Lecture 3: Boolean Algebra

- Logistics
- Last lecture --- Numbers
  - Binary numbers
  - Base conversion
  - Number systems for negative numbers
  - A/D and D/A conversion
- Today's lecture
  - Boolean algebra
    - Axioms
    - Useful laws and theorems
    - Examples

The "WHY" slide

- Boolean Algebra
  - When we learned numbers like 1, 2, 3, we also then learned how to add, multiply, etc. with them. Boolean Algebra covers operations that we can do with 0's and 1's. Computers do these operations ALL THE TIME and they are basic building blocks of computation inside your computer program.
- Axioms, laws, theorems
  - We need to know some rules about how those 0's and 1's can be operated on together. There are similar axioms to decimal number algebra, and there are some laws and theorems that are good for you to use to simplify your operation.

How does Boolean Algebra fit into the big picture?

- It is part of the Combinational Logic topics (memoryless)
  - Different from the Sequential logic topics (can store information)
- Learning Axioms and theorems of Boolean algebra
    - Allows you to do design logic functions
    - Allows you to know how to combine different logic gates
    - Allows you to simplify or optimize on the complex operations

Boolean algebra

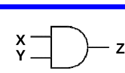
- A Boolean algebra comprises...
  - A set of elements B
  - Binary operators {+ , •} Boolean sum and product
  - A unary operation { ' } (or { ¯ }) example: A' or $\overline{A}$
- ...and the following axioms
  - 1. The set B contains at least two elements {a b} with a ≠ b
  - 2. Closure: a+b is in B a•b is in B
  - 3. Commutative: a+b = b+a a•b = b•a
  - 4. Associative: a+(b+c) = (a+b)+c a•(b•c) = (a•b)•c
  - 5. Identity: a+0 = a a•1 = a
  - 6. Distributive: a+(b•c)=(a+b)•(a+c) a•(b+c)=(a•b)+(a•c)
  - 7. Complementarity: a+a' = 1 a•a' = 0

Digital (binary) logic is a Boolean algebra

- Substitute
  - {0, 1} for B
  - AND for • Boolean Product. In CSE 321 this was ∧
  - OR for + Boolean Sum. In CSE 321 this was ∨
  - NOT for ' Negation. In CSE 321 this was ¬
- All the axioms hold for binary logic
- Definitions
  - Boolean function
    - Maps inputs from the set {0,1} to the set {0,1}
  - Boolean expression
    - An algebraic statement of Boolean variables and operators

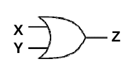
Logic Gates (AND, OR, Not) & Truth Table

- AND X•Y XY



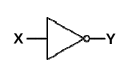
X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

- OR X+Y



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

- NOT $\overline{X}$ X'



X	Y
0	1
1	0

Logic functions and Boolean algebra

- *Any* logic function that is expressible as a truth table can be written in Boolean algebra using +, •, and '

Z=X•Y

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

Z=X'•Y

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

Z=(X•Y)+(X' •Y')

X	Y	X'	Y'	X • Y	X' •Y'	Z
0	0	1	1	0	1	1
0	1	1	0	0	0	0
1	0	0	1	0	0	0
1	1	0	0	1	0	1

Some notation

- Priorities: $\overline{A} \bullet B + C = ((\overline{A}) \bullet B) + C$
- Variables are sometimes called literals

Two key concepts

- Duality (a meta-theorem— *a theorem about theorems*)
  - All Boolean expressions have logical duals
  - Any theorem that can be proved is also proved for its dual
  - Replace: • with +, + with •, 0 with 1, and 1 with 0
  - Leave the variables unchanged
- de Morgan's Theorem
  - Procedure for complementing Boolean functions
  - Replace: • with +, + with •, 0 with 1, and 1 with 0
  - Replace all variables with their complements

Useful laws and theorems

Identity:	X + 0 = X	Dual: X • 1 = X
Null:	X + 1 = 1	Dual: X • 0 = 0
Idempotent:	X + X = X	Dual: X • X = X
Involution:	(X')' = X	
Complementarity:	X + X' = 1	Dual: X • X' = 0
Commutative:	X + Y = Y + X	Dual: X • Y = Y • X
Associative:	(X+Y)+Z=X+(Y+Z)	Dual: (X•Y)•Z=X•(Y•Z)
Distributive:	X•(Y+Z)=(X•Y)+(X•Z)	Dual: X+(Y•Z)=(X+Y)•(X+Z)
Uniting:	X•Y+X•Y'=X	Dual: (X+Y)•(X+Y')=X

Useful laws and theorems (con't)

Absorption:	X+X•Y=X	Dual: X•(X+Y)=X
Absorption (#2):	(X+Y')•Y=X•Y	Dual: (X•Y')+Y=X+Y
de Morgan's:	(X+Y+...)'=X'•Y'•...	Dual: (X•Y•...)'=X'+Y'+...
Duality:	$(X+Y+...)^D$=X•Y•...	Dual: $(X•Y•...)^D$=X+Y+...

Multiplying & factoring: (X+Y)•(X'+Z)=X•Z+X'•Y
Dual: X•Y+X'•Z=(X+Z)•(X'+Y)

Consensus: (X•Y)+(Y•Z)+(X'•Z)= X•Y+X'•Z
Dual: (X+Y)•(Y+Z)•(X'+Z)=(X+Y)•(X'+Z)

Proving theorems

- Example 1: Prove the uniting theorem-- X•Y+X•Y'=X

Distributive	X•Y+X•Y' = X•(Y+Y')
Complementarity	= X•(1)
Identity	= X

- Example 2: Prove the absorption theorem-- X+X•Y=X

Identity	X+X•Y = (X•1)+(X•Y)
Distributive	= X•(1+Y)
Null	= X•(1)
Identity	= X

Proving theorems

- Example 3: Prove the consensus theorem--
  (XY)+(YZ)+(X'Z)= XY+X'Z

Complementarity	XY+YZ+X'Z = XY+(X+X')YZ + X'Z
Distributive	= XYZ+XY+X'YZ+X'Z

➪ *Use absorption {AB+A=A} with A=XY and B=Z*

	= XY+X'YZ+X'Z
Rearrange terms	= XY+X'ZY+X'Z

➪ *Use absorption {AB+A=A} with A=X'Z and B=Y*

XY+YZ+X'Z = XY+X'Z

de Morgan's Theorem

- Use de Morgan's Theorem to find complements
- Example: F=(A+B)•(A'+C), so F'=(A'•B')+(A•C')

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

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

One more example of logic simplification

- Example:

Z = A'BC + AB'C' + AB'C + ABC' + ABC

= A'BC + AB'(C' + C) + AB(C' + C)	distributive
= A'BC + AB' + AB	complementary
= A'BC + A(B' + B)	distributive
= A'BC + A	complementary
= BC + A	absorption #2 Duality

(X•Y')+Y=X+Y with X=BC and Y=A