Combinational circuit

Vikas Dhiman for ECE275

September 8, 2023

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.

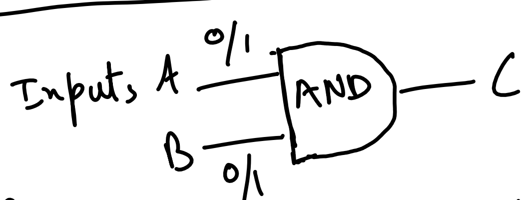
Building blocks: Gates

Basic gates

1. AND gate
2. OR gate
3. NOT gate

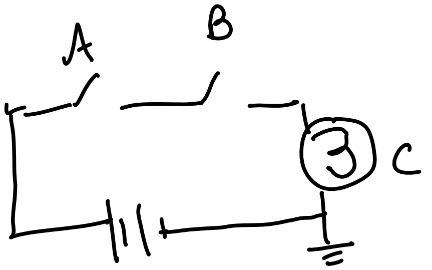
("Grass is green" AND "sky is blue") is true iff both statements are true

---



ANSI symbol

$$C = \underbrace{A \cdot B}_{\text{Boolean algebra}} = \underbrace{A \,\&\, B}_{\text{C/verilog}}$$



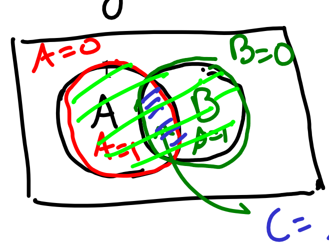
Truth table

A	B	C
0	0	0
0	1	0
1	0	0
1	1	1

0 = False
1 = True

In a circuit AND gate ≡ Switches in Series

Venn diagram

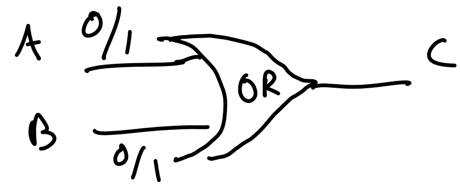


$C = A + B$

$C = A \cdot B$

OR gate

① ANSI symbol



② Boolean algebra symbol $C = \underbrace{A + B}_{\text{Boolian algebra}} = \underbrace{A \mid B}_{\text{C/Verilog}}$

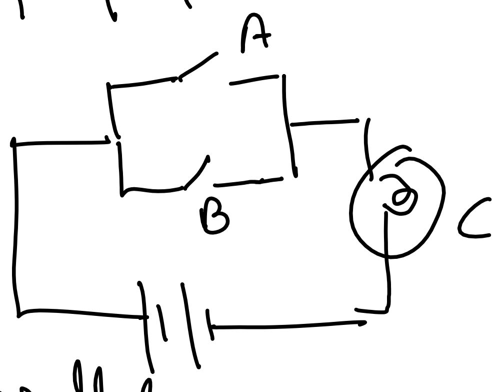
Diff from binary addition

③ Truth table

A	B	C
0	0	0
0	1	1
1	0	1
1	1	1

0 = False
1 = True

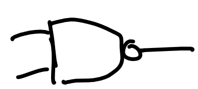
④ Switching circuit



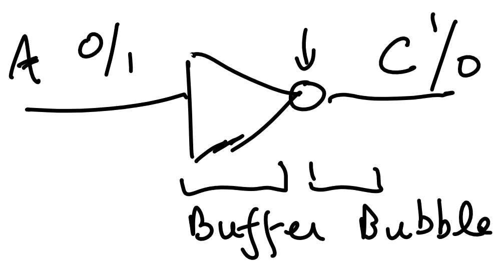
In circuit, OR gate

≡ Switches in parallel

NOT gate



① ANSI symbol



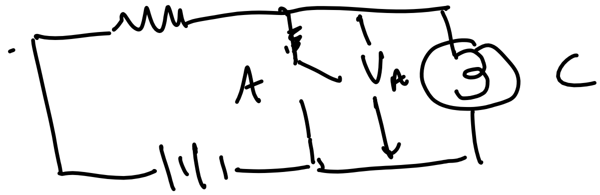
② Boolean algebra symbol $C = \bar{A}$
$= A'$

$\underbrace{C = \sim A}_{\text{C/Verilog}}$

③ Truth table

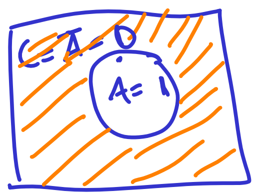
A	C
0	1
1	0

(4) Switching circuit for NOT gate

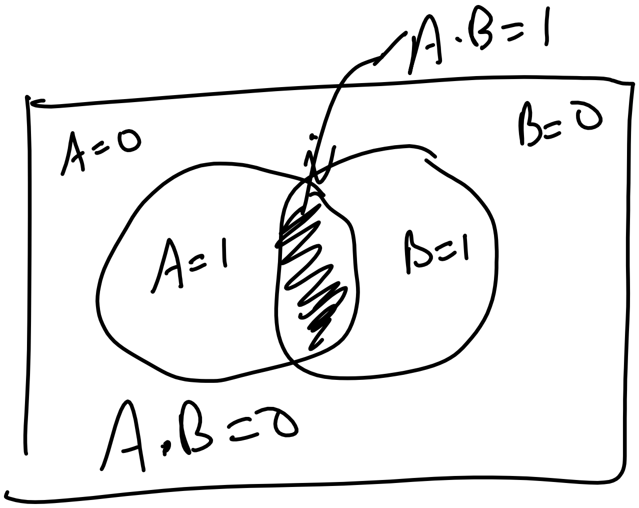


$C = \bar{A}$

Venn Diagram for not gate

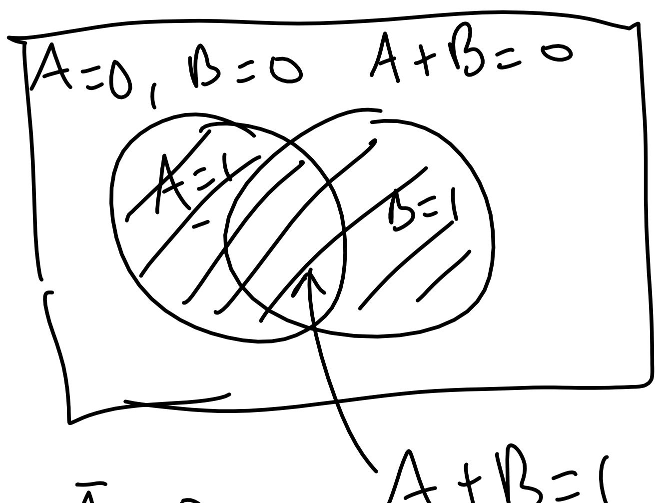


AND gate



A	B	$C = A \cdot B$
0	0	0
0	1	0
1	0	0
1	1	1

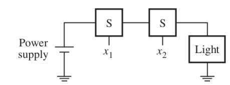
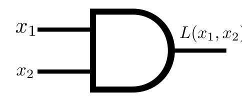
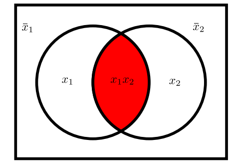
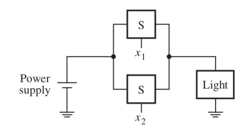
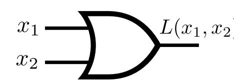
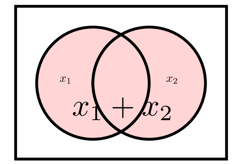
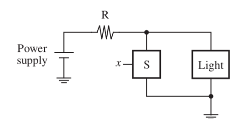
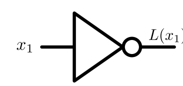
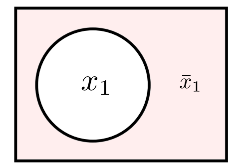
OR gate



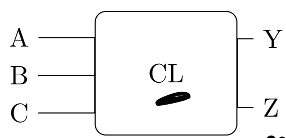
A	B	A+B
0	0	0
0	1	1
1	0	1
1	1	1

Not gate

2 Basic Gates and notations summary

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

3 Digital circuits or networks



CL = Combinational Logic

Inputs on Left

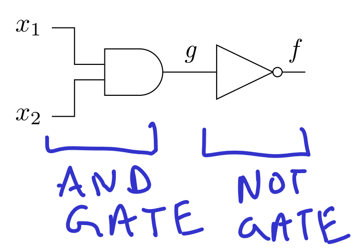
Output on the right

function notation

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

4 Two input networks

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



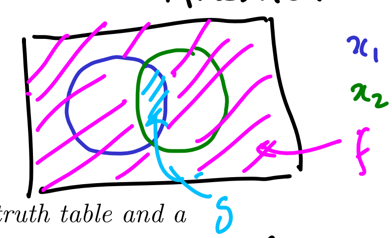
$g = x_1 \cdot x_2$

$f = \overline{g} = g'$

$\Rightarrow f = \overline{x_1 \cdot x_2} = (x_1 \cdot x_2)'$ — Boolean expression

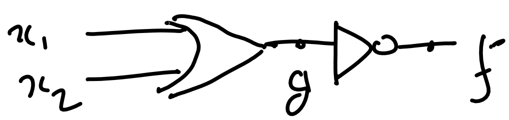
Truth table

x_1	x_2	g	f
0	0	0	1
0	1	0	1
1	0	0	1
1	1	1	0

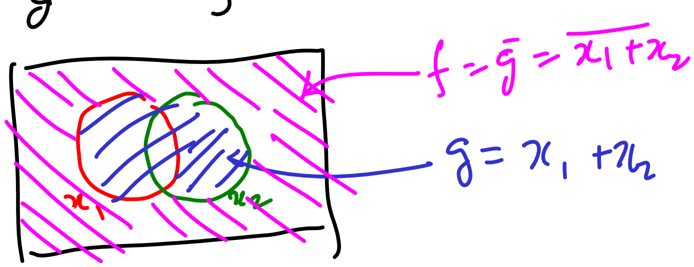


Example 2. *Convert the following Boolean expression into a (ANSI) network, a truth table and a Venn diagram:*

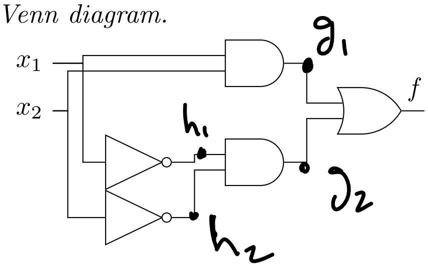
$$f = \overline{x_1 + x_2}$$



x_1	x_2	g	f
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0



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



$g_1 = ? = x_1 \cdot x_2$

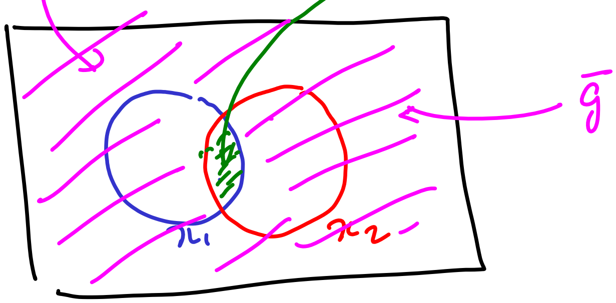
$h_1 = \overline{x_1}$

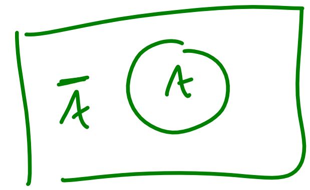
$h_2 = \overline{x_2}$

$g_2 = h_1 \cdot h_2 = \overline{x_1} \cdot \overline{x_2}$

$f = g_1 + g_2 = (x_1 \cdot x_2) + (\overline{x_1} \cdot \overline{x_2})$

$f = \overline{x_1 \cdot x_2}$ $\quad g = x_1 \cdot x_2$

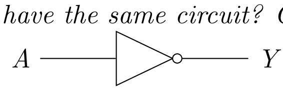




Example 3. *Convert the following Boolean expression into a network, a truth table and a Venn diagram:*

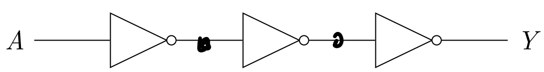
$$f = x_1\bar{x}_2 + \bar{x}_1 x_2$$

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*



$= \bar{A}$

$Y = \bar{A}$



$Y = \bar{\bar{\bar{A}}}$

How about Venn digrams?

g_1 g_2

$g_1 = \bar{A}$

$g_2 = \bar{g}_1 = \bar{\bar{A}} = A$

$y = \bar{g}_2 = \bar{A}$

What

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.*

How

circuit

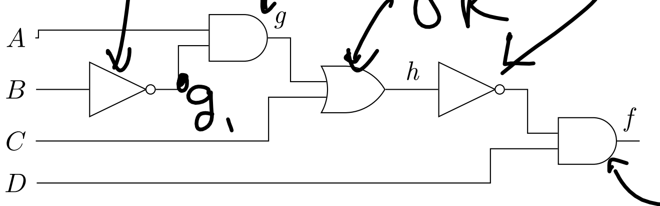
NOT

NOT

5 Multi-input networks

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

$n=4$



$g = ? = A \cdot \overline{B}$

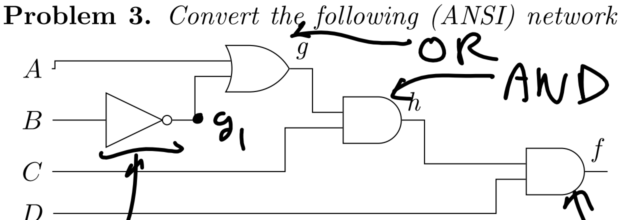
$h = ? = g + C = A \cdot \overline{B} + C$

$f = \overline{h} \cdot D = \overline{(A \cdot \overline{B} + C)} \cdot D$

	A	B	C	D	g	h	f
0	0	0	0	0			
1	0	0	0	1			
2	0	0	1	0			
3	0	0	1	1			
4	⋮						
5							
⋮							
15							

\# of row in Truth table $= 2^n = 2^4 = 16$

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



2 min

$g_1 = \overline{B}$

$g = A + g_1 = A + \overline{B}$

$h = g \cdot C = (A + \overline{B}) \cdot C$

$f = h \cdot D = ((A + \overline{B}) \cdot C) \cdot D$

$= (A + \overline{B}) \cdot C \cdot D$

Row num	A	B	C	D	$(A+\bar{B})\cdot C\cdot D = f$
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	1
4	0	1	0	0	0
5	0	1	0	1	0
6	0	1	1	0	0
7	0	1	1	1	0
8	1	0	0	0	0
9	1	0	0	1	0
10	1	0	1	0	0
11	1	0	1	1	1
12	1	1	0	0	0
13	1	1	0	1	0
14	1	1	1	0	0
15	1	1	1	1	1

summation

$$f(A,B,C,D) = m_3 + m_{11} + m_{15} = \sum m(3,11,15)$$

MINTERMS

$$m_3 = \begin{cases} 1 & \text{iff for row } \underline{3} \\ 0 & \text{otherwise} \end{cases}$$

for Row 3

$A = 0 \quad \bar{A} = 1$

$B = 0 \quad \bar{B} = 1$

$C = 1 \quad C = 1$

$D = 1 \quad D = 1$

$$m_3 = (A + \bar{B}) \cdot C \cdot D$$

$$m_3 = \bar{A} \cdot \bar{B} \cdot C \cdot D$$

3th minterm

$$= \begin{cases} 1 & \text{for row 3} \\ 0 & \text{otherwise} \end{cases}$$

$$A \bar{B} C D$$

$$11_{10} = (1011)_2$$

$$m_{11} = A \cdot \bar{B} \cdot C \cdot D$$

11th Minterm

$$= \begin{cases} 1 & \text{for row 11} \\ 0 & \text{otherwise} \end{cases}$$

$$15_{10} = (1111)_2$$

$$m_{15} = A \cdot B \cdot C \cdot D$$

Row num	A	B	C	D	$\overline{(A+\overline{B})\cdot C\cdot D} = f = p$
0	0	0	0	0	1
1	0	0	0	1	1
2	0	0	1	0	1
3	0	0	1	1	0
4	0	1	0	0	1
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1
10	1	0	1	0	1
11	1	0	1	1	0
12	1	1	0	0	1
13	1	1	0	1	1
14	1	1	1	0	1
15	1	1	1	1	0

$$f(A,B,C,D) = \Sigma m(0, 1, 2 \ldots\ldots, 14)$$

$$f(A,B,C,D) = M_3 \cdot M_{11} \cdot M_{15} = \prod M(3, 11, 15)$$

Maxterm

$$M_3 = \begin{cases} 0 & \text{for row 3} \\ 1 & \text{otherwise} \end{cases}$$

$$M_3 = \overline{(\bar{A} \cdot \bar{B} \cdot C \cdot D)} \qquad 3_{10} = (0011)_2$$

$$= (A + B + \bar{C} + \bar{D})$$

$$M_{11} = \bar{A} + B + \bar{C} + \bar{D} \qquad 11_{10} = (1011)_2$$

$$= \overline{(A \cdot \bar{B} \cdot C \cdot D)}$$

$$M_{15} = (\bar{A} + \bar{B} + \bar{C} + \bar{D}) \qquad 15_{10} = (1111)_2$$

$$= \overline{(A \cdot B \cdot C \cdot D)}$$

maxterms

Truth tables is that they are cumbersome to write

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

Problem 4. *What is the minterm m_{23} for a 5-input circuit with inputs a, b, c, d, e (ordered from MSB to LSB).*

Example 6. *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 7. *Write the maxterm M_{11} for 4-input Boolean function with the ordered inputs A, B, C, D.*

Example 8. *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 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*

Example 9. *Write the 3-input truth table for the function* $f = m_2 + m_3 + m_7$.

Problem 7. *Write the 3-input truth table for the function* $f = M_4 M_5 M_7$.

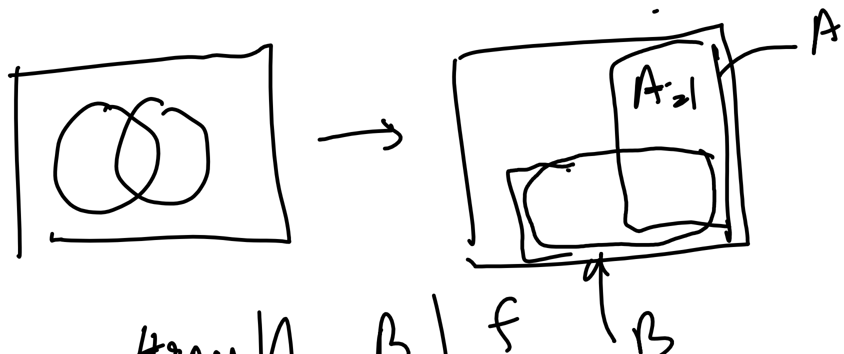
Problem 8. *Write the truth table for the function* $f = \bar{A}B\bar{C} + AB\bar{C}$.

Venn Diagram + Truth Table

7 Karnaugh maps

7.1 Two input K-maps

B \ A	0	1
0	m_0 0	m_2 2
1	m_1 1	m_3 3



#row	A	B	f
0	0	0	
1	0	1	
2	1	0	
3	1	1	

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

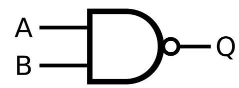
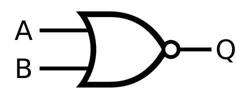
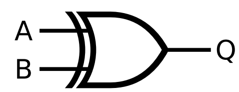
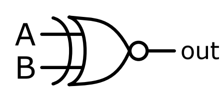
A = 0

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

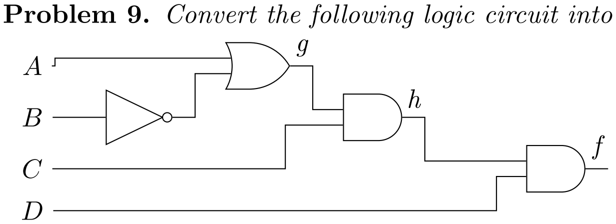
DE \ BC	00	01	11	10
00	m_{16}	m_{20}	m_{28}	m_{24}
01	m_{17}	m_{21}	m_{29}	m_{25}
11	m_{19}	m_{23}	m_{31}	m_{27}
10	m_{18}	m_{22}	m_{30}	m_{26}

8 More Gates and notations summary

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

Example 10. *Convert the following Boolean expression into a K-map.* $f = \overline{A\bar{B} + CD}$

Problem 9. *Convert the following logic circuit into a K-map.*



9 Boolean Algebra

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 $x \neq 0$

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 2 (Duality). *Swap + with $\cdot$ and 0 with 1 to get another theorem*

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:

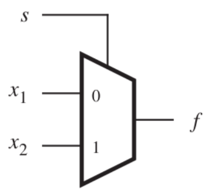
   (a) $x + \bar{x} \cdot y = x + y$

   (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 11 (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 12. *Simplify $f = \bar{A}\bar{B}\bar{C} + A\bar{B}\bar{C} + A\bar{B}\bar{C}$ using boolean algebra.*

Example 13. *Simplify $f = \bar{A}\bar{A}\bar{C} + \bar{A}\bar{B}C$ using K-maps.*