Binary Adders

By

P. K. Roy

Half Adder

- Addition of 2 bits only.
- No requirement of initial carry.

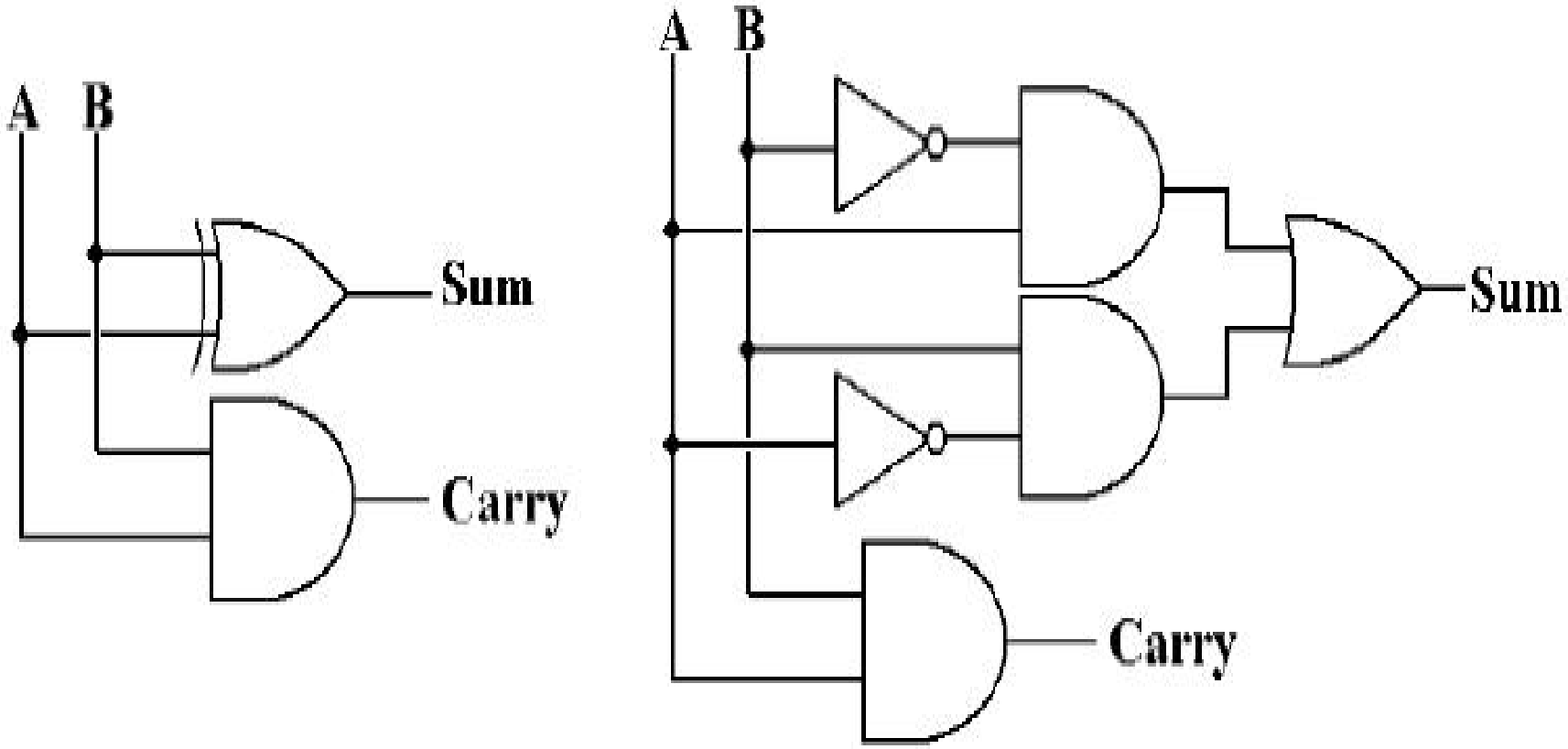
A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Sum represents Ex-OR gate and Carry represents AND gate.

$$\mathbf{Sum} = \overline{\mathbf{A}}\cdot\mathbf{B} + \mathbf{A}\cdot\overline{\mathbf{B}}$$

$$\mathbf{Carry} = \mathbf{A.B}$$

Cntd…



Full Adder

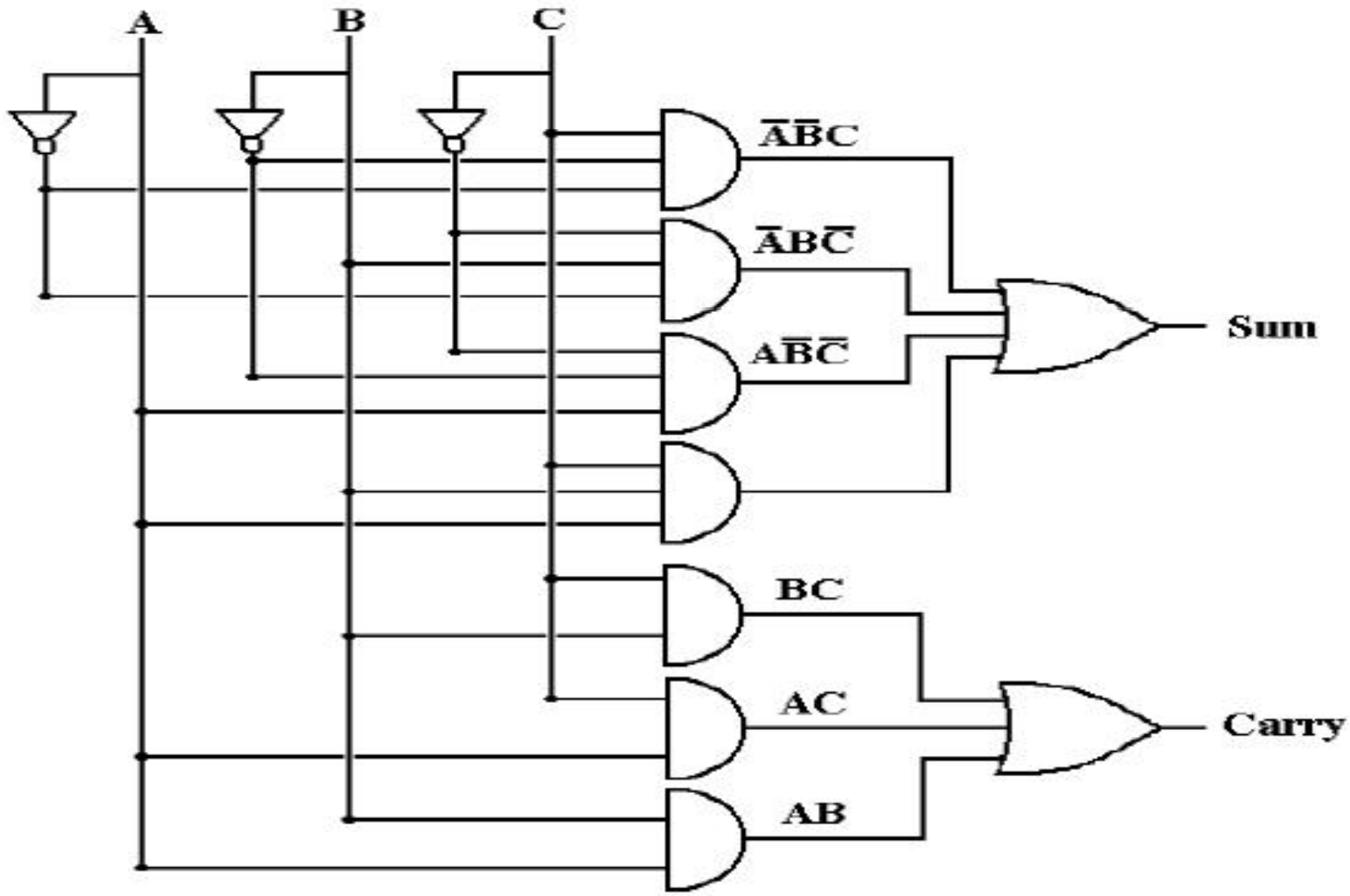
- Addition of 3 bits only.
- Requires an initial carry.

A	B	C	Sum	Carry
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

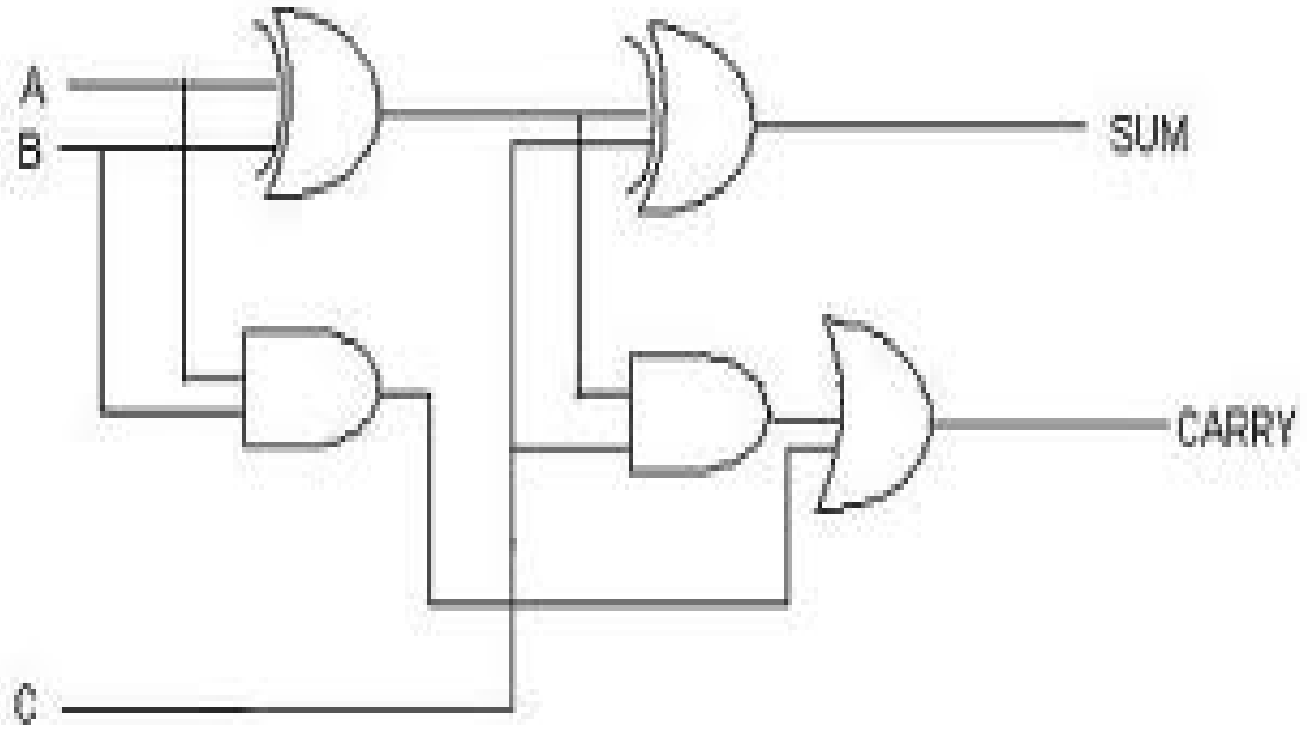
$$\text{Sum} = \overline{A} \bullet \overline{B} \bullet C + \overline{A} \bullet B \bullet \overline{C} + A \bullet \overline{B} \bullet \overline{C} + A \bullet B \bullet C$$

$$\text{Carry} = A \bullet B + A \bullet C + B \bullet C$$

Cntd…



Designing FA Using HA



Home Work: HS & FS

When FA Behaves Like HA?

$$\text{Sum} = \overline{A} \bullet \overline{B} \bullet C + \overline{A} \bullet B \bullet \overline{C} + A \bullet \overline{B} \bullet \overline{C} + A \bullet B \bullet C$$

$$\text{Carry} = A \bullet B + A \bullet C + B \bullet C$$

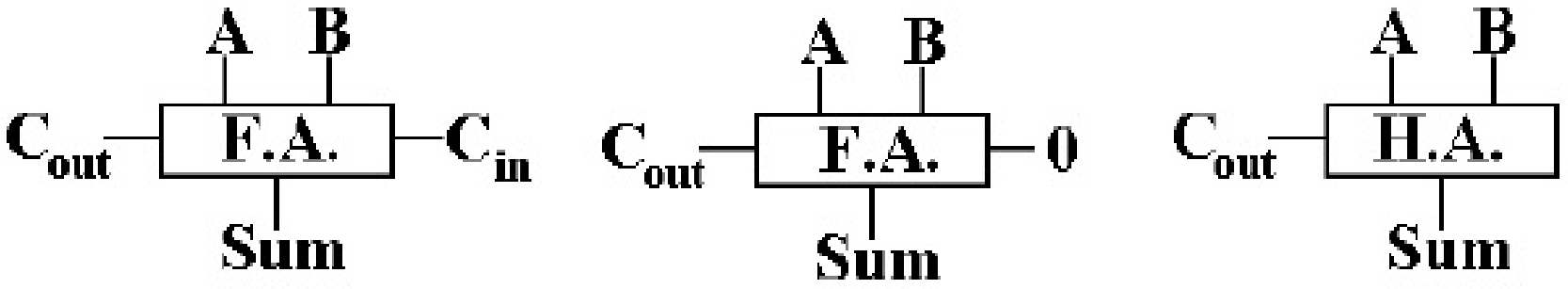
Suppose we let the carry–in C = 0. Then $\overline{C} = 1$.
What we have then is as follows.

$$\begin{aligned} \text{Sum} &= \overline{A} \bullet \overline{B} \bullet 0 + \overline{A} \bullet B \bullet 1 + A \bullet \overline{B} \bullet 1 + A \bullet B \bullet 0 \\ &= \overline{A} \bullet B + A \bullet \overline{B} \end{aligned}$$

$$\begin{aligned} \text{Carry} &= A \bullet B + A \bullet 0 + B \bullet 0 \\ &= A \bullet B \end{aligned}$$

The Full–Adder and Half–Adder as Circuit Elements

- Each circuit is denoted as a simple box with inputs and outputs.

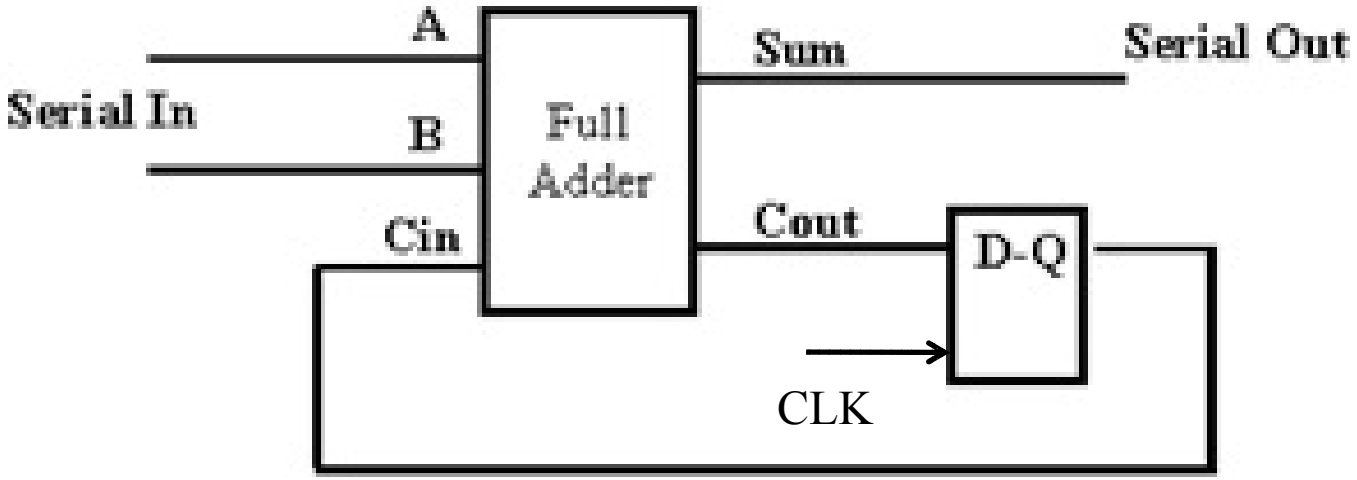


(a) **(b)** **(c)**

(a) full–adder with carry–in
(b) full–adder acting as a half–adder
(c) half–adder with no carry–in

Serial Adder

For addition of 2 binary numbers.



Disadvantage:

1. Addition process is performed by bit-by-bit.
2. Time required for addition depends on number of bits (for n no. of bits it will be n clock cycles).

Parallel Adder

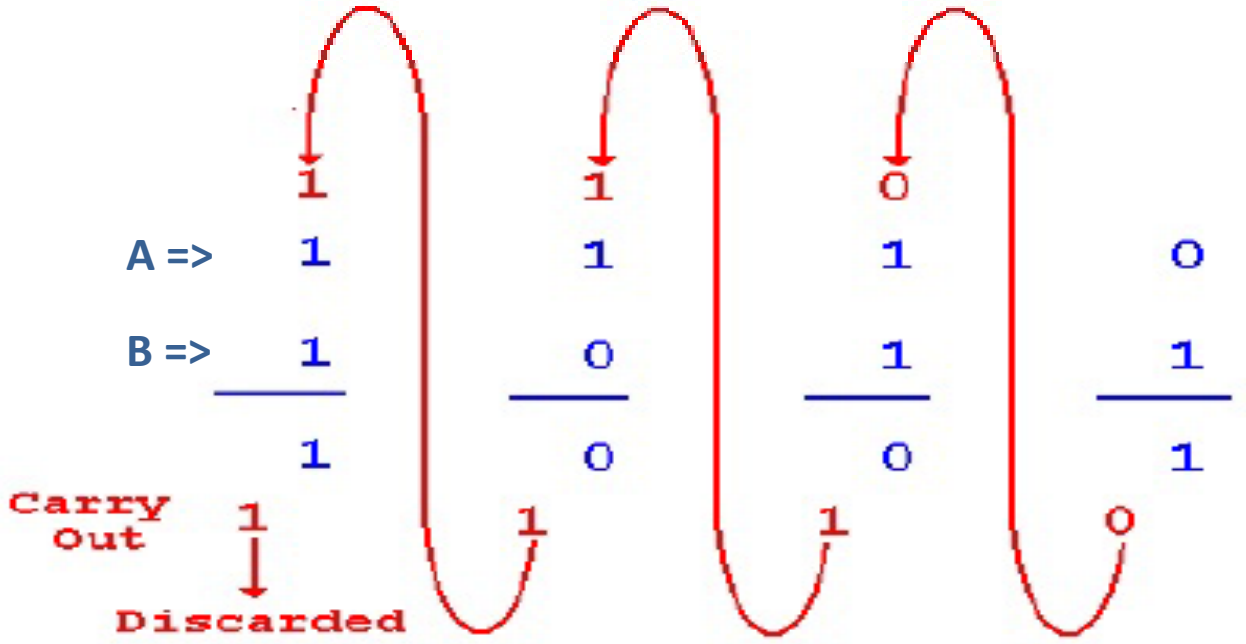
- For addition of 2 binary numbers.
- All bits are added concurrently.
- Time required does not depend on the number of bits.

Are of 2 types –

1. Carry Propagate Adder (CPA) or Ripple Carry Adder
2. Carry Look-Ahead Adder

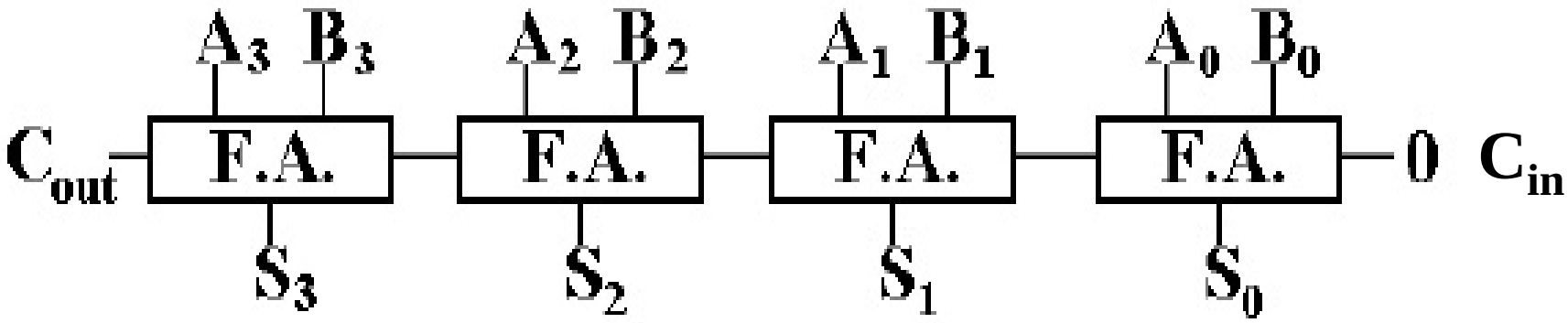
Ripple Carry Adder or CPA

- ***Propagating the Carry Bits*** – carry-out from one stage is carried into the next stage.



Cntd…

Considering both A & B are of 4bits



4 – bit Ripple Carry Adder

Disadvantage:

1. Each FA has to wait for its Carry-in generated from the previous FA (except the initial one.)
2. ***Max. Delay*** – $n\Delta$ for n-bit nos.

Carry Look-Ahead Adder

A	B	C	Sum	Carry	
0	0	0	0	0	*No Generate , No Propagate*
0	0	1	1	0	
0	1	0	1	0	*Carry Propagate*
0	1	1	0	1	
1	0	0	1	0	
1	0	1	0	1	
1	1	0	0	1	*Carry Generate*
1	1	1	1	1	

Cntd…

- A carry signal will be generated in two cases:

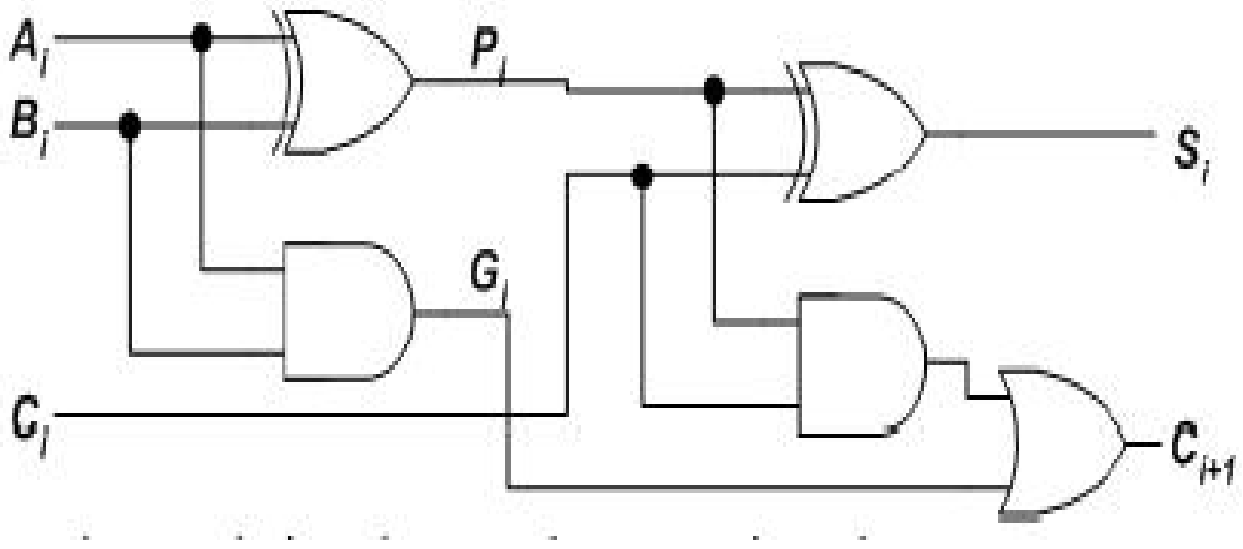
  (1) when both bits A^i and B^i are 1, or

  (2) when one of the two bits is 1 and the carry-in (carry of the previous stage) is 1.

Hence,

$$P_i = A_i \oplus B_i \quad \text{.....(1)}$$

$$G_i = A_i B_i \quad \text{.....(2)}$$



Cntd…

The output sum and carry can be defined as :

$$S_i = P_i \oplus C_i \quad \text{.....(3)}$$

$$C_{i+1} = G_i + P_i C_i \quad \text{.....(4)}$$

G_i is known as the ***carry Generate*** signal since a carry (C_{i-1}) is generated whenever G_i =1, regardless of the input carry (C_i).

P_i is known as the *carry propagate* signal since whenever P_i =1, the input carry is propagated to the output carry, i.e., $C_{i+1} = C_i$ (note that whenever P_i =1, G_i =0).

Cntd…

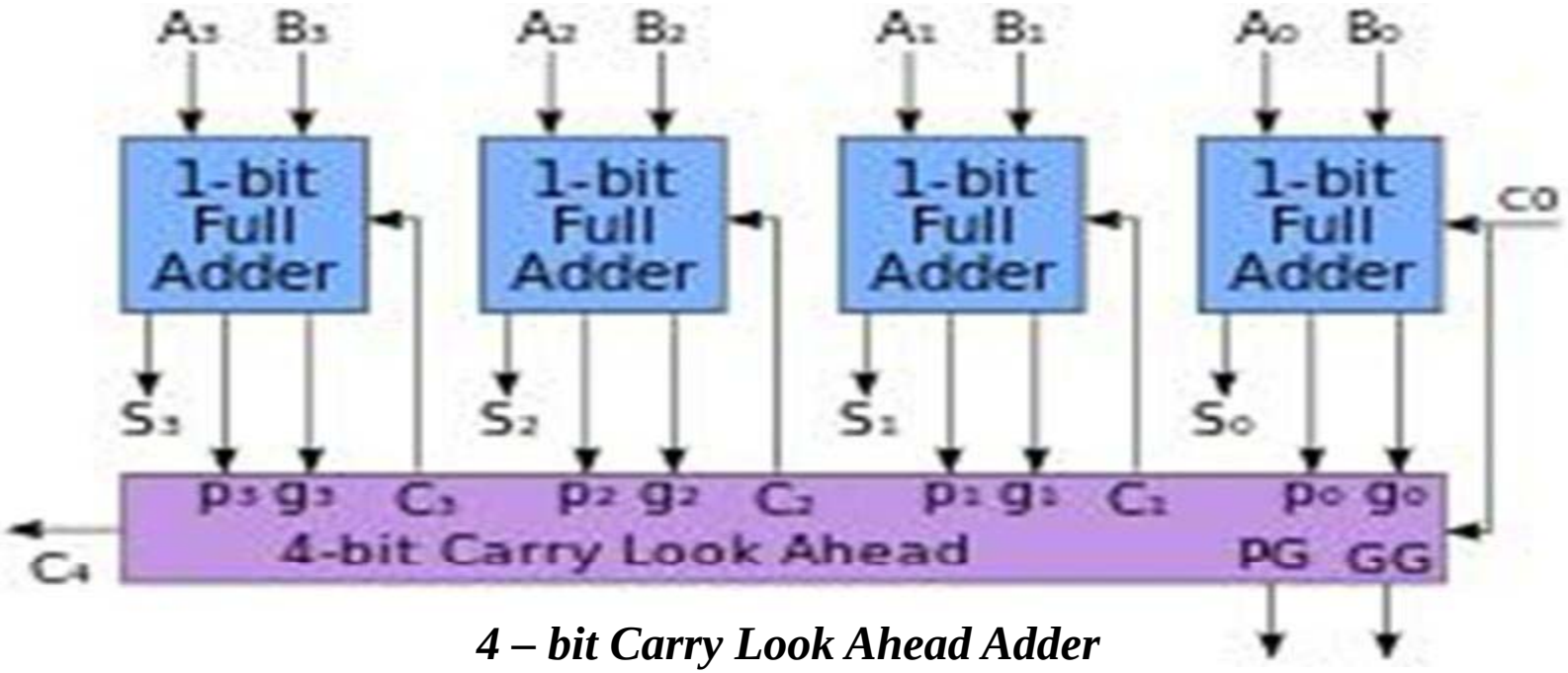
Considering 4-bit numbers –

The Boolean expression of the carry outputs of various stages can be written as follows:

$$C_1 = G_0 + P_0C_0$$
$$C_2 = G_1 + P_1C_1 = G_1 + P_1 (G_0 + P_0C_0)$$
$$= G_1 + P_1G_0 + P_1P_0C_0$$
$$C_3 = G_2 + P_2C_2 = G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0C_0$$
$$C_4 = G_3 + P_3C_3$$
$$= G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0 + P_3P_2P_1P_0C_0$$

Conclusion: All of the $Carry_{out}$ signals can be derived from the initial $Carry_{in}$ signal only.

Cntd…



4 – bit Carry Look Ahead Adder

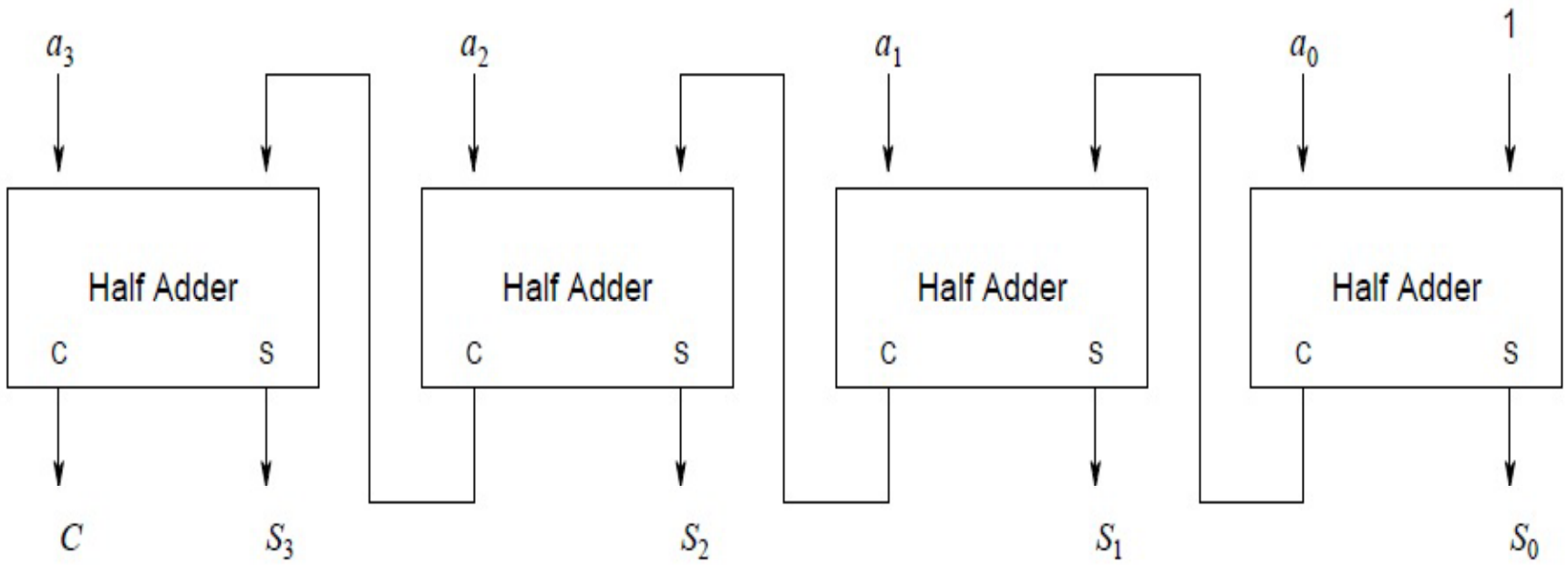
Max. Delay – 6Δ for n-bit nos. : For G_i & P_i => Δ ;
For C_i => 2Δ ; for S_i => 3Δ

HW: *1. Design a 16-bit CPA using 4-bit CPAs*
2. Derive the carry generate-propagate expressions for 8-bit CLA

4-Bit Binary Incrementer Unit

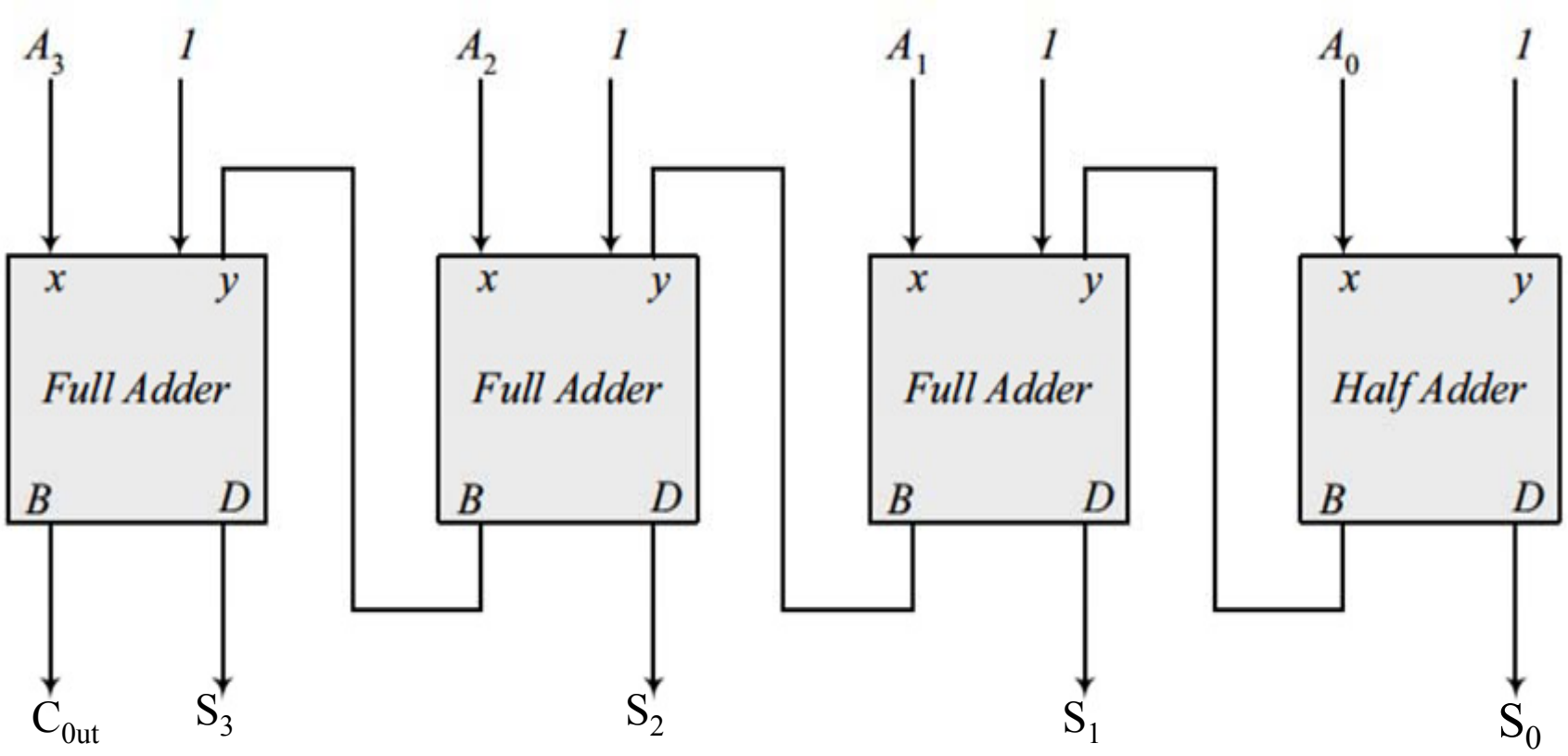
$$
\begin{array}{cccccc}
c & c & c & c & & \\
 & a_3 & a_2 & a_1 & a_0 & \\
+ & & & & 1 & \\
\hline
C & S_3 & S_2 & S_1 & S_0 &
\end{array}
$$

Addition of 1 with the actual number



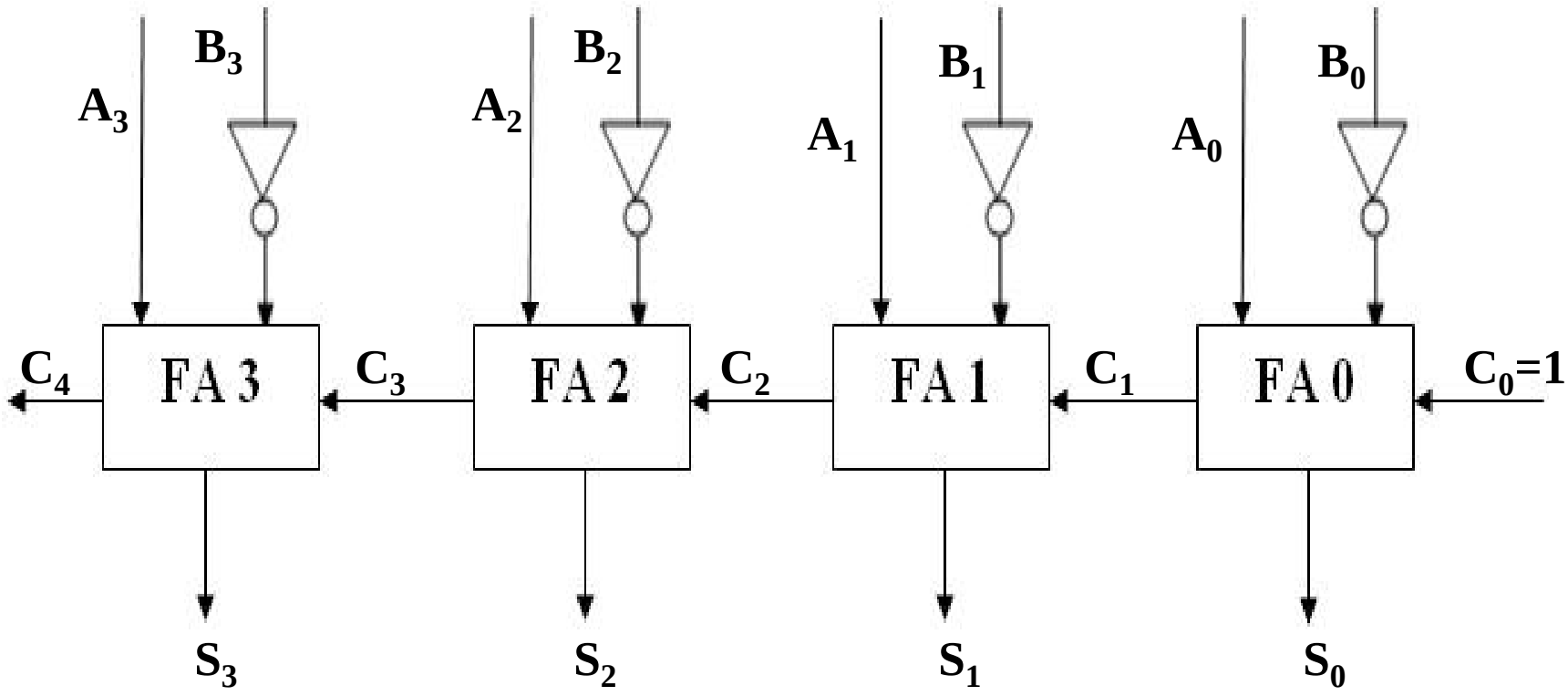
4-Bit Binary Decrementer Unit

$A_3A_2A_1A_0 - 1 \Rightarrow A_3A_2A_1A_0 + 1111$



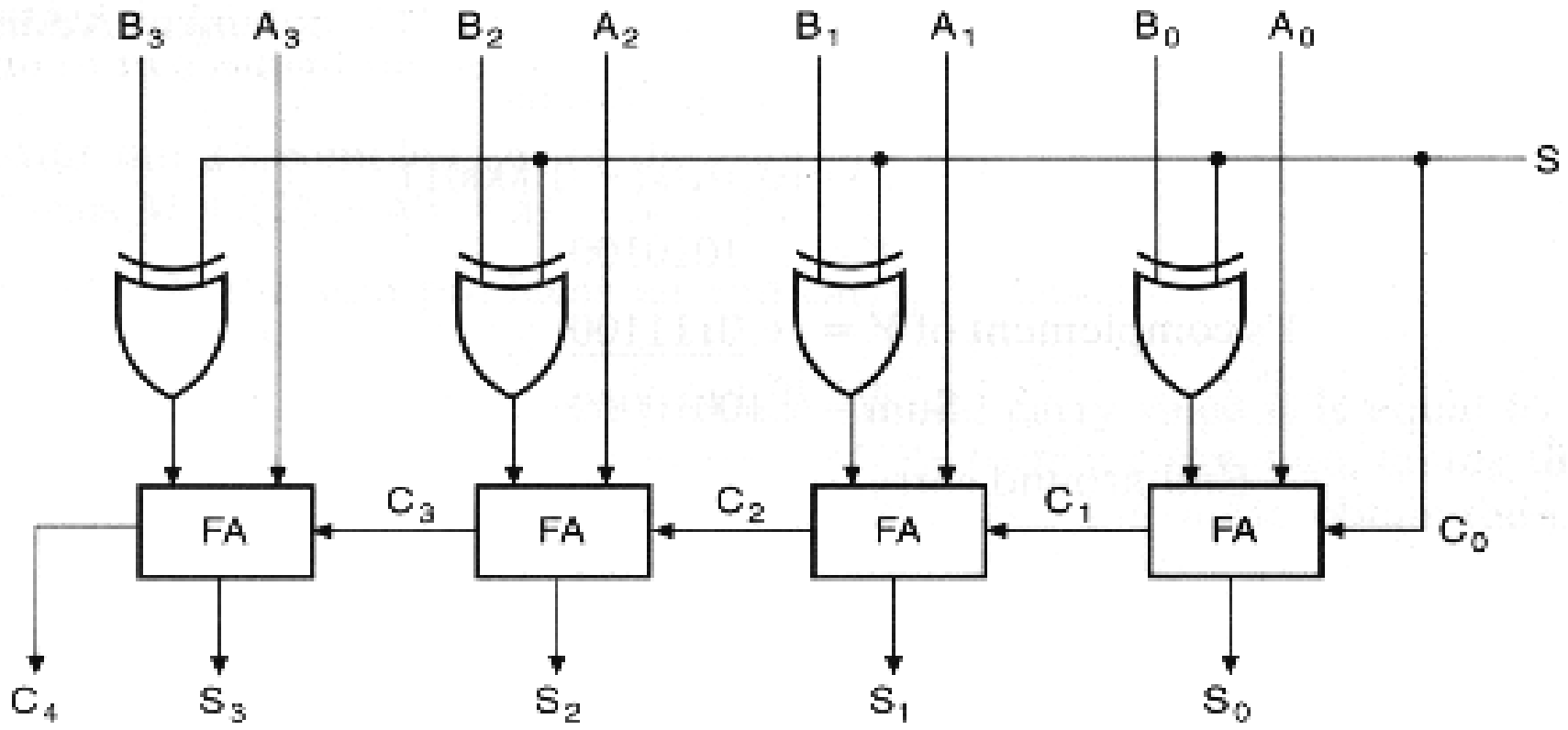
4-bit Parallel Subtractor

A-B => A + complement of B + 1



4-bit Adder/Subtractor Composite Unit

S=1 => Subtraction => A + complement of B + 1 => A – B
S=0 +> Addition => A + B + 0 => A + B




Q+A

1. If both of the inputs in CLA are 0, carry is –

   a) generated b) propagated

   c) neither generated nor propagated d) none of these

2. Basic FA performs addition of –

   a) 2 bits b) 3 bits c) 4bits d) 2 multi-bit numbers

3. FA is equivalent to HA only when carry-in is-

   a) 0 b) 1 c) 2 d) 3

4. To add 2 n-bit nos. using parallel adder, the no. of clock cycle is –

   a) 1 b) 2 c) n d) 2n

5. To add 2 n-bit nos. in serial adder, the no. of clock cycle is –

   a) 1 b) 2 c) n d) 2n

6. To add 2 n-bit nos. using CPA ,the max. delay is –

   a) Δ b) 2Δ c) 6Δ d) $n\Delta$

7. To add 2 n-bit nos. using CLA ,the max. delay is –

   a) Δ b) 2Δ c) 6Δ d) $n\Delta$

8. Which one is faster than the others –

a) Serial adder b) CPA c) CLA d) Can't be determined

9. FA has …….. Outputs

a) 1 b) 2 c) 3 d) 4

10. HA has …….. Outputs

a) 1 b) 2 c) 3 d) 4

11. FA has …….. Inputs

a) 1 b) 2 c) 3 d) 4

12. HA has …….. Inputs

a) 1 b) 2 c) 3 d) 4

13. If both of the inputs in CLA are 1, carry is –

a) generated b) propagated

c) neither generated nor propagated d) none of these

14. If both of the inputs in CLA are different, carry is –

a) generated b) propagated

c) neither generated nor propagated d) none of these

15. The initial FA for n-bit CPA can be considered as an HA

a) True b) False

16. $C_{i+1} = C_i$ is equivalent to –

a) generate b) propagate c) both d) none

17. A_iB_i is equivalent to -

a) generate b) propagate c) both d) none

References:

1. Salivahanan
2. T. K. Ghosh

Thank You