Chapter 1

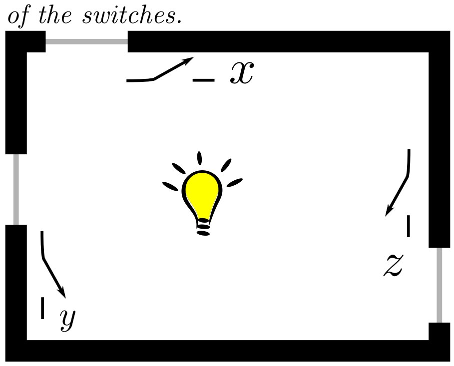
Boolean Algebra

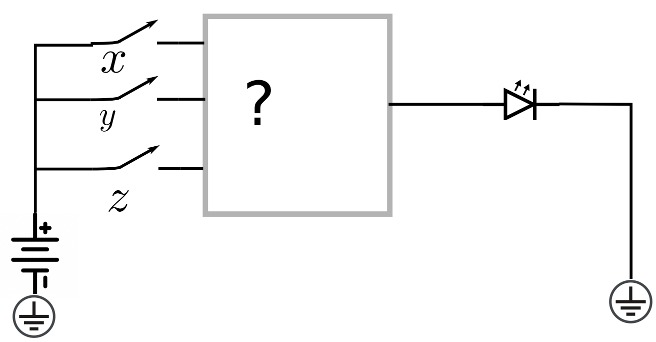
1.1 Learning objectives

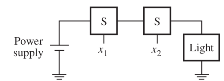
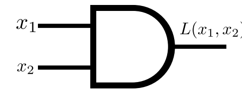
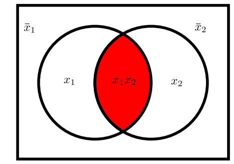
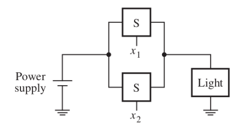
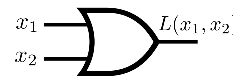
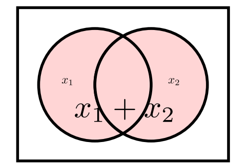
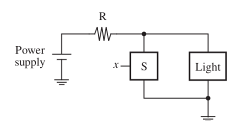
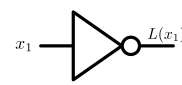
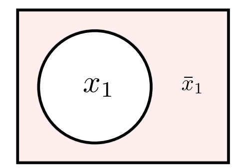
1. Representing digital circuits
2. Converting between different notations: Boolean expression, logic networks and switching circuits
3. Converting between different logic network specifications: truth table, minterm, maxterms, product of sums canonical form and sum of product canonical form.
4. Introduce truth tables as Behavioral Verilog
5. This handout has 11 homework problems totaling to 140 marks

1.2 Motivating Problem

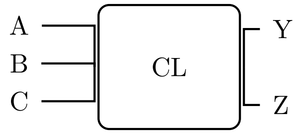
Example 1.1. *Assume that a large room has three doors and that a switch near each door controls a light in the room. It has to be possible to turn the light on or off by changing the state of any one of the switches.*





<table>
<tr><th>Name</th><th>C/Verilog</th><th>Boolean expr.</th><th>Truth Table</th><th>Switching circuit</th><th>(ANSI) symbol</th><th>Venn diagram</th></tr>
<tr><td>AND Gate</td><td><code>L = x1 & x2</code></td><td>$L = x_1 \cdot x_2 = x_1 x_2$</td><td>$\begin{array}{cc|c} x_1 & x_2 & x_1 \cdot x_2 \\ \hline 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{array}$</td><td></td><td></td><td></td></tr>
<tr><td>OR Gate</td><td><code>L = x1 | x2</code></td><td>$L = x_1 + x_2$</td><td>$\begin{array}{cc|c} x_1 & x_2 & x_1 + x_2 \\ \hline 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{array}$</td><td></td><td></td><td></td></tr>
<tr><td>NOT Gate</td><td><code>L = ∼ x1</code></td><td>$L = \bar{x}_1 = x_1'$</td><td>$\begin{array}{c|c} x_1 & \bar{x}_1 \\ \hline 0 & 1 \\ 1 & 0 \end{array}$</td><td></td><td></td><td></td></tr>
</table>

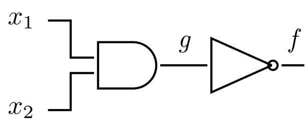
1.3 Digital circuits or networks



$$Y = F(A, B, C) \qquad Z = G(A, B, C)$$

1.4 Two input networks

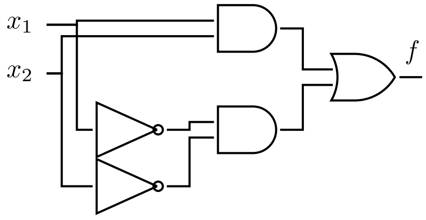
Example 1.2. *Convert the following (ANSI) network into a Boolean expression, a truth table and a Venn diagram.*



Example 1.3. *Convert the following Boolean expression into a (ANSI) network, a truth table and a Venn diagram:*

$$f = \overline{x_1 + x_2}$$

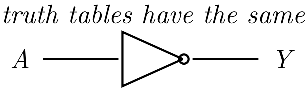
Problem 1.1 (10 marks)**.** *Convert the following (ANSI) network into a Boolean expression, a truth table and a Venn diagram.*

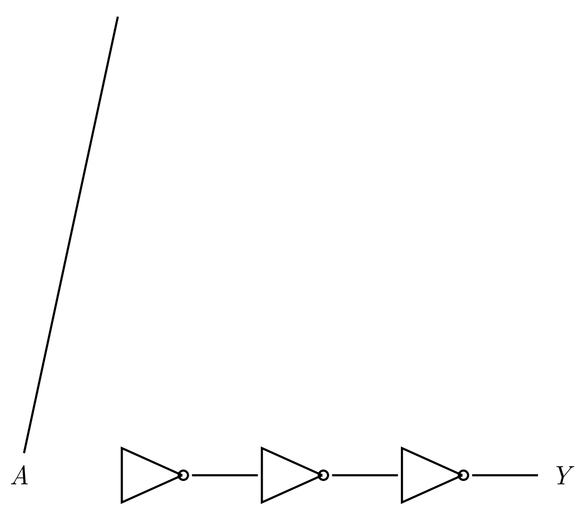


Example 1.4. *Convert the following Boolean expression into a network, a truth table and a Venn diagram:*

$$f = x_1\bar{x}_2 + \bar{x}_1 x_2$$

Problem 1.2 (5 marks)**.** *Can two different circuits have the same truth table? Can two different truth tables have the same circuit? Consider the following two circuits for example*

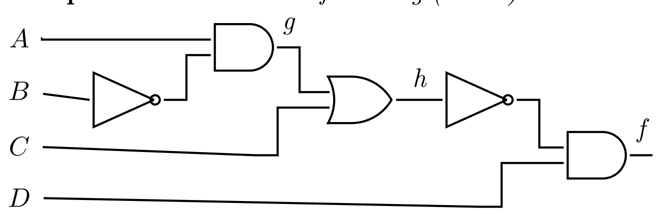




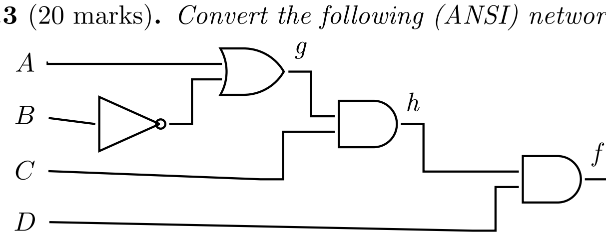
Remark 1.1. *Truth tables and Venn diagrams define* what *the combinational circuit should do. Truth tables define output for every input. Boolean expression and networks define* how *to achieve the desired input output relationship.*

1.5 Multi-input networks

Example 1.5. *Convert the following (ANSI) network into a Boolean expression and a truth table.*



Problem 1.3 (20 marks)**.** *Convert the following (ANSI) network into a Boolean expression and a truth table.*



1.6 Minterms and Maxterms

1.6.1 Minterms

Minterm is a product involving all inputs (or complements) to a function. Every row of a truth table has a corresponding minterm. Minterm is true if and only if the corresponding row in the table is active.

Minterms defined as follows for each row of a two input truth table:

A	B	minterm	minterm name
0	0	$\bar{A}\bar{B}$	m_0
0	1	$\bar{A}B$	m_1
1	0	$A\bar{B}$	m_2
1	1	AB	m_3

Consider a two input circuit whose output Y is given by the truth table:

A	B	Y	minterm	minterm name
0	0	0	$\bar{A}\bar{B}$	m_0
0	1	1	$\bar{A}B$	m_1
1	0	0	$A\bar{B}$	m_2
1	1	1	AB	m_3

then $Y = \bar{A}B + AB = m_1 + m_3 = \sum(1,3)$.
This also gives the *sum of products canonical form.*

Example 1.6. *What is the minterm m_{13} for a 4-input circuit with inputs x, y, z, w (ordered from MSB to LSB).*

Problem 1.4 (5 marks)**.** *What is the minterm m_{23} for a 5-input circuit with inputs a, b, c, d, e (ordered from MSB to LSB).*

Example 1.7. *Convert the following 4-input truth table into sum of minterms and sum of products canonical form.*

minterm name	*A*	*B*	*C*	*D*	*f*
m_0	0	0	0	0	0
m_1	0	0	0	1	1
m_2	0	0	1	0	0
m_3	0	0	1	1	0
m_4	0	1	0	0	0
m_5	0	1	0	1	1
m_6	0	1	1	0	0
m_7	0	1	1	1	0
m_8	1	0	0	0	0
m_9	1	0	0	1	0
m_{10}	1	0	1	0	0
m_{11}	1	0	1	1	0
m_{12}	1	1	0	0	0
m_{13}	1	1	0	1	1
m_{14}	1	1	1	0	0
m_{15}	1	1	1	1	0

Problem 1.5 (10 marks)**.** *Convert the following 4-input truth table into sum of minterms and sum of products canonical form.*

minterm name	*A*	*B*	*C*	*D*	*f*
m_0	0	0	0	0	0
m_1	0	0	0	1	0
m_2	0	0	1	0	0
m_3	0	0	1	1	1
m_4	0	1	0	0	0
m_5	0	1	0	1	0
m_6	0	1	1	0	0
m_7	0	1	1	1	1
m_8	1	0	0	0	0
m_9	1	0	0	1	0
m_{10}	1	0	1	0	0
m_{11}	1	0	1	1	1
m_{12}	1	1	0	0	0
m_{13}	1	1	0	1	1
m_{14}	1	1	1	0	1
m_{15}	1	1	1	1	0

1.6.2 Maxterms

Maxterm is a sum involving all inputs (or complements) to a function. Every row of a truth table has a corresponding maxterm. Minterm is false if and only if the corresponding row in the table is active.

Maxterms are defined as follows for each row of a two input truth table:

A	B	maxterm	maxterm name
0	0	$A+B$	M_0
0	1	$A+\bar{B}$	M_1
1	0	$\bar{A}+B$	M_2
1	1	$\bar{A}+\bar{B}$	M_3

Consider a two input circuit whose output Y is given by the truth table:

A	B	Y	maxterm	maxterm name
0	0	0	$A+B$	M_0
0	1	1	$A+\bar{B}$	M_1
1	0	0	$\bar{A}+B$	M_2
1	1	1	$\bar{A}+\bar{B}$	M_3

then $Y = (A+B)(\bar{A}+B) = M_0M_2$.
Writing a functional specification in terms of minterms is also called product of sums canonical form.

Example 1.8. *Write the maxterm M_{11} for 4-input Boolean function with the ordered inputs A, B, C, D.*

Example 1.9. *Convert the following 4-input truth table into product of maxterms and product of sums canonical form.*

maxterm name	*A*	*B*	*C*	*D*	*f*
M_0	0	0	0	0	0
M_1	0	0	0	1	0
M_2	0	0	1	0	0
M_3	0	0	1	1	1
M_4	0	1	0	0	0
M_5	0	1	0	1	0
M_6	0	1	1	0	0
M_7	0	1	1	1	1
M_8	1	0	0	0	0
M_9	1	0	0	1	0
M_{10}	1	0	1	0	0
M_{11}	1	0	1	1	1
M_{12}	1	1	0	0	0
M_{13}	1	1	0	1	1
M_{14}	1	1	1	0	1
M_{15}	1	1	1	1	0

Problem 1.6 (10 marks)**.** *Convert the following 4-input truth table into product of maxterms and products of sums canonical form.*

maxterm name	*A*	*B*	*C*	*D*	*f*
M_0	*0*	*0*	*0*	*0*	*0*
M_1	*0*	*0*	*0*	*1*	*1*
M_2	*0*	*0*	*1*	*0*	*1*
M_3	*0*	*0*	*1*	*1*	*1*
M_4	*0*	*1*	*0*	*0*	*1*
M_5	*0*	*1*	*0*	*1*	*0*
M_6	*0*	*1*	*1*	*0*	*1*
M_7	*0*	*1*	*1*	*1*	*1*
M_8	*1*	*0*	*0*	*0*	*0*
M_9	*1*	*0*	*0*	*1*	*1*
M_{10}	*1*	*0*	*1*	*0*	*1*
M_{11}	*1*	*0*	*1*	*1*	*1*
M_{12}	*1*	*1*	*0*	*0*	*0*
M_{13}	*1*	*1*	*0*	*1*	*1*
M_{14}	*1*	*1*	*1*	*0*	*1*
M_{15}	*1*	*1*	*1*	*1*	*0*

Example 1.10. *Write the 3-input truth table for the function* $f = m_2 + m_3 + m_7$.

Problem 1.7 (10 marks)**.** *Write the 3-input truth table for the function* $f = M_4 M_5 M_7$.

Problem 1.8 (10 marks)**.** *Write the truth table for the function* $f = \bar{A}B\bar{C} + AB\bar{C}$.

1.7 Karnaugh maps

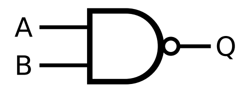
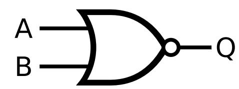
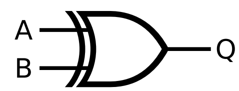
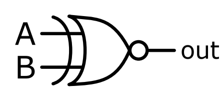
Two input K-maps

Three input K-maps

Four input K-maps

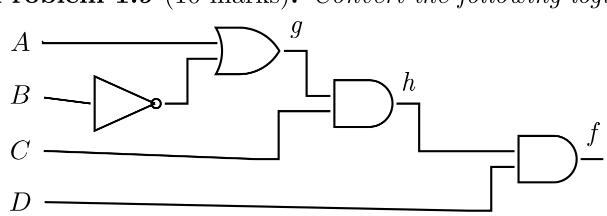
Five input K-maps

1.8 More Gates and notations summary

Name	C/Verilog	Boolean expr.	Truth Table	(ANSI) symbol	K-map
NAND Gate	`Q = ~(x1 & x2)`	$Q = \overline{x_1 \cdot x_2} = \overline{x_1 x_2}$	x_1 x_2 | $\overline{x_1 \cdot x_2}$ 0 0 | 1 0 1 | 1 1 0 | 1 1 1 | 0		B\A: 0 1 0: 1 1 1: 1 0
NOR Gate	`Q = ~(x1 | x2)`	$Q = \overline{x_1 + x_2}$	x_1 x_2 | $\overline{x_1 + x_2}$ 0 0 | 1 0 1 | 0 1 0 | 0 1 1 | 0		B\A: 0 1 0: 1 0 1: 0 0
XOR Gate	`Q = x1 ^ x2`	$Q = x_1 \oplus x_2$	x_1 x_2 | $x_1 \oplus x_2$ 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 0		B\A: 0 1 0: 0 1 1: 1 0
XNOR Gate	`Q = ~(x1 ^ x2)`	$Q = \overline{x_1 \oplus x_2}$	x_1 x_2 | $\overline{x_1 \oplus x_2}$ 0 0 | 1 0 1 | 0 1 0 | 0 1 1 | 1		B\A: 0 1 0: 1 0 1: 0 1

Example 1.11. *Convert the following Boolean expression into a K-map.* $f = \overline{A\bar{B} + C}D$

Problem 1.9 (10 marks)**.** *Convert the following logic circuit into a K-map.*



1.9 Boolean Algebra

1.9.1 Axioms of Boolean algebra

1. $0 \cdot 0 = 0$
2. $1 + 1 = 1$

3. $1 \cdot 1 = 1$
4. $0 + 0 = 0$
5. $0 \cdot 1 = 1 \cdot 0 = 0$
6. $\bar{0} = 1$
7. $\bar{1} = 0$
8. $x = 0$ if $x \neq 1$
9. $x = 1$ if $x \neq 0$

1.9.2 Single variable theorems (Prove by drawing K-maps)

1. $x \cdot 0 = 0$
2. $x + 1 = 1$
3. $x \cdot 1 = x$
4. $x + 0 = x$
5. $x \cdot x = x$
6. $x + x = x$
7. $x \cdot \bar{x} = 0$

8. $x + \bar{x} = 1$

9. $\bar{\bar{x}} = x$

Remark 1.2 (Duality)**.** *Swap + with · and 0 with 1 to get another theorem*

1.9.3 Two and three variable properties (Prove by K-maps)

1. Commutative: $x \cdot y = y \cdot x$, $x + y = y + x$

2. Associative: $x \cdot (y \cdot z) = (x \cdot y) \cdot z$, $x + (y + z) = (x + y) + z$

3. Distributive: $x \cdot (y + z) = x \cdot y + x \cdot z$, $x + y \cdot z = (x + y) \cdot (y + z)$

4. Absorption: $x + x \cdot y = x$, $x \cdot (x + y) = x$

5. Combining: $x \cdot y + x \cdot \bar{y}$, $(x + y) \cdot (x + \bar{y}) = x$

6. DeMorgan's theorem: $\overline{x \cdot y} = \bar{x} + \bar{y}$, $\overline{x + y} = \bar{x} \cdot \bar{y}$.

7. Concensus:

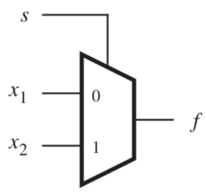
   (a) $x + \bar{x} \cdot y = x + y$

   (b) $x \cdot (\bar{x} + y) = x \cdot y$

   (c) $x \cdot y + y \cdot z + \bar{x} \cdot z = x \cdot y + \bar{x} \cdot z$

   (d) $(x + y) \cdot (y + z) \cdot (\bar{x} + z) = (x + y) \cdot (\bar{x} + z)$

Example 1.12 (Multiplexer). *Multiplexer is a circuit used to select one of the input lines x_1 and x_2 based only select input s. When $s = 0$, x_1 is selected, x_2 is selected otherwise. Find a boolean expression and a circuit for multiplexer*



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

Example 1.13. *Simplify* $f = \bar{A}\bar{B}\bar{C} + A\bar{B}\bar{C} + A\bar{B}\bar{C}$ *using boolean algebra.*

Problem 1.10 (30 marks, Exercise 2.14 [1]). *Simplify the following Boolean equations using Boolean theorems.*

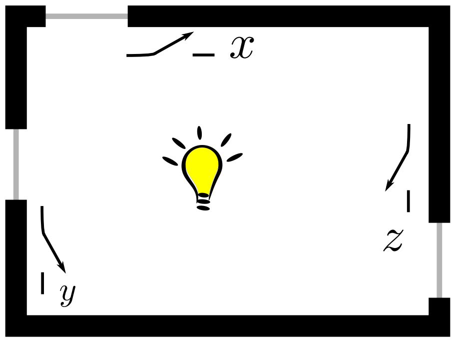
$$Y = \bar{A}BC + \bar{A}B\bar{C} \tag{1.1}$$

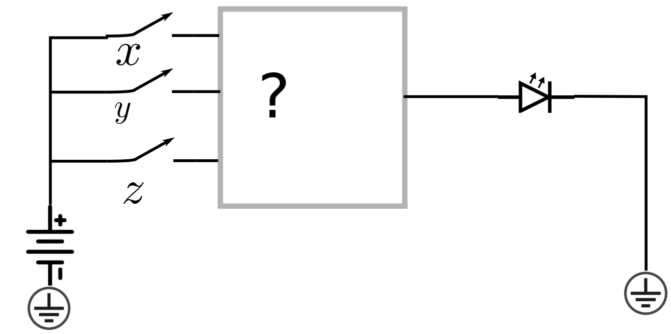
$$Y = \overline{ABC} + A\bar{B} \tag{1.2}$$

$$Y = ABC\bar{D} + A\overline{BCD} + (\overline{A + B + C + D}) \tag{1.3}$$

Example 1.14. *Simplify* $f = \bar{A}\bar{A}\bar{C} + \bar{A}\bar{B}C$ *using K-maps.*

Example 1.15. *Assume that a large room has three doors and that a switch near each door controls a light in the room. It has to be possible to turn the light on or off by changing the state of any one of the switches.*





Problem 1.11 (20 marks, Exercise 2.38 [1]). *An M-bit thermometer code for the number k consists of k 1's in the least significant bit positions and M – k 0's in all the more significant bit positions. A binary-to-thermometer code converter has N inputs and* 2^{N-1} *outputs. It produces a* 2^N-1 *bit thermometer code for the number specified by the input. For example, if the input is 110, the output should be 0111111. Design a 3:7 binary-to-thermometer code converter. Give a simplified Boolean equation for each output.*