SNS COLLEGE OF ENGINEERING

Kurumbapalayam (PO), Coimbatore – 641 107
Accredited by NAAC-UGC with 'A' Grade
Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

DEPARTMENT OF ECE

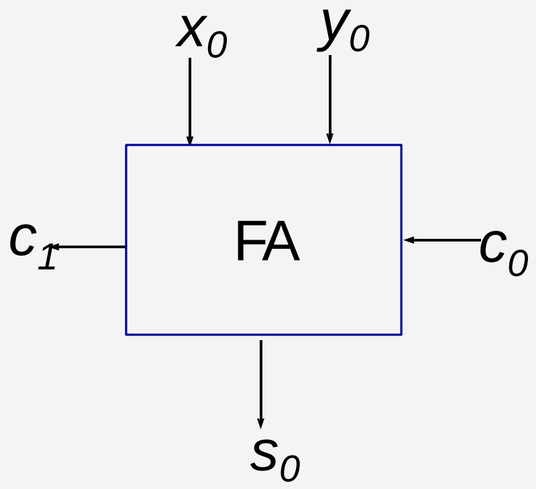
COURSE NAME: 19IT301 COMPUTER ORGANIZATION AND ARCHITECTURE

II YEAR/ III SEM

Unit 2 : ARITHMETIC OPERATIONS

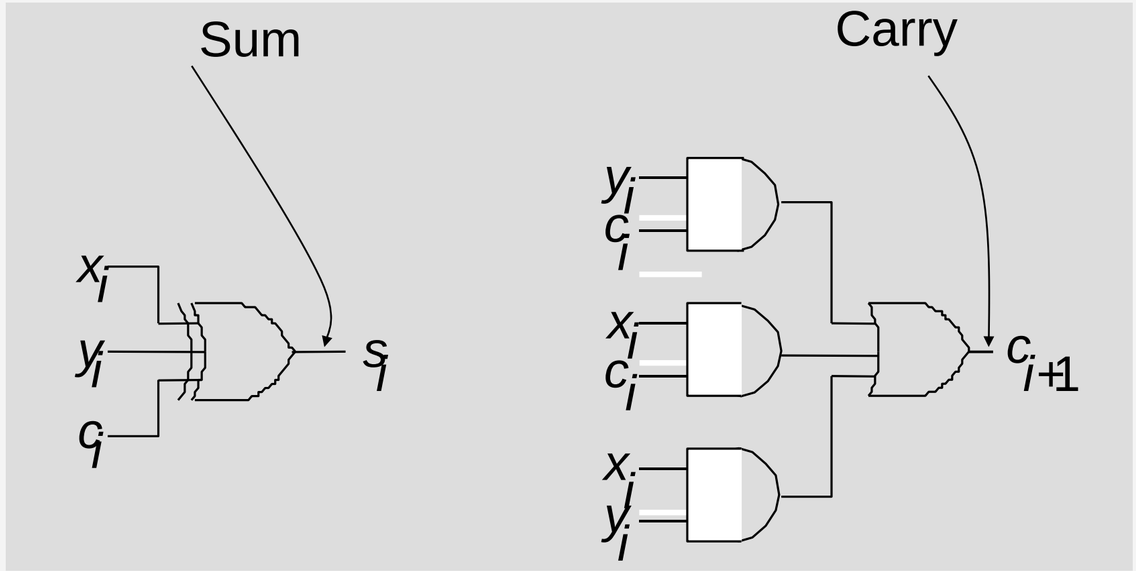
Topic 2: Design of Fast Adders

Computing the add time



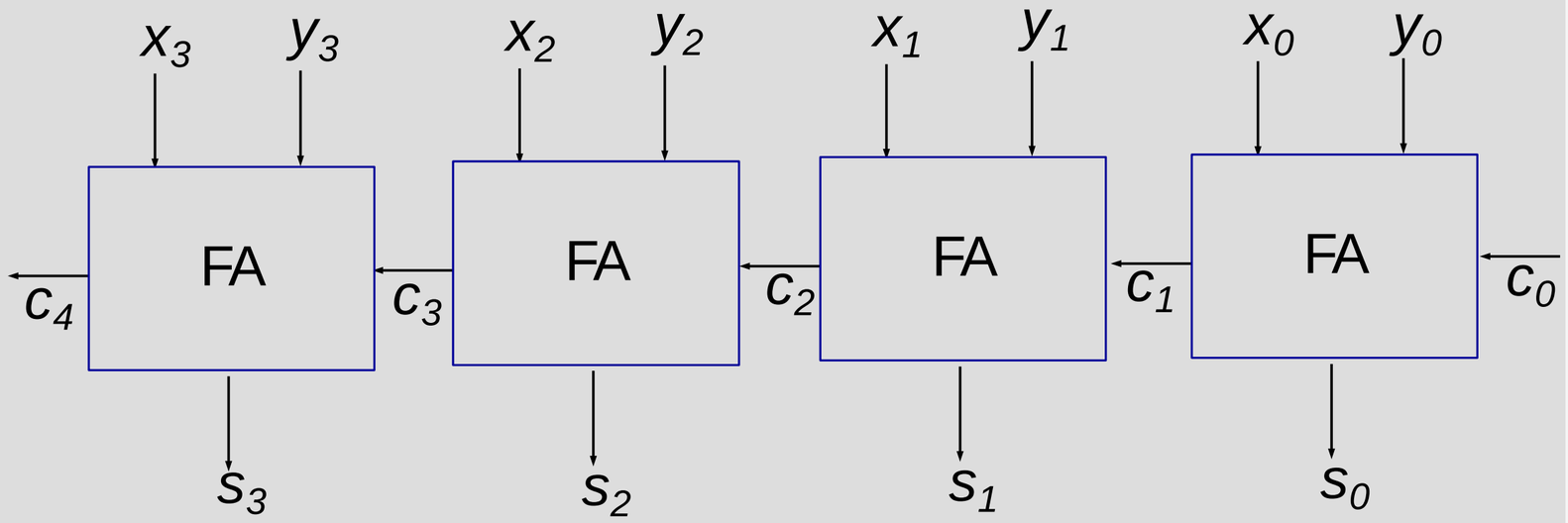
Consider 0^{th} stage:

- c_1 is available after 2 gate delays.
- s_1 is available after 1 gate delay.



Computing the add time

Cascade of 4 Full Adders, or a 4-bit adder



- s_0 available after 1 gate delays, c_1 available after 2 gate delays.
- s_1 available after 3 gate delays, c_2 available after 4 gate delays.
- s_2 available after 5 gate delays, c_3 available after 6 gate delays.
- s_3 available after 7 gate delays, c_4 available after 8 gate delays.

For an *n*-bit adder, s_{n-1} is available after *2n-1* gate delays
c_n is available after *2n* gate delays.

Drawback of ripple carry adder

All sum bits are available in 2n gate delays.

- Two approaches can be used to reduce delay in adders:

1) Use the fastest possible electronic technology in implementing the ripple-carry design.
2) Use an augmented logic-gate network structure.

Design of fast adders

Recall the equations:

$$s_i = x_i \oplus y_i \oplus c_i$$

$$c_{i+1} = x_i y_i + x_i c_i + y_i c_i$$

Second equation can be written as:

$$c_{i+1} = x_i y_i + (x_i + y_i) c_i$$

We can write:

$$c_{i+1} = G_i + P_i c_i$$

$$where\ G_i = x_i y_i\ and\ P_i = x_i + y_i$$

- G_i is called **generate function** and P_i is called **propagate functio**
- G_i and P_i are computed only from x_i and y_i and not c_i, thus they can be computed in one gate delay after X and Y are applied to the inputs of an n-bit adder.

Carry-Lookahead Addition

$$c_{i+1} = G_i + P_i c_i$$

$$c_i = G_{i-1} + P_{i-1} c_{i-1}$$

$$\Rightarrow c_{i+1} = G_i + P_i (G_{i-1} + P_{i-1} c_{i-1})$$

continuing

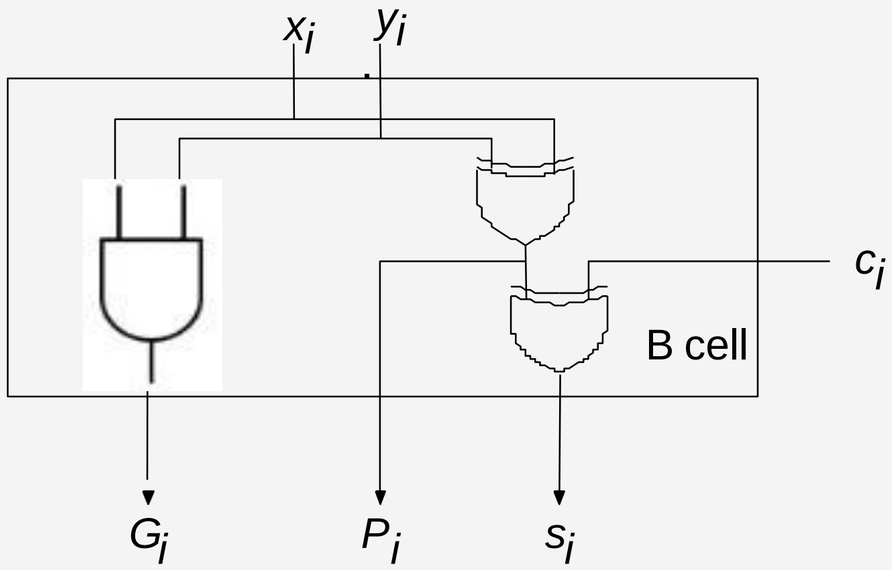
$$\Rightarrow c_{i+1} = G_i + P_i (G_{i-1} + P_{i-1} (G_{i-2} + P_{i-2} c_{i-2}))$$

until

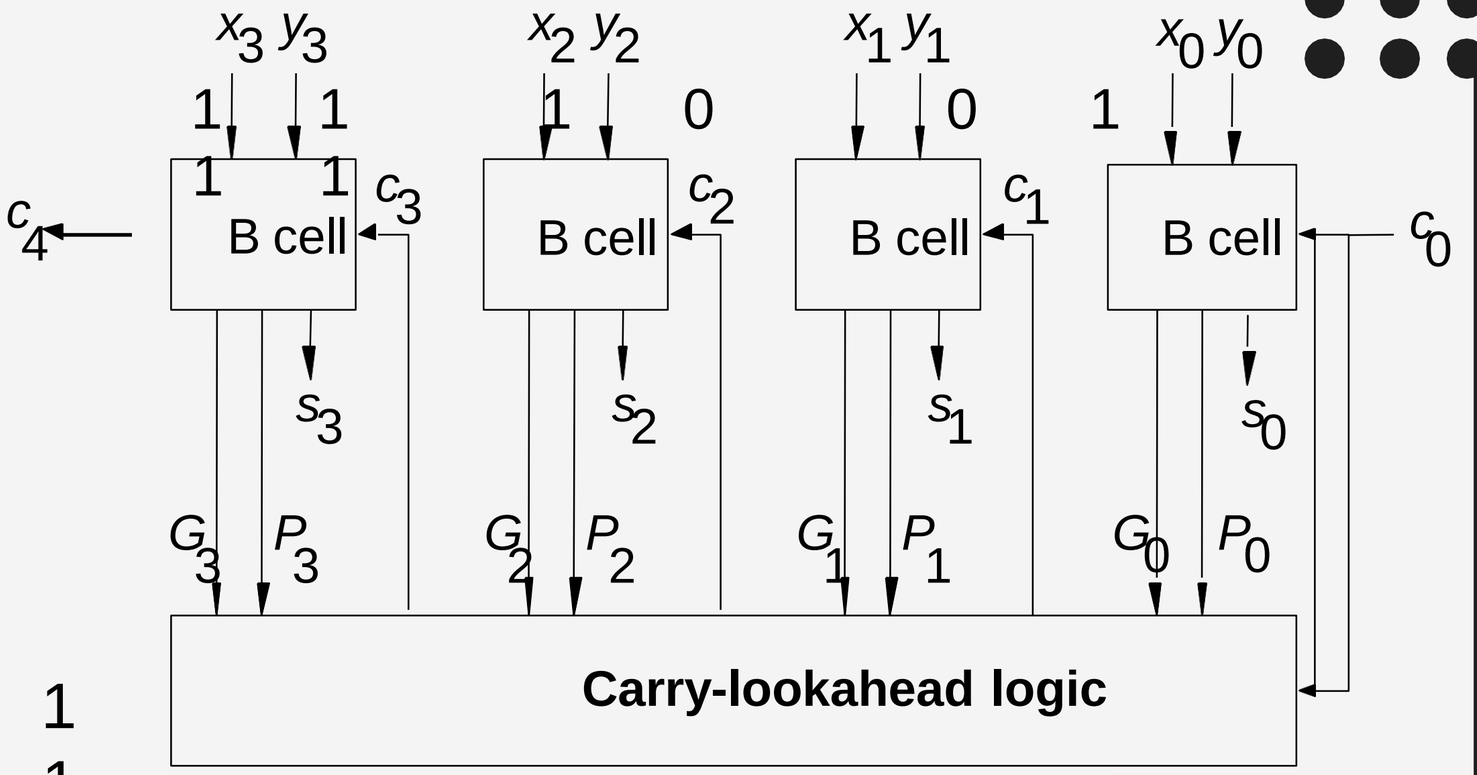
$$c_{i+1} = G_i + P_i G_{i-1} + P_i P_{i-1} G_{i-2} + .. + P_i P_{i-1} .. P_1 G_0 + P_i P_{i-1} ... P_0 c_0$$

- All carries can be obtained 3 gate delays after *X*, *Y* and c_0 are applied.
  - One gate delay for P_i and G_i
  - Two gate delays in the AND-OR circuit for c_{i+1}
- All sums can be obtained 1 gate delay after the carries are computed.
- Independent of *n*, *n*-bit addition requires only 4 gate delays.
- This is called Carry look-ahead adder.

Carry-Lookahead Adder



Bit-stage cell



4-bit carry look-ahead adder

Example:

X : 1 1 0 1

Y: 1 0 1 1

1 1 0 0

0

Actually $P_i = x_i + y_i$

But, $P_i = x_i \oplus y_i$

Example

$s_0 = x_0 \oplus y_0 \oplus c_0 = 1 \oplus 1 \oplus 0 = 0$

$c_0 = 0$

$G_0 = x_0 y_0 = 1 . 1 = 1$

$P_0 = x_0 \oplus y_0 = 1 \oplus 1 = 0$

$c_{i+1} = G_i + P_i c_i$

$c_1 = G_0 + P_0 c_0$

$c_1 = 1 + 0 = 1$

$s_1 = x_1 \oplus y_1 \oplus c_1 = 0 \oplus 1 \oplus 1 = 0$

$c_2 = G_1 + P_1 c_1$

$= G_1 + P_1(G_0 + P_0 c_0)$

$c_2 = G_1 + P_1 G_0 + P_1 P_0 c_0$

$G_1 = x_1 y_1 = 0 . 1 = 0$

$P_1 = x_1 \oplus y_1 = 0 \oplus 1 = 1$

$c_2 = 0 + 1.1 + 1.0.0 = 1$

Truth table for OR

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

Truth table for Ex-or

A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

Example(cntd)

$s_2 = x_2 \oplus y_2 \oplus c_2 = 1 \oplus 0 \oplus 1 = 0$

$c_3 = G_2 + P_2c_2$

$= G_2 + P_2(G_1 + P_1G_0 + P_1P_0c_0)$

$c_3 = G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0c_0$

$G_2 = x_2y_2 = 1.0 = 0$

$P_2 = x_2 \oplus y_2 = 1 \oplus 0 = 1$

$c_3 = 0 + 1.0 + 1.1.1 + 1.1.0.0 = 1$

$s_3 = x_3 \oplus y_3 \oplus c_3 = 1 \oplus 1 \oplus 1 = 1$

$c_4 = G_3 + P_3c_3$

$= G_3 + P_3(G_2 + P_2G_1 + P_2P_1G_0 + P_2P_1P_0c_0)$

$c_4 = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0 + P_3P_2P_1P_0c_0$

$G_3 = x_3y_3 = 1.1 = 1$

$P_3 = x_3 \oplus y_3 = 1 \oplus 1 = 0$

$c_4 = 1 + 0.0 + 0.1.0 + 0.1.1.1 + 0.1.1.0.0 = 1 + 0 + 0 + 0 + 0 = 1$

Carry-Lookahead Adder

- Performing n-bit addition in 4 gate delays independent of n is good only theoretically because of fan-in constraints.
- Last AND gate and OR gate require a fan-in of (n+1) for a n-bit adder.
  - For a 4-bit adder (n=4) fan-in of 5 is required.
  - Practical limit for most gates.
- In order to add operands longer than 4 bits, we can cascade 4-bit carry look-ahead adders.
- Cascade of carry look-ahead adders is called **Blocked Carry look-ahead adder.**

Higher-level Generate & Propagate Functions

Carry-out from a 4-bit block can be given as:

$$c_4 = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0 + P_3P_2P_1P_0c_0$$

$c_3 = G_2 + P_2G_1 + P_2 P_1G_0 + P_2 P_1P_0c_0$

Rewrite this as:

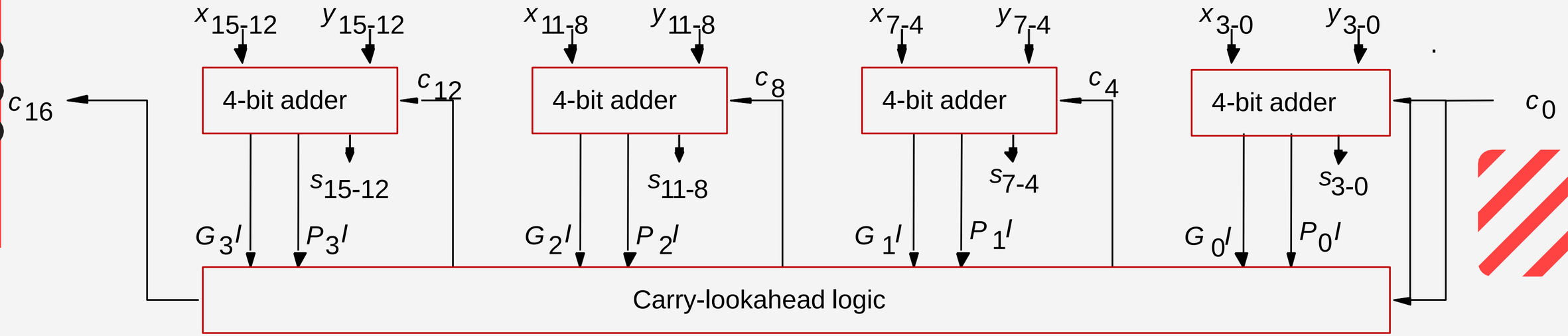
$$P_0^I = P_3P_2P_1P_0$$

$$G_0^I = G_3 + P_3G_2 + P_3P_2G_1 + P_3P_2P_1G_0$$

Subscript *I* denotes the blocked carry look-ahead and identifies the block cascade 4 *4*-bit adders, c_{16} can be expressed as:

$$c_{16} = G_3^I + P_3^I G_2^I + P_3^I P_2^I G_1^I + P_3^I P_2^I P_1^0 G_0^I + P_3^I P_2^I P_1^0 P_0^0 c_0$$

Blocked Carry Look-ahead adder



After x_i, y_i and c_0 are applied as inputs:

- G_i and P_i for each stage are available after 1 gate delay.
- P^I is available after 2 and G^I after 3 gate delays.
- All carries are available after 5 gate delays.
- c_{16} is available after 5 gate delays.

Assessment

1. What distinguishes the look-ahead-carry adder?
a) It is slower than the ripple-carry adder
b) It is easier to implement logically than a full adder
c) It is faster than a ripple-carry adder
d) It requires advance knowledge of the final answer

2. Carry look-ahead logic uses the concepts of _____
a) Inverting the inputs
b) Complementing the outputs
c) Generating and propagating carries
d) Ripple factor

Assessment

3. What is one disadvantage of the ripple-carry adder?
a) The interconnections are more complex
b) More stages are required to a full adder
c) It is slow due to propagation time
d) All of the mentioned

4. The carry propagation delay in 4-bit full-adder circuits _____
a)Is cumulative for each stage and limits the speed at which arithmetic operations are performed
b)Is normally not a consideration because the delays are usually in the nanosecond range
c) Decreases in direct ratio to the total number of full-adder stages
d)Increases in direct ratio to the total number of full-adder stages, but is not a factor in limiting the speed of arithmetic operations

Answers

1. C
2. C
3. C
4. A

Thank You