Combinational circuit

Vikas Dhiman for ECE275

September 7, 2022

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.

① AND Gate

In C

```
bool x1;
bool x2;
```

$L = x1 \;\&\&\; x2$

$L = x1 \;\&\; x2$

$L = x1' \;\&\; x2$

SystemVerilog

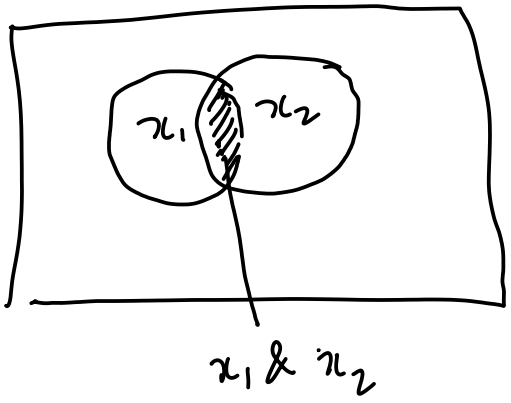
0 = false
1 = true

x1	x2	L
0	0	0
0	1	0
1	0	0
1	1	1

Truth table

x_1 = Blueberry
x_2 = Pecan

Venn Diagram



$\{0, 1\}$

$+$ = OR gate

$\cdot$ Boolean algebra

$\circ$ = AND gate

$0+0=0$

$0+1=1$

$1+0=1$

$1+1=1$

$0 \cdot 0 = 0$

$0 \cdot 1 = 0$

$1 \cdot 0 = 0$

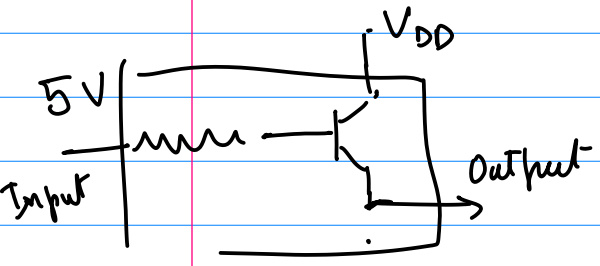
$1 \cdot 1 = 1$

AND gate $L = x_1 \cdot x_2 = x_1 x_2$

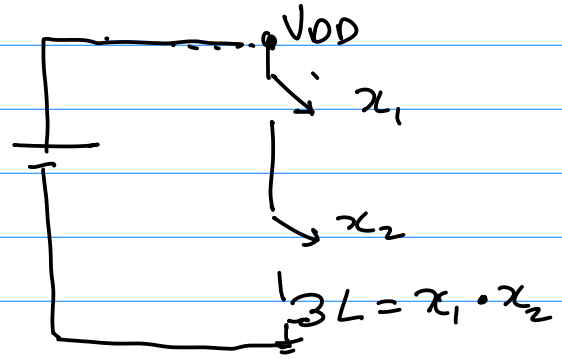
Switching circuit

$0 = GND \leq V_{IL}, V_{OL}$

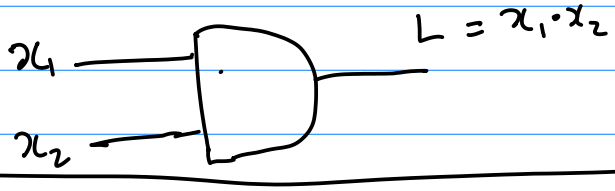
$1 = V_{DD}\ (5V, 3.5V) \geq V_{IH}, V_{OH}$



AND gate from switches?



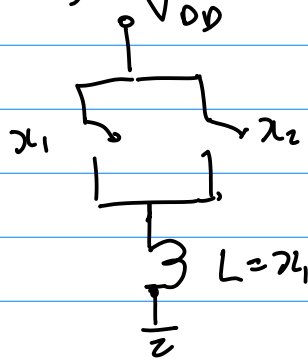
ANSI symbol



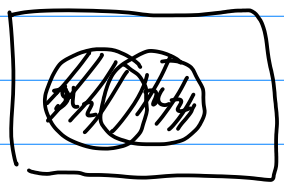
OR gate Truth table

x_1	x_2	$L = x_1 + x_2$
0	0	0
0	1	1
1	0	1
1	1	1

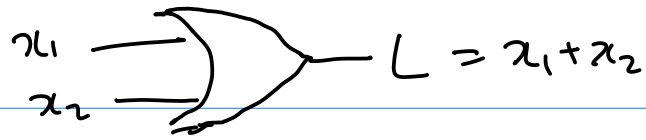
Switching circuit



Venn diagram



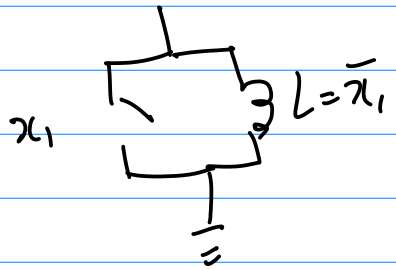
ANSI Symbol

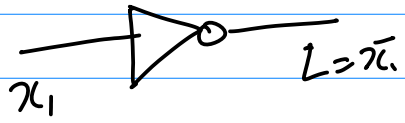


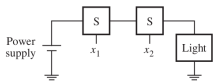
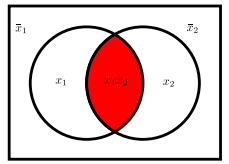
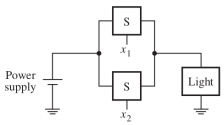
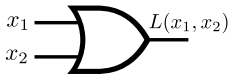
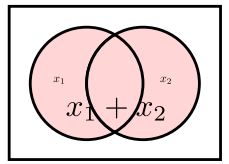
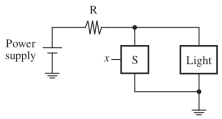
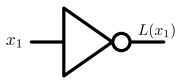
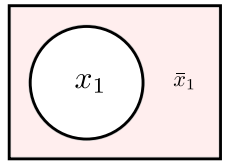
NOT gate

x_1	$L = \bar{x}_1 = x_1'$
0	1
1	0

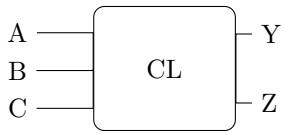
Truth Table





Name	C/Verilog	Boolean expr.	Truth Table	Switching circuit	(ANSI) symbol	Venn diagram
AND Gate	`L = x1 & x2`	$L = x_1 \cdot x_2 = x_1 x_2$	x_1 x_2 | $x_1 \cdot x_2$ 0 0 | 0 0 1 | 0 1 0 | 0 1 1 | 1			
OR Gate	`L = x1 | x2`	$L = x_1 + x_2$	x_1 x_2 | $x_1 + x_2$ 0 0 | 0 0 1 | 1 1 0 | 1 1 1 | 1			
NOT Gate	`L = ~ x1`	$L = \bar{x}_1 = x_1'$	x_1 | $\bar{x}_1$ 0 | 0 0 | 1			

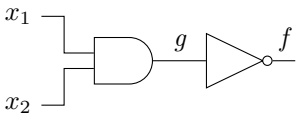
3 Digital circuits or networks



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

4 Two input networks

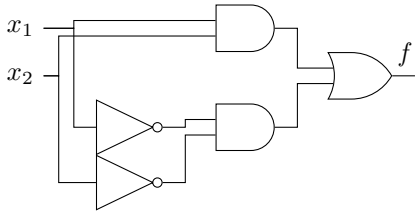
Example 1. *Convert the following (ANSI) network into a Boolean expression, a truth table and a Venn diagram.*



Example 2. *Convert the following Boolean expression into a (ANSI) network, a truth table and a Venn diagram:*

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

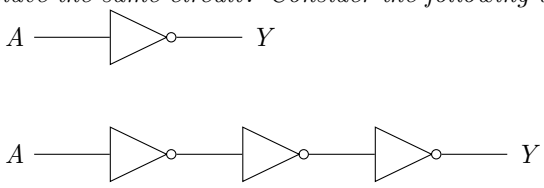
Problem 1. *Convert the following (ANSI) network into a Boolean expression, a truth table and a Venn diagram.*



Example 3. *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 2. *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*

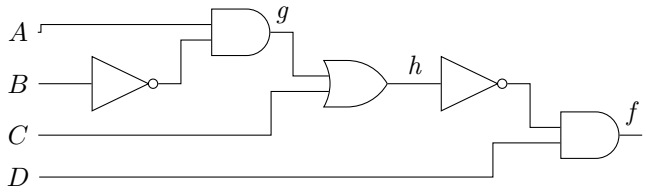


How about Venn digrams?

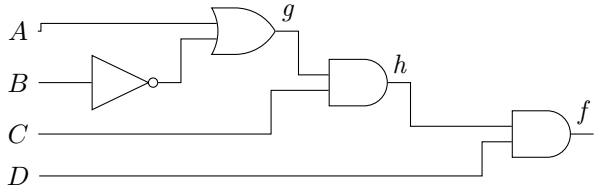
Remark 1. *Truth tables and Venn diagrams define* what *the combinational circuit should do. Truth tables define output for every input. Boolean expression and networks define* how *to achieve the desired input output relationship.*

5 Multi-input networks

Problem 3. *Convert the following (ANSI) network into a Boolean expression and a truth table.*



Problem 4. *Convert the following (ANSI) network into a Boolean expression and a truth table.*



6 Minterms and Maxterms

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 4. *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 5. *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*

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 $Y = (A + B)(\bar{A} + B) = M_0 M_2$.
Writing a functional specification in terms of minterms is also called product of sums canonical form.

Example 5. *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*

Example 6.

Problem 6. *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*

7 Karnaugh maps

7.1 Two input K-maps

B \ A	0	1
0	m_0	m_2
1	m_1	m_3

7.2 Three input K-maps

C \ AB	00	01	11	10
0	m_0	m_2	m_6	m_4
1	m_1	m_3	m_7	m_5

7.3 Four input K-maps

CD \ AB	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

7.4 Five input K-maps

A = 0

DE \ BC	00	01	11	10
00	m_0	m_4	m_{12}	m_8
01	m_1	m_5	m_{13}	m_9
11	m_3	m_7	m_{15}	m_{11}
10	m_2	m_6	m_{14}	m_{10}

A = 1

DE \ BC	00	01	11	10
00	m_{16}	m_{20}	m_{28}	m_{24}
01	m_{17}	m_{21}	m_{29}	m_{25}
11	m_{19}	m_{23}	m_{31}	m_{27}
10	m_{18}	m_{22}	m_{30}	m_{26}