Lecture 11: Multi-Level Logic

CSE 370, Autumn 2007
Benjamin Ylvisaker

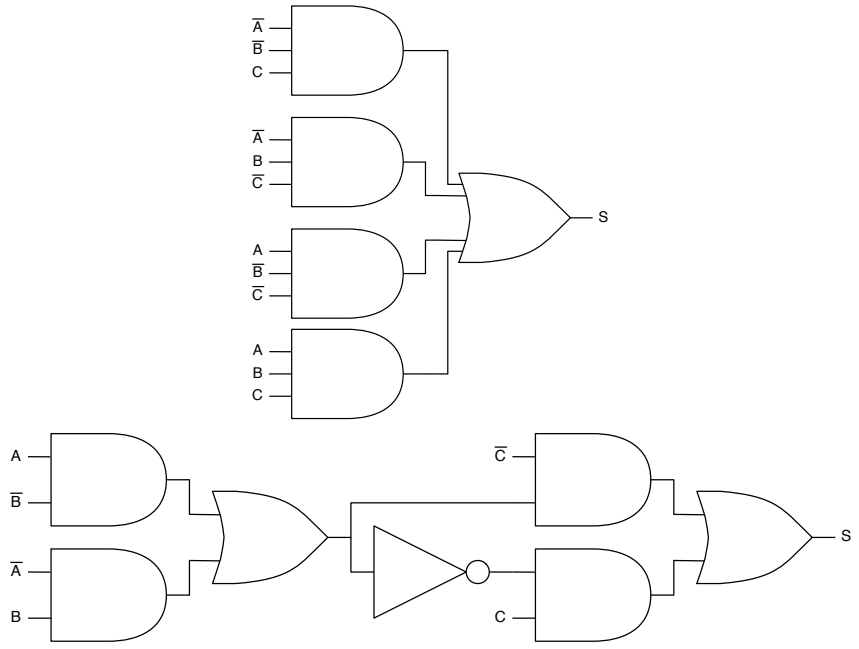
Where We Are

- Last lecture: Quine-McCluskey Minimization
- This lecture: Multi-Level Logic
- Next lecture: Circuit Delay and Timing
- Homework 4 in progress
- Lab 3 done; lab 4 next week

2-Level Minimum Circuits are Not Always the Best Solution

- Important circuit metrics:
  - Size
  - Speed
  - Complexity
  - Energy efficiency
- How we approximate these metrics:
  - Number and kind of gates
  - Number of gate inputs
  - Circuit depth

Example: Full Adder Sum Output



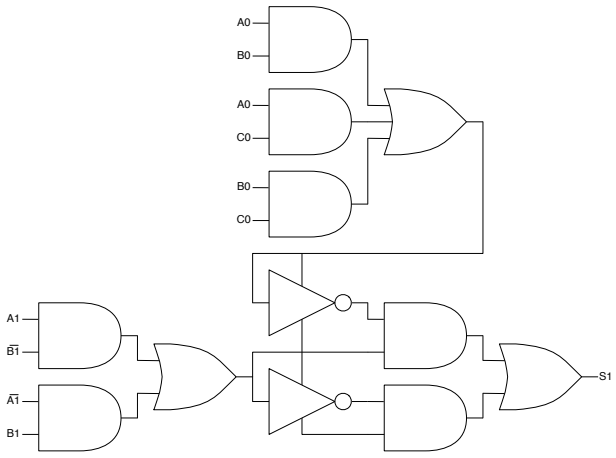
More Extreme Example: 2-Bit Adder

A1=0

$B_1 A_0$ / $B_0 C$	00	01	11	10
00	0	0	1	0
01	0	1	1	1
11	1	1	0	1
10	0	1	1	1

A1=1

$B_1 A_0$ / $B_0 C$	00	01	11	10
00	0	1	1	1
01	1	1	0	1
11	1	0	0	0
10	1	1	0	1



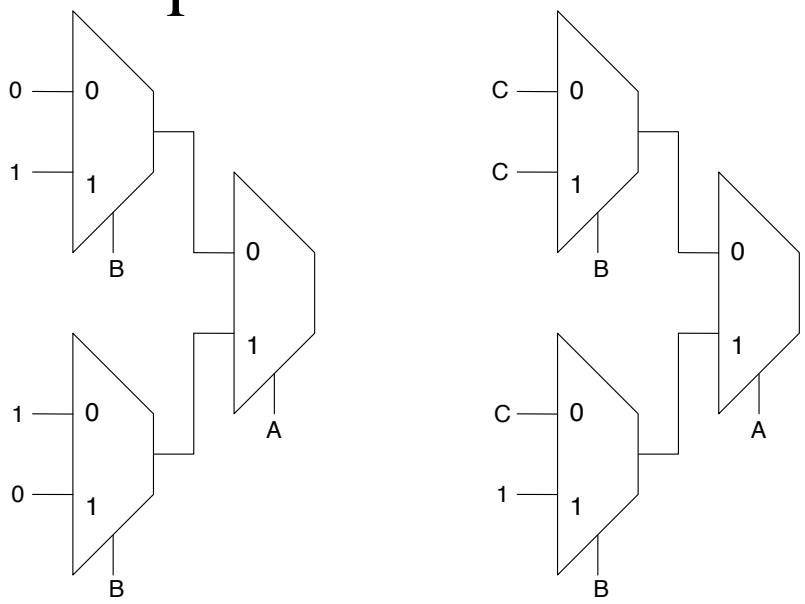
No Simple Methods

- For 2-level minimization we have:
  - K-maps
  - Quine-McCluskey
  - Espresso
- For multi-level minimization we have:
  - Lots of heuristics
  - SIS

Factoring

- Z = ADF + AEF + BDF + BEF + CDF + CEF + G
  - AND3: 6 OR7: 1 Depth: 2
- Z = (AD + AE + BD + BE + CD + CE)F + G
  - AND2: 7 OR6: 1 OR2: 1 Depth: 4
- Z = (AD + BD + CD + AE + BE + CE)F + G
  - AND2: 7 OR6: 1 OR2: 1 Depth: 4
- Z = [(A + B + C)D + (A + B + C)E]F + G
  - OR3: 2 AND2: 3 OR2: 2 Depth: 5
- Z = (A + B + C)(D + E)F + G
  - OR3: 1 OR2: 2 AND3: 1 Depth: 3

Using Multiplexors to Implement Functions

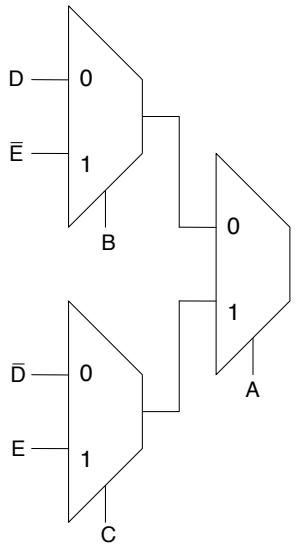


Cofactoring

- Z = ACE + A¬C¬D + ¬AB¬E + ¬A¬BD
  - Cofactor A
- Z = A(CE + ¬C¬D) + ¬A(B¬E + ¬BD)
  - Cofactor C in the left expression and B in the right expression
- Z = A(C(E) + ¬C(¬D)) + ¬A(B(¬E) + ¬B(D))

Translating to Muxes

- A(C(E) + ¬C(¬D)) + ¬A(B(¬E) + ¬B(D))



Thank You for Your Attention

- Start reading lab 4
- Start looking at homework 4
- Continue reading the book