Boolean Algebra

Boolean Algebra

- A **Boolean algebra**
  - A set of operators (e.g. the binary operators: +, •, INV)
  - A set of axioms or postulates

Postulates

> Commutative	$x+y = y+x$
	$x \bullet y = y \bullet x$
> Distributive	$x+(y \bullet z)=(x+y) \bullet (x+z)$
	$x \bullet (y+z)=x \bullet y+x \bullet z$
> Identities	$x+0=x$
	$x \bullet 1=x$
> Complement	$x+x'=1$ $x \bullet x'=0$
Closure	$x+y$ $x \bullet y$
Associative	$(x+y)+z=x+(y+z)$
	$(x \bullet y) \bullet z=x \bullet (y \bullet z)$

Properties of Boolean Algebra

- > Complement of a variable is unique.
- > $(x')' = x$ -- involution
- > $x+x = x$ $\quad x \cdot x = x$ --idempotent
- > $x+1=1$ $\quad x \cdot 0 = 0$
- > $x+x \cdot y = x$ $\quad x \cdot (x+y) = x$ -- absorption
- > $(x+y)' = x' \cdot y'$ $\quad (x \cdot y)' = x' + y'$ -- DeMorgan's Law
- > $xy+x'z+yz = xy + x'z$
  $(x+y) \cdot (x'+z) \cdot (y+z) = (x+y) \cdot (x'+z)$ -- consensus

Duality: $+ \leftrightarrow \cdot$ and $0 \leftrightarrow 1$

Proof of Consensus

- a <= b essentially implies that if a =1 then b =1 and if a = 0 then b could be anything

Theorem: In Consensus xy + x'z + yz = xy + x'z (How?)

We Prove that xy + x'z + yz >= xy + x'z and xy + x'z + yz <= xy + x'z (This would imply equality and prove the theorem)

Proof: First: xy + x'z + yz >= xy + x'z. Let xy + x'z = a and yz = b;

So we want to prove that a + b >= a which is true by definition of the inequality

Second: xy + x'z + yz <= xy + x'z. This inequality could be split into xy + x'z <= xy + x'z and yz <= xy + x'z

All we need to do is to prove yz <= xy + x'z

Proof of Consensus

yz <= xy + x'z

We know that a <= b iff ab' = 0 (How?)

So if the above inequality is true yz(xy + x'z)' = 0 must be true. Which is always true.

Hence Proved.

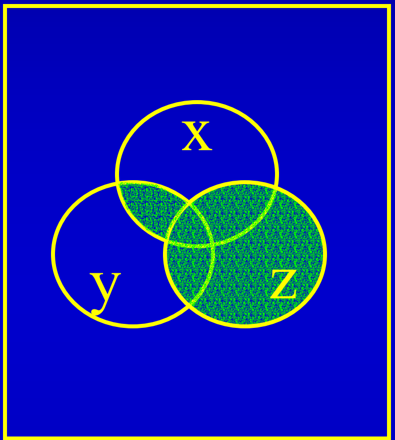
Boolean Functions

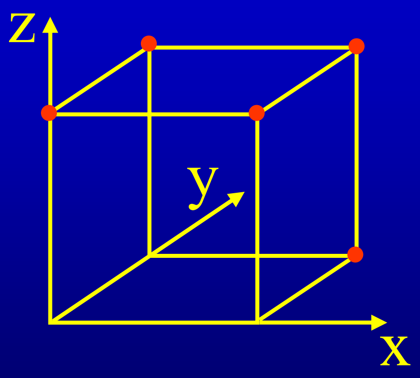
- A **Boolean function** is a mapping $f(x): B^n \rightarrow B$.
  - Constant function: $f(x_1,\ldots,x_n) = b.$
  - Projection (to the *i*-th axis): $f(x_1,\ldots,x_n) = x_i.$
- A Boolean function is **complete** if $f(x)$ is defined for all $x \in B^n$. Otherwise the point x that $f(x)$ is not defined is called a **don't care** condition.
- Operations on Boolean functions:
  - Sum: $(f+g)(x) = f(x) + g(x)$
  - Product: $(f \bullet g)(x) = f(x) \bullet g(x)$
  - Complement: $(f')(x) = (f(x))'$

Representations of Boolean Functions

- Algebraic expressions
  - $f(x,y,z) = xy+z$
- Tabular forms
- Venn diagrams
- Cubical representations
- Binary decision diagrams (BDD)

x	y	z	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

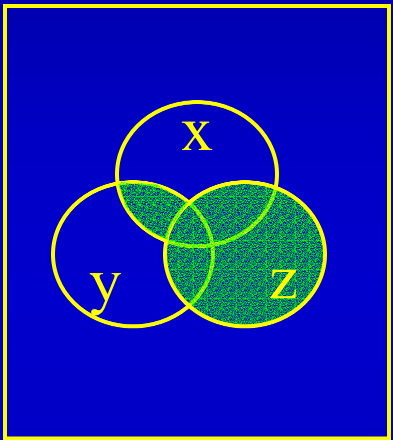


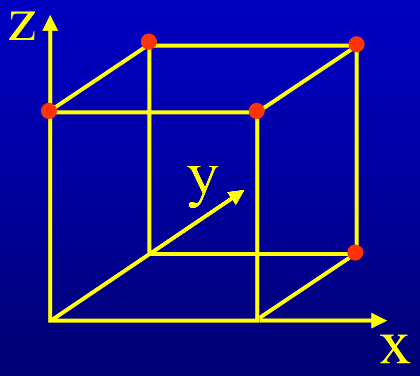


Representations of Boolean Functions

- Algebraic expressions
  - $f(x,y,z) = xy+z$
- Tabular forms
- Venn diagrams
- Cubical representations
- Binary decision diagrams (BDD)

x	y	z	f
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1





Boole's Expansion Theorem

- The cofactor of $f(x_1,\ldots,x_n)$ w.r.t. x_i (or x'_i) is a Boolean function $f_{x_i}\ (or\ \ f_{x'_i}) : B^{n-1} \rightarrow B$, s.t.

$$f_{x_i / x'_i}(x_1,\cdots,x_{i-1},x_{i+1},\cdots,x_n) = f(x_1,\cdots,x_{i-1},1/0,x_{i+1},\cdots,x_n)$$

- Boole's Expansion Theorem:

$$f(x_1,\cdots,x_{i-1},x_i,x_{i+1},\cdots,x_n) = x_i \bullet f_{x_i} + x'_i \bullet f_{x'_i} = (x_i + f_{x'_i}) \bullet (x'_i + f_{x_i})$$

Boole's Expansion: Examples

- $f(x) = x \cdot f(1) + x' \cdot f(0)$ for single-variable x

  $= (x + f(0)) \cdot (x' + f(1))$ (from duality)

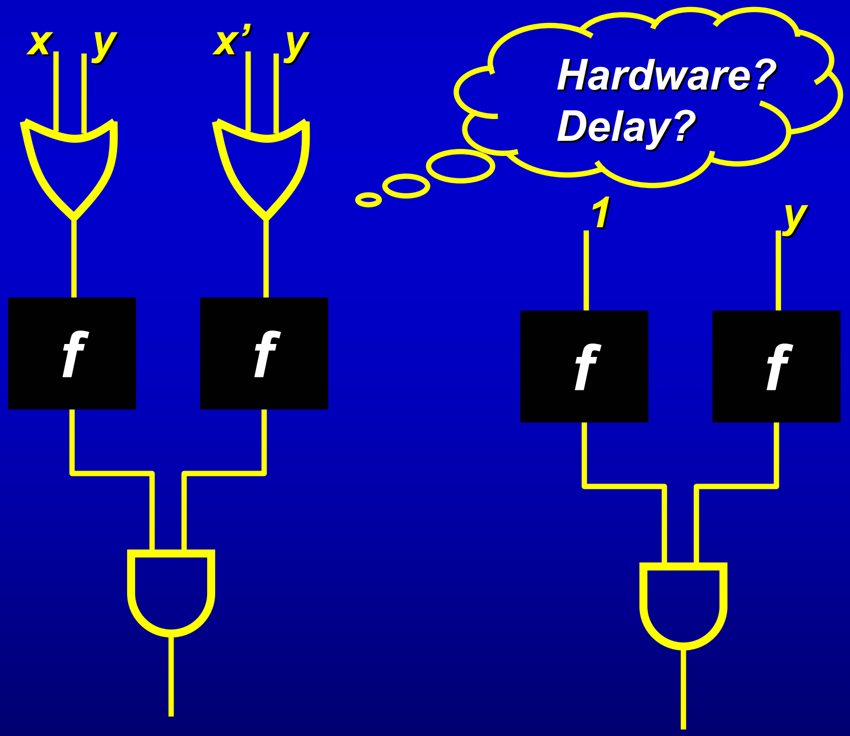
- $f(x+y) \cdot f(x'+y) = f(1) \cdot f(y)$
  - Define $g(x,y) = f(x+y)f(x'+y)$

    $$g(x,y) = xg(1,y) + x'g(0,y)$$
    $$= xf(1)f(y) + x'f(y)f(1)$$
    $$= f(1)f(y)$$

  - What if expand w.r.t variable y?

    $$f(x+y) \cdot f(x'+y) = yf(1)f(1) + y'f(x)f(x')$$
    $$= yf(1) + y'f(1)f(0) = f(1)(yf(1) + y'f(0))$$
    $$= f(1)f(y)$$

Boole's Expansion: Examples



Complete Expansion

minterms

$$f(x_1, \cdots, x_{n-1}, x_n) = f(0, \cdots, 0, 0)\, x_1' \cdots x_{n-1}' x_n'$$

$$+ f(0, \cdots, 0, 1)\, x_1' \cdots x_{n-1}' x_n + \cdots$$

$$+ f(1, \cdots, 1, 1)\, x_1 \cdots x_{n-1} x_n$$

discriminants

Canonical Forms

- A form is called canonical if the representation of the function in that form is unique.
- Minterm Canonical Form
  - AND-OR circuits
- Pro and Cons of Canonical Forms
  - Unique up to permutation
  - Inefficient

Normal (Standard) Forms

> SOP (Disjunctive Normal Form)

= A disjunction of product terms

= A product term

= 0

> The Primary Objective During Logic Minimization is to Remove the Redundancy in the Representation.

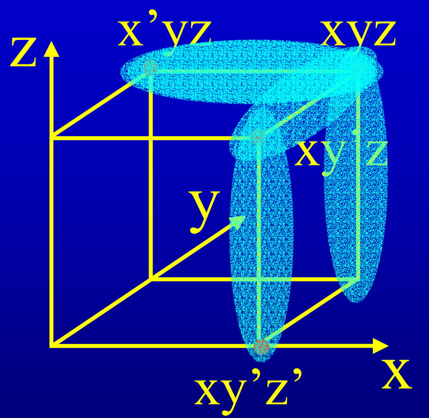
Implicants

- An implicant of a function is a product term that is included in the function.
- An implicant is prime if it cannot be included in any other implicants.
- A prime implicant is essential if it is the only one that includes a minterm.

Example: *f(x,y,z) = xy' + yz*

xy(not I),xyz(I, not PI),

xz(PI,not EPI), yz(EPI)



Specification for Incompleteness

- A Boolean function is incomplete if *f(x)* is NOT defined for some $x \in B^n$. Such point *x* is called a don't care condition.
- Tabular representation
- {1-set, 0-set, don't care set}
  - 1-set = {xy}
  - 0-set = {x'y'}
  - Don't care set = {x'y, xy'}

x	y	f
0	0	0
0	1	-
1	0	-
1	1	1

Don't Care Conditions

- Satisfiability don't cares of a subcircuit consist of all input patterns that will never occur.
- Observability don't cares of a subcircuit are the input patterns that represent situations when an output is not observed.

Example: