Today

- Design Works Demo
- Sum of Products
- DeMorgan's Law
- Boolean simplification
- Reverse Engineering Example

Sum-of-Product & Product-of-Sum

- Sum of Product
  - see entries with value 1.
  - Ensure output 1 for those inputs. So **OR**
  - A'B+AB'
- Product of Sum
  - see entries with value 0
  - Ensure output 0 for those input. So **AND**
  - (A+B)(A'+B')

A	B	A **XOR** B
0	0	0
0	1	1
1	0	1
1	1	0

DeMorgan's Law

$$\overline{X+Y} = \overline{X} \cdot \overline{Y}$$

- Thus, the inversion doesn't distribute directly: it also changes the "or" to an "and"
- Dual:

$$\overline{X \cdot Y} = \overline{X} + \overline{Y}$$

DeMorgan's Examples

- Simplify: (A′ + B + C)′

  (A′ + B + C)′

  = AB′C′
- Simplify: ((A′B)′ + C)′

  (A(B′C + D))′

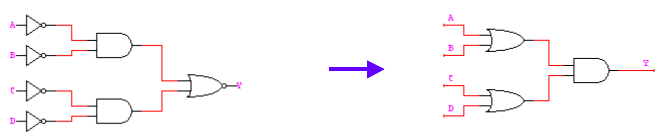
  = A′ + (B′C + D)′

  = A′ + (B′C)′D′

  = A′ + (B + C′)D′

DeMorgan's law on circuits

- You can do DeMorgan's law directly on the circuit:



Simplification

- Some important rules for simplification (how do you prove these?):
  - AB + AB′ = A
  - A + AB = A
- Note that you can use the rules in either direction, to remove terms, or to add terms. Indeed, sometimes you need to add some terms in order to get to the simples solution.

Examples

- Simplify: ab'c + abc + a'bc

  ab'c + abc + a'bc

  = ab'c + abc + abc + a'bc = ac + bc

- Show that X + X'Y = X + Y

  X + X'Y

  = X(1 + Y) + X'Y

  = X + XY + X'Y

  = X + Y

Examples (cont'd)

- Simplify:WX + XY + X'Z' + WY'Z'

  WX + XY + X'Z' + WY'Z'

  = WX + XY + X'Z' + WY'Z'X + WY'Z'X'

  = WX(1 + Y'Z') + XY + X'Z'(1 + WY')

  = WX + XY + X'Z'

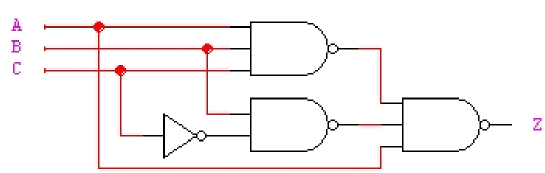
Long example

- Simplify:

A'B'C'D' + A'BC'D' + A'BD + A'BC'D + ABCD + ACD' + B'CD'
= A'C'D'(B' + B) + A'BD(1 + C') + ABCD + ACD' + B'CD'
= A'C'D' + A'BD + ABCD + ACD' + B'CD'
= A'C'D' + BD(A' + AC) + ACD' + B'CD'
= A'C'D' + BD(A' + C) + ACD' + B'CD' (Since X + X'Y = X + Y)
= A'C'D' + A'BD + (BCD + ACD') + B'CD'
= A'C'D' + A'BD + (BCD + ACD' + ABC) + B'CD'
(Added ABC by consensus)
= A'C'D' + (A'BD + ABC + BCD) + (ABC + B'CD' + ACD')
= AC'D' + A'BD + ABC + B'CD'

Reverse Engineering Example

- Write down the Boolean expression:



$$f(A,B,C) = \overline{\overline{ABC} \bullet \overline{B\overline{C}} \bullet A}$$

Reverse Engineering Ex (cont.)

- Simplify the function using Boolean algebra:

$f(A,B,C) = \overline{\overline{ABC} \bullet \overline{B\overline{C}} \bullet A}$	
$= ABC + B\overline{C} + \overline{A}$	DeMorgan's/Involution
$= B(AC + \overline{C}) + \overline{A}$	Distributive
$= B(A + \overline{C}) + \overline{A}$	Simplification
$= AB + B\overline{C} + \overline{A}$	Distributive
$= \overline{A} + B + B\overline{C}$	Simplification
$= \overline{A} + B$	Simplification

Reverse Engineering Ex (cont.)

- Write the complete truth table for the circuit.

A	B	C	Z
0	0	0	1
	0	1	1
	1	0	1
	1	1	1
1	0	0	0
	0	1	0
	1	0	1
	1	1	1

Reverse Engineering Ex (cont.)

- Write the canonical SOP form.

$$f(A,B,C) = \overline{A}\,\overline{B}\,\overline{C} + \overline{A}\,\overline{B}C + \overline{A}B\overline{C} + \overline{A}BC + AB\overline{C} + ABC$$

Reverse Engineering Ex (cont.)

- Re-implement the better design.