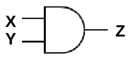
Lecture 4

- Logic gates and truth tables
- Implementing logic functions
- Canonical forms
  - Sum-of-products
  - Product-of-sums

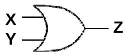
Logic gates and truth tables

- AND X•Y XY



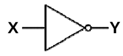
X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

- OR X + Y



X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

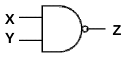
- NOT X' $\overline{X}$



X	Y
0	1
1	0

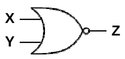
Logic gates and truth tables

- NAND $\overline{X \bullet Y}$ $\overline{XY}$



X	Y	Z
0	0	1
0	1	1
1	0	1
1	1	0

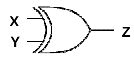
- NOR $\overline{X + Y}$



X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	0

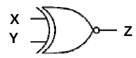
Logic gates and truth tables

- XOR $X \oplus Y$



X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

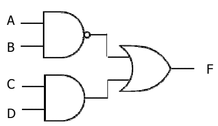
- XNOR $\overline{X \oplus Y}$



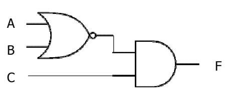
X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	1

Realizing Boolean formulas

- F = (A•B)' + C•D



- F = C•(A+B)'



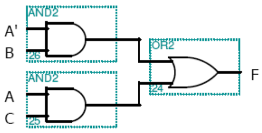
Realizing truth tables

- Given a truth table
  1. Write the Boolean expression
  2. Minimize the Boolean expression
  3. Draw as gates

Example

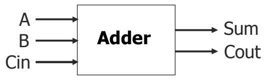
A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

F = A′BC′+A′BC+AB′C+ABC
  = A′B(C′+C)+AC(B′+B)
  = A′B+AC



Example: Binary full adder

- 1-bit binary adder
  - Inputs: A, B, Carry-in
  - Outputs: Sum, Carry-out



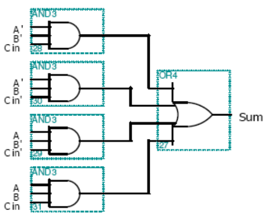
A	B	Cin	Cout	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Sum = A'B'Cin + A'BCin' + AB'Cin' + ABCin
Cout = A'BCin + AB'Cin + ABCin' + ABCin

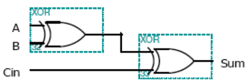
Both Sum and Cout can be minimized.

Full adder: Sum

Before Boolean minimization
Sum = A'B'Cin + A'BCin'
  + AB'Cin' + ABCin

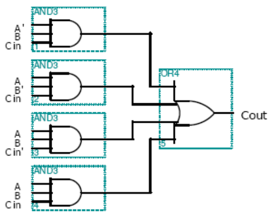


After Boolean minimization
Sum = (A⊕B) ⊕ Cin

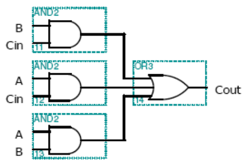


Full adder: Carry-out

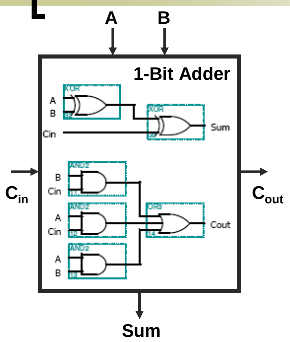
Before Boolean minimization
Cout = A'BCin + AB'Cin
  + ABCin' + ABCin

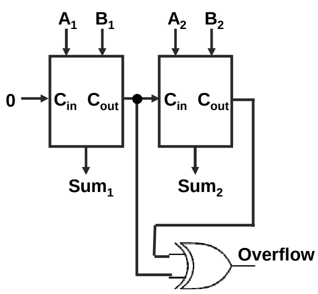


After Boolean minimization
Cout = BCin + ACin + AB



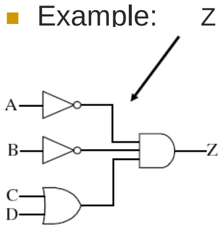
Preview: 2-bit ripple-carry adder

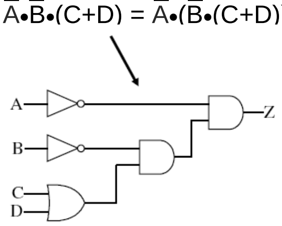




Many possible mappings

- Many ways to map expressions to gates
- Example: Z = $\overline{A}$•$\overline{B}$•(C+D) = $\overline{A}$•($\overline{B}$•(C+D))





What is the optimal realization?

- We use the axioms and theorems of Boolean algebra to “optimize” our designs
- Design goals vary
  - Reduce the number of gates?
  - Reduce the number of gate inputs?
  - Reduce the number of cascaded levels of gates?

What is the optimal realization?

- How do we explore the tradeoffs?
  - Logic minimization: Reduce number of gates and complexity
  - Logic optimization: Maximize speed and/or minimize power
  - CAD tools

Canonical forms

- Canonical forms
  - Standard forms for Boolean expressions
  - Derived from truth table
  - Generally not the simplest forms (can be minimized)
- Two canonical forms
  - Sum-of-products (minterms)
  - Product-of-sums (maxterms)

Sum-of-products (SOP)

- Also called *disjunctive normal form* (DNF) or *minterm expansion*

A	B	C	F	F'
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

001 011 101 110 111

F = A'B'C + A'BC + AB'C + ABC' + ABC

minterm

F' = A'B'C' + A'BC' + AB'C'

Minterms

- Variables appear exactly once in each minterm in true or inverted form (but not both)

A	B	C	minterms	
0	0	0	A'B'C'	m0
0	0	1	A'B'C	m1
0	1	0	A'BC'	m2
0	1	1	A'BC	m3
1	0	0	AB'C'	m4
1	0	1	AB'C	m5
1	1	0	ABC'	m6
1	1	1	ABC	m7

short-hand notation

F in canonical form:
F(A,B,C) = Σm(1,3,5,6,7)
= m1 + m3 + m5 + m6 + m7
= A'B'C+A'BC+AB'C+ABC'+ABC

Product-of-sums (POS)

- Also called *conjunctive normal form* (CNF) or *maxterm expansion*

A	B	C	F	F'
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

000 010 100

F = (A + B + C) (A + B' + C) (A' + B + C)

maxterm

F' = (A+B+C')(A+B'+C')(A'+B+C')(A'+B'+C)(A'+B'+C')

Maxterms

- Variables appear exactly once in each maxterm in true or inverted form (but not both)

A	B	C	maxterms	
0	0	0	A+B+C	M0
0	0	1	A+B+C'	M1
0	1	0	A+B'+C	M2
0	1	1	A+B'+C'	M3
1	0	0	A'+B+C	M4
1	0	1	A'+B+C'	M5
1	1	0	A'+B'+C	M6
1	1	1	A'+B'+C'	M7

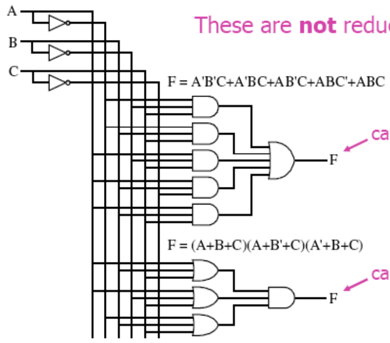
short-hand notation

F in canonical form:

F(A,B,C) = ∏M(0,2,4)
= M0 • M2 • M4
= (A+B+C)(A+B'+C)(A'+B+C)

Example: F = AB+C

These are **not** reduced forms for F



canonical sum-of-products

canonical product-of-sums

From SOP to POS and back

- Minterm to maxterm
  - Use maxterms that aren't in minterm expansion
  - F(A,B,C) = ∑m(1,3,5,6,7) = ∏M(0,2,4)
- Maxterm to minterm
  - Use minterms that aren't in maxterm expansion
  - F(A,B,C) = ∏M(0,2,4) = ∑m(1,3,5,6,7)

From SOP to POS and back

- Minterm of F to minterm of F'
  - Use minterms that don't appear
  - F(A,B,C) = ∑m(1,3,5,6,7)   F' = ∑m(0,2,4)
- Maxterm of F to maxterm of F'
  - Use maxterms that don't appear
  - F(A,B,C) = ∏M(0,2,4)   F' = ∏M(1,3,5,6,7)

SOP, POS, and DeMorgan's

- Sum-of-products
  - F' = A'B'C' + A'BC' + AB'C'
- Apply DeMorgan's to get POS
  - (F')' = (A'B'C' + A'BC' + AB'C')'
  - F = (A+B+C)(A+B'+C)(A'+B+C)

SOP, POS, and DeMorgan's

- Product-of-sums
  - F' = (A+B+C')(A+B'+C')(A'+B+C')(A'+B'+C')
- Apply DeMorgan's to get SOP
  - (F')' = ((A+B+C')(A+B'+C')(A'+B+C')(A'+B'+C'))'
  - F = A'B'C + A'BC + AB'C + ABC