BOOLEAN ALGEBRA (1)

- Algebraic properties of functions of logical variables
  - ▷ Every Boolean function can be implemented by a combinational logic circuit
  - ▷ Every combinational logic circuit implements a Boolean function
- Fundamental Boolean operations are + (OR), · (AND) and NOT
- Approaches:
  - ▷ Axiomatic
    - ∘ Algebraic proofs
  - ▷ Practical
    - ∘ Truth-table demonstrations

BOOLEAN ALGEBRA (2)

AXIOMS AND BASIC THEOREMS

Description	OR form	AND form
Axiom 2	$x + 0 = x$	$x \cdot 1 = x$
Axiom 3	$x + y = y + x$	$x \cdot y = y \cdot x$
Axiom 4	$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$	$x + y \cdot z = (x + y) \cdot (x + z)$
Axiom 5	$x + \overline{x} = 1$	$x \cdot \overline{x} = 0$
Theorem 1	$x + x = x$	$x \cdot x = x$
Theorem 2	$x + 1 = 1$	$x \cdot 0 = 0$
Theorem 3	$\overline{\overline{x}} = x$	
Associativity	$x + (y + z) = (x + y) + z$	$x \cdot (y \cdot z) = (x \cdot y) \cdot z$
Absorption	$x + x \cdot y = x$	$x \cdot (x + y) = x$
DeMorgan's laws	$\overline{x + y} = \overline{x} \cdot \overline{y}$	$\overline{x \cdot y} = \overline{x} + \overline{y}$

BOOLEAN ALGEBRA (3)

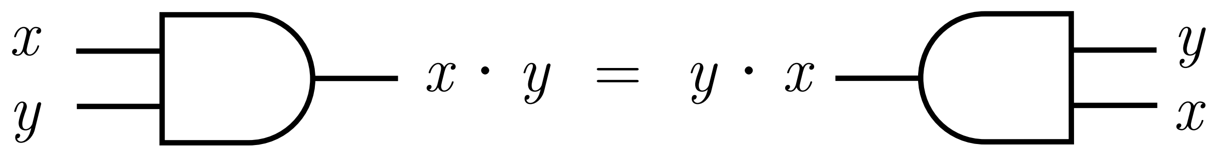
Verification of Axiom 3 (Commutativity):

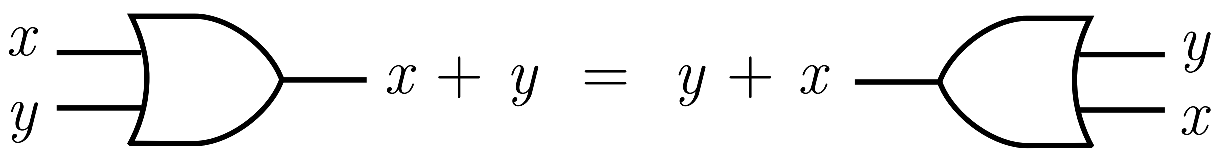
x	y	$x + y$	$y + x$
0	0	0	0
0	1	1	1
1	0	1	1
1	1	1	1

x	y	$x \cdot y$	$y \cdot x$
0	0	0	0
0	1	0	0
1	0	0	0
1	1	1	1

BOOLEAN ALGEBRA (4)

Illustration of Axiom 3 (Commutativity):





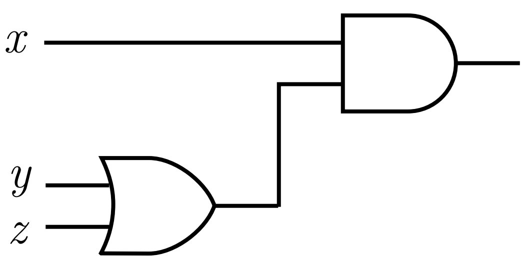
BOOLEAN ALGEBRA (5)

Verification of Axiom 4 (Distributivity of AND):

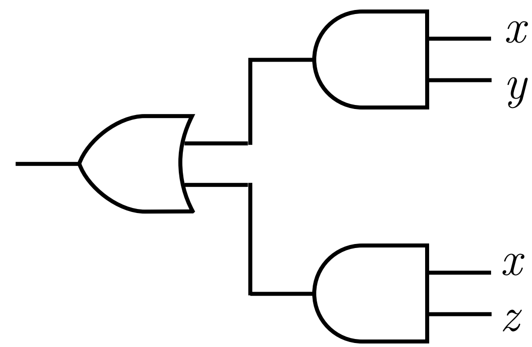
x	y	z	$y+z$	$x \cdot (y+z)$	$x \cdot y$	$x \cdot z$	$(x \cdot y) + (x \cdot z)$
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

BOOLEAN ALGEBRA (6)

Illustration of Axiom 4 (Distributivity of AND over OR):



$$x \cdot (y + z) = (x \cdot y) + (x \cdot z)$$



BOOLEAN ALGEBRA (7)

• DeMorgan's laws

▷ How to interchange NOT with AND or OR

▷ OR form:

$$\overline{x+y} = \overline{x} \cdot \overline{y}$$

▷ AND form:

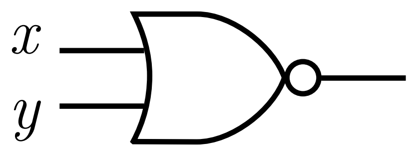
$$\overline{x \cdot y} = \overline{x} + \overline{y}$$

▷ Truth table:

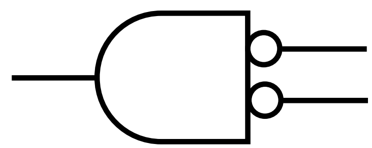
x	y	$\overline{x+y}$	$\overline{x} \cdot \overline{y}$	$\overline{x \cdot y}$	$\overline{x} + \overline{y}$
0	0	1	1	1	1
0	1	0	0	1	1
1	0	0	0	1	1
1	1	0	0	0	0

BOOLEAN ALGEBRA (8)

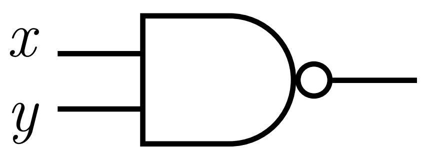
- OR form of DeMorgan's laws:



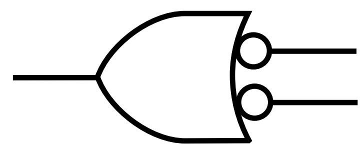
$$\overline{x + y} = \overline{x} \cdot \overline{y}$$



- AND form of DeMorgan's laws:



$$\overline{x \cdot y} = \overline{x} + \overline{y}$$



BOOLEAN ALGEBRA (9)

- A **Boolean function** $f: \{0,1\}^n \to \{0,1\}$
  - ▷ maps n inputs $x_1, \ldots, x_n$ to one output y,
  - ▷ takes only the values 0 or 1, and
  - ▷ accepts only inputs that take the values 0 or 1
- Truth-table representation:

x_1	x_2	$\cdots$	x_n	$y = f(x_1, \ldots, x_n)$	term
0	0	$\cdots$	1	0	
0	1	$\ldots$	0	1	$\overline{x}_1 \cdot x_2 \cdots \overline{x}_n$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	
0	1	$\cdots$	1	1	$\overline{x}_1 \cdot x_2 \cdots x_n$
1	1	$\cdots$	1	0	

- Algebraic representation (for example, as a sum of products):

$$f(x_1, \ldots, x_n) = \overline{x}_1 \cdot x_2 \cdots \overline{x}_n + \cdots + \overline{x}_1 \cdot x_2 \cdots x_n$$

BOOLEAN ALGEBRA (10)

- A Boolean function that is expressed in algebraic form can be simplified using the axioms and theorems of Boolean algebra
  - ▷ Example 1: OR form of the Absorption Theorem

$$x + x \cdot y = x \cdot 1 + x \cdot y = x \cdot (1 + y) = x \cdot 1 = x$$

  - ▷ Example 2:

$$\begin{aligned} x + \overline{x} \cdot y &= x + x \cdot y + \overline{x} \cdot y \\ &= x \cdot x + x \cdot y + \overline{x} \cdot x + \overline{x} \cdot y \\ &= (x + \overline{x}) \cdot (x + y) = 1 \cdot (x + y) \\ &= x + y \end{aligned}$$

  - ▷ Example 3:

$$x \cdot (x + y) = x \cdot x + x \cdot y = x + x \cdot y = x$$

BOOLEAN ALGEBRA (11)

- The **Consensus Theorem**:

$$x \cdot y + \overline{x} \cdot z + y \cdot z = x \cdot y + \overline{x} \cdot z$$

  - ▷ Intuitively obvious, for if $y \cdot z$ is true, then both y and z are true, and therefore either $x \cdot y$ is true or $\overline{x} \cdot z$ is true
  - ▷ For an algebraic proof, introduce $x + \overline{x} = 1$ into the last term:

$$\begin{aligned} x \cdot y + \overline{x} \cdot z + y \cdot z &= x \cdot y + \overline{x} \cdot z + y \cdot z \cdot (x + \overline{x}) \\ &= [(x \cdot y) + (x \cdot y) \cdot z] + [(\overline{x} \cdot z) + (\overline{x} \cdot z) \cdot y] \\ &= x \cdot y + \overline{x} \cdot z \end{aligned}$$

  (The last line follows from the Absorption Theorem)

BOOLEAN ALGEBRA (12)

- A product (multiple AND) $x_1\overline{x}_2 \cdots \overline{x}_n$ is called a **minterm** because it takes the value "true" a minimal number of times (for one and only one set of values of $x_1, \ldots, x_n$)
  - ▷ The minterm $x_1\overline{x}_2 \cdots \overline{x}_n$ is realized as an AND gate with n inputs
  - ▷ In a sum-of-minterms representation of a Boolean function,
    - ◦ No. of logical variables in each minterm
      = no. of inputs to the equivalent AND gate
    - ◦ No. of terms in sum
      = no. of AND gates in 1st level of logic
      = no. of inputs to 2nd level OR
- A sum (multiple OR) $x_1 + \overline{x}_2 + \cdots + \overline{x}_n$ is called a **maxterm** because it takes the value "true" a maximal number of times (whenever at least one of $x_1, \overline{x}_2, \ldots, \overline{x}_n$ is true)

BOOLEAN ALGEBRA (13)

- Truth-table representation of the sum of minterms $f(x, y, z) = \overline{x} \cdot y \cdot z + x \cdot \overline{y} \cdot \overline{z} = \sum m(3, 4)$:

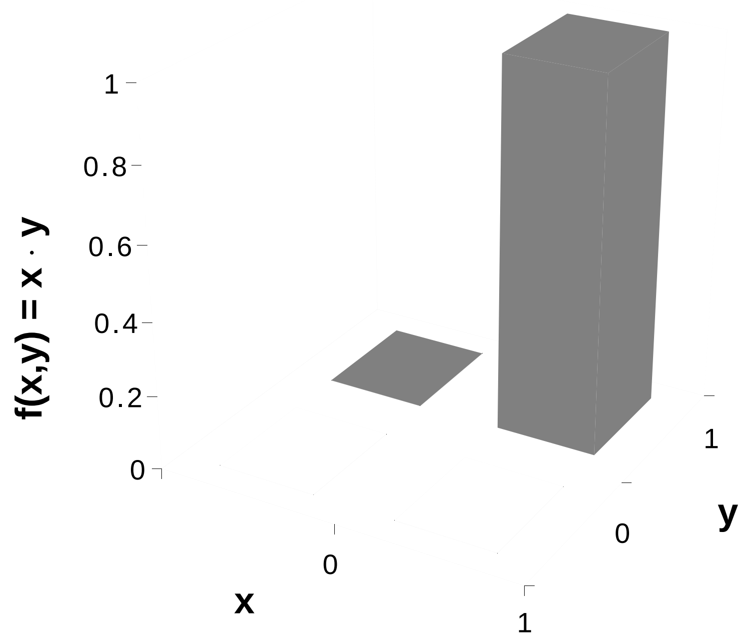
x	y	z	$f(x, y, z)$	Label
0	0	0	0	0
0	0	1	0	1
0	1	0	0	2
0	1	1	1	3
1	0	0	1	4
1	0	1	0	5
1	1	0	0	6
1	1	1	0	7

  ▷ Each minterm (row) is labeled by the string of values of x, y, z, considered as an unsigned 3-bit integer

KARNAUGH MAPS (1)

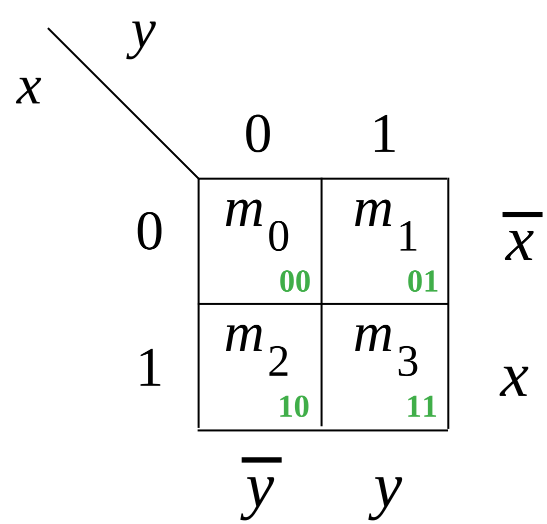
- A **Karnaugh map** is a pictorial representation of a truth table
  - ▷ Purpose: Make it easier to transform a logic function to a minimal sum of products
  - ▷ Used to:
    - ◦ Reduce the number of gates in the critical path
    - ◦ Reduce the number of (and number of inputs to) 1st level gates
  - ▷ Advantage: Uses humans' ability to interpret visual information
  - ▷ Disadvantages:
    - ◦ Visual information is not easy for computers to analyze
    - ◦ Not useful for more than five variables
    - ◦ Not used in industry
    - ◦ Essentially an academic teaching tool

Karnaugh map for logical “and”, viewed as a bar chart



Two-Variable Karnaugh map

Minterms in x,y



Three-Variable Karnaugh map

Minterms in x,y,z

x \ $y\,z$	$\bar{z}$	z		$\bar{z}$	
	00	01	11	10	
0	m_0 000	m_1 001	m_3 011	m_2 010	$\bar{x}$
1	m_4 100	m_5 101	m_7 111	m_6 110	x
	$\bar{y}$		y		

Four -Variable Karnaugh map

Minterms in *w,x,y,z*

$wx \backslash yz$	00	01	11	10
00	m_0 0000	m_1 0001	m_3 0011	m_2 0010
01	m_4 0100	m_5 0101	m_7 0111	m_6 0110
11	m_{12} 1100	m_{13} 1101	m_{15} 1111	m_{14} 1110
10	m_8 1000	m_9 1001	m_{11} 1011	m_{10} 1010

Columns 00: $\overline{z}$; columns 01, 11: z; column 10: $\overline{z}$. Columns 00, 01: $\overline{y}$; columns 11, 10: y.

Rows 00, 01: $\overline{w}$; rows 11, 10: w. Row 00: $\overline{x}$; rows 01, 11: x; row 10: $\overline{x}$.

KARNAUGH MAPS (2)

- Rules for assigning minterms to cells in Karnaugh maps:
  - ▷ Each cell corresponds to one minterm (one AND of all of the logical variables or their complements)
    - ◦ Example: Cell 3 in a 4-variable Karnaugh map corresponds to the minterm $w \cdot x \cdot \overline{y} \cdot \overline{z}$
  - ▷ For each cell and its minterm, the values of the logical variables are written as a binary integer, the bits of which are the values of the logical variables for which the minterm is true
    - ◦ Example: For cell 3 in a 4-variable Karnaugh map, the binary integer is `0011`, corresponding to $w = 0$, $x = 0$, $y = 1$, $z = 1$
  - ▷ Only one bit changes when one goes to an adjacent cell
    - ◦ Example: Cells 3 (`0011`) and 11 (`1011`) are adjacent
  - ▷ The top and bottom of the map are adjacent
  - ▷ The right and left sides of the map are adjacent

KARNAUGH MAPS (3)

- Rules for displaying a Boolean function:
  - ▷ Put a 1 in each cell for which the function is true
    - ∘ Example: For the Boolean function

$$f(w,x,y,z) = \sum m(4,5,12,13)$$
$$= \overline{w}\cdot x\cdot\overline{y}\cdot\overline{z} + \overline{w}\cdot x\cdot\overline{y}\cdot z + w\cdot x\cdot\overline{y}\cdot\overline{z} + w\cdot x\cdot\overline{y}\cdot z,$$

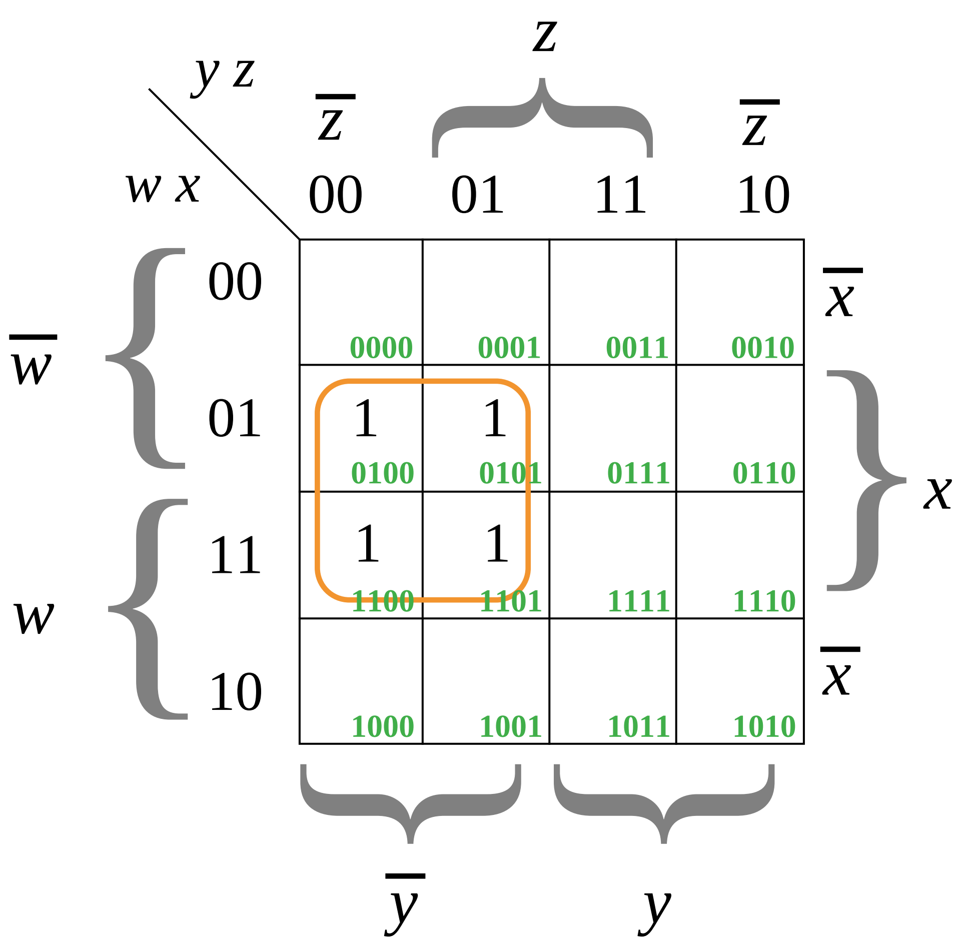
put 1's in the cells numbered 4, 5, 12, and 13 (see next slide), because the function is true when any one of the following minterms is true:

$m_4 = \overline{w}\cdot x\cdot\overline{y}\cdot\overline{z}$, $m_5 = \overline{w}\cdot x\cdot\overline{y}\cdot z$, $m_{12} = w\cdot x\cdot\overline{y}\cdot\overline{z}$, $m_{13} = w\cdot x\cdot\overline{y}\cdot z$

  - ▷ Do not put 0's in the cells for which the function is false

Four -Variable Karnaugh map

$f(w,x,y,z) = \Sigma\, m(4,5,12,13) = x \cdot \bar{y}$



KARNAUGH MAPS (4)

- Rules for finding prime implicants of a Boolean function f:
  - ▷ Circle only cells that contain 1's
    - ◦ Each circled group of cells corresponds to one implicant of f
    - ◦ Example: In the preceding slide, the circled cells are 4, 5, 12, and 13, corresponding to the implicant $x \cdot \overline{y}$
  - ▷ The number of cells enclosed in one circle must be a power of 2 (1, 2, 4, 8, or 16)
    - ◦ Example: In the preceding slide, $2^2 = 4$ cells are circled
  - ▷ To find a prime implicant, circle the maximum number of cells consistent with the above rules
- The more cells a given circle encloses, the smaller the number of logical variables needed to specify the implicant

KARNAUGH MAPS (5)

- Definitions for Karnaugh maps:
  - ▷ **Minimal sum**: Expression of a Boolean function f as a sum of products such that
    - ◦ No sum of products for f has fewer terms
    - ◦ Any sum of products with the same no. of terms has at least as many logical variables
  - ▷ A Boolean function g **implies** a Boolean function f if & only if for every set of input values such that g is 1, then f is 1
  - ▷ **Prime implicant** of a Boolean function f: A product term that implies f, such that if any logical variable is removed from the product, then the resulting product does not imply f
  - ▷ **Prime implicant theorem**: A minimal sum is a sum of prime implicants

KARNAUGH MAPS (6)

- More definitions and theorems for Karnaugh maps:
  - ▷ **Distinguished 1-cell**: A combination of inputs to a Boolean function that is implied by one & only one prime implicant
  - ▷ **Essential prime implicant** of a Boolean function f: A prime implicant that covers one or more distinguished 1-cells
  - ▷ Every minimal sum for a Boolean function f must include all of the essential prime implicants of f imply f
  - ▷ **Steps in simplifying a Boolean function**:
    - ◦ Determine the distinguished 1-cells and the prime implicants that imply them
    - ◦ Find a minimal set of prime implicants that imply that f is true for the input combinations that are not implied by essential prime implicants