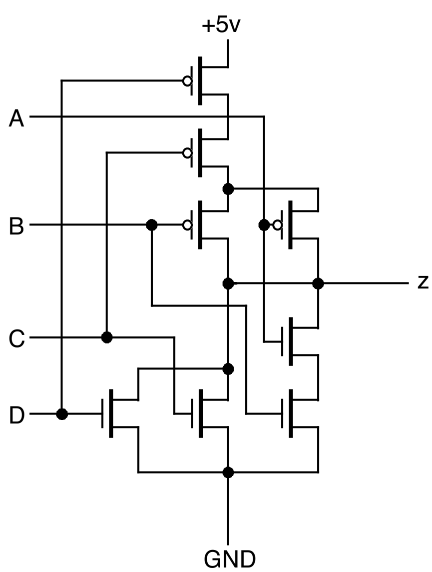
CMSC 313 Lecture 17

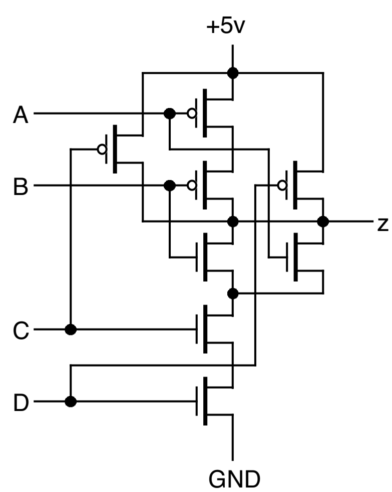
- Focus Groups
  - Need good sample of all types of CS students
  - Mon 11/17 & Thu 11/20, 12:30p-2:00p & 6:00p-7:30p
- Announcement: in-class lab Thu 10/30
- Homework 3 Questions
- Circuits for Addition
- Midterm Exam returned

Due: October 30, 2003

1. (20 points) Draw schematics for the following functions using AND, OR and NOT gates. (Do not simplify the formulas.)
   (a) $X(Y + Z)$
   (b) $\overline{X} + \overline{Y}\,\overline{Z}$
   (c) $\overline{X(Y + Z)}$
   (d) $W(X + YZ)$
2. (10 points) Question A.3, page 493, Murdocca & Heuring
3. (10 points) Prove the Consensus Theorem $AB + \overline{A}C + BC = AB + \overline{A}C$ using the postulates and theorems of Boolean algebra (except the Consensus Theorem itself) in Table A-1 (p. 451). *Hint:* use absorption creatively.
4. (40 points) For each CMOS circuit below,
   (a) Provide a truth table for the circuit's function.
   (b) For diagram (a), write down the Sum-of-Products (SOP) Boolean formula for the truth table. For diagram (b), wrtie down the Product-of-Sums (POS) Boolean formula.
   (c) Simplify the SOP or POS formula using the postulates and theorems of Boolean Algebra (p. 451). *Show all work.*
   (d) Draw the logic diagram of the simplified formula using AND, OR, NAND, NOR and NOT gates.



(a)



(b)

Last Time

- Postulates & Theorems of Boolean Algebra
- Periodic Table & Semiconductors
- P-N junction
- Field-Effect Transistors
- CMOS Logic Gates

Half Adder

- Inputs: A and B
- Outputs: $S =$ lower bit of $A + B$, $c_{\text{out}} =$ carry bit

A	B	S	c_{out}
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

- Using Sum-of-Products: $S = \overline{A}B + A\overline{B}$, $c_{\text{out}} = AB$.
- Alternatively, we could use XOR: $S = A \oplus B$.

Full Adder

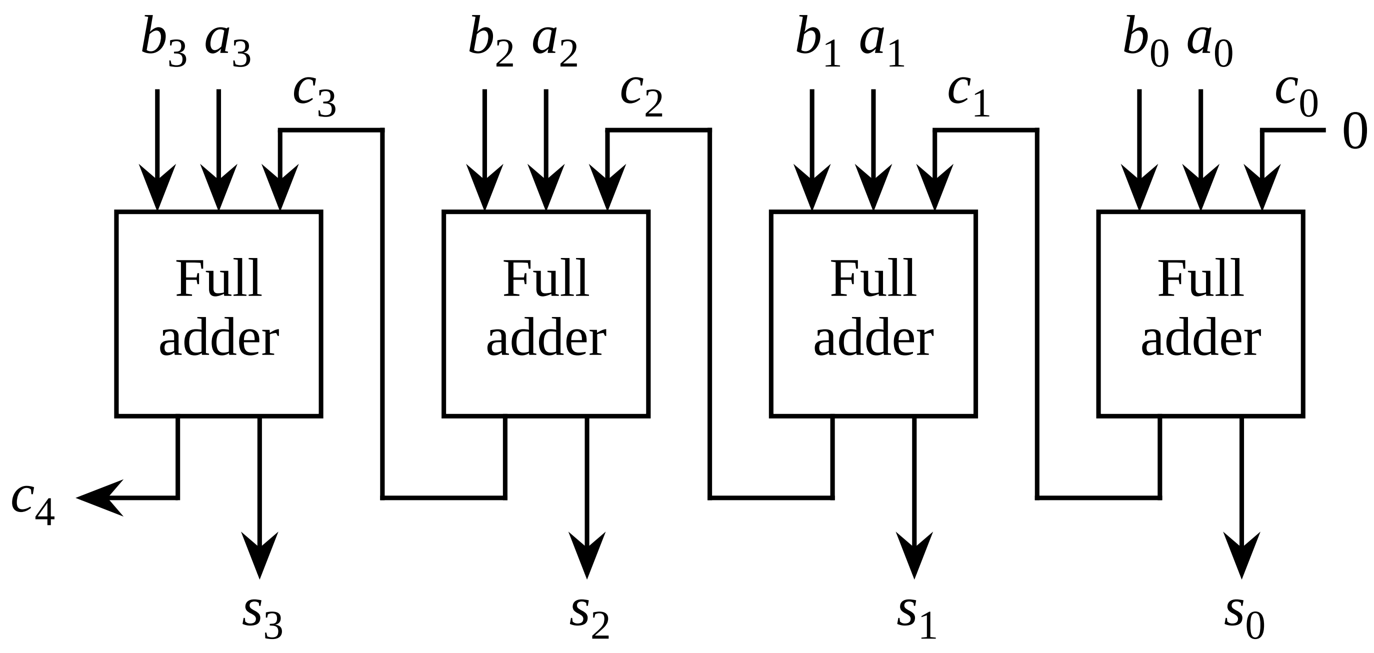
- Inputs: A, B and c_{in}
- Outputs: $S =$ lower bit of $A + B$, $c_{\text{out}} =$ carry bit

A	B	c_{in}	S	c_{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

- $S = \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + A\overline{B}\,\overline{C} + ABC = A \oplus B \oplus C$.
- $c_{\text{out}} = \mathrm{MAJ3} = AB + BC + AC$.

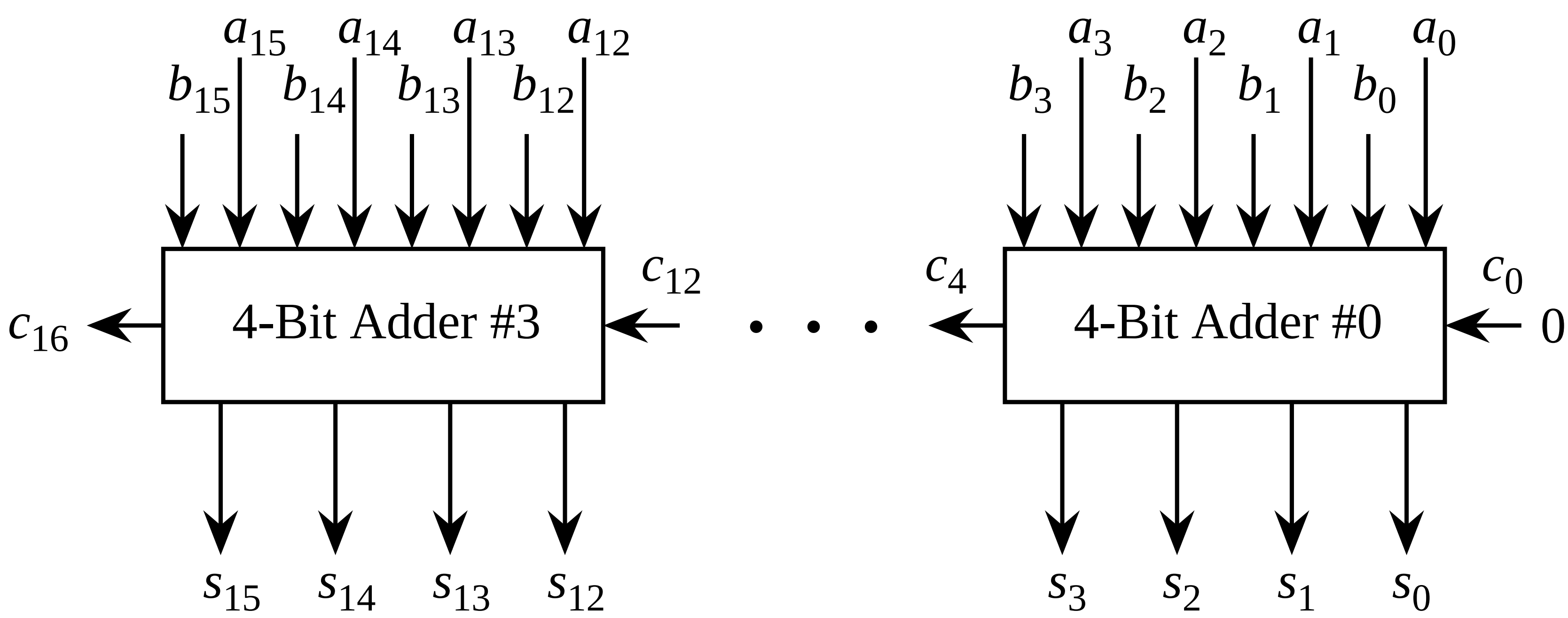
Ripple Carry Adder

- Two binary numbers *A* and *B* are added from right to left, creating a sum and a carry at the outputs of each full adder for each bit position.



Constructing Larger Adders

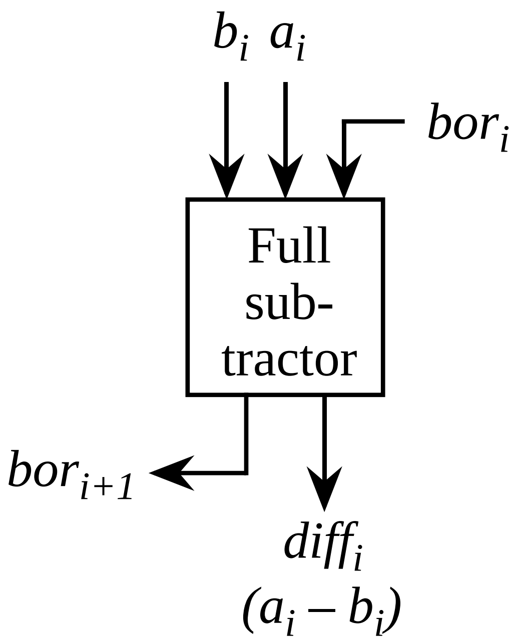
- A 16-bit adder can be made up of a cascade of four 4-bit ripple-carry adders.



Full Subtractor

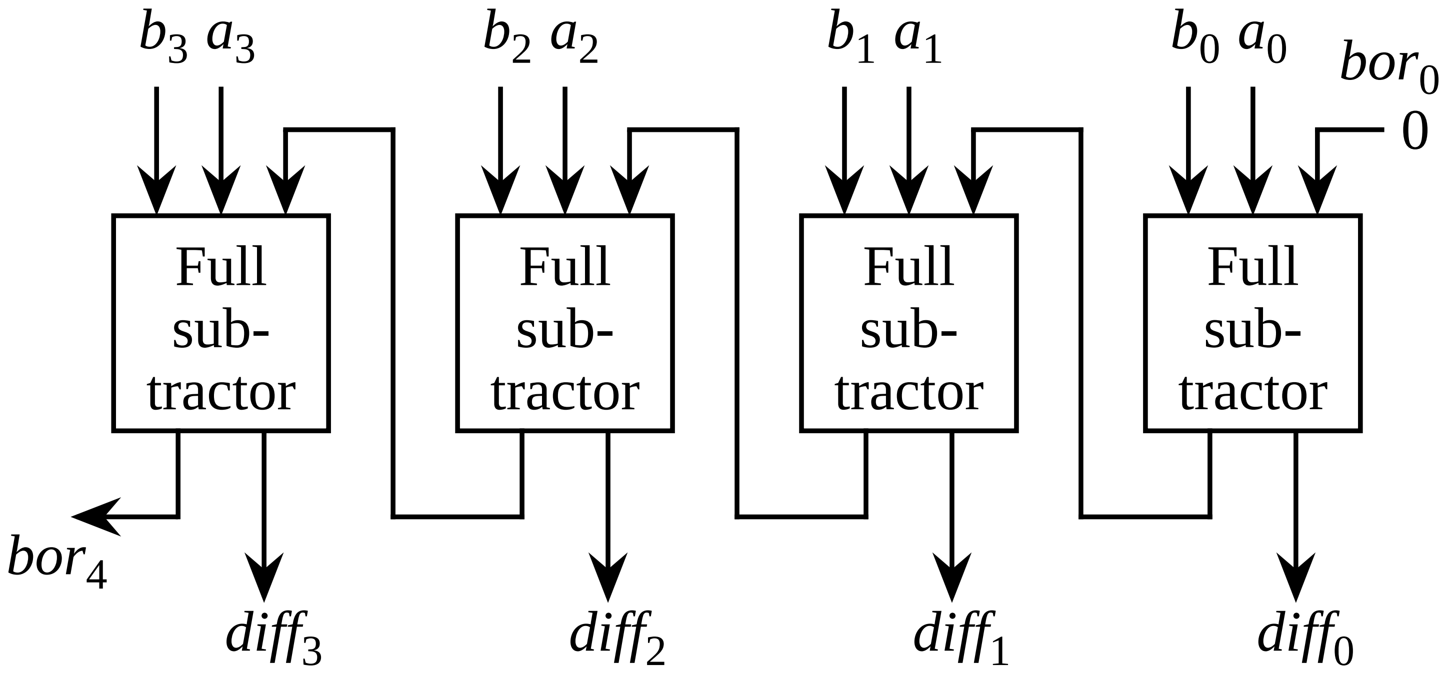
- **Truth table and schematic symbol for a ripple-borrow subtractor:**

a_i	b_i	bor_i	$diff_i$	bor_{i+1}
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1



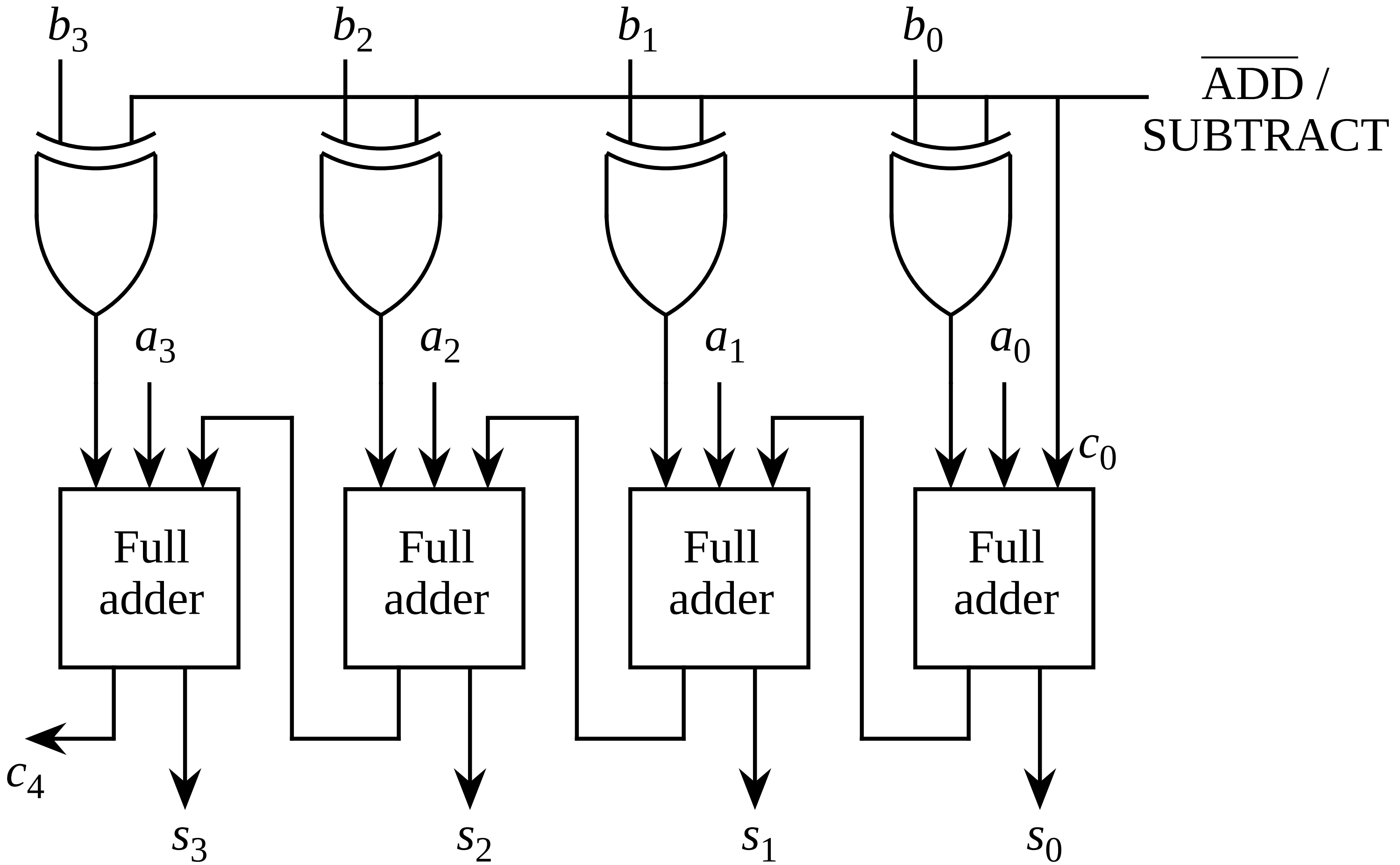
Ripple-Borrow Subtractor

- **A ripple-borrow subtractor can be composed of a cascade of full subtractors.**
- **Two binary numbers *A* and *B* are subtracted from right to left, creating a difference and a borrow at the outputs of each full subtractor for each bit position.**



Combined Adder/Subtractor

- A single ripple-carry adder can perform both addition and subtraction, by forming the two's complement negative for *B* when subtracting. (Note that +1 is added at c_0 for two's complement.)



Carry-Lookahead Addition

$$s_i = \bar{a}_i\bar{b}_i c_i + \bar{a}_i b_i \bar{c}_i + a_i \bar{b}_i \bar{c}_i + a_i b_i c_i$$

$$c_{i+1} = b_i c_i + a_i c_i + a_i b_i$$

$$c_{i+1} = a_i b_i + (a_i + b_i) c_i$$

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

- Carries are represented in terms of G_i (generate) and P_i (propagate) expressions.

$$G_i = a_i b_i \quad \text{and} \quad P_i = a_i + b_i$$

$$c_0 = 0$$

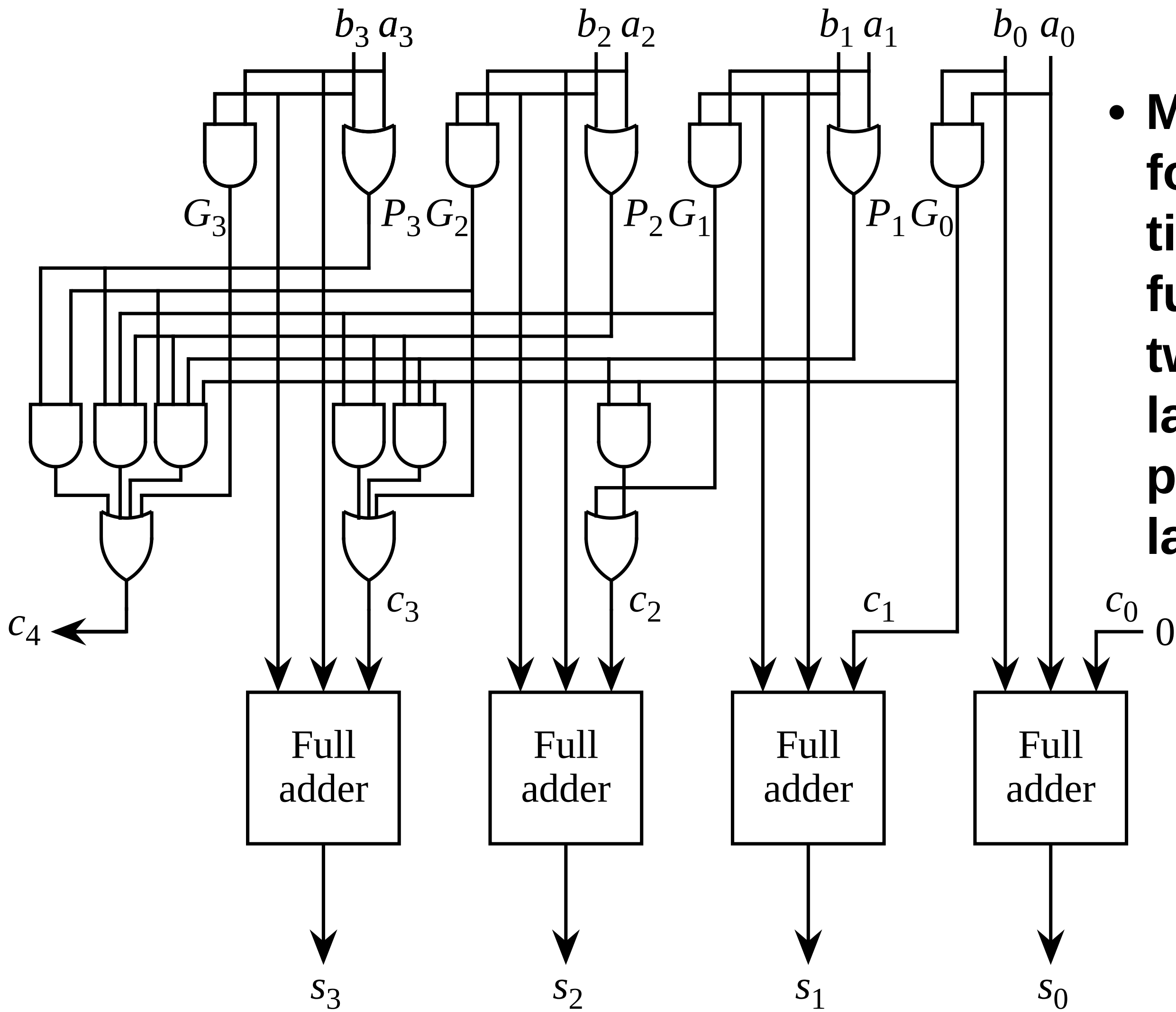
$$c_1 = G_0$$

$$c_2 = G_1 + P_1 G_0$$

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

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

Carry Lookahead Adder



- **Maximum gate delay for the carry generation is only 3. The full adders introduce two more gate delays. Worst case path is 5 gate delays.**

Next Time

- In-class Labs
- Homework 3 due
- Homework 4 assigned