Computer Organization & Architecture

Chapter 9 – Design of Fast Adders

Zhang Yang　张杨

cszyang@scut.edu.cn

Autumn 2025
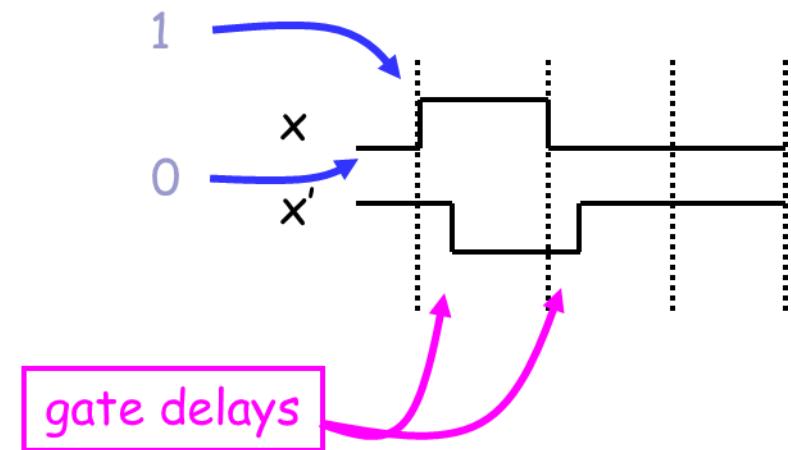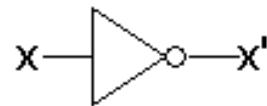
Content of this lecture

- **9.2 Design of Fast Adders**
  - ☐ Gate Delays
  - ☐ Carry-Lookahead Addition
  - ☐ 4-bit Carry-Lookahead Adder
  - ☐ Extension to Longer Adders
  - ☐ Solved Problem Example 9.1 (P374)
  - ☐ Summary

Gate Delays (1)

■ Every gate takes some small fraction of a second between the time inputs are presented and the time the correct answer appears on the outputs. This little fraction of a second is called a gate delay.

■ We can use a timing diagram to show gate delays graphically.

Gate Delays (2)

- There are actually detailed ways of calculating gate delays that can get quite complicated, but for this class, let's just assume that there's some small constant delay that's the same for all gates.
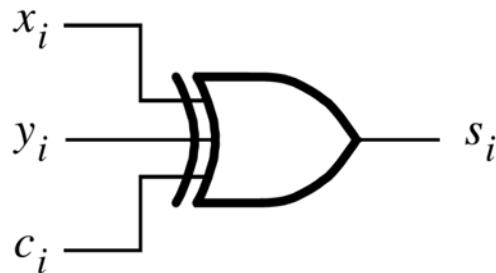
Delays in n-bit Ripple Carry Adder (1)

- The delay through a network of logic gates depends on the integrated circuit electronic technology used in fabricating the network and on the number of gates in the paths from inputs to outputs.

- The delay through any combinational logic network constructed from gates in a particular technology is determined by adding up the number of logic-gate delays along the longest signal propagation path through the network.

- In the case of the n-bit ripple-carry adder, the longest path is from inputs x_0, y_0 and c_0 at the LSB position to outputs c_n and s_{n-1} at the MSB position.
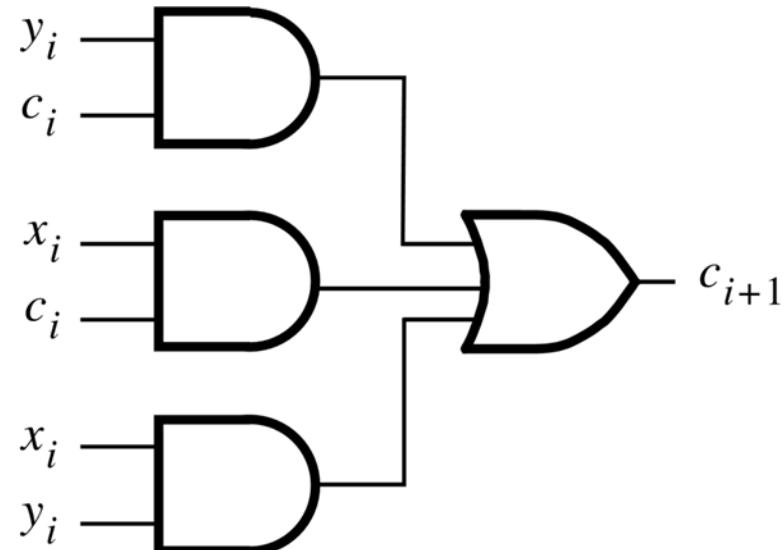
Delays in n-bit Ripple Carry Adder (2)

- ## 1-bit Full Adder Delay Analysis
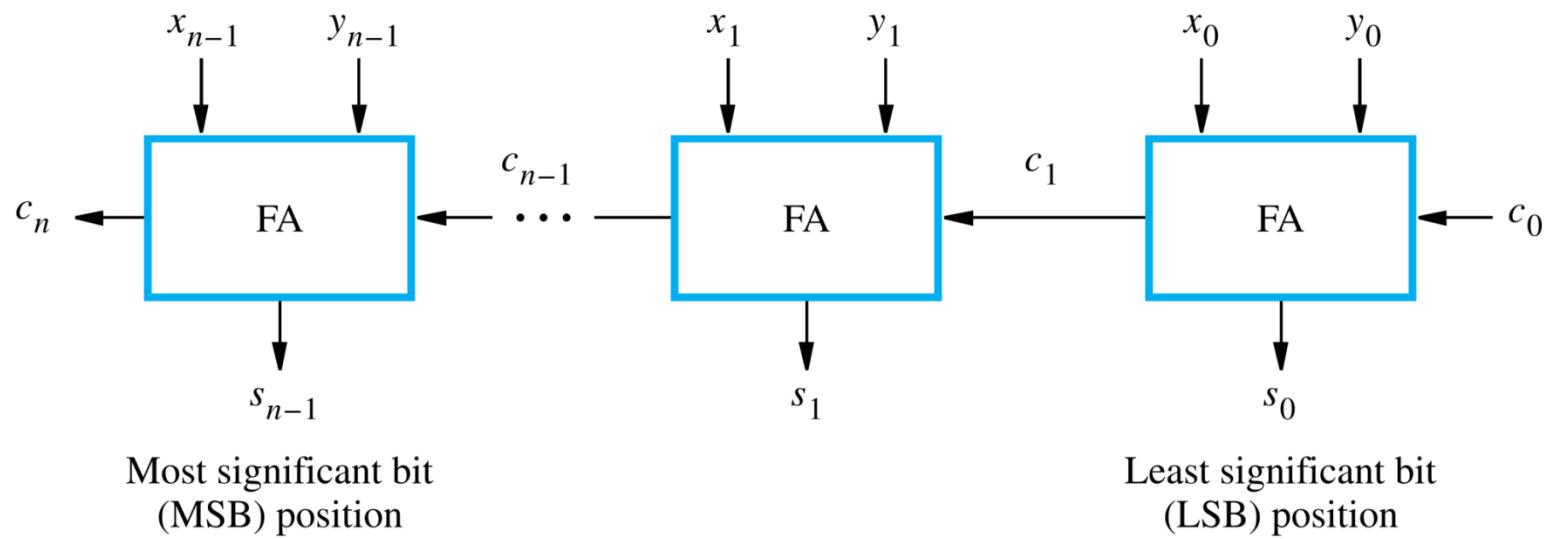 - Assume that the gate delay of an AND gate or an OR gate or a XOR gate is T.

c_{i+1}	2T
Si	T

Delays in n-bit Ripple Carry Adder (3)

■ n-bit Ripple Carry Adder Delay Analysis

c_{n-1} $2(n-1)T$

s_{n-1} $2(n-1)T+T=(2n-1)T$

c_n $2(n-1)T+2T=(2n)T$

Drawback of n-bit ripple-carry adder

- Drawback
  - It may have too much delay in developing its outputs, s_0 through s_{n-1} and c_n.
    - s_{n-1} $(2n-1)T$
    - c_n $(2n)T$

- Solutions
  - Use the fastest possible electronic technology in implementing the ripple-carry logic design or variations of it.

  - Use an augmented logic gate network structure.

Content of this lecture

- **9.2 Design of Fast Adders**
  - ☐ Gate Delays
  - ☐ Carry-Lookahead Addition
  - ☐ 4-bit Carry-Lookahead Adder
  - ☐ Extension to Longer Adders
  - ☐ Solved Problem Example 9.1 (P374)
  - ☐ Summary

Carry-Lookahead Addition (1)

- Generate Function & Propagate Function

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

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

$$= x_i y_i + (x_i + y_i) c_i$$

$$G_i = x_i y_i \qquad \text{(generate function for stage i)}$$

$$P_i = x_i + y_i \qquad \text{(propagate function for stage i)}$$

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

Carry-Lookahead Addition (2)

- Generate Function & Propagate Funcation (ctd.)

$$P_i = x_i + y_i \quad \longrightarrow \quad P_i = x_i \oplus y_i$$

x_i	y_i	$x_i + y_i$	$x_i \oplus y_i$
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	0

if $\quad x_i = 1 \quad$ and $\quad y_i = 1$

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

Carry-Lookahead Addition (3)

- A simpler circuit of bit stage
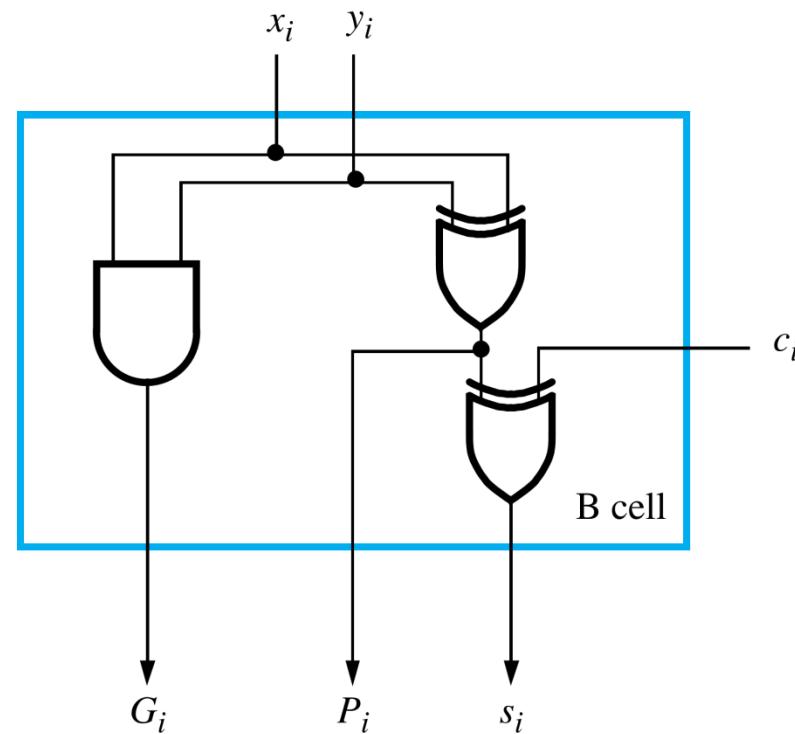

Figure 9.4   (a) Bit-stage cell

Carry-Lookahead Addition (4)

- Logic Functions Extension

$$S_i = x_i \oplus y_i \oplus c_i$$

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

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

$$= G_i + P_i G_{i-1} + P_i P_{i-1} c_{i-1}$$

$$\ldots$$

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

Carry-Lookahead Addition (5)

- Delays of n-bit CLA

  □ All carries can be obtained 3 gate delays after the input signal X, Y and c_0 are applied.

  □ All sum bits are available 4 gate delays after the input signal X, Y and c_0 are applied.

- CLA Summary

  □ Provide supplemental logic circuits that form a carry signal into each adder stage

    - To accelerate the propagation of carries

    - Carry propagation becomes concurrent instead of sequential

Content of this lecture

- 9.2 Design of Fast Adders
  - Gate Delays
  - Carry-Lookahead Addition
  - 4-bit Carry-Lookahead Adder
  - Extension to Longer Adders
  - Solved Problem Example 9.1 (P374)
  - Summary

4-bit Carry-Lookahead Adder (1)

- Logic Functions of 4-bit Carry-Lookahead Adder

$$s_i = x_i \oplus y_i \oplus c_i \qquad i = 0,1,2,3$$

$$c_1 = G_0 + P_0 c_0$$

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

$$c_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 c_0$$

$$c_4 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 c_0$$

4-bit Carry-Lookahead Adder (2)

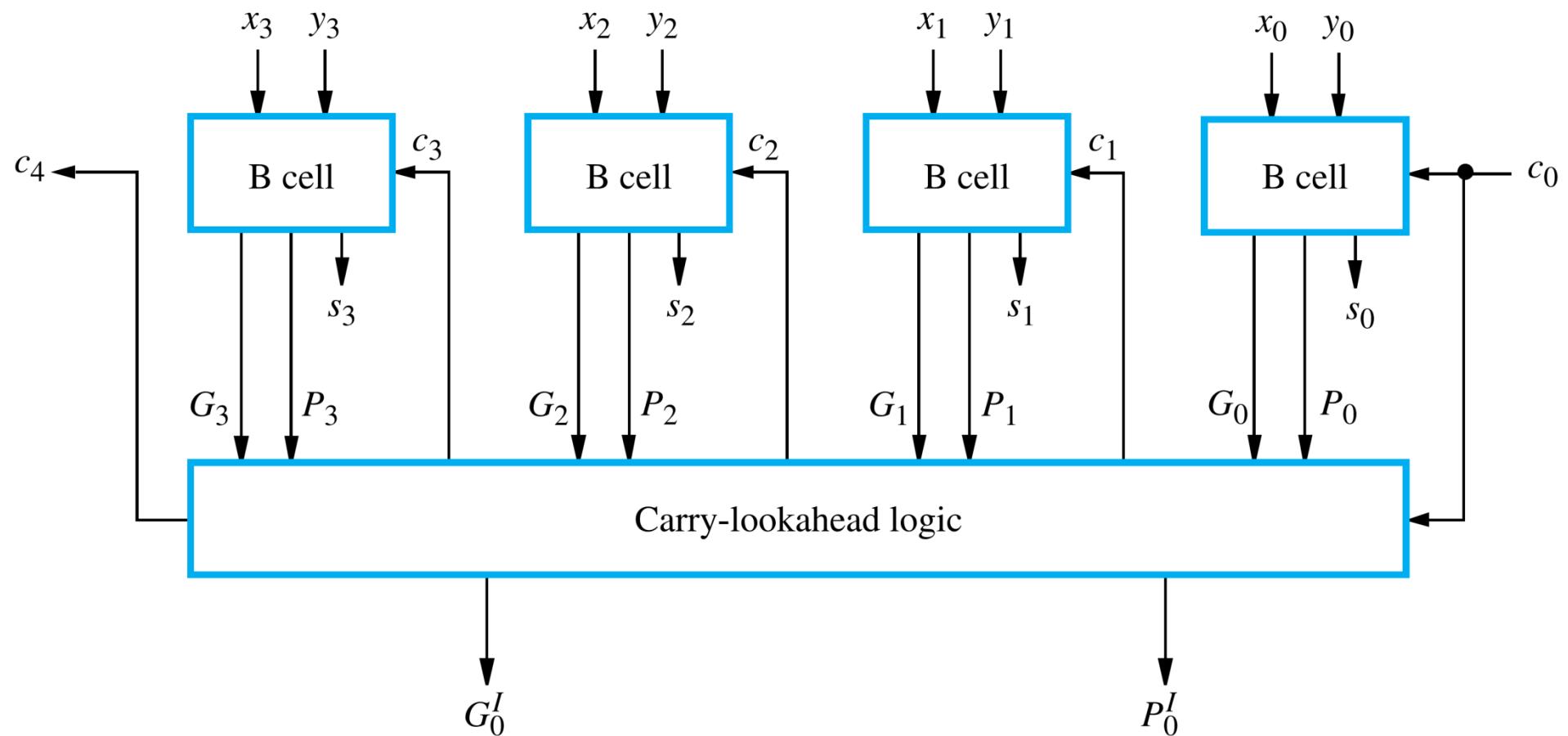■ Figure of 4-bit Carry-Lookahead Adder

Figure 9.4 (b) 4-bit carry-lookahead adder

4-bit Carry-Lookahead Adder (3)

- **Delay Comparison**
  - 4-bit CLA
    - c_4, c_3, c_2, c_1 3T
    - s_3, s_2, s_1, s_0 4T
  - 4-bit Ripple-Carry Adder
    - c_4 8T
    - s_3 7T

Content of this lecture

- **9.2 Design of Fast Adders**
  - ☐ Gate Delays
  - ☐ Carry-Lookahead Addition
  - ☐ 4-bit Carry-Lookahead Adder
  - ☐ **Extension to Longer Adders**
  - ☐ Solved Problem Example 9.1 (P374)
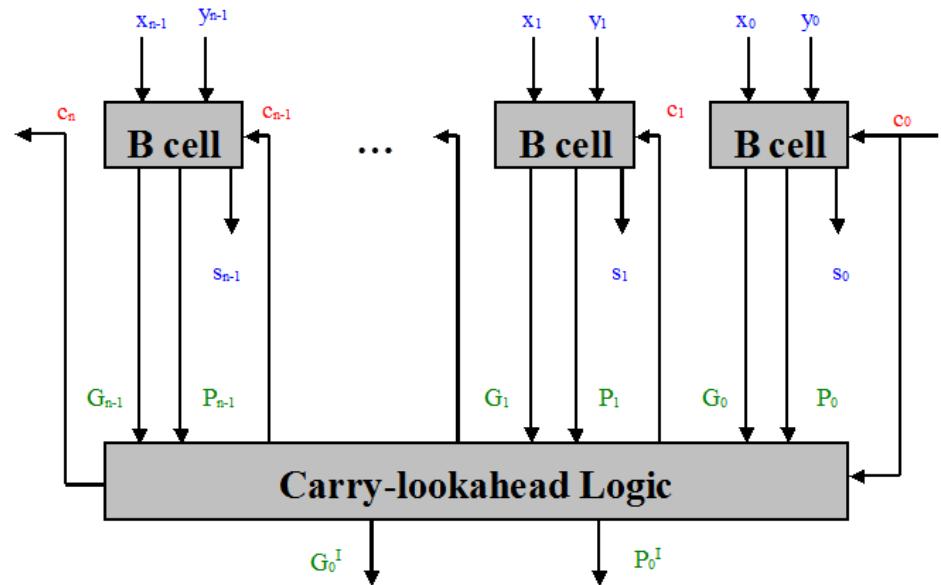 - ☐ Summary

Can the 4-bit carry-lookahead adder design be directly extended to longer operand sizes?

A   Yes, it can.

B   No, it can't.

Extension to Longer Adders (1)

- **Fan-In Constraints**
  - ☐ Fan-In
    - ■ The number of inputs to a logic gate is called its fan-in.
  - ☐ Fan-Out
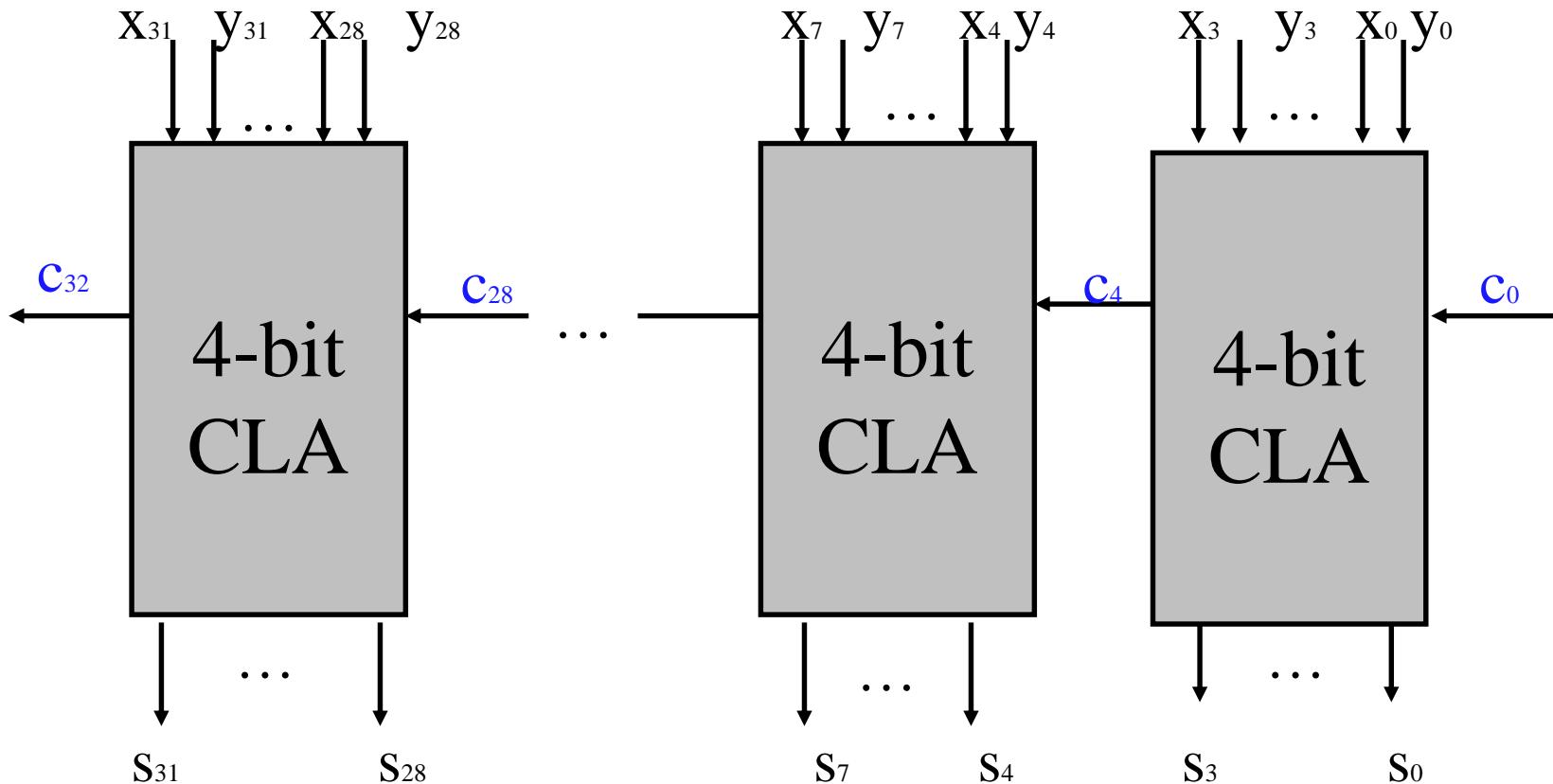   - ■ The number of gate inputs that the output of a logic gate drives is called its fan-out.
  - ☐ Practical circuits do not allow large fan-in and fan-out because they both have an adverse effect on the propagation delay and hence the speed of the circuit.
  - ☐ Example: Fan-In of the last AND gate is i + 2.

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

Extension to Longer Adders (2)

- Build 32-bit Adders with 4-bit Adders
  - Cascade of 8 4-bit Carry-Lookahead Adders

Extension to Longer Adders (3)

- Build 32-bit Adders with 4-bit Adders (ctd.)
  - Cascade of 8 4-bit Carry-Lookahead Adders (ctd.)
    - Calculate the delays in generating sum bits s28 , s29 , s30 s31, and c32 in the high-order 4-bit adder.

c_4 3T c_8 3T+2T=5T

c_{12} 5T+2T   …

c_{28} $(6 \times 2)T+3T=15T$

c_{29} c_{30} c_{31} c_{32} 15T+2T=17T

s_{28} s_{29} s_{30} s_{31} 17T+T=18T

Content of this lecture

- 9.2 Design of Fast Adders
  - Gate Delays
  - Carry-Lookahead Addition
  - 4-bit Carry-Lookahead Adder
  - Extension to Longer Adders
  - Solved Problem Example 9.1 (P374)
  - Summary

Solved Problems (1)

■ Example 9.1

Problem: How many logic gates are needed to build the 4-bit carry-lookahead adder shown in Figure 9.4?

Solution: Each B cell requires 3 gates as shown in Figure 9.4a. Hence, 12 gates are needed for all four B cells.

The carries c_1, c_2, c_3, and c_4, produced by the carry-lookahead logic, require 2, 3, 4, and 5 gates, respectively, according to the four logic expressions in Section 9.2.1. The carry-lookahead logic also produces G_0^I, using 4 gates, and P_0^I, using 1 gate, as also shown in Section 9.2.1. Hence, a total of 19 gates are needed to implement the carry-lookahead logic.

The complete 4-bit adder requires $12 + 19 = 31$ gates, with a maximum fan-in of 5.
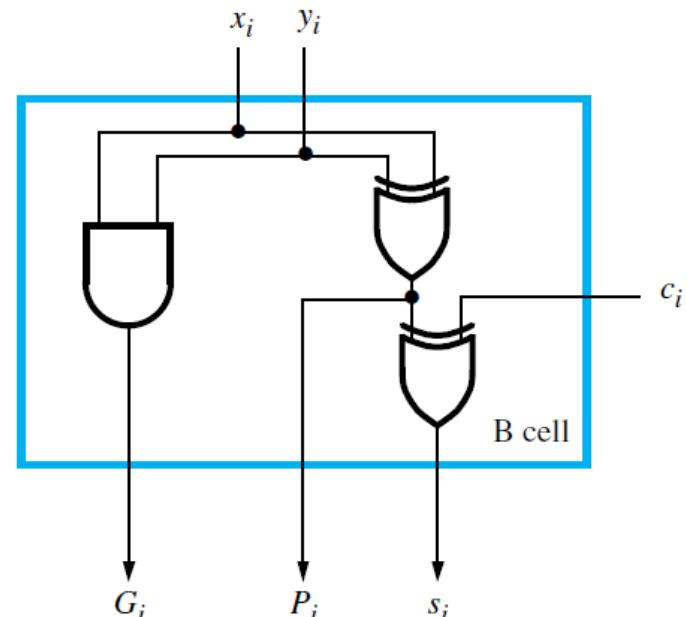
Solved Problems (2)
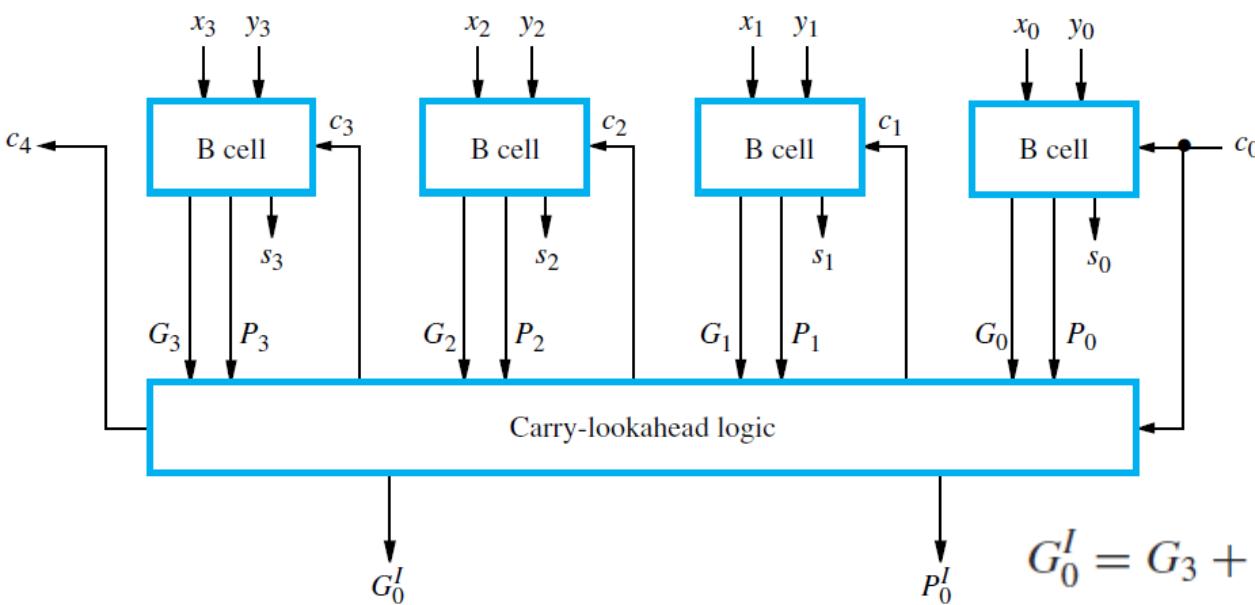
- ## Example 9.1(ctd.)

$$c_1 = G_0 + P_0 c_0$$
$$c_2 = G_1 + P_1 G_0 + P_1 P_0 c_0$$
$$c_3 = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 c_0$$
$$c_4 = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 c_0$$

(a) Bit-stage cell

(b) 4-bit adder

$$P_0^I = P_3 P_2 P_1 P_0$$

$$G_0^I = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0$$

Summary (1)

- 知识点 Gate Delays

  □ Delay analysis of n-bit ripple-carry adder

- 掌握程度

  □ 给定位数的行波进位加法器，能够分析各位和以及进位的时间延迟

Summary (2)

- 知识点 Carry-lookahead adder
  - Logic function of Gi and Pi
  - Figure of Bit-Stage cell
  - Logic function of 4-bit carry-lookahead adder
  - Figure of 4-bit carry-lookahead adder
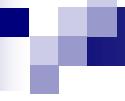 - Delay analysis of n-bit carry-lookahead adder

Summary (3)

- 掌握程度
  - 熟练写出Gi和Pi的逻辑表达式
  - 能够画出bit-stage cell的逻辑框图
  - 熟练写出4位先行进位加法器中各进位的表达式
  - 能够画出4位先行进位加法器的逻辑框图
  - 掌握理论上n位先行进位加法器中和与进位的时间延迟

Summary (4)

- 知识点 Hierarchical Adder Design

- 掌握程度

  - 能够用较少位数的先行进位加法器构造更多位数的加法器并分析出各位和与进位的时间延迟

Homework

- P377 9.1

- Reading P342 Higher-Level Generate and Propagate Functions

Exercise (1)

- 1. A 16-bit ripple-carry adder is used to add two numbers X(x15x14…x1x0) and Y(y15y14…y1y0) to generate a 16-bit sum S(s15s14…s1s0). How many gate delays are required to compute s13 after all the inputs have been applied?

  - ☐ A. 8
  - ☐ B. 25
  - ☐ C. 27
  - ☐ D. 31

Exercise (2)

- 2. In carry-lookahead adder,which expression is called the generate function Gi for stage i?
  - A. $x_i + y_i$
  - B. $x_i \oplus y_i$
  - C. $\underline{x_i y_i}$
  - D. $x_i \oplus y_i \oplus c_i$

Exercise (3)

- 3. In carry-lookahead adder, which expression is called the propagate function Pi for stage i?
  - ☐ A. $x_i + y_i$
  - ☐ B. $x_i \oplus y_i$
  - ☐ C. $x_i y_i$
  - ☐ D. $x_i \oplus y_i \oplus c_i$

Exercise (4)

- 4. A 12-bit hierarchical adder is used to add two numbers X(x11x10…x1x0) and Y(y11y10…y1y0) to generate a 12-bit sum S(s11s10…s1s0).This adder is constituted with three 4-bit carry-lookahead adders. How many gate delays are required to compute the most significant bit s11 of the sum S?

  - A. 5 gate delays
  - B. 7 gate delays
  - C. 8 gate delays
  - D. 9 gate delays