Lecture 8: Finite Boolean Algebra

Xiaoxing Ma

Nanjing University

xxm@nju.edu.cn

November 20, 2017

Acknowledgement: These Beamer slides are totally based on the textbook *Discrete Mathematical Structures*, by B. Kolman, R. C. Busby and S. C. Ross, and Prof. Daoxu Chen's PowerPoint slides.

At the Last Class

1. Partial Order
   - Order relations and Hasse Diagrams
   - Extremal elements in partially ordered sets
2. Lattice
   - Lattices as a mathematical structure
   - Isomorphic lattices
   - Properties of lattices

Overview

1. Finite Boolean Algebra
   - Boolean algebra: a special type of lattice
   - Substitution rule for Boolean algebra
2. Logical Design
   - Boolean expressions
   - Circuit Design

Lattice $(P(S), \subseteq)$

For a finite set S:

- The power set of S, $P(S)$, is a finite set of $2^{|S|}$ elements.
- Set inclusion is a partial order on $P(S)$.
- $(P(S), \subseteq)$ is a lattice
  For any subsets of S, S_1 and S_2, $S_1 \cup S_2$ is the (unique) least upper bound of S_1 and S_2; and $S_1 \cap S_2$ is the (unique) greatest lower bound of S_1 and S_2.

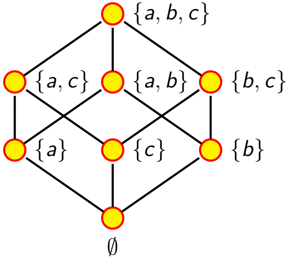
Isomorphism of Finite Lattices

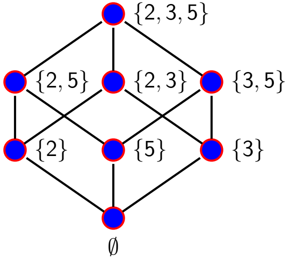
If $S_1 = \{x_1, x_2, \cdots, x_n\}$ and $S_2 = \{y_1, y_2, \cdots, y_n\}$ are any two finite sets with the same number of elements, then $(P(S_1), \subseteq)$ and $(P(S_2), \subseteq)$ are isomorphic.

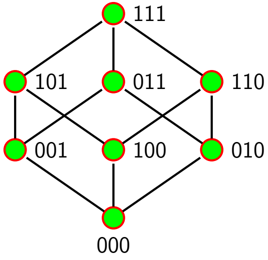
Proof.

A one-to-one correspondence: $f(x_i) = y_i$ for $i = 1, 2, \cdots, n$.
For any subset A, B of S_1, $A \subseteq B$ iff. $f(A) \subseteq f(B)$. □

Hasse Diagrams of Isomorphic Lattices



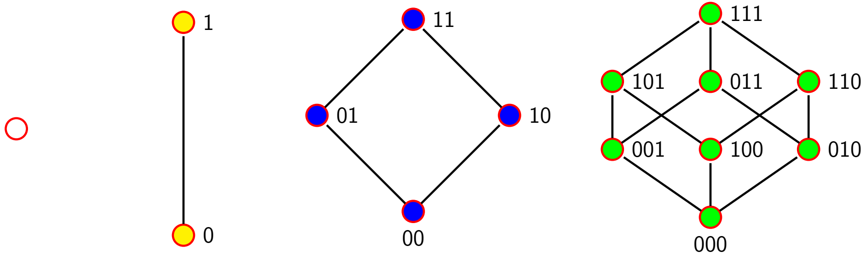




Lattice B_n

- B_n has 2^n elements.
- Each element is labeled by a sequence of 0's and 1's of length n.
- For any elements $x = a_1 a_2 \cdots a_n$, $y = b_1 b_2 \cdots b_n$, in B_n (each a_i,b_i is 0 or 1):
  - $x \preccurlyeq y$ iff. $a_k \preccurlyeq b_k$ for $k = 1, 2, \cdots, n$.
  - $x \wedge y = c_1 c_2 \cdots c_n$, where $c_k = min\{a_k, b_k\}$.
  - $x \vee y = d_1 d_2 \cdots d_n$, where $d_k = max\{a_k, b_k\}$

Hasse Diagrams of B_n



$n = 0$ $n = 1$ $n = 2$ $n = 3$

Boolean Algebra

Definition

A finite lattice isomorphic with B_n is called a **Boolean Algebra**.

Example

An example, D_6

- The set of D_6 is all positive integer divisors of 6.
- The partial order with D_6 is divisibility.
- D_6 is isomorphic with B_2:

$$f : D_6 \to B_2 : \quad f(1) = 00, f(2) = 10, f(3) = 01, f(6) = 11$$

B_n is distributive and Complemented

Proof.

For any x in B_n, x has a complement $x' = z_1 z_2 \cdots z_n$, where $z_k = 1$ if $x_k = 0$, and $z_k =_0$ if $x_k = 1$. □

Proof.

For any elements $x = a_1 a_2 \cdots a_n, y = b_1 b_2 \cdots b_n, z = c_1 c_2 \cdots c_n$, in B_n , (each a_i ,b_i, c_i is 0 or 1):

- $x \wedge (y \vee z) = (min\{a_1, max\{b_1, c_1\}\}))(min\{a_2, max\{b_2, c_2\}\})\cdots$ $(min\{a_n, max\{b_n, c_n\}\}) =$ $(max\{min\{a_1, b_1\}, min\{a_1, c_1\}\})(max\{min\{a_2, b_2\}, min\{a_2, c_2\}\})$ $\cdots\ (max\{min\{a_n, b_n\}, min\{a_n, c_n\}\}) = (x \wedge y) \vee (x \wedge z)$
- Similarly, $x \vee (y \wedge z) = (x \vee y) \wedge (x \vee z)$
- So, B_n is distributive.

□

A General Definition of Boolean Algebra

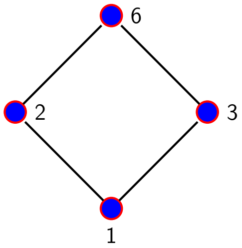
Definition

A distributive and complemented lattice is called a **Boolean Algebra**.

We can define a boolean algebra directly:

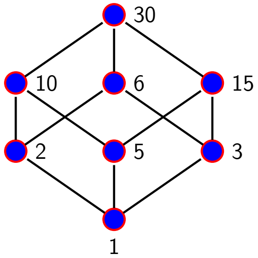
- A set B and a group of operations: join, meet, and complement
- Associative, Commutative and Distributive
- 0 and 1
- Complementary element

D_n is the poset of all positive divisors of n with the partial order "divisibility".



D_6

D_{20} is not a Boolean Algebra.



D_{30}

D_n as Boolean Algebra

Theorem

Let $n = p_1 p_2 \cdots p_k$, where the p_i are distinct primes. Then D_n is a Boolean algebra.

D_n as Boolean Algebra

Theorem

Let $n = p_1 p_2 \cdots p_k$, where the p_i are distinct primes. Then D_n is a Boolean algebra.

Sketch of proof.

- Let $S = \{p_1, p_2, \cdots p_k\}$, and for any subset T of S, a_T is the product of the primes in T.
- Note: any divisor of n must be some a_T. And we have $a_T | n$ for any T.
- For any subsets V, T, $V \subseteq T$ iff. $a_V | a_T$, and $a_V \wedge a_T = \text{GCD}(a_V, a_T)$ and $a_V \vee a_T = \text{LCM}(a_V, a_T)$.
- $f : P(S) \to D_n$ given by $f(T) = a_T$ is an isomorphism from $P(S)$ to D_n.

□

D_n as Boolean Algebra (cont.)

Theorem

If n is a positive integer and $p^2|n$, where p is a prime number, then D_n is not a Boolean algebra.

D_n as Boolean Algebra (cont.)

Theorem

If n is a positive integer and $p^2|n$, where p is a prime number, then D_n is not a Boolean algebra.

Proof.

Since $p^2|n$, $n = p^2q$ for some positive integer q. Note that p is also a element of D_n, then if D_n is a Boolean algebra, p must have a complement p', which means $\text{GCD}(p, p') = 1$ and $\text{LCM}(p, p') = n$. So, $pp' = n$, which leads to $p' = pq$. So, $\text{GCD}(p, pq) = 1$, contradiction. □

D_n as Boolean Algebra (cont.)

Theorem

If n is a positive integer and $p^2|n$, where p is a prime number, then D_n is not a Boolean algebra.

Proof.

Since $p^2|n$, $n = p^2q$ for some positive integer q. Note that p is also a element of D_n, then if D_n is a Boolean algebra, p must have a complement p', which means $\text{GCD}(p, p') = 1$ and $\text{LCM}(p, p') = n$. So, $pp' = n$, which leads to $p' = pq$. So, $\text{GCD}(p, pq) = 1$, contradiction. □

So, D_n is a Boolean algebra if and only if $n = p_1p_2 \cdots p_k$, where the p_i 's are distinct primes.

Operation Correspondence

Any formula involving $\cup$ or $\cap$ that holds for arbitrary subsets of a set S will continue to hold for arbitrary elements of a Boolean algebra L if $\wedge$ is substituted for $\cap$ and $\vee$ for $\cup$.

Proof.

- $(x')' = x \Leftrightarrow \overline{(\overline{A})} = A$
- $(x \wedge y)' = x' \vee y' \Leftrightarrow \overline{(A \cap B)} = \overline{A} \cup \overline{B}$
- $(x \vee y)' = x' \wedge y' \Leftrightarrow \overline{(A \cup B)} = \overline{A} \cap \overline{B}$
- $x \preccurlyeq y$ iff. $x \vee y = y \Leftrightarrow A \subseteq B$ iff. $A \cup B = B$
- $x \preccurlyeq y$ iff. $x \wedge y = x \Leftrightarrow A \subseteq B$ iff. $A \cap B = A$
- $x \vee 0 = x, x \wedge 0 = 0 \Leftrightarrow A \cup \emptyset = A, A \cap \emptyset = \emptyset$
- $x \vee 1 = 1, x \wedge 1 = x \Leftrightarrow A \cup S = S, A \cap S = A$
- ...

□

B_n as Product of n B's

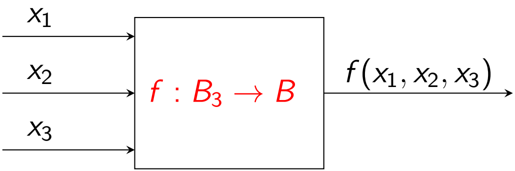
- B_1, $(\{0,1\}, \wedge, \vee, 1, 0, ')$, is denoted as B.
- For any $n \geq 1$, B_n is the product $B \times B \times \cdots \times B$ of B, n factors, where $B \times B \times \cdots \times B$ is given the product partial order.

Product partial order

$x \leq y$ if and only if $x_k \leq y_k$ for all k.

Truth Table

x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0



Boolean Polynomials

Definition

- $x_1, x_2, \cdots x_n$ are all Boolean polynomials (expressions).
- The symbols 0 and 1 are Boolean polynomials.
- If $p(x_1, x_2, \cdots x_n)$ and $q(x_1, x_2, \cdots x_n)$ are two Boolean polynomials, then so are:
  - $p(x_1, x_2, \cdots x_n) \vee q(x_1, x_2, \cdots x_n)$
  - $p(x_1, x_2, \cdots x_n) \wedge q(x_1, x_2, \cdots x_n)$
  - $(p(x_1, x2, \cdots x_n))'$
- There are no Boolean polynomials in the variables x_k other than those that can be obtained by repeated use of the rules above.

Interpreting Boolean Polynomials

- Boolean polynomials may be interpreted as representing Boolean computations with unspecified elements of B, that is, with 0's and 1's.
- Boolean polynomials are subject to the rules of Boolean algebra.
- Two Boolean polynomials are considered equivalent if one can be turned into the other with Boolean manipulations. Or, two boolean polynomials are equivalent if they have the truth tables with the same structure.

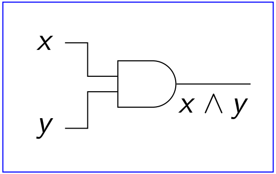
Truth Table: an Example

$$p(x_1, x_2, x_3) = (x_1 \wedge x_2) \vee (x_1 \vee (x_2' \wedge x_3))$$

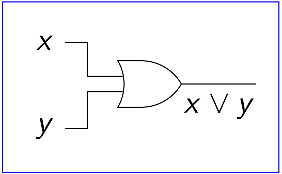
x_1	x_2	x_3	$(x_1 \wedge x_2) \vee (x_1 \vee (x_2' \wedge x_3))$
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Logic Diagrams for Boolean Polynomials

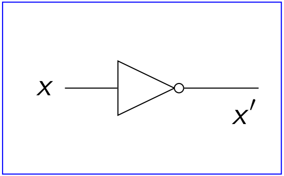
Basic Components:



and gate



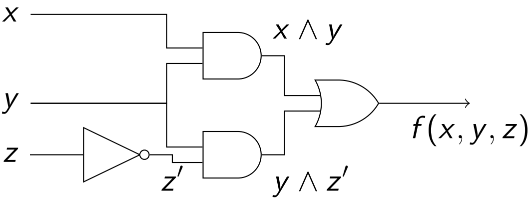
or gate



inverter

Logic Diagrams for Boolean Polynomials

$f(x, y, z) = (x \wedge y) \vee (y \wedge z')$



Subset of B_n Mapping to 1

Theorem

If $f : B_n \to B$, *define* $S(f) = \{b | b \in B_n, \text{ and } f(b) = 1\}$, *then, for three functions from* B_n *to* B, f, f_1, f_2, *we have:*

- *If* $S(f) = S(f_1) \cup S(f_2)$, *then* $f(b) = f_1(b) \vee f_2(b)$ *for all* b *in* B_n.
- *If* $S(f) = S(f_1) \cap S(f_2)$, *then* $f(b) = f_1(b) \wedge f_2(b)$ *for all* b *in* B_n.

Subset of B_n Mapping to 1

Theorem

If $f : B_n \rightarrow B$, define $S(f) = \{b | b \in B_n, \text{ and } f(b) = 1\}$, then, for three functions from B_n to B, f, f_1, f_2, we have:

- *If $S(f) = S(f_1) \cup S(f_2)$, then $f(b) = f_1(b) \vee f_2(b)$ for all b in B_n.*
- *If $S(f) = S(f_1) \cap S(f_2)$, then $f(b) = f_1(b) \wedge f_2(b)$ for all b in B_n.*

Proof.

For any b in B_n, if $b \in S(f)$, then $f(b) = 1$. Either b is in $S(f_1)$ or in $S(f_2)$, or both. In either cases $f_1(b) \vee f_2(b) = 1$. On the other hand, if $b \notin S(f)$, then $f(b) = 0$. Since neither $b \in S(f_1)$ nor $b \in S(f_2)$, so, $f_1(b) \vee f_2(b) = 0$. Thus, for all $b \in B_n$, $f(b) = f_1(b) \vee f_2(b)$.

Same for the second part. □

Minterm

Minterm expression: for $b = (c_1 c_2 \cdots c_n) \in B_n$,
$E_b = \overline{x_1} \wedge \overline{x_2} \wedge \cdots \wedge \overline{x_n}$, where $\overline{x_k} = x_k$ if $c_k = 1$, $\overline{x_k} = {x_k}'$ if $c_k = 0$.

x	y	$f(x,y)$	
0	0	0	
0	1	1	$x' \wedge y$
1	0	0	
1	1	0	

x_1	x_2	x_3	$f(x_1,x_2,x_3)$	
0	0	0	0	
0	0	1	0	
0	1	0	1	$x_1' \wedge x_2 \wedge x_3'$
0	1	1	0	
1	0	0	0	
1	0	1	0	
1	1	0	0	
1	1	1	0	

All Functions Expressible

Any function $f : B_n \to B$ can be produced by a Boolean expression

- Union of minterms.
- Proof:
  - For any given boolean function $f : B_n \to B$, let $S(f) = \{b_1, b_2, \cdots, b_k\}$.
  - For each $i = 1, 2, \cdots, k$, define function $f_i : B_n \to B$, as, $f_i(b_i) = 1$ and $f_i(b) = 0$ for any other b.
  - Then $S(f_i) = \{b_i\}$, so, $S(f) = S(f_1) \cup \cdots \cup S(f_k)$.
  - So, $f = f_1 \vee f_2 \vee \cdots, \vee f_k$, which is produced by the union of all minterms E_{b_i}.

Karnaugh Map of f for $n = 2$

$f : B_2 \to B$

Basic positions:

00	01
10	11

$$\begin{array}{ccc} & y' & y \\ x' & x' \wedge y' & x' \wedge y \\ x & x \wedge y' & x \wedge y \end{array}$$

$$f(x, y) = (x' \wedge y') \vee (x' \wedge y)$$

x	y	$f(x, y)$
0	0	1
0	1	1
1	0	0
1	1	0

$$\begin{array}{ccc} & y' & y \\ x' & 1 & 1 \\ x & 0 & 0 \end{array}$$

Karnaugh Map of f for $n = 2$

$f : B_2 \to B$

Basic positions:

00	01
10	11

$$\begin{array}{ccc} & y' & y \\ x' & x' \wedge y' & x' \wedge y \\ x & x \wedge y' & x \wedge y \end{array}$$

$f(x, y) = (x' \wedge y') \vee (x' \wedge y)$

x	y	$f(x, y)$
0	0	1
0	1	1
1	0	0
1	1	0

$$\begin{array}{ccc} & y' & y \\ x' & 1 & 1 \\ x & 0 & 0 \end{array}$$

$f(x, y) = x'$!

Simplifying Using Karnaugh Map

$f : B_2 \to B$

Basic positions:

00	01
10	11

	y'	y
x'	$x' \wedge y'$	$x' \wedge y$
x	$x \wedge y'$	$x \wedge y$

$$f(x, y) = (x' \wedge y') \vee (x' \wedge y) \vee (x \wedge y')$$

x	y	$f(x, y)$
0	0	1
0	1	1
1	0	1
1	1	0

	y'	y
x'	1	1
x	1	0

$f(x, y) = x' \vee y'$

Karnaugh Map with $n = 3$

	00	01	11	10
0	000	001	011	010
1	100	101	111	110

	$y' \wedge z'$	$y' \wedge z$	$y \wedge z$	$y \wedge z'$
x'	$x' \wedge y' \wedge z'$	$x' \wedge y' \wedge z$	$x' \wedge y \wedge z$	$x' \wedge y \wedge z'$
x	$x \wedge y' \wedge z'$	$x \wedge y' \wedge z$	$x \wedge y \wedge z$	$x \wedge y \wedge z'$

Can you find the areas corresponding to $x, x', y, y', z,$ and z', respectively?

Simplifying 3-Variable Expression

x	y	z	$f(x,y,z)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

The expression:

$$(x' \wedge y' \wedge z) \vee (x' \wedge y \wedge z') \vee$$
$$(x \wedge y' \wedge z') \vee (x \wedge y \wedge z') \vee (x \wedge y \wedge z)$$

Simplifying 3-Variable Expression

x	y	z	$f(x,y,z)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

The expression:

$$(x' \wedge y' \wedge z) \vee (x' \wedge y \wedge z') \vee (x \wedge y' \wedge z') \vee (x \wedge y \wedge z') \vee (x \wedge y \wedge z)$$

	$y'z'$	$y'z$	yz	yz'
x'	1	0	0	1
x	1	0	1	1

Simplifying 3-Variable Expression

x	y	z	$f(x,y,z)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

	$y'z'$	$y'z$	yz	yz'
x'	1	0	0	1
x	1	0	1	1

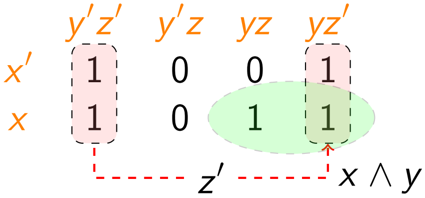
z'

The expression:

$$(x' \wedge y' \wedge z) \vee (x' \wedge y \wedge z') \vee$$
$$(x \wedge y' \wedge z') \vee (x \wedge y \wedge z') \vee (x \wedge y \wedge z)$$

Simplifying 3-Variable Expression

x	y	z	$f(x,y,z)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



The expression:

$$(x' \wedge y' \wedge z) \vee (x' \wedge y \wedge z') \vee$$
$$(x \wedge y' \wedge z') \vee (x \wedge y \wedge z') \vee (x \wedge y \wedge z)$$

Simplifying 3-Variable Expression

x	y	z	$f(x,y,z)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

The expression:

$$(x' \wedge y' \wedge z) \vee (x' \wedge y \wedge z') \vee (x \wedge y' \wedge z') \vee (x \wedge y \wedge z') \vee (x \wedge y \wedge z)$$

	$y'z'$	$y'z$	yz	yz'
x'	1	0	0	1
x	1	0	1	1

z'

$x \wedge y$

So,

$$f(x,y,z) = z' \vee (x \wedge y)$$

Logic Circuit at Work

For each try in a contest of weight lifting, it is assumed success only if at least 2 of 3 referees decide it a success. Design a logic circuit for use in the situation.

Logic Circuit at Work

For each try in a contest of weight lifting, it is assumed success only if at least 2 of 3 referees decide it a success. Design a logic circuit for use in the situation.

The function: $f(x, y, z) = 1$ iff.
there are at least two **1**s in x, y, x.

The expression:
$(x' \wedge y \wedge z) \vee (x \wedge y' \wedge z) \vee$
$(x \wedge y \wedge z') \vee (x \wedge y \wedge z)$

Logic Circuit at Work

For each try in a contest of weight lifting, it is assumed success only if at least 2 of 3 referees decide it a success. Design a logic circuit for use in the situation.

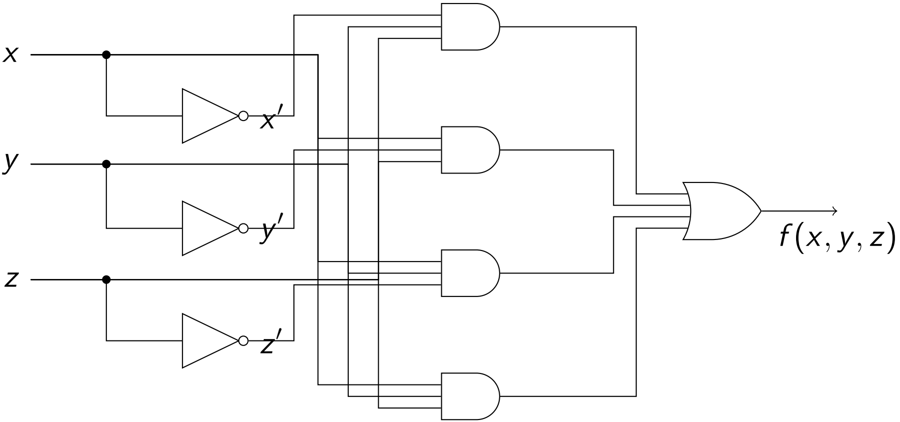
The function: $f(x, y, z) = 1$ iff. there are at least two **1**s in x, y, x.

The expression:
$(x' \wedge y \wedge z) \vee (x \wedge y' \wedge z) \vee$
$(x \wedge y \wedge z') \vee (x \wedge y \wedge z)$

x	y	z	$f(x, y, z)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

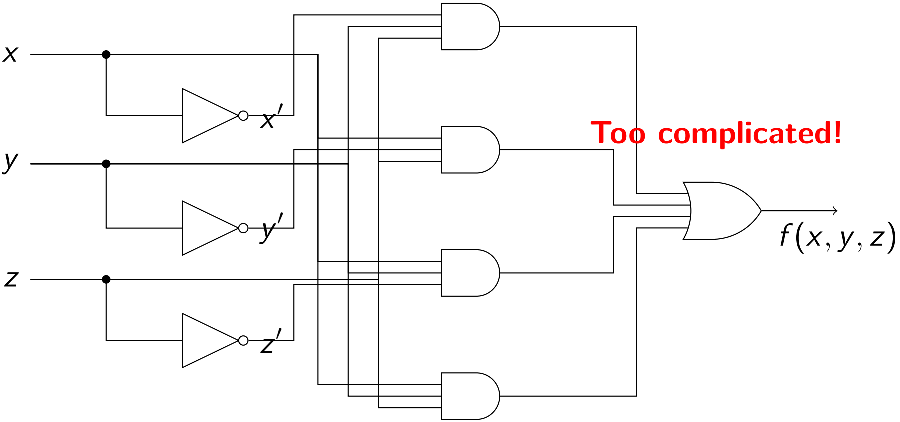
The Circuit

The expression: $(x' \wedge y \wedge z) \vee (x \wedge y' \wedge z) \vee (x \wedge y \wedge z') \vee (x \wedge y \wedge z)$



The Circuit

The expression: $(x' \wedge y \wedge z) \vee (x \wedge y' \wedge z) \vee (x \wedge y \wedge z') \vee (x \wedge y \wedge z)$



Make it Simpler

x	y	z	$f(x,y,z)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

	$y'z'$	$y'z$	yz	yz'
x'	0	0	1	0
x	0	1	1	1

Make it Simpler

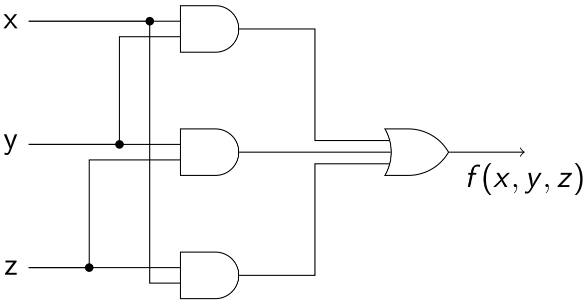
x	y	z	$f(x,y,z)$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

	$y'z'$	$y'z$	yz	yz'
x'	0	0	1	0
x	0	1	1	1

$$f(x,y,z) = (y \wedge z) \vee (x \wedge z) \vee (x \wedge y)$$

Looks Better

The expression: $f(x, y, z) = (y \wedge z) \vee (x \wedge z) \vee (x \wedge y)$



K-map of 4-Variable Expressions

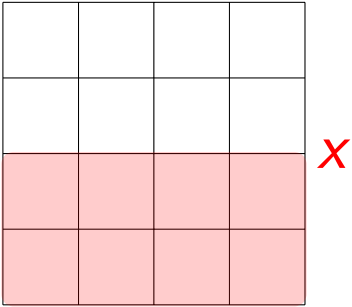
For 4-variable Boolean expression $f(x, y, z, w)$

	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010

K-map of 4-Variable Expressions

For 4-variable Boolean expression $f(x, y, z, w)$

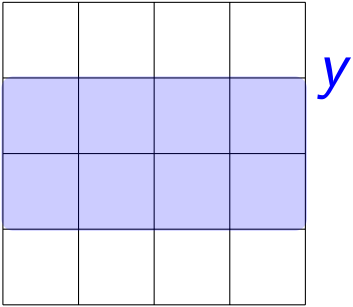
	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010



K-map of 4-Variable Expressions

For 4-variable Boolean expression $f(x, y, z, w)$

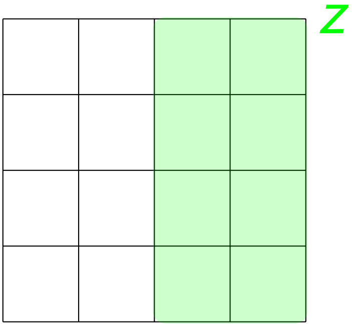
	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010



K-map of 4-Variable Expressions

For 4-variable Boolean expression $f(x, y, z, w)$

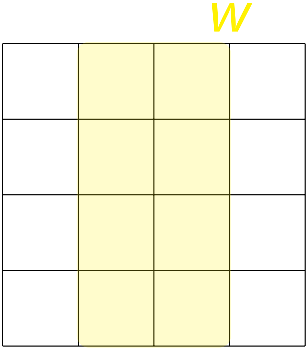
	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010



K-map of 4-Variable Expressions

For 4-variable Boolean expression $f(x, y, z, w)$

	00	01	11	10
00	0000	0001	0011	0010
01	0100	0101	0111	0110
11	1100	1101	1111	1110
10	1000	1001	1011	1010



An Example

1	0	0	1
0	1	1	0
0	1	1	0
1	0	0	1

An Example

w

1	0	0	1	
0	1	1	0	y
0	1	1	0	
1	0	0	1	

An Example

		w		
1	0	0	1	y
0	1	1	0	
0	1	1	0	
1	0	0	1	

So, it is $(y \wedge w) \vee (y' \wedge w')$

Another Example

1	1	1	1
0	0	0	0
0	0	1	0
1	1	0	0

Another Example

z' y'

1	1	1	1
0	0	0	0
0	0	1	0
1	1	0	0

2×2 square gives 2-literal minterm: $y' \wedge z'$

Another Example

z' y'

1	1	1	1
0	0	0	0
0	0	1	0
1	1	0	0

2×2 square gives 2-literal minterm: $y' \wedge z'$

1×2 square gives 3-literal minterm: $x' \wedge y' \wedge z$

Another Example

z' y'

1	1	1	1
0	0	0	0
0	0	1	0
1	1	0	0

2×2 square gives 2-literal minterm: $y' \wedge z'$

1×2 square gives 3-literal minterm: $x' \wedge y' \wedge z$

1×1 square gives 4-literal minterm: $x \wedge y \wedge z \wedge w$

Another Example

z' y'

1	1	1	1
0	0	0	0
0	0	1	0
1	1	0	0

2×2 square gives 2-literal minterm: $y' \wedge z'$

1×2 square gives 3-literal minterm: $x' \wedge y' \wedge z$

1×1 square gives 4-literal minterm: $x \wedge y \wedge z \wedge w$

So, it is $(y' \wedge z') \vee (x' \wedge y' \wedge z) \vee (x \wedge y \wedge z \wedge w)$

Same, or Different

The same Boolean function may take different forms, and, ...

The same circuit can implement different Boolean functions, maybe with some exchanges on inputs.

Home Assignments

To be checked

- Ex.6.4: 6, 8, 10, 16-21, 27, 29, 32
- Ex.6.5: 11-14, 18-23
- Ex.6.6: 8, 12, 14, 16, 24, 25-26

Experiment 6

The End