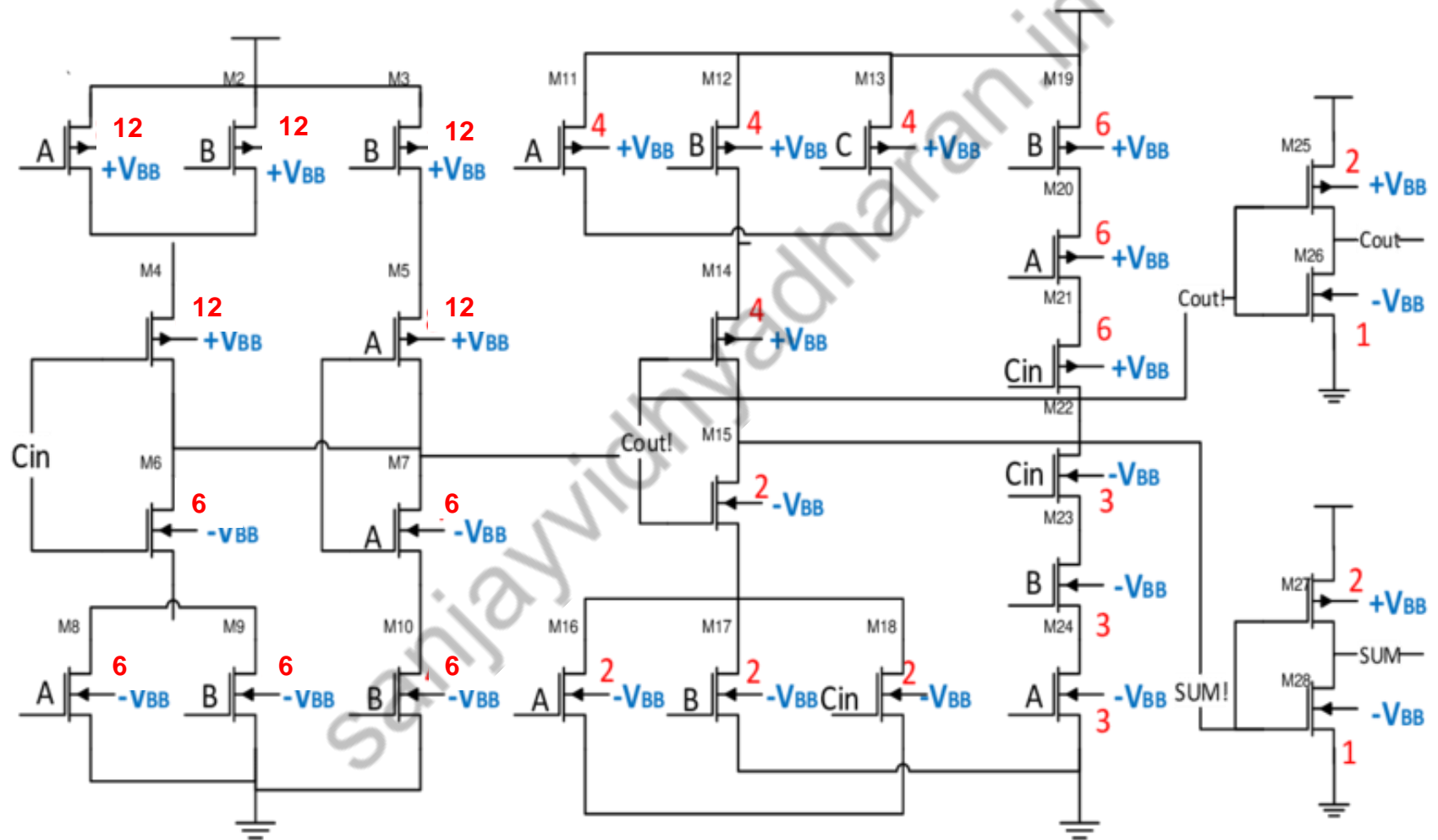
BCi \ A	00	01	11	10
0	0	0	1	0
1	0	1	1	1

$$\text{Carry} = AB + BC + CA = AB + C(A + B)$$

$$\text{Pull - Down Network for Carry Bar} = AB + C(A + B)$$

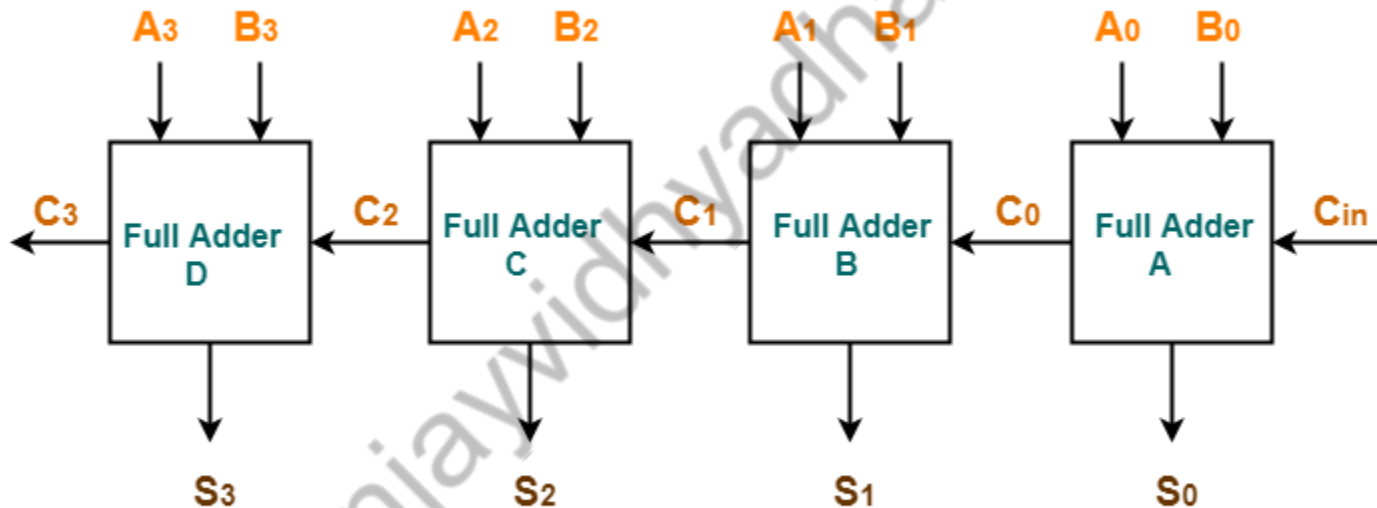
$$\begin{aligned}\text{Pull - UP Network for Carry Bar} &= A'B' + B'C' + C'A' \\ &= A'B' + C'(A' + B')\end{aligned}$$

CMOS 28T Mirror Adder



Ripple Carry Adder

This is called Ripple Carry Adder, because of the construction with full adders are connected in cascade.



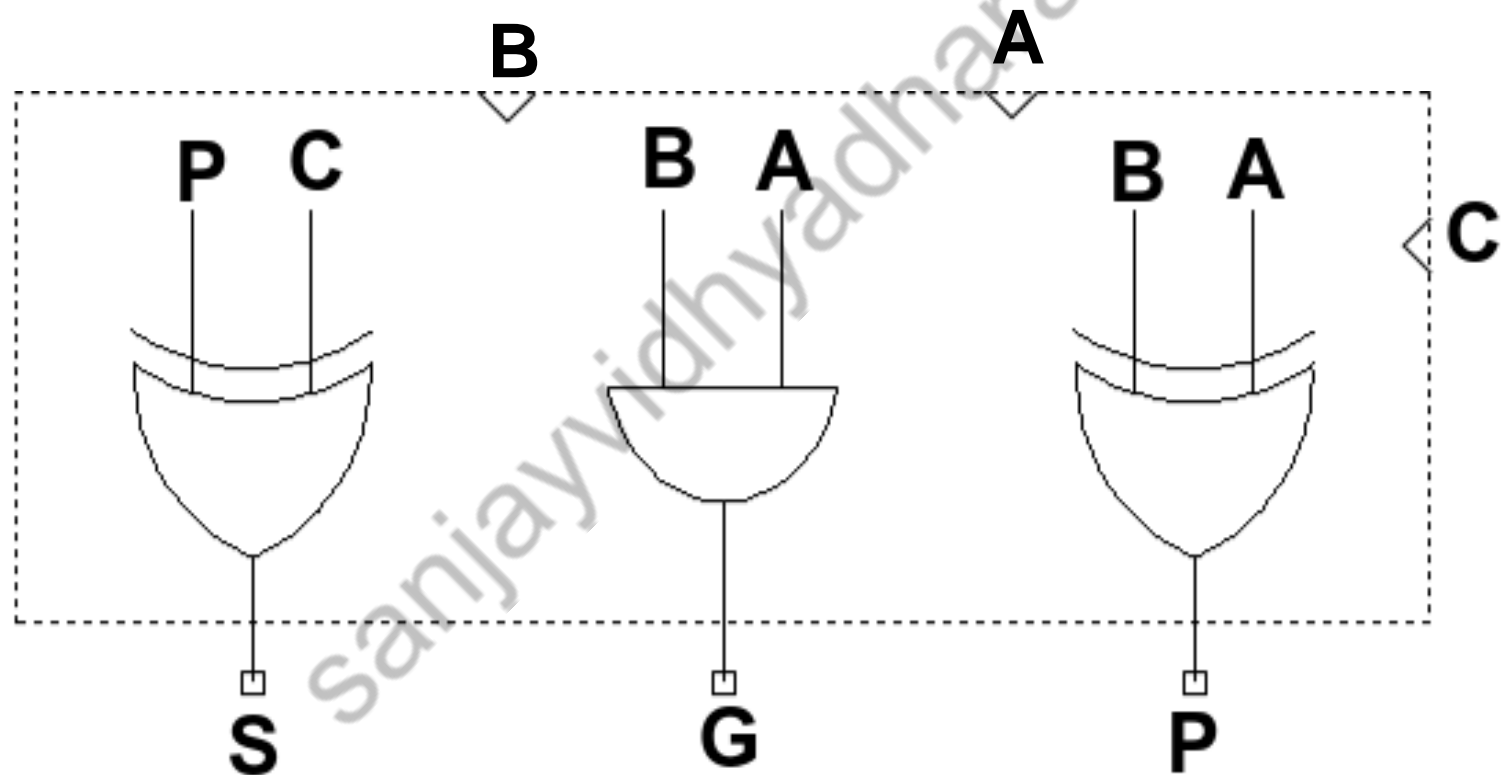
4-bit Ripple Carry Adder

Delay = 4 X Full Adder Delay = 8 Gate Delays

$$\text{Delay} = (N-1) t_{\text{carry}} + t_{\text{sum}}$$

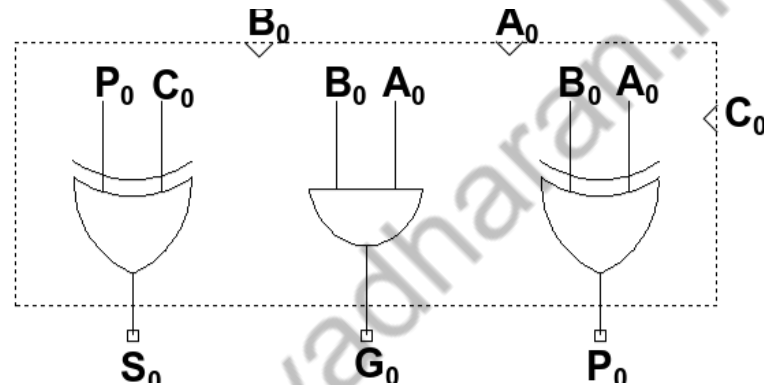
Carry Look-Ahead Adder

1-bit CLA

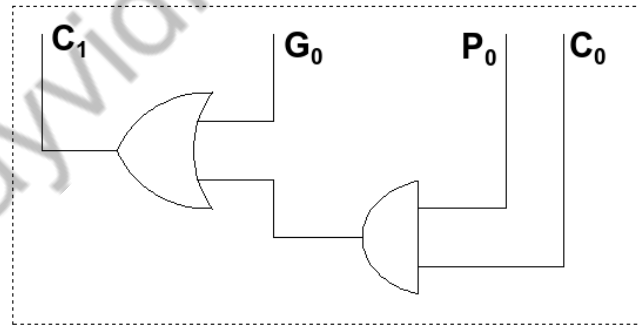


Carry Look-Ahead Adder

CLA

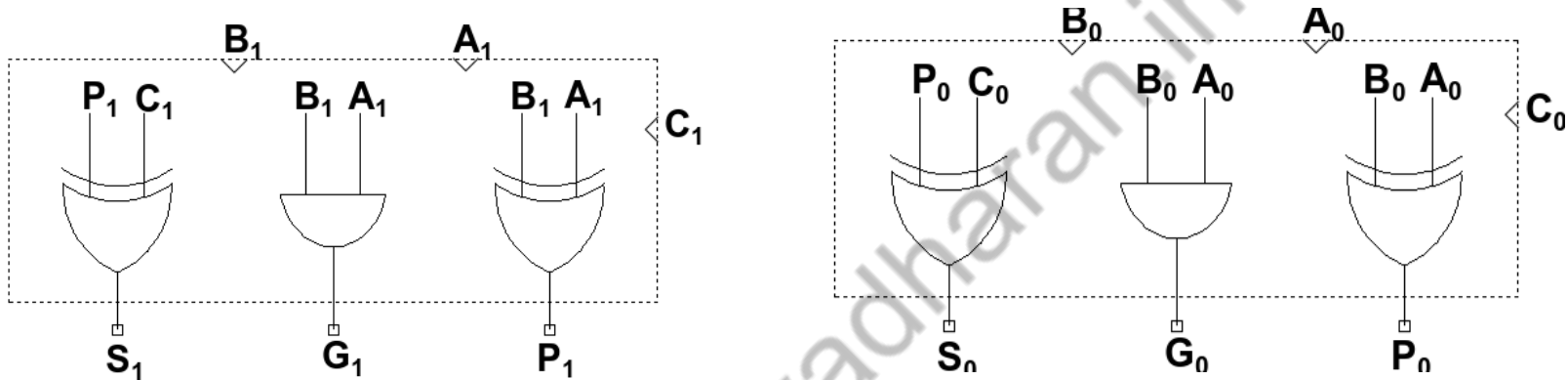


CLLB



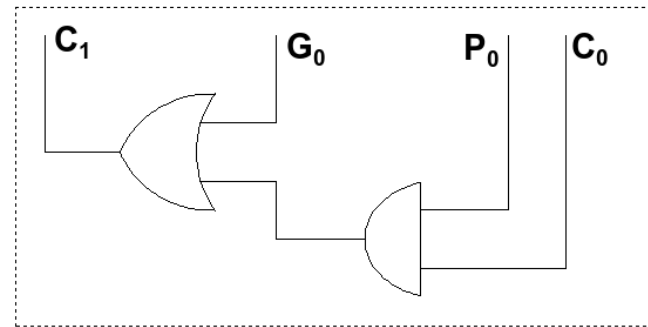
$$C_1 = G_0 + P_0 C_0$$

Carry Look-Ahead Adder

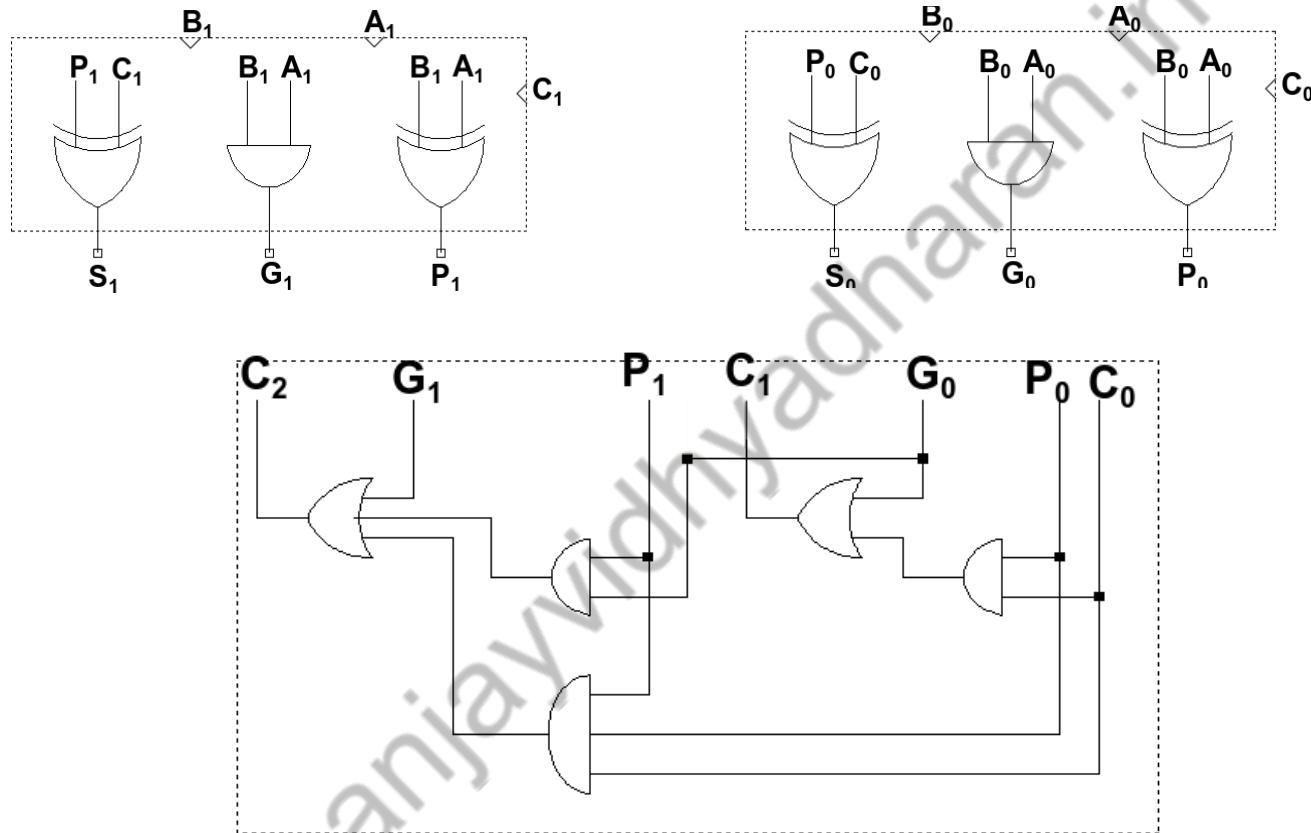


$$C_2 = G_1 + P_1 C_1$$

$$\begin{aligned} C_2 &= G_1 + P_1 (G_0 + P_0 C_0) \\ &= G_1 + P_1 G_0 + P_1 P_0 C_0 \end{aligned}$$



Carry Look-Ahead Adder



$$C_2 = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$C_1 = G_0 + P_0 C_0$$

Carry Look-Ahead Adder

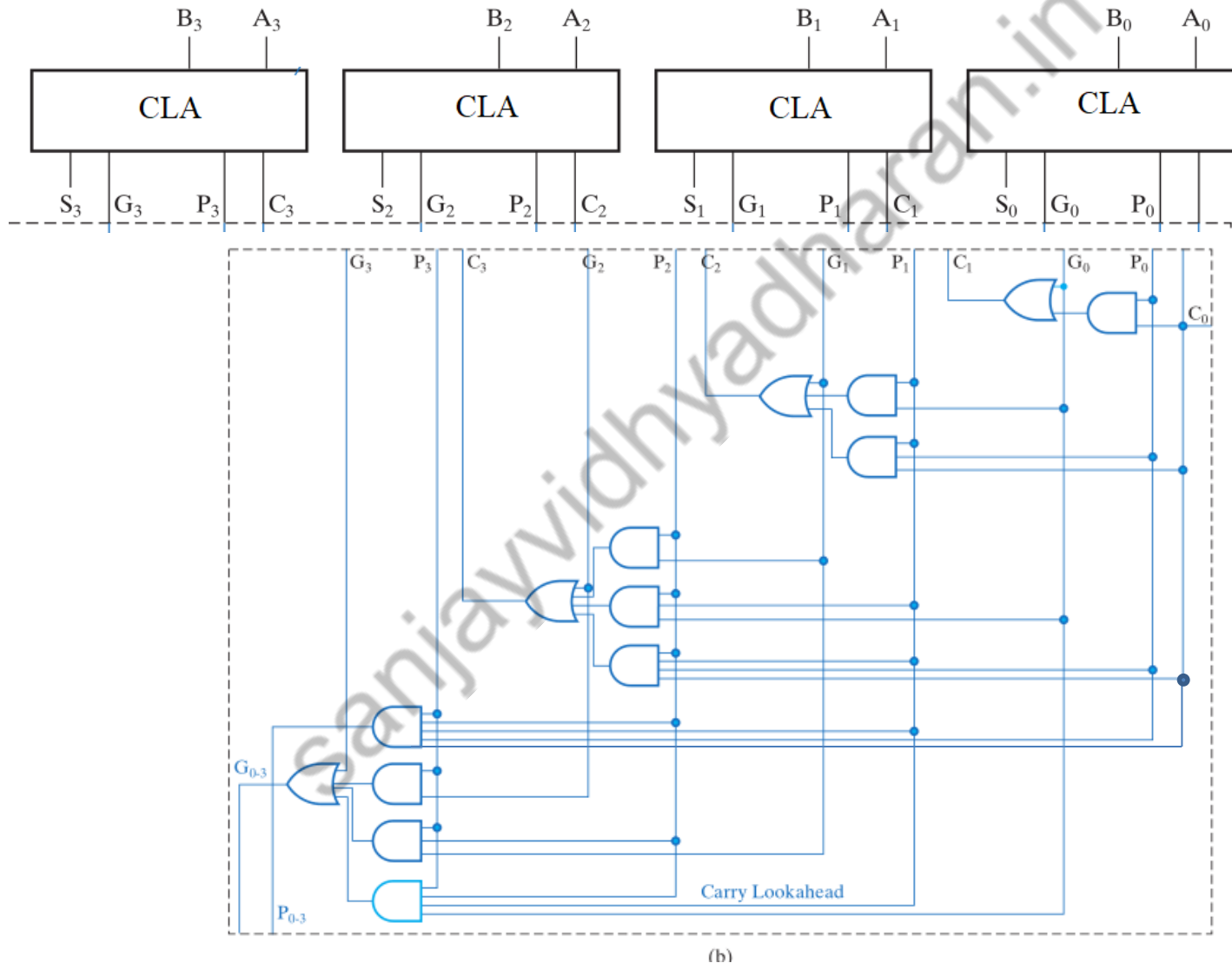
$$C_1 = G_0 + P_0 C_0$$

$$C_2 = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$C_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0$$

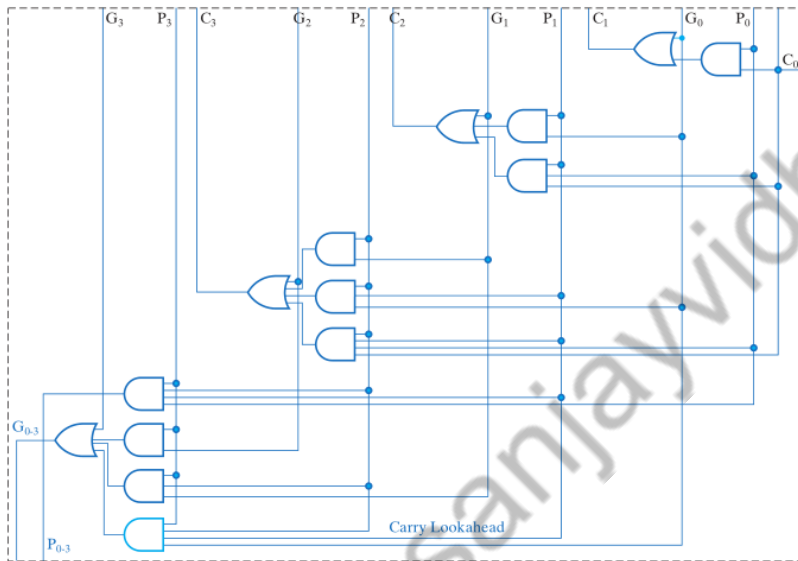
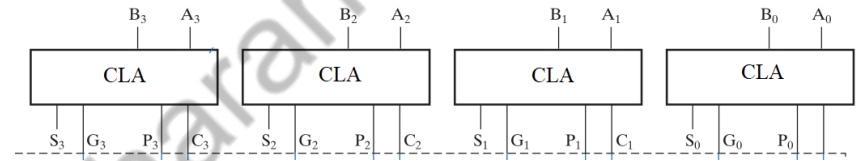
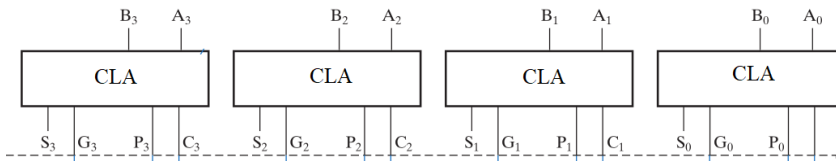
$$C_4 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0$$

Carry Look-Ahead Adder

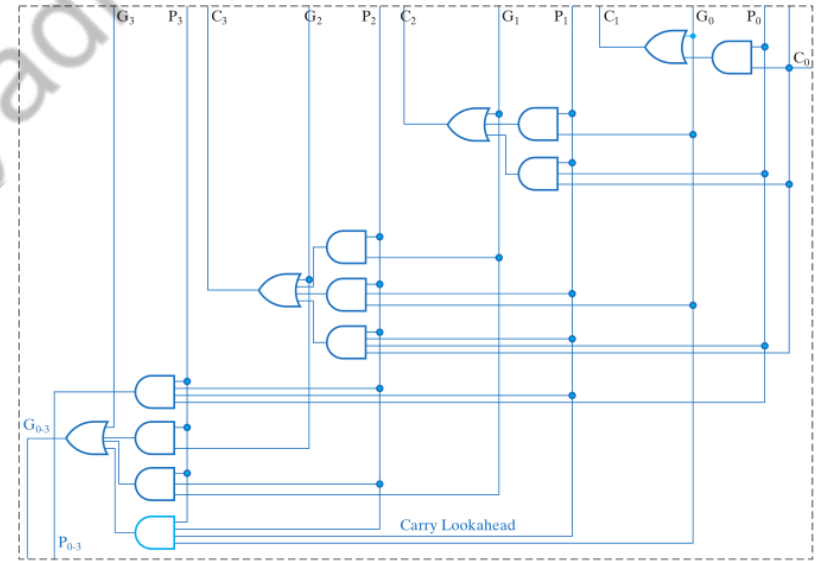


Carry Look-Ahead Adder

8 Bit Full Adder



(b)

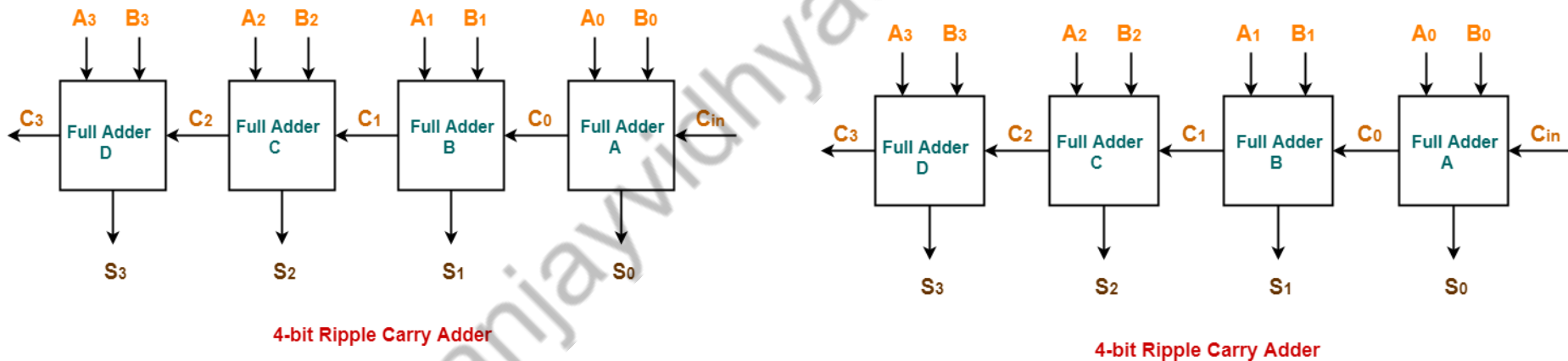


(b)

Delay = $2 \times 3 = 6$ Gate Delay

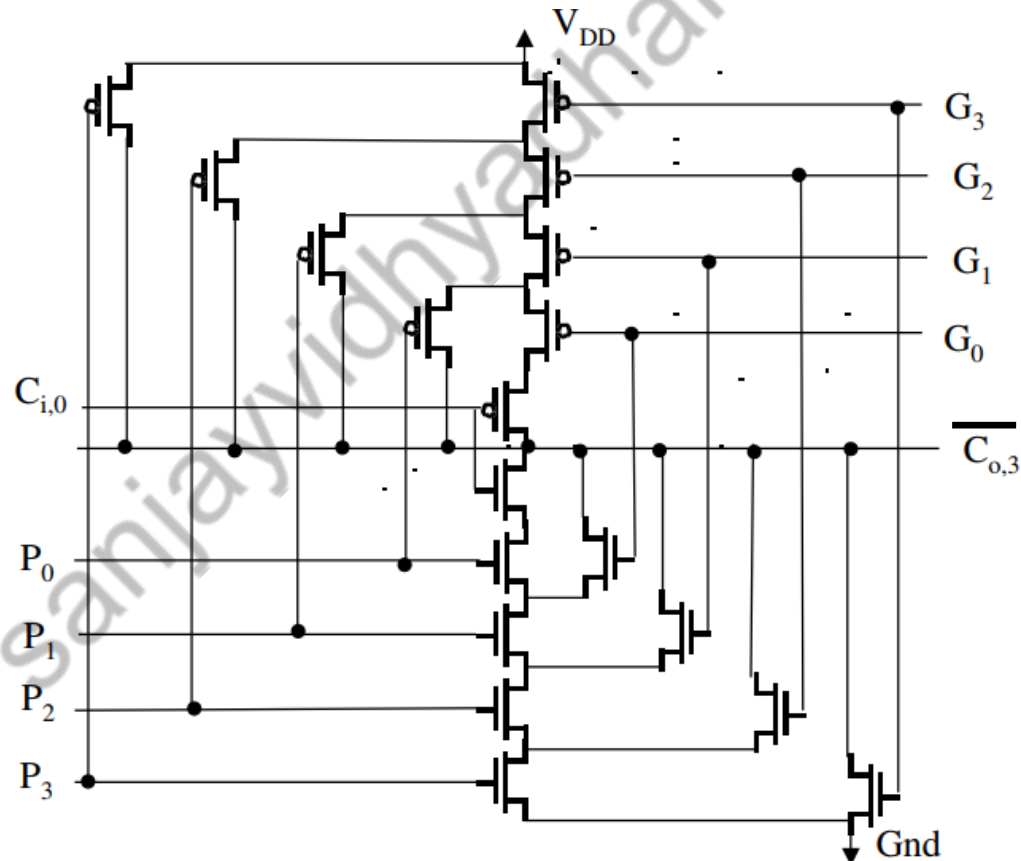
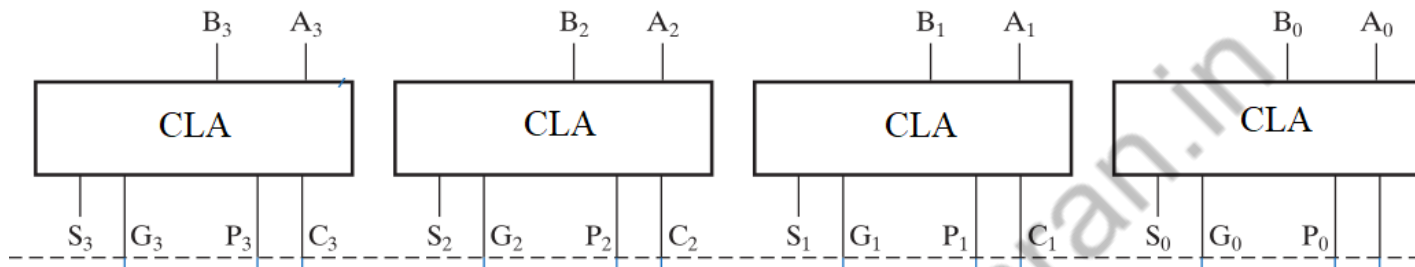
Ripple Carry Adder

8 Bit Full Adder

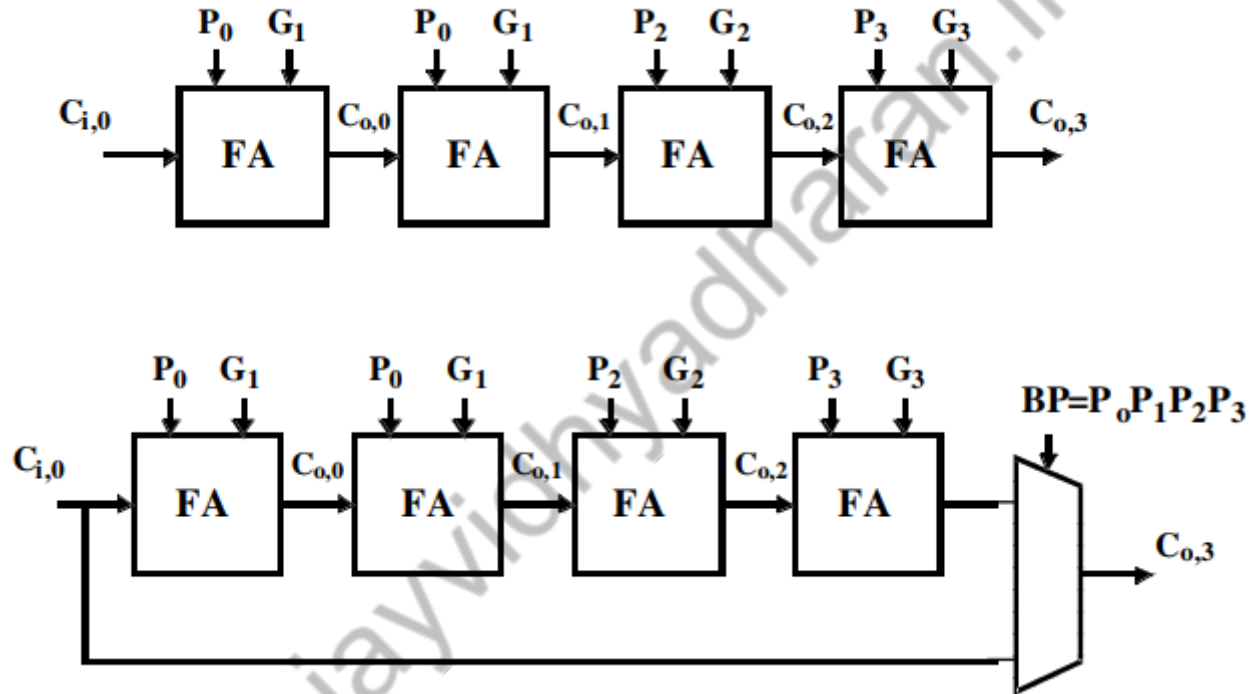


Delay = 8 X 2 Gate Delay

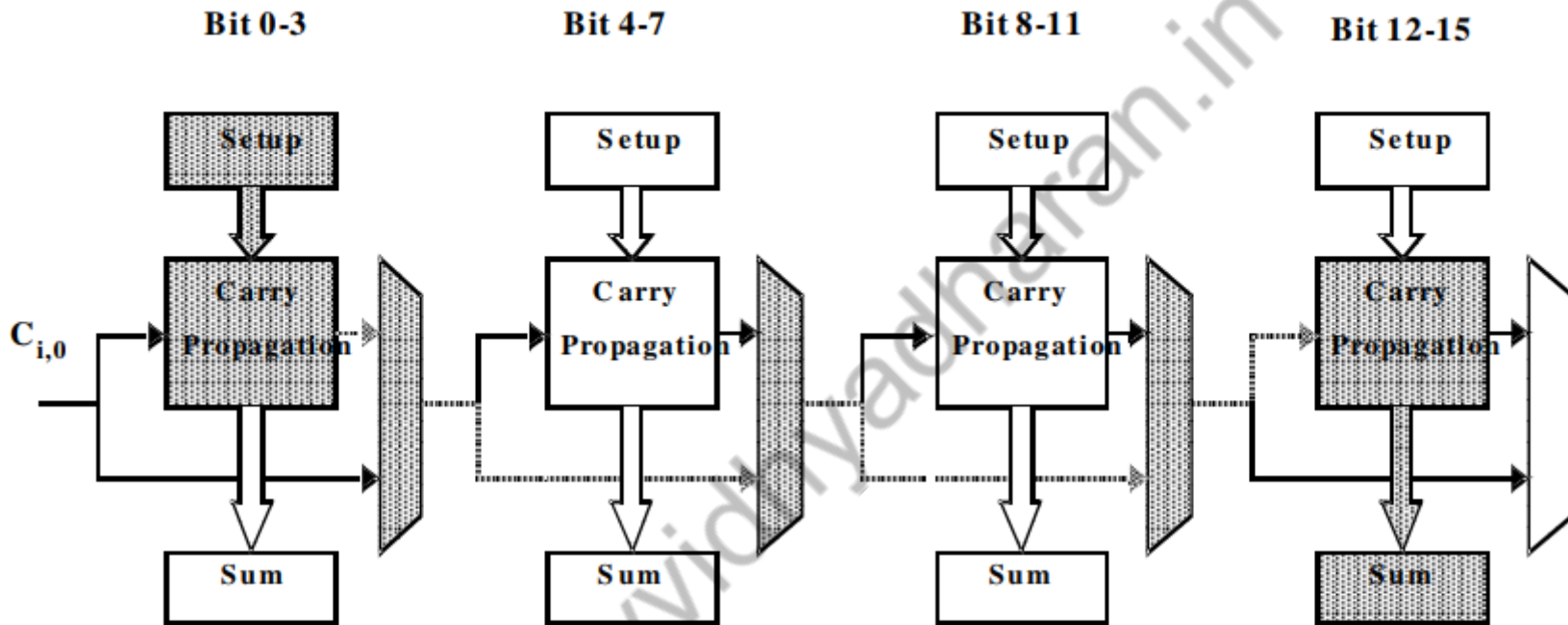
Carry Look-Ahead Adder



Carry Bypass or Carry Skip Adder



Carry Bypass or Carry Skip Adder



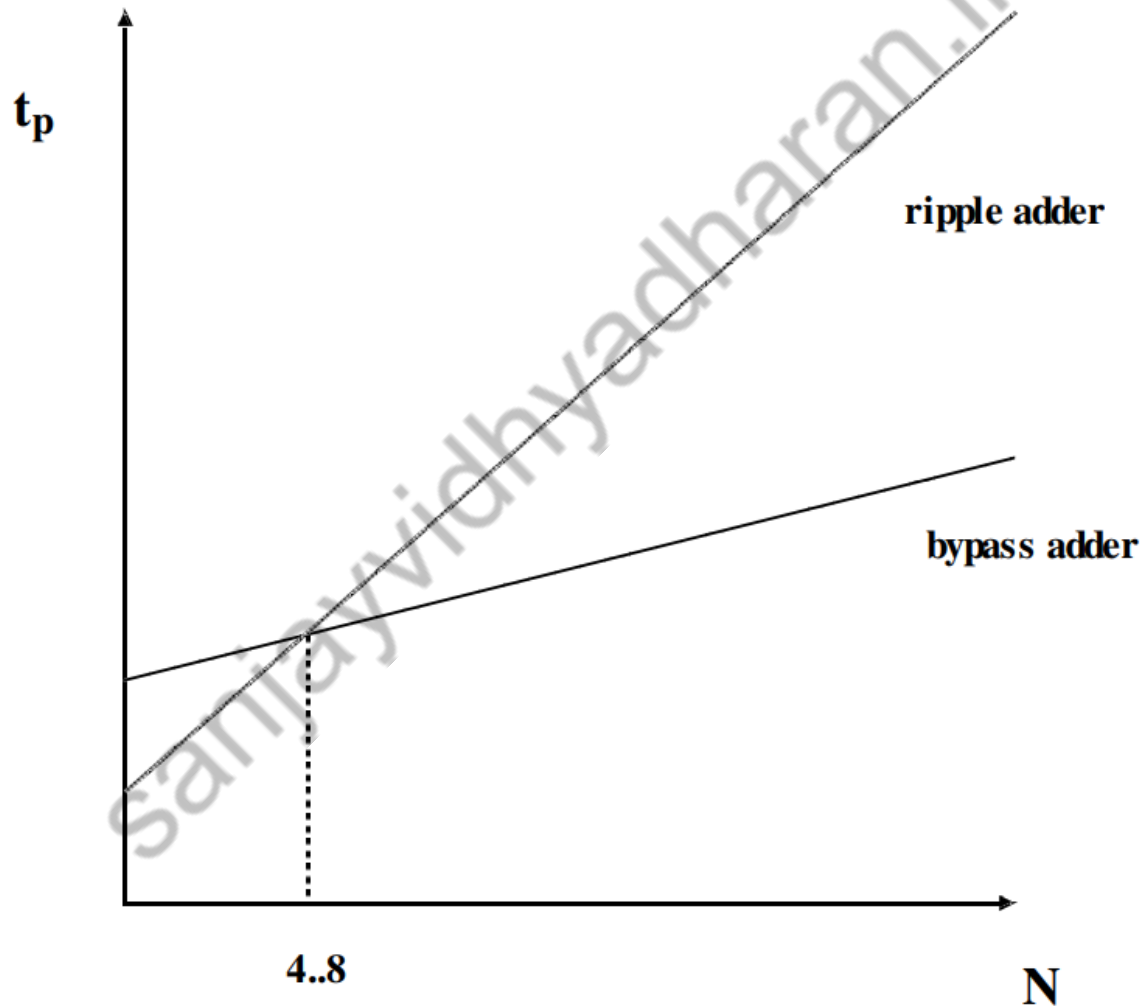
Design N-bit adder using N/M equal length stages

e.g. $N = 16, M = 4$

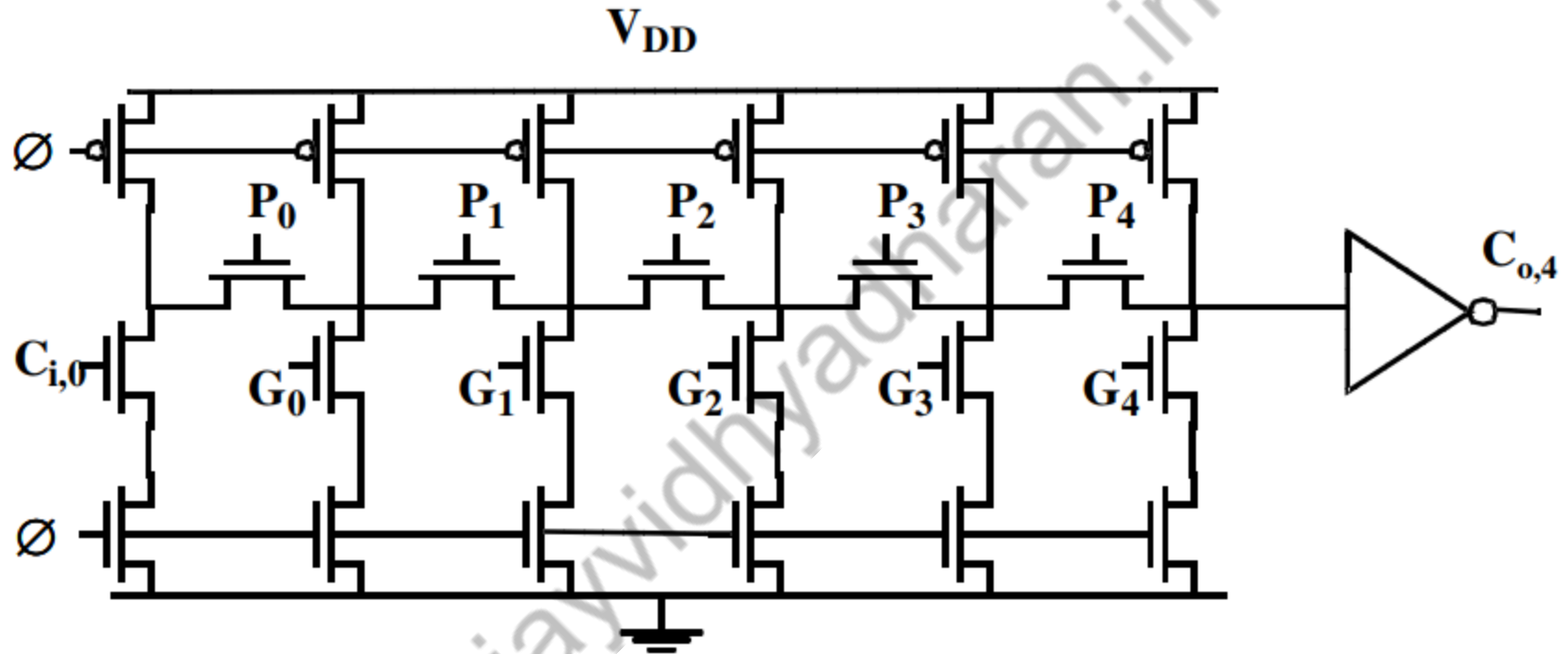
What is the critical path?

$$t_p = t_{\text{setup}} + Mt_{\text{carry}} + (N/M-1)t_{\text{bypass}} + Mt_{\text{carry}} + t_{\text{sum}}, \text{ i.e. } O(N)$$

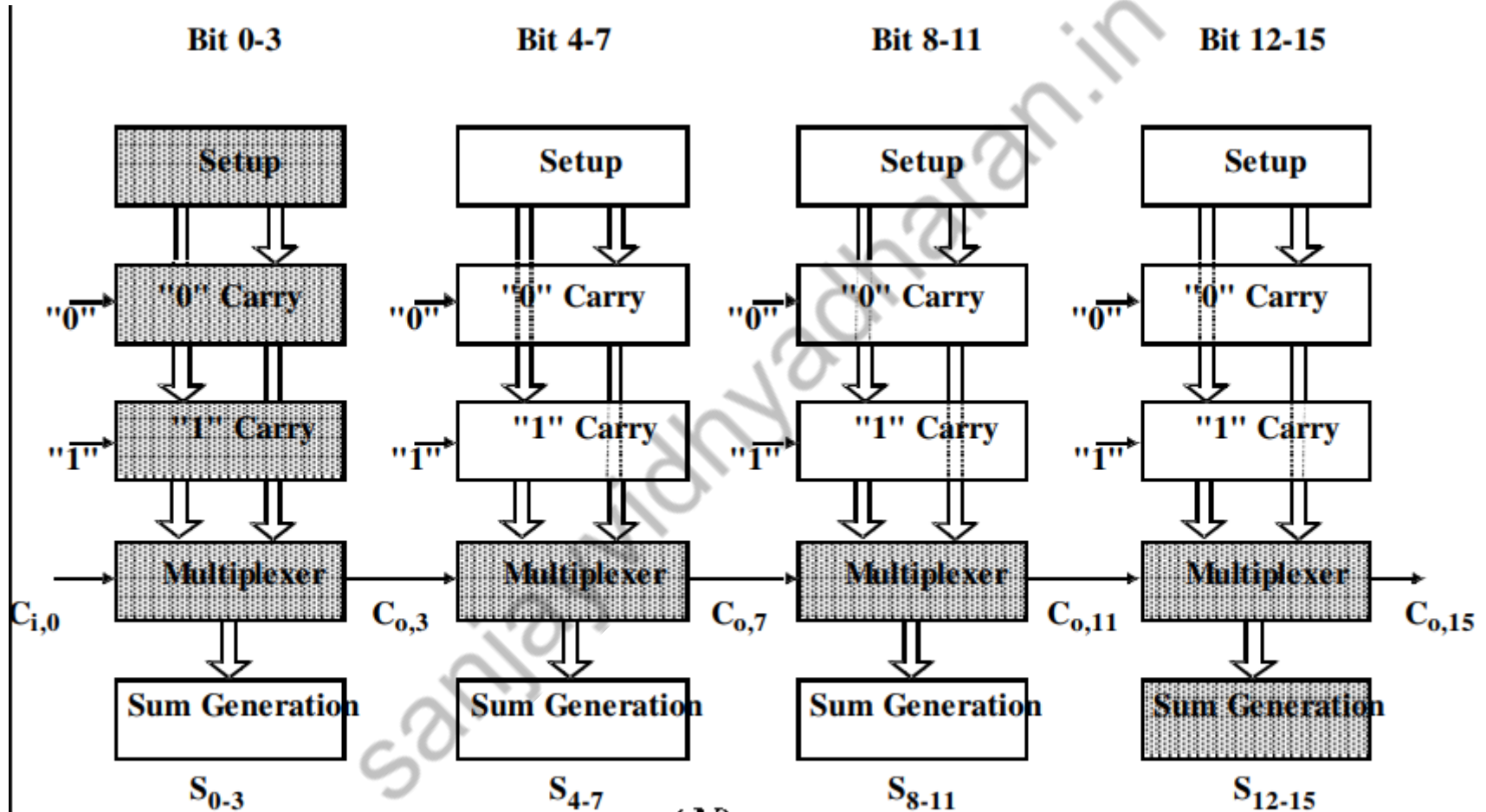
Carry Ripple versus Carry Bypass



Manchester Carry Chain

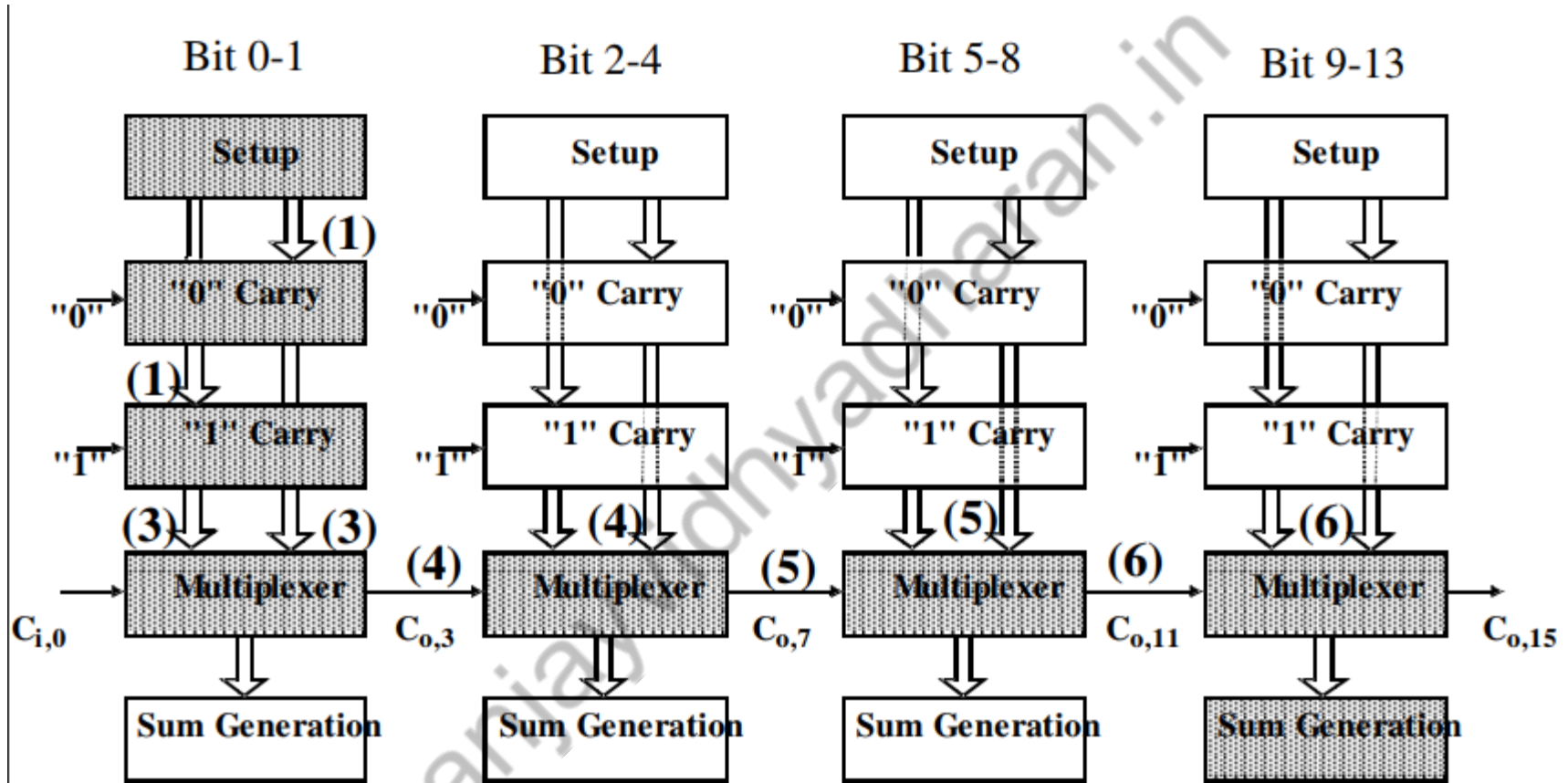


Linear Carry-Select Adder



$$t_{add} = t_{setup} + \left(\frac{N}{M}\right)t_{carry} + Mt_{mux} + t_{sum}$$

Square Root Carry-Select Adder



$$t_{add} = t_{setup} + Mt_{carry} + (\sqrt{2N})t_{mux} + t_{sum}$$

Square Root Carry-Select Adder

N Bit adder, M – Bits in First Stage, P – Number of Stages

$$N = M + (M + 1) + (M + 2) + (M + 3) + \dots + (M + P - 1)$$

$$N = MP + \frac{P(P - 1)}{2}$$

$$N = \frac{P^2}{2} + P\left(M - \frac{1}{2}\right)$$

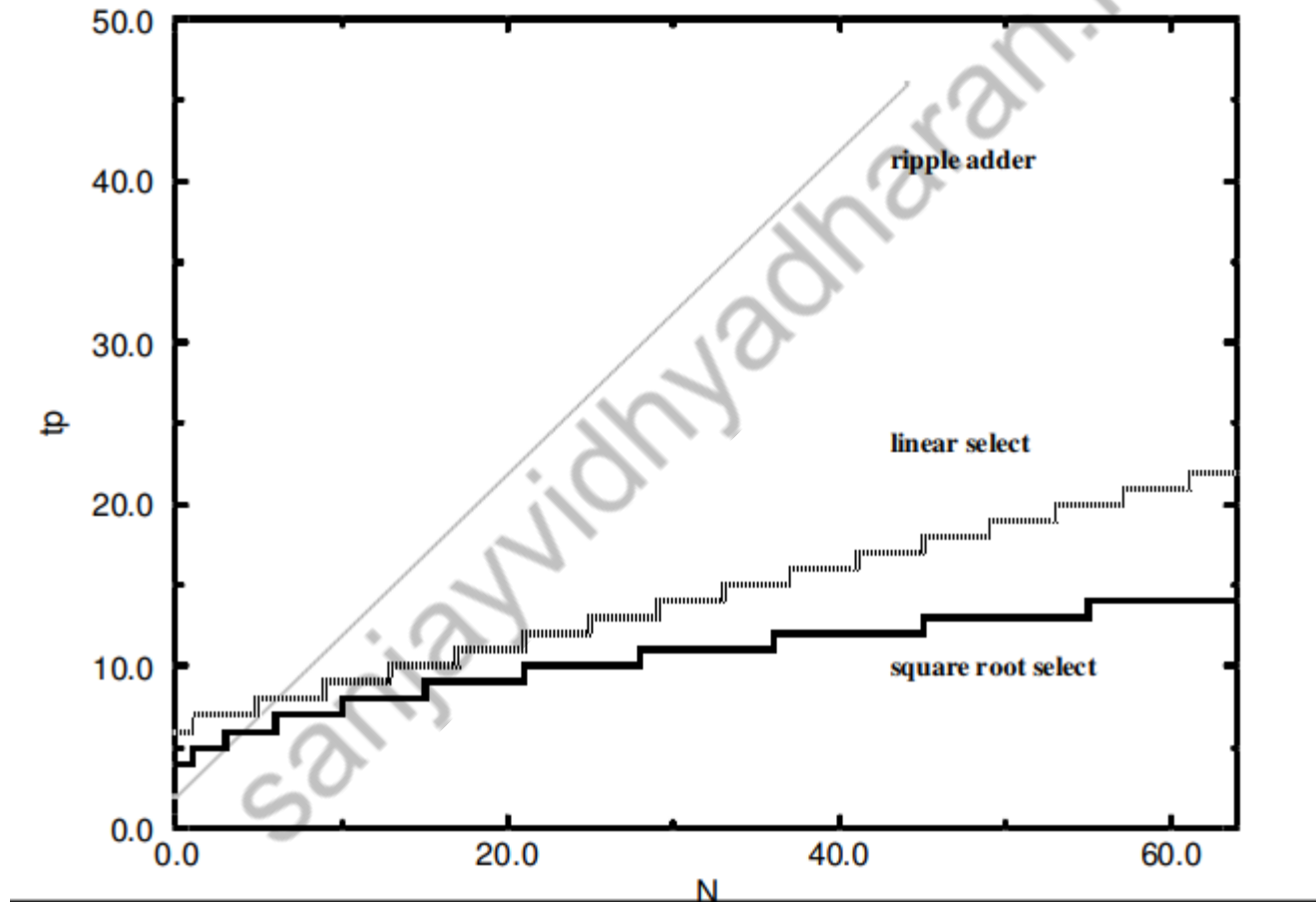
$$M \ll N \text{ (e.g. } M = 2 \text{ and } N = 64)$$

$$N \approx \frac{P^2}{2}$$

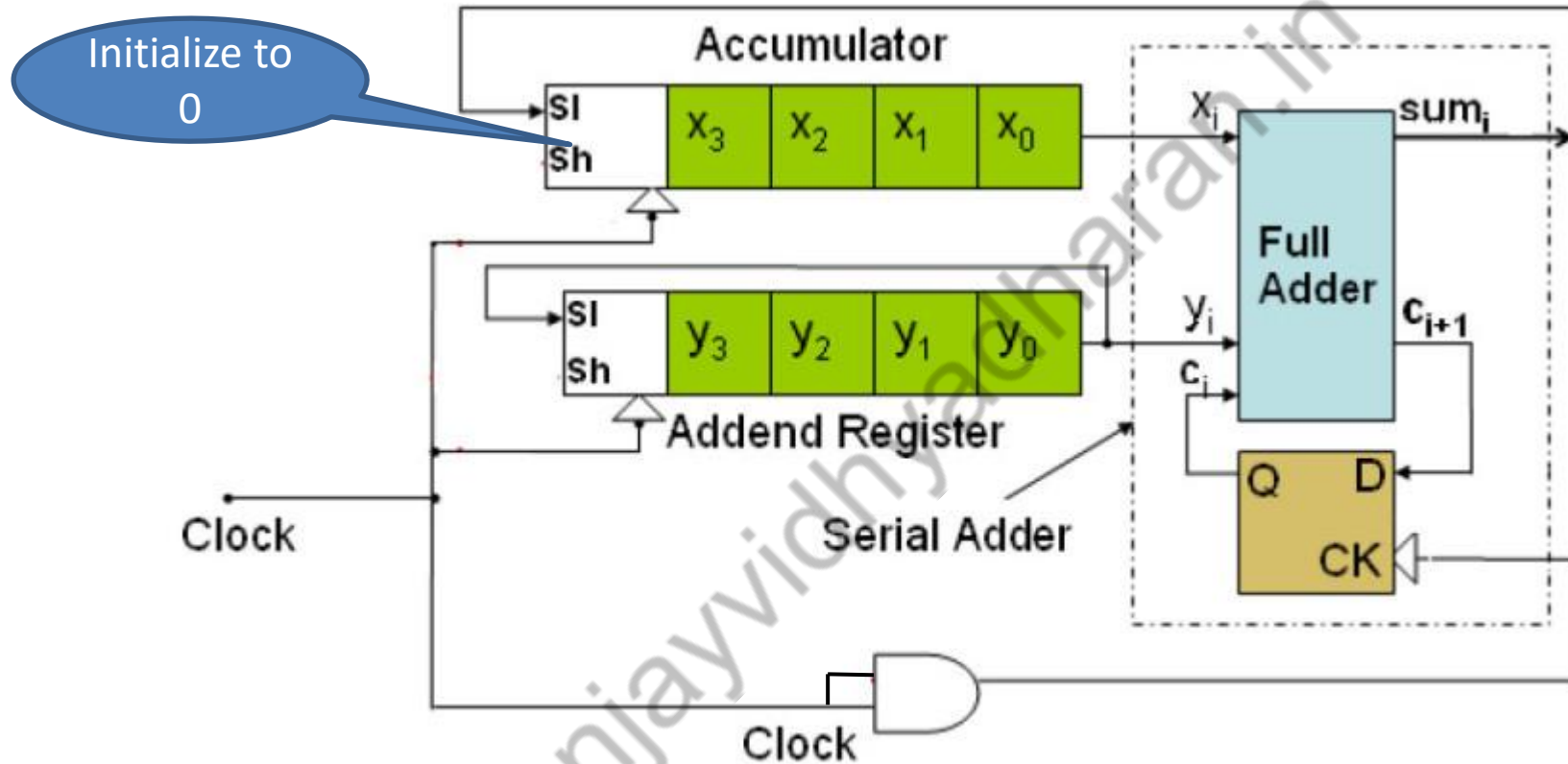
Series: $a, a+d, a+2d, \dots, a+(n-1)d$

$$s_n = n/2(2a + (n-1)d)$$

Adder Delays - Comparison



Serial Adder



Block Diagram of a 4-bit Serial Adder with Accumulator

4 Bit-Adder Subtractor

Add 4 & -3

```

0100
1101
1 0001
    
```

Add -4 & -5

```

1100
0111
1 0111
    
```

Add -8 & 4

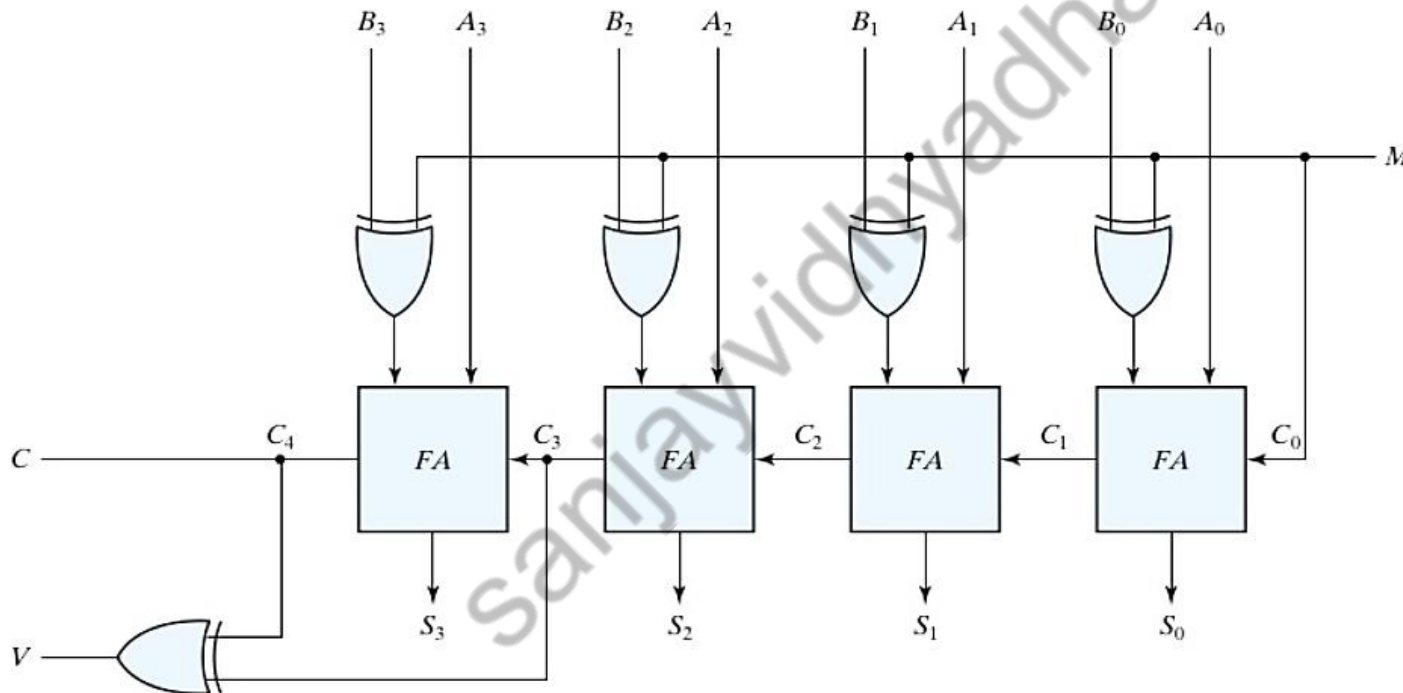
```

1000
0100
1100
    
```

Add 4 & 4

```

0100
0100
1000
    
```



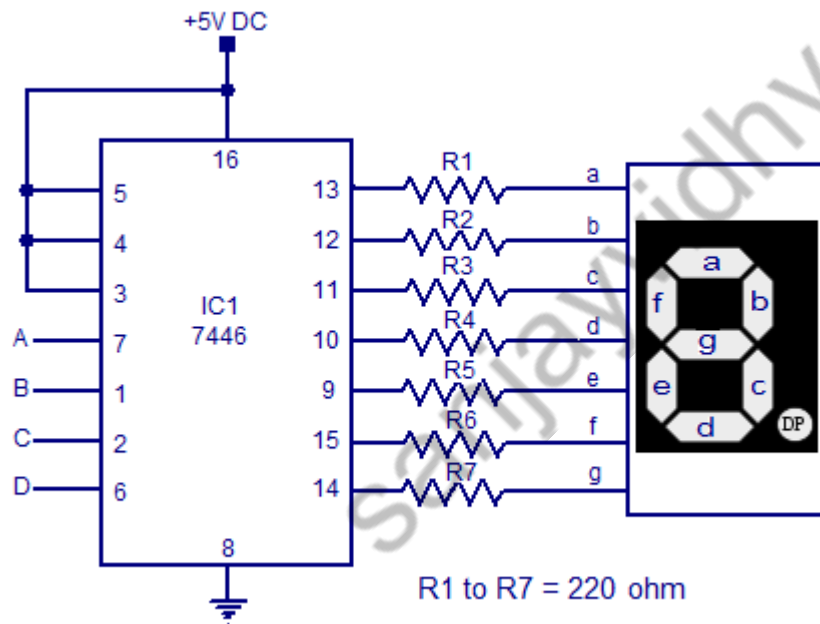
Overflow

Decimal	2's comp.
7	0111
6	0110
5	0101
4	0100
3	0011
2	0010
1	0001
0	0000
-0	-
-1	1111
-2	1110
-3	1101
-4	1100
-5	1011
-6	1010
-7	1001
-8	1000

Binary Coded Decimal

General digital systems

User enters decimal \rightarrow BCD i/p \rightarrow Binary i/p \rightarrow compute in binary \rightarrow Binary o/p \rightarrow BCD o/p \rightarrow Decimal output shown to user



Binary Coded Decimal

BCD addition

$$4 + 5$$

4 0 1 0 0

5 0 1 0 1

9 1 0 0 1 Expected Result

$$4 + 8$$

4 0 1 0 0

8 1 0 0 0

1 1 0 0 Is this expected Result ?

Expected answer 0001 0010
is BCD of 12

Binary Coded Decimal

BCD addition

4 + 8 4 0 1 0 0

8 1 0 0 0

Greater than 9

1 1 0 0

0 1 1 0

0 0 0 1 0 0 1 0

1 2

Add correction of +6

= To skip 6 invalid
states (10 - 15) BCDs

Binary Coded Decimal

BCD addition

9 + 9 9 1 0 0 1

9 1 0 0 1

Carry out generated 1 0 0 1 0

0 1 1 0

0 0 0 1 1 0 0 0

1 8

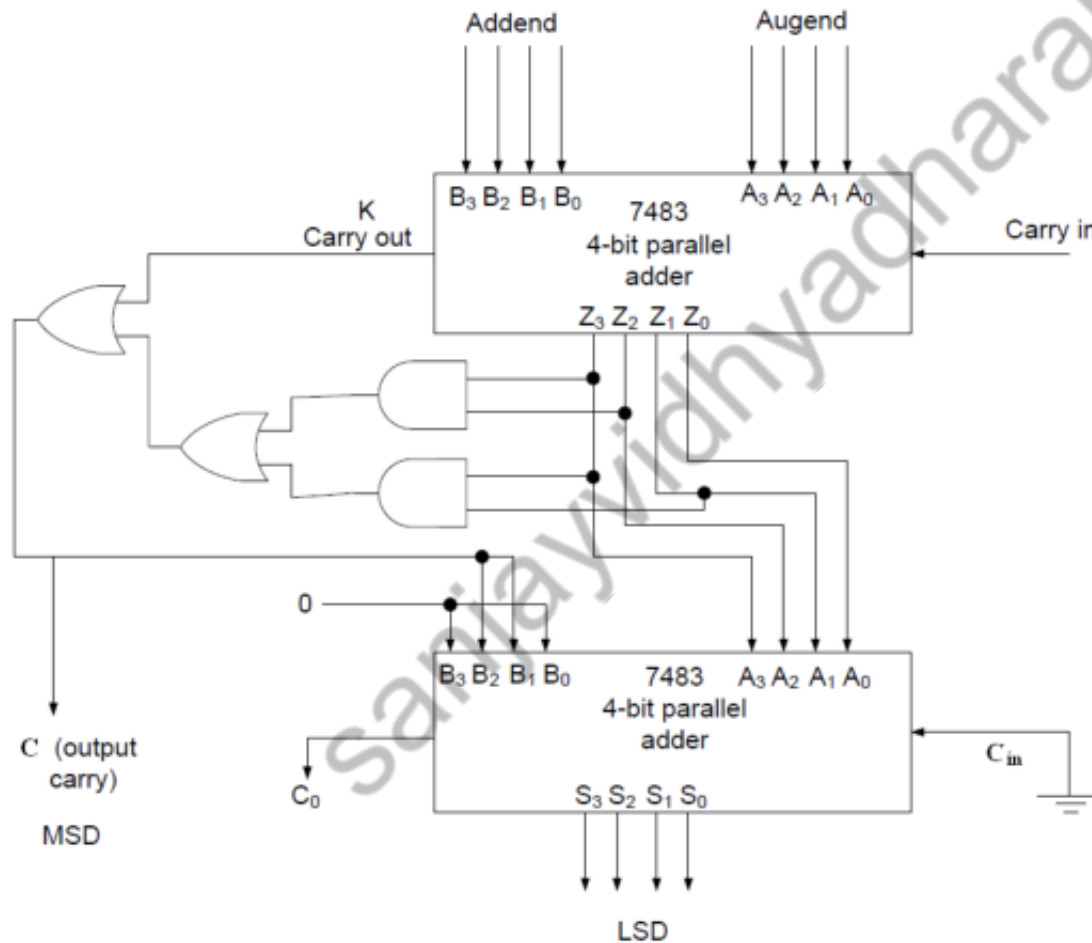
Expected result ?

Add correction of +6

After addition if carry out is generated or if sum is greater than 9 there is need for correction

Binary Coded Decimal

BCD addition



0 0 0 0
 0 0 0 1
 0 0 1 0
 0 0 1 1
 0 1 0 0
 0 1 0 1
 0 1 1 0
 0 1 1 1
 1 0 0 0
 1 0 0 1

Thank you