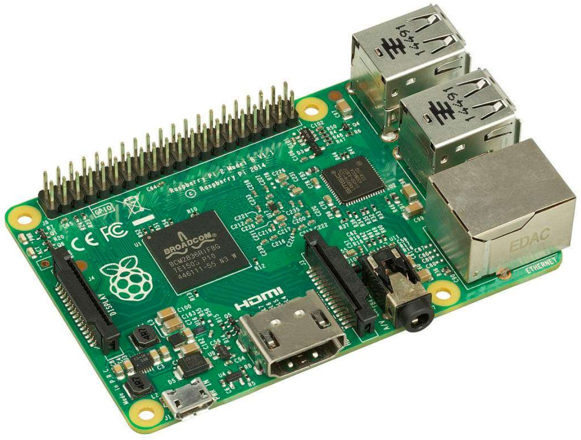
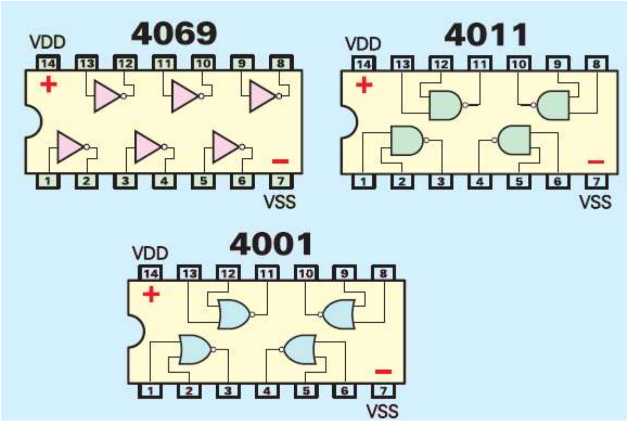
BOOLEAN LOGIC GATES

Chapter 10

INTRODUCTION

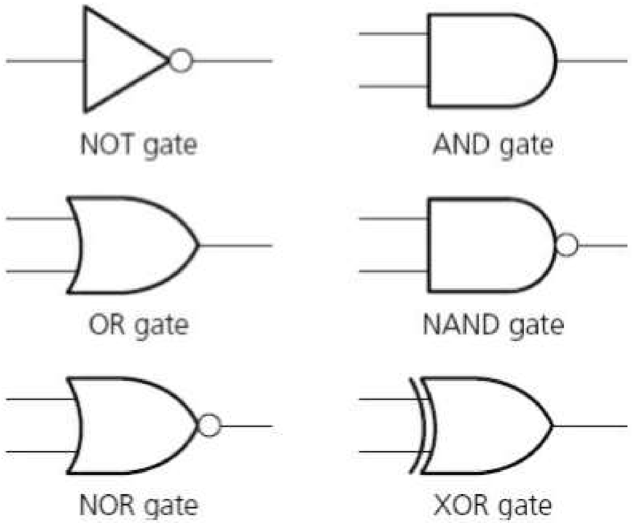
- electronic circuits, memory devices and controlling devices are made of thousands of logic gates



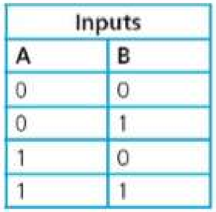


INTRODUCTION

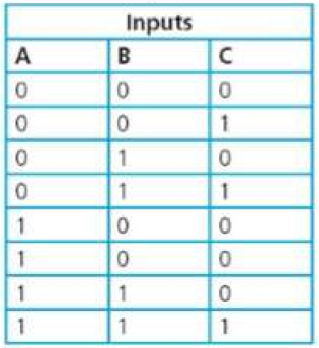
- types of logic gates



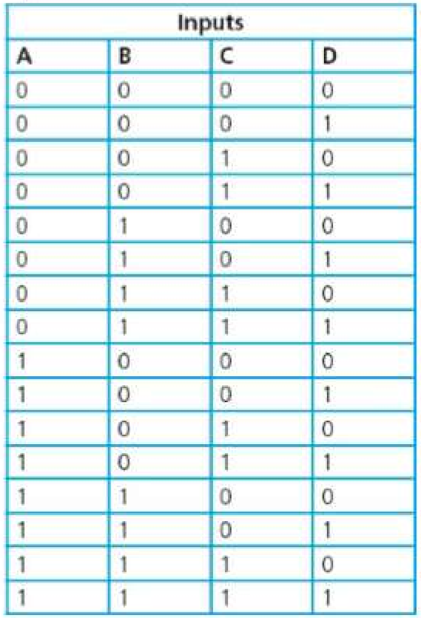
- truth table
  - to trace the output from a logic circuit



Inputs	
A	B
0	0
0	1
1	0
1	1



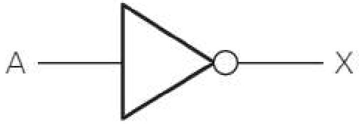
Inputs		
A	B	C
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	0
1	1	0
1	1	1



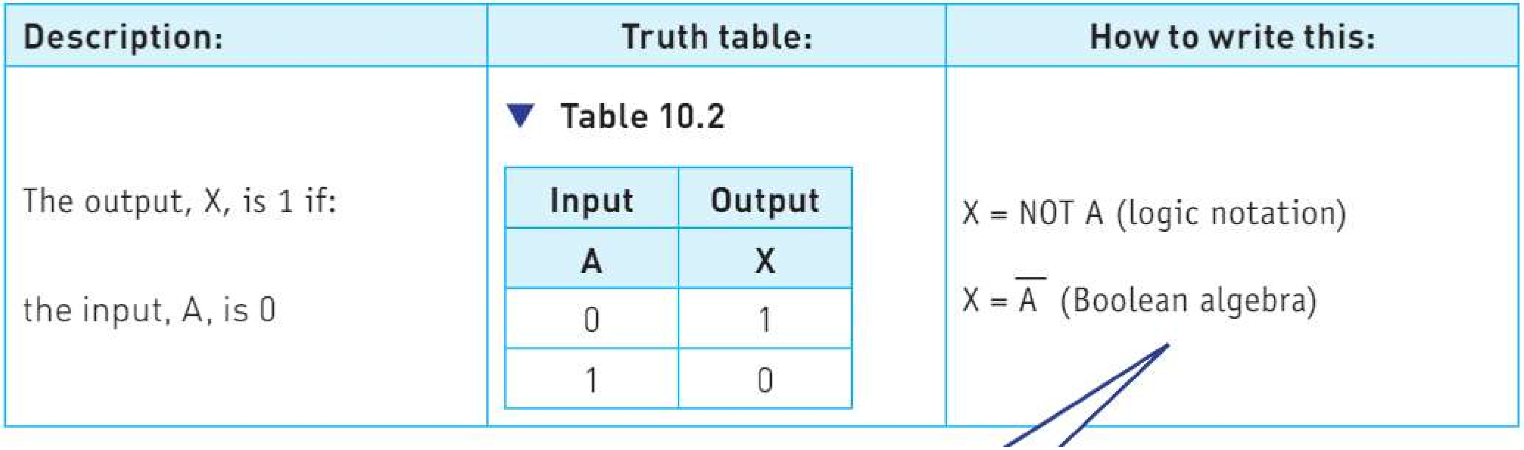
Inputs			
A	B	C	D
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1
0	1	0	0
0	1	0	1
0	1	1	0
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

NOT GATE

- example, NOT A



▲ **Figure 10.2**

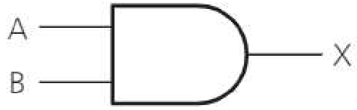


Description:	Truth table:	How to write this:
The output, X, is 1 if: the input, A, is 0	▼ **Table 10.2**	X = NOT A (logic notation) $X = \overline{A}$ (Boolean algebra)

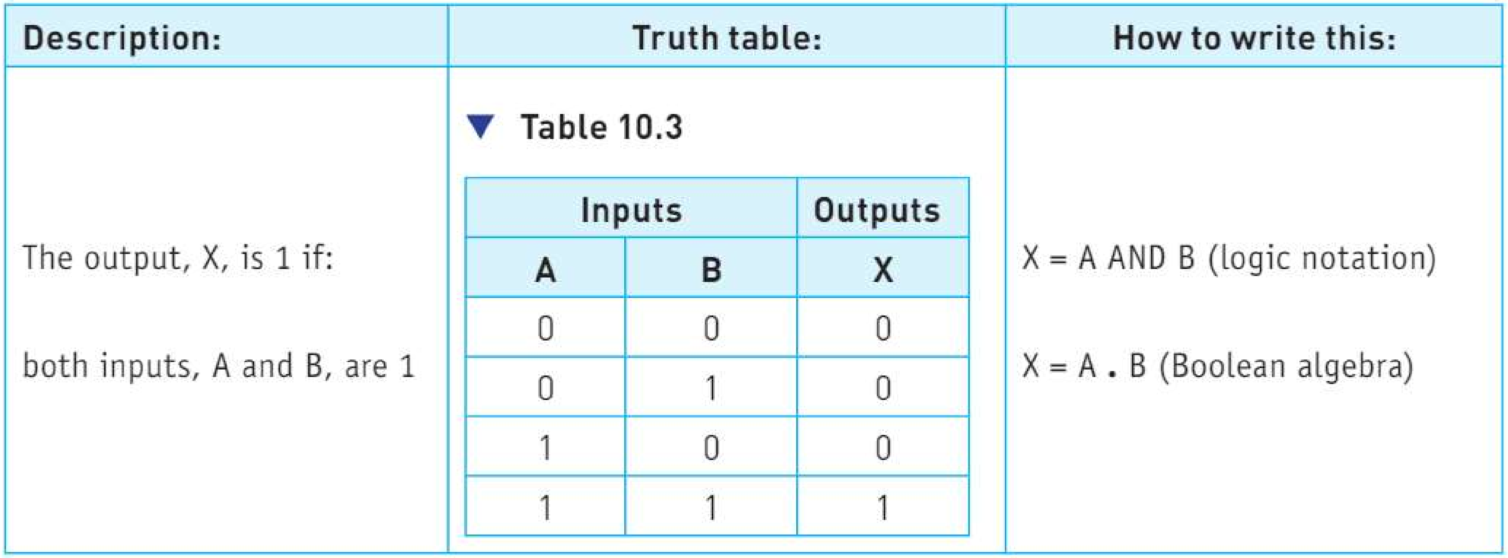
Input	Output
A	X
0	1
1	0

AND GATE

- example, A AND B



▲ **Figure 10.3**



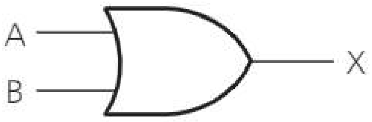
<table>
<tr><th>Description:</th><th>Truth table:</th><th>How to write this:</th></tr>
<tr>
<td>The output, X, is 1 if:

both inputs, A and B, are 1</td>
<td>▼ Table 10.3
<table>
<tr><th colspan="2">Inputs</th><th>Outputs</th></tr>
<tr><th>A</th><th>B</th><th>X</th></tr>
<tr><td>0</td><td>0</td><td>0</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>X = A AND B (logic notation)

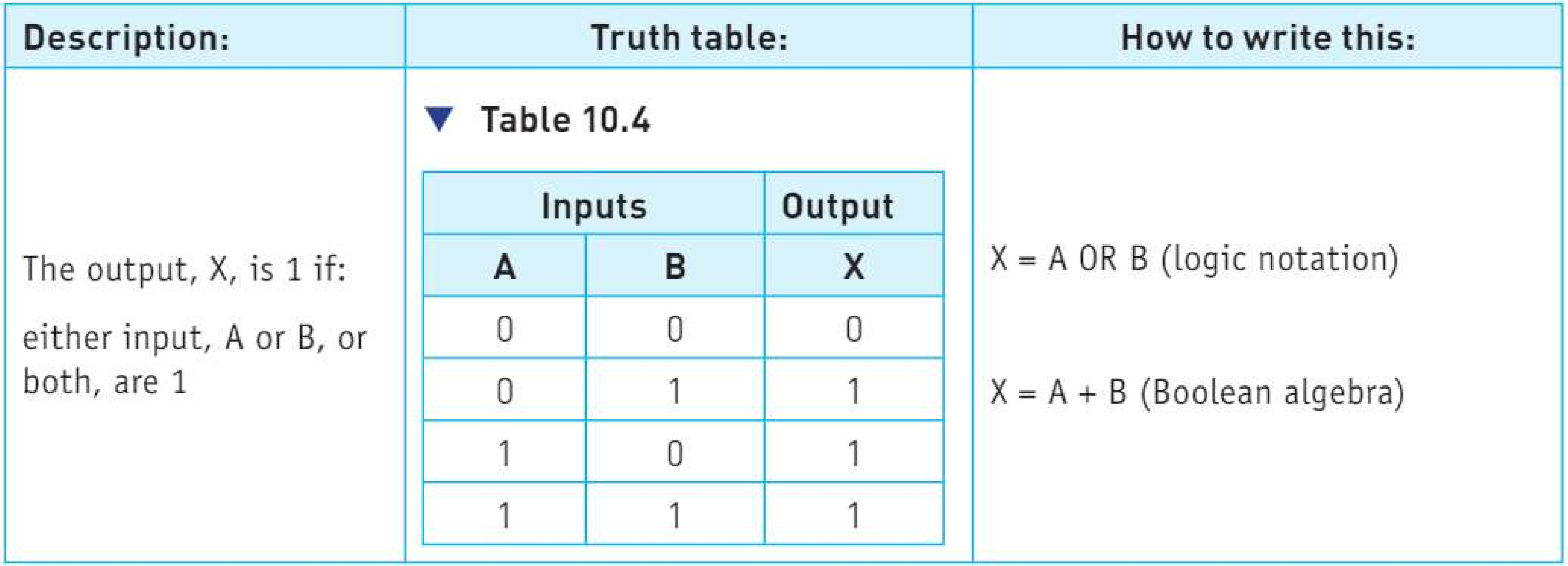
X = A . B (Boolean algebra)</td>
</tr>
</table>

OR GATE

- example, A OR B



▲ **Figure 10.4**



<table>
<tr><th>Description:</th><th>Truth table:</th><th>How to write this:</th></tr>
<tr>
<td>The output, X, is 1 if:
either input, A or B, or both, are 1</td>
<td>▼ Table 10.4
<table>
<tr><th colspan="2">Inputs</th><th>Output</th></tr>
<tr><th>A</th><th>B</th><th>X</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>1</td></tr>
</table>
</td>
<td>X = A OR B (logic notation)
X = A + B (Boolean algebra)</td>
</tr>
</table>

NAND GATE (NOT AND)

- example, A NAND B

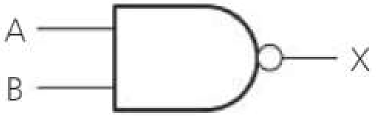
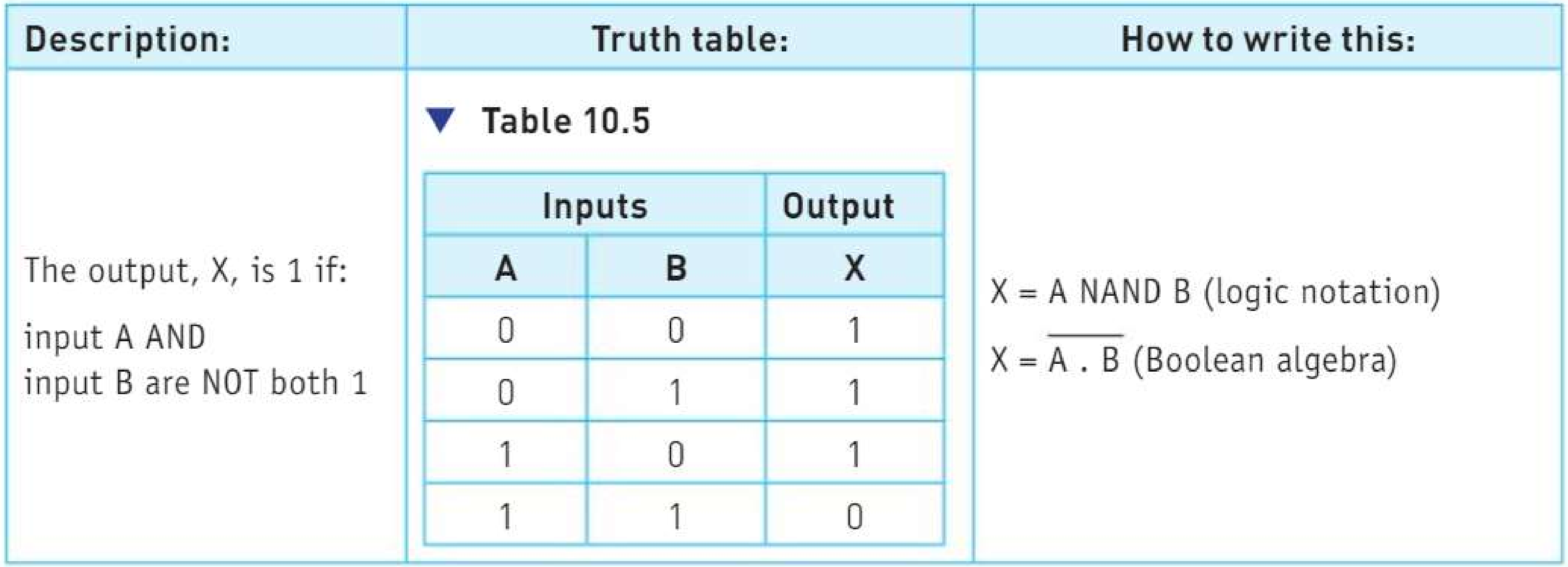


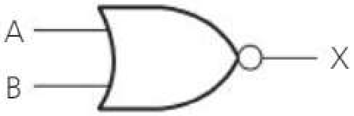
Figure 10.5



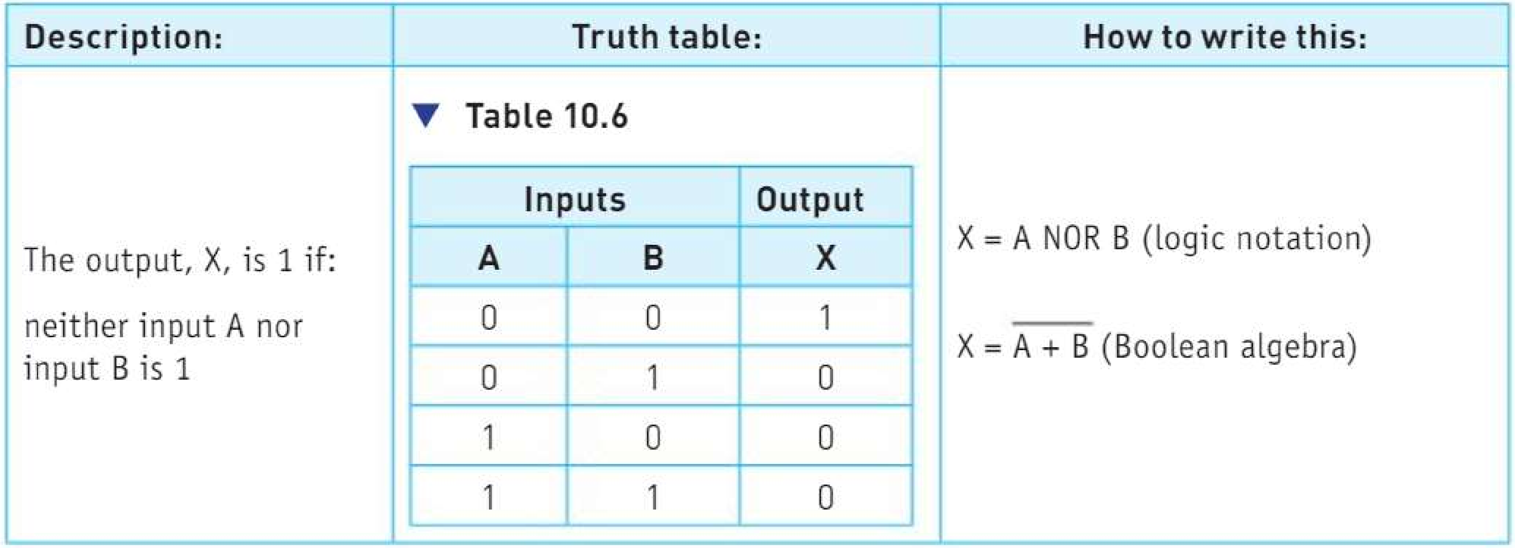
<table>
<tr><th>Description:</th><th>Truth table:</th><th>How to write this:</th></tr>
<tr>
<td>The output, X, is 1 if:
input A AND
input B are NOT both 1</td>
<td>Table 10.5
<table>
<tr><th colspan="2">Inputs</th><th>Output</th></tr>
<tr><th>A</th><th>B</th><th>X</th></tr>
<tr><td>0</td><td>0</td><td>1</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>X = A NAND B (logic notation)
$X = \overline{A . B}$ (Boolean algebra)</td>
</tr>
</table>

NOR GATE (NOT OR)

- example, A NOT OR B



▲ Figure 10.6

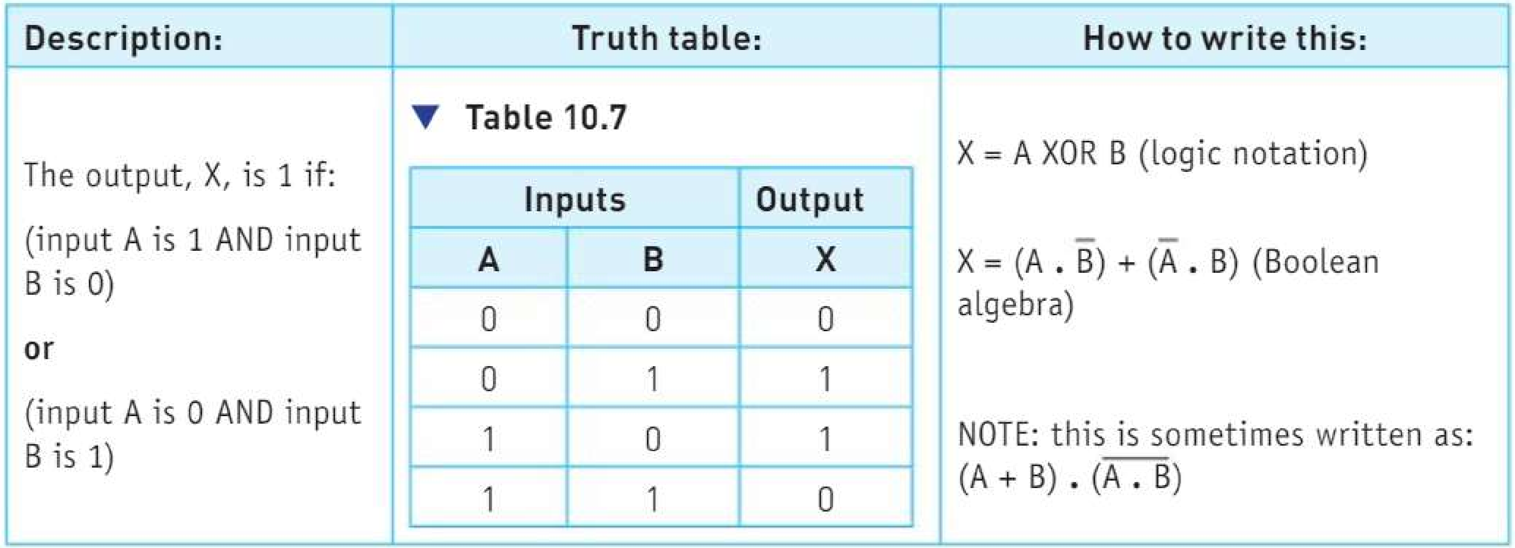


<table>
<tr><th>Description:</th><th>Truth table:</th><th>How to write this:</th></tr>
<tr>
<td>The output, X, is 1 if:
neither input A nor input B is 1</td>
<td>▼ Table 10.6
<table>
<tr><th colspan="2">Inputs</th><th>Output</th></tr>
<tr><th>A</th><th>B</th><th>X</th></tr>
<tr><td>0</td><td>0</td><td>1</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>X = A NOR B (logic notation)
$X = \overline{A + B}$ (Boolean algebra)</td>
</tr>
</table>

XOR GATE (EXCLUSIVE OR)

- example, A XOR B

▲ Figure 10.7



Description:

The output, X, is 1 if:

(input A is 1 AND input B is 0)

or

(input A is 0 AND input B is 1)

Truth table:

▼ **Table 10.7**

Inputs		Output
A	**B**	**X**
0	0	0
0	1	1
1	0	1
1	1	0

How to write this:

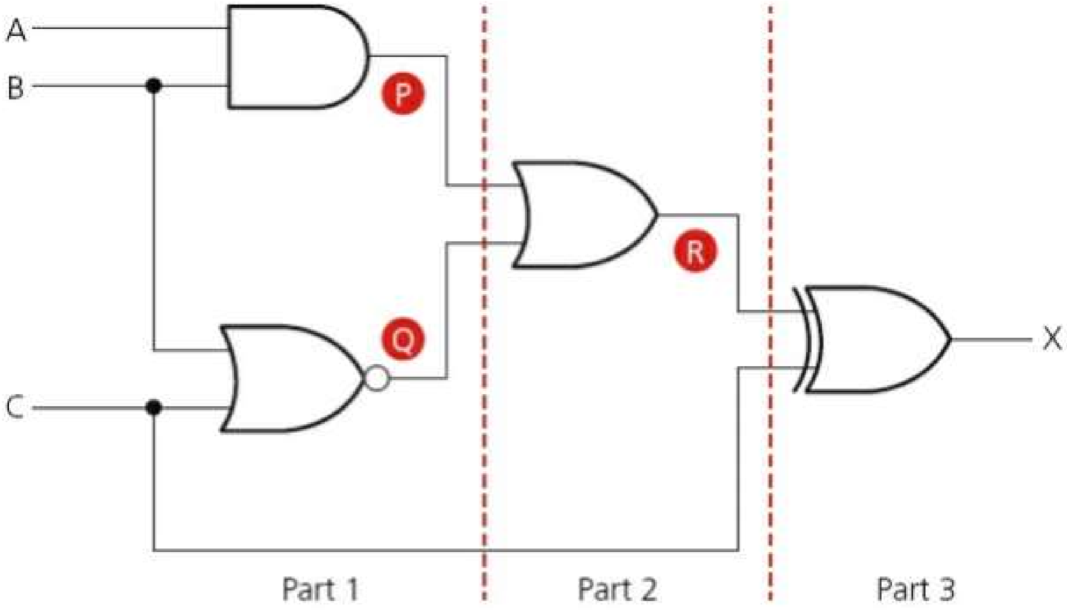
X = A XOR B (logic notation)

$X = (A \cdot \bar{B}) + (\bar{A} \cdot B)$ (Boolean algebra)

NOTE: this is sometimes written as: $(A + B) \cdot (\overline{A \cdot B})$

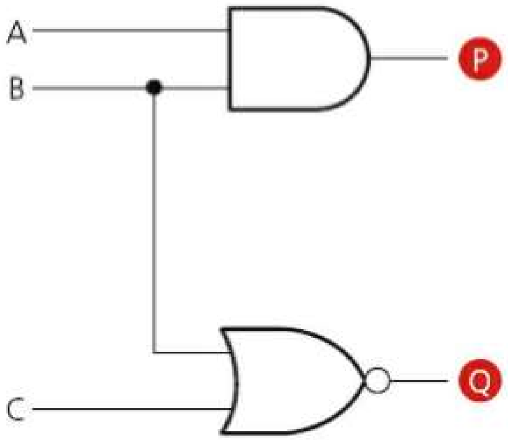
LOGIC CIRCUIT (TYPE I)

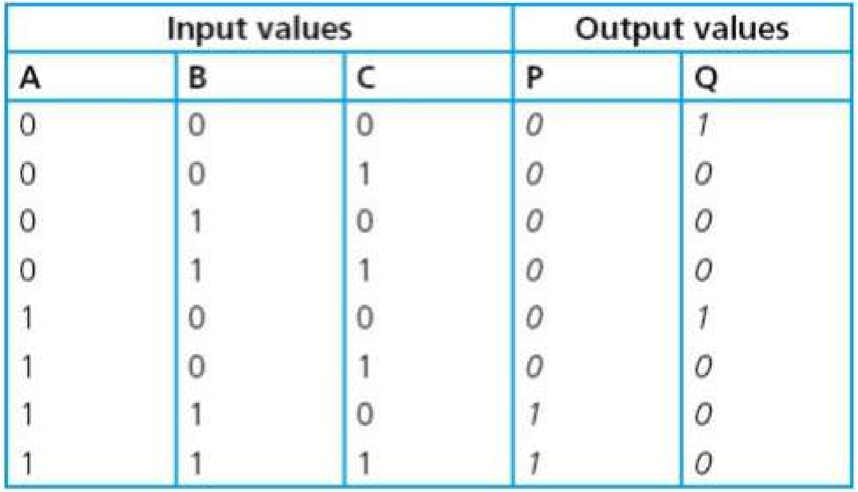
- Produce truth table from this logic circuit



LOGIC CIRCUIT (TYPE I)

- Part one





Input values			Output values	
A	B	C	P	Q
0	0	0	0	1
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	1
1	0	1	0	0
1	1	0	1	0
1	1	1	1	0

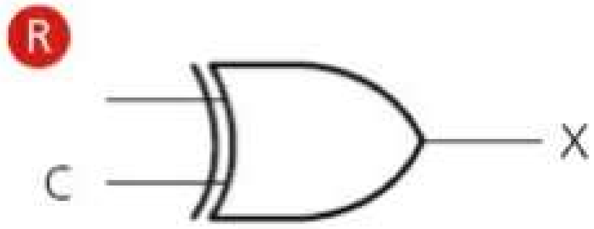
LOGIC CIRCUIT (TYPE I)

- Part two

Input		Output
P	Q	R
0	1	1
0	0	0
0	0	0
0	0	0
0	1	1
0	0	0
1	0	1
1	0	1

LOGIC CIRCUIT (TYPE I)

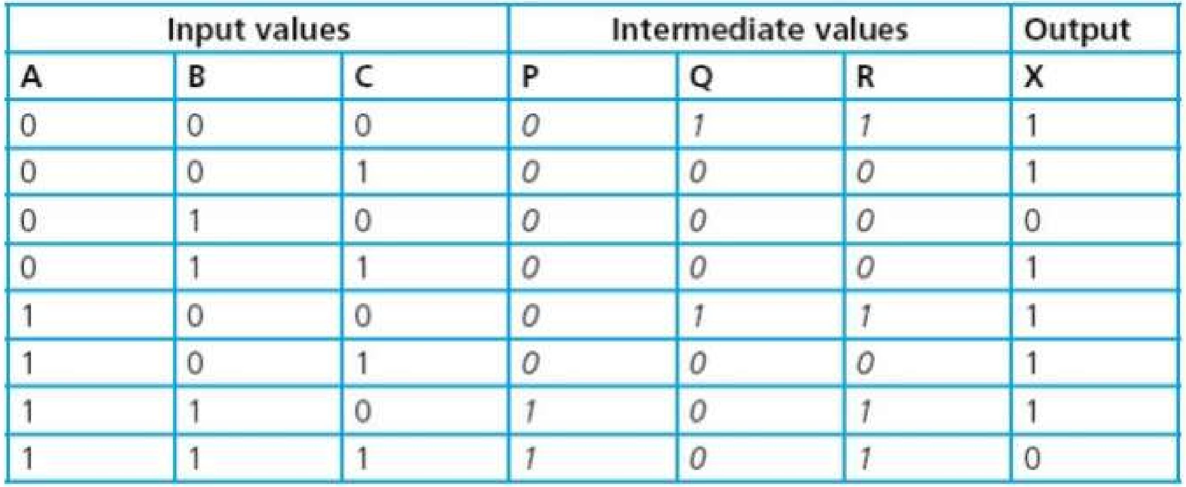
- Part three



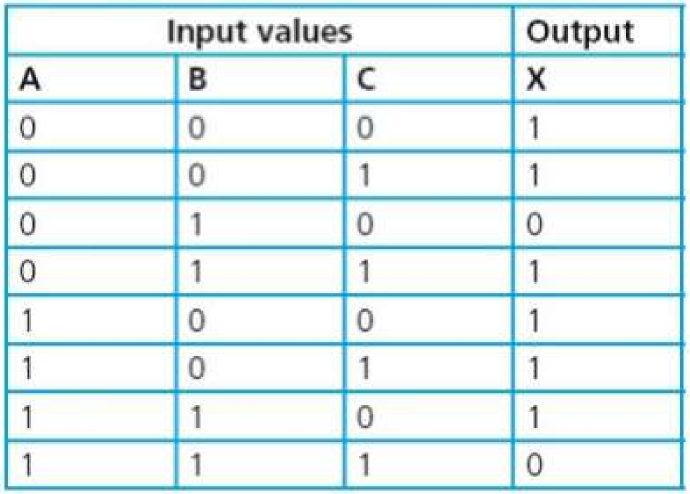
Inputs		Output
R	**C**	**X**
1	0	*1*
0	1	*1*
0	0	*0*
0	1	*1*
1	0	*1*
0	1	*1*
1	0	*1*
1	1	*0*

LOGIC CIRCUIT (TYPE I)

- Truth table

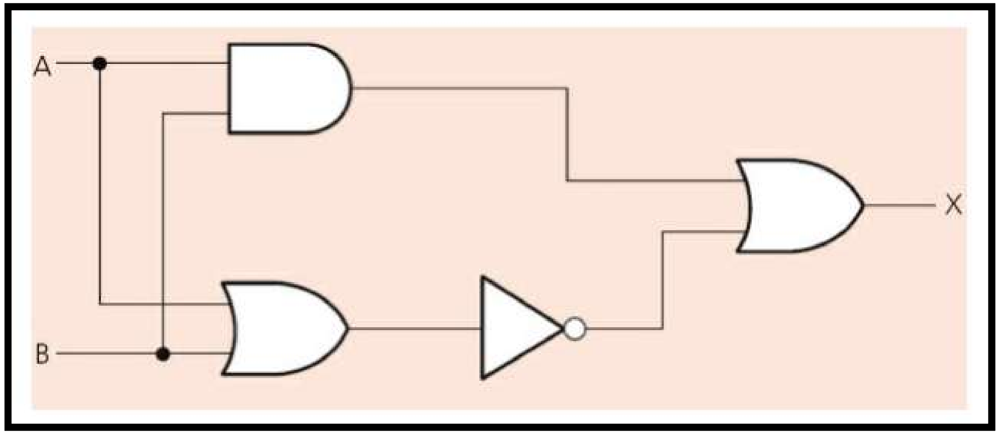


Input values			Intermediate values			Output
A	B	C	P	Q	R	X
0	0	0	0	1	1	1
0	0	1	0	0	0	1
0	1	0	0	0	0	0
0	1	1	0	0	0	1
1	0	0	0	1	1	1
1	0	1	0	0	0	1
1	1	0	1	0	1	1
1	1	1	1	0	1	0

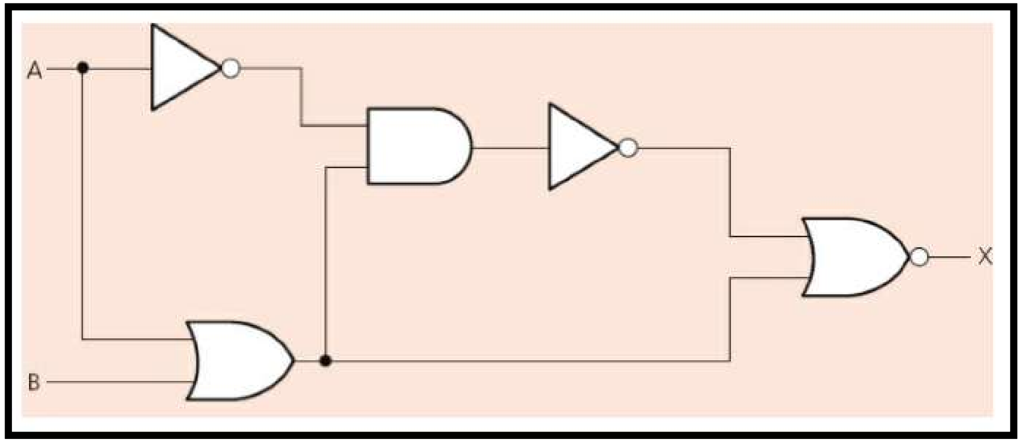


Input values			Output
A	B	C	X
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

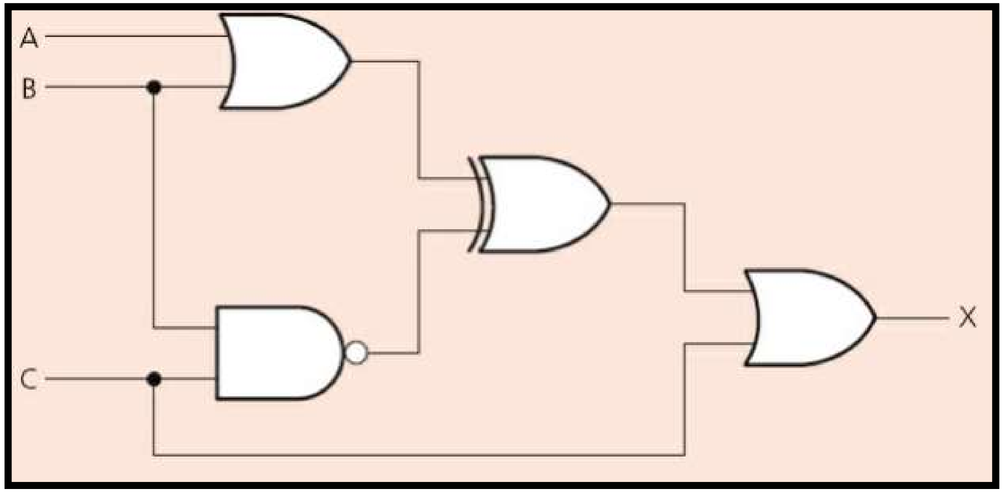
LOGIC CIRCUIT (TYPE I) EXERCISE



LOGIC CIRCUIT (TYPE I) EXERCISE



LOGIC CIRCUIT (TYPE I) EXERCISE



LOGIC CIRCUIT (TYPE I) EXERCISE