Summary

A. Number systems:
decimal, binary, hexadecimal, and octal

B. Subtraction:
Subtraction was converted to addition by introducing two's-complement.

C. To obtain two's-complement:

1) Convert 1's to 0's and 0's to 1's
2) Add 1 to the result

Boolean Algebra

The algebra deals with binary variables and logic operations

Variables : A, B, C, ...
Functions : F, G, H, ...
Values(variables or functions): 0, 1
Operations : AND(.), OR(+), NOT('), XOR(⊕)

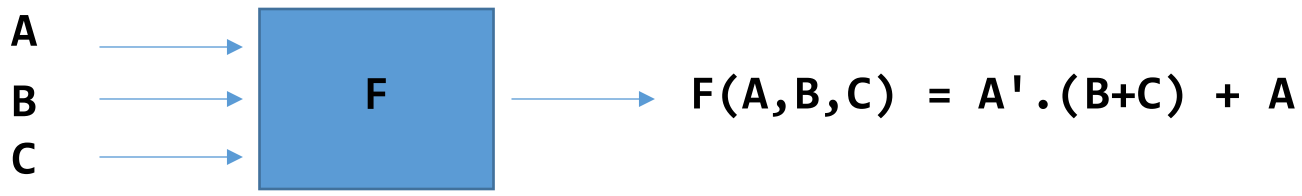
Boolean Algebra

Example : Operations and variables

A.B, A+B, A⊕B, A.(B⊕C), A + 1, A + 0

Example : Function

F(A,B,C) = A'.(B+C) + A



Boolean Algebra

Table of truth: the value of any Boolean variable is 0 or 1, so a table can be created to display all possible values for one or more variables

$$\text{Number of rows} = 2^{\text{Number of variables}}$$

A
0
1

2^1

A	B
0	0
0	1
1	0
1	1

2^2

A	B	c
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

2^3

Boolean Algebra

Operations : NOT(')

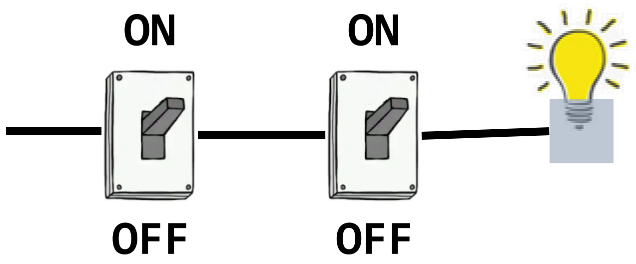
A	A'(NOT)
0	1
1	0

Example : NOT

(B + C)', A'+ C, F', A'+ B'+ C'

Boolean Algebra

Operations : AND(.)



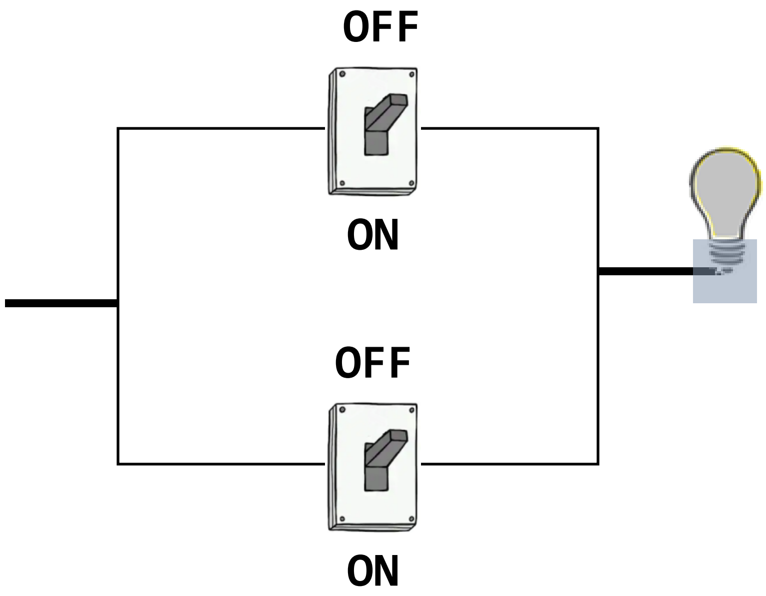
A	B	A.B (AND)
0	0	0
0	1	0
1	0	0
1	1	1

Example : AND

A.B.C , A'.B, B'.C.A'

Boolean Algebra

Operations : OR(+)



A	B	A+B(OR)
0	0	0
0	1	1
1	0	1
1	1	1

Example : OR

A.(B + C) , A'+ B, B'+ C.A'

Boolean Algebra

Operations : XOR(⊕)

A	B	A⊕B(XOR)
0	0	0
0	1	1
1	0	1
1	1	0

Example : XOR

$(A \oplus B).C$, $A' \oplus B$, $(B' \oplus C) + A$

Boolean Algebra

Example : Show that

$$A \oplus B = A'B + B'A$$

A	B	A⊕B(XOR)	A'	B'	A'.B	B'.A	A'.B + B'.A
0	0	0	1	1	0	0	0
0	1	1	1	0	1	0	1
1	0	1	0	1	0	1	1
1	1	0	0	0	0	0	0

Example :

$$1 + 0 = 1$$

$$1 + 1 = 1$$

$$1 \cdot 1 = 1$$

Boolean Algebra
Number of Boolean functions

Operations : F

A	F			
0	0	1	0	1
1	1	0	0	1

F:

A , A', A'.A , A'+ A

Boolean Algebra
Number of Boolean functions

Operations : F

A	B	F
0	0	
0	1	?
1	0	
1	1	

16 functions

F ?

Boolean Algebra
Number of Boolean functions

Operations : F

A	B	F
0	0	
0	1	?
1	0	
1	1	

16 functions

F:

0, A'B', A'B, A', AB', B', A⊕B, A'+B', AB, A'B'+A.B, B, A'+B, A, A+B', A+B, 1

Boolean Algebra
Number of Boolean functions

How many different Boolean functions are there with n inputs?

With n inputs: $$2^{2^n}$$

Example :

$$n = 1 \quad 2^{2^1} = 2^2 = 4$$

$$n = 2 \quad 2^{2^2} = 2^4 = 16$$

Boolean Algebra

Identities table

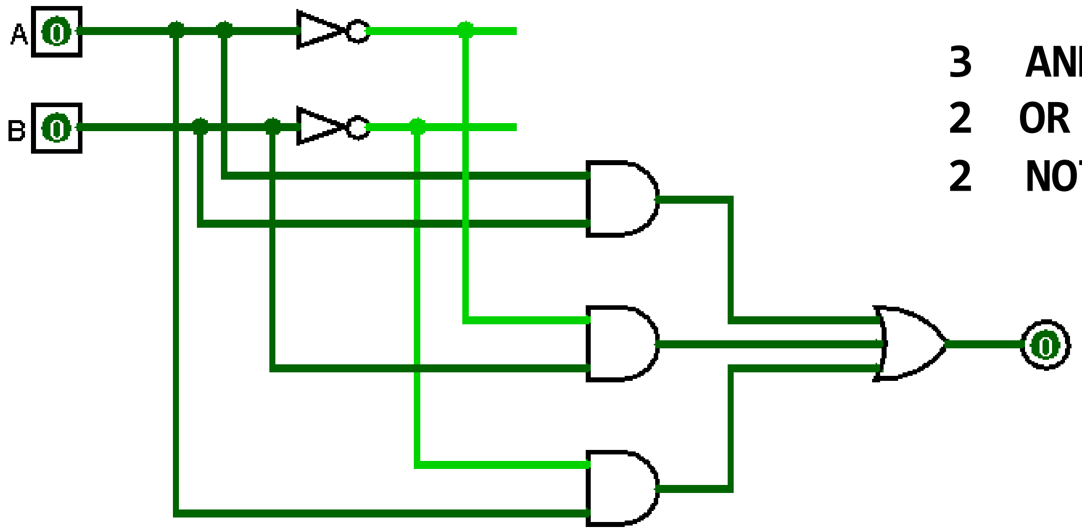
Name	AND form	OR form
Identity law	$1A = A$	$0 + A = A$
Null law	$0A = 0$	$1 + A = 1$
Idempotent law	$AA = A$	$A + A = A$
Inverse law	$AA' = 0$	$A + A' = 1$
Commutative law	$AB = BA$	$A + B = B + A$
Associative law	$(AB)C = A(BC)$	$(A + B) + C = A + (B + C)$
Distributive law	$A + BC = (A + B)(A + C)$	$A(B + C) = AB + AC$
Absorption law	$A(A + B) = A$	$A + AB = A$
De Morgan's law	$(AB)' = A' + B'$	$(A + B)' = A'B'$

Boolean Algebra

Use identities table to simplify Boolean expressions

Example : Simplify Boolean expression F.

F = A'B + AB' + AB



3 AND gates (? transistors)
2 OR gates (? transistors)
2 NOT gates (? transistors)

Boolean Algebra

Use identities table to simplify Boolean expressions

Example : Simplify Boolean expression F.

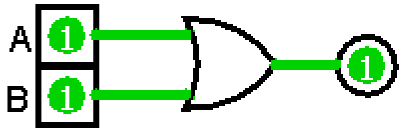
$$F = A'B + AB' + AB$$

$$F = A'B + AB' + AB = (A' + A)B + AB'$$

$$F = (1)B + AB' = B + AB' = (B+A)(B+B')$$

$$F = A + B$$

2 transistors



Boolean Algebra

Example : Show that

$$(A + B)' = A'.\ B'$$

A	B	A + B	(A + B)'	A'	B'	A' B'
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

Boolean Algebra

Use identities table to simplify Boolean expressions

Example : Simplify Boolean expression F and G.

G = A'.(B + C) + A

F = A'B + AB' + AB

Boolean Algebra

Use identities table to simplify Boolean expressions

Example : Simplify Boolean expressions

$F1 = AB' + A(A + C)' + B(B + C)'$

$F2 = A'BC + AB'C' + A'B'C' + AB'C + ABC$

$F3 = (AB'(C + BD) + A'B')C$

$$
\begin{aligned}
F2 &= A'BC + B'C'(A + A') + AC(B' + B) \\
&= A'BC + B'C'(1) + AC(1) \\
&= A'BC + B'C' + AC
\end{aligned}
$$

Boolean Algebra

Obtaining F from the truth table

Example : Obtain F for the following truth table

A	B	F
0	0	0
0	1	1
1	0	0
1	1	1

There are two ways to do that:

1. Sum of Product (**SOP**)
2. Product of Sum (**POS**)

Boolean Algebra

1. Sum of Products (SOP)

1) Only outputs with the value of one (1) in the truth table are considered
2) Minterm for each case is obtained by designating A(B, C, ..) to value 1 and A'(B', C', ..) to value 0 and considering **AND** combination of variables
3) F = m0 + m1 + . . . = $\sum$ Minterms

Boolean Algebra

1. Sum of Product (SOP)

Example : Obtain F for the following truth table

	A	B	F	Minterms	
0	0	0	0		
1	0	1	1	A'B	m1
2	1	0	0		
3	1	1	1	AB	m3

$F = m1 + m3 = A'B + AB$

$F = (A' + A)B$

$F = B$

Boolean Algebra

Example : Obtain F(SOP) for the following truth table

	A	B	C	F
0	0	0	0	0
1	0	0	1	0
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	0

Boolean Algebra

1. Product of Sums (POS)

1) Only outputs with the value of zero (0) in the truth table are considered
2) Maxterm for each case is obtained by designating A(B, C, ..) to value 0 and A'(B', C', ..) to value 1 and considering **OR** combination of variables
3) F = m0.m1.m2. . . . = $\prod$ Maxterms

Boolean Algebra

1. Product of Sums (POS)

Example : Obtain F for the following truth table

	A	B	F	Maxterms	
0	0	0	0	A+B	m0
1	0	1	1		
2	1	0	0	A'+B	m2
3	1	1	1		

$$F = m0 \cdot m2 = (A + B)(A' + B)$$
$$F = AA' + AB + BA' + BB$$
$$F = 0 + B(A + A') + B = B + B$$
$$F = B$$

Boolean Algebra

Example : Express the Boolean function $F = (C' + A + B)(A' + C)$ as standard product of Maxterms (POS).

$F = (C' + A + B)(A' + C + 0)$
$F = (C' + A + B)(A' + C + BB')$
$F = (C' + A + B)((A' + C) + BB')$
$F = (C' + A + B)((A' + C) + B)((A' + C) + B')$

$$F = (C' + A + B)(A' + C + B)(A' + C + B')$$

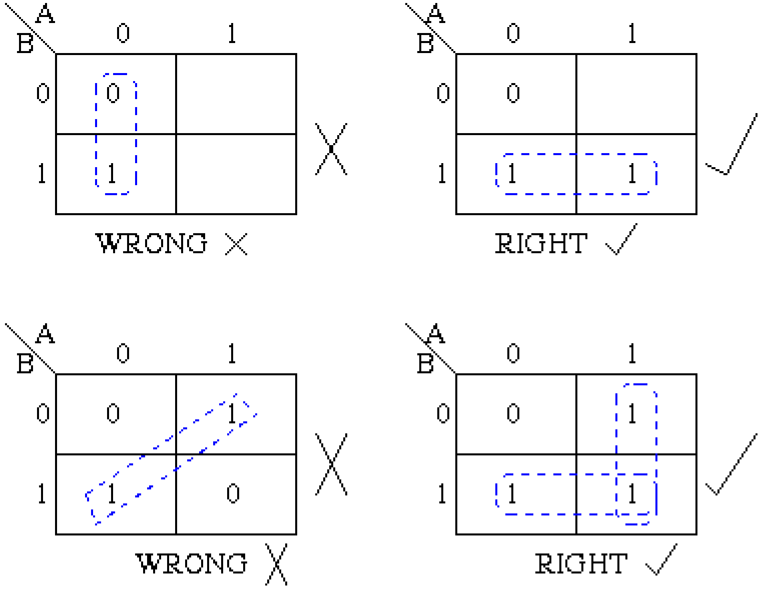
Boolean Algebra

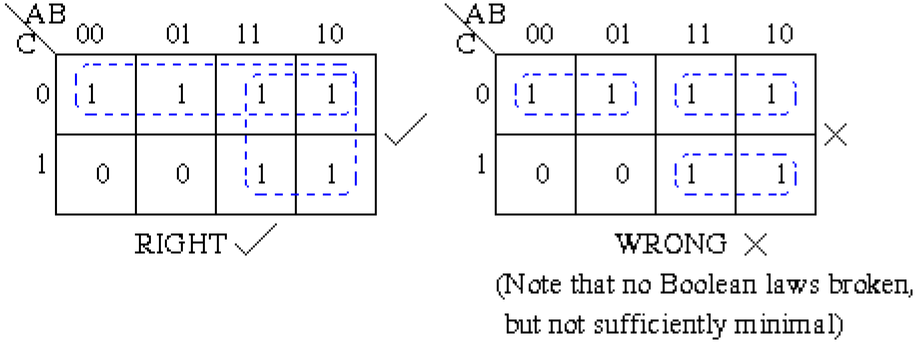
Karnaugh map (K-map)

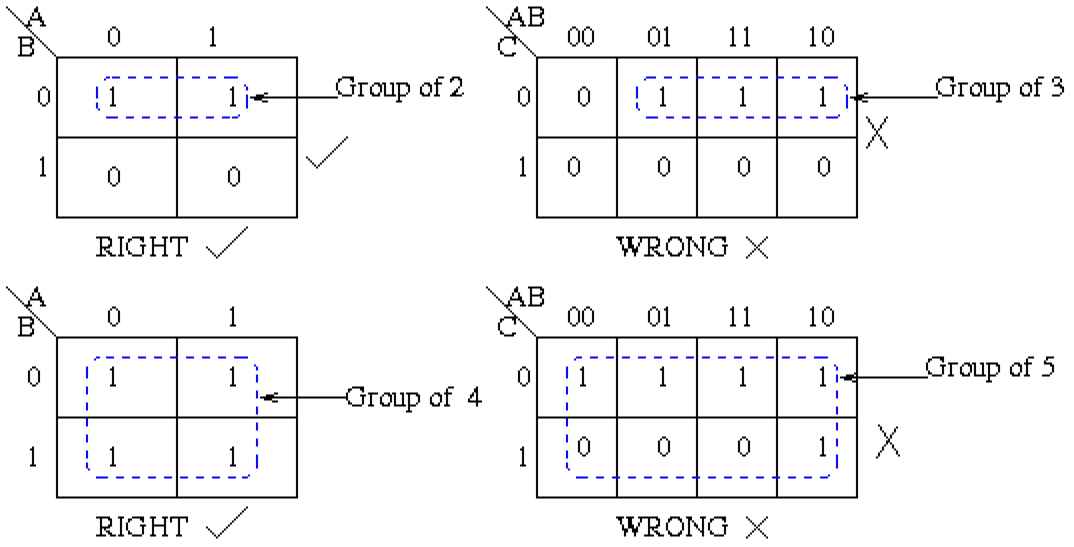
Used to simplify Boolean terms, in order to create a K-map

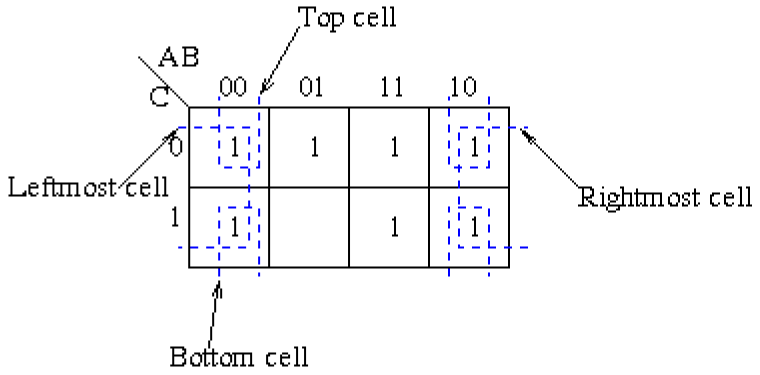
- A K-map is a grid of squares, one square for each row in truth table
- The value for each square comes from F values, just 1'es (SOP)
- Make rectangular groups of 1'es and look for unchanging variables
- Allowed rectangular groups : 1, 2, 4, .. just power of 2
- Each group should be as large as possible
- Group may overlap and also may wrap around a table
- There should be as few groups as possible
- Every 1 must be in at least one group
- Just horizontal or vertical rectangular groups are allowed

RefLink


WRONG
RIGHT
WRONG
RIGHT


RIGHT
WRONG
(Note that no Boolean laws broken,
but not sufficiently minimal)


Group of 2
RIGHT
Group of 3
WRONG
Group of 4
RIGHT
Group of 5
WRONG


Top cell
Leftmost cell
Rightmost cell
Bottom cell

Boolean Algebra

Example : Obtain a simplified form of F(SOP) for the following truth table.

A	B	F
0	0	0
0	1	1
1	0	1
1	1	1

A\B	0	1
0		1
1	1	1

AB
01
11

B

AB
10
11

A

$F = A + B$

Boolean Algebra

Example : Obtain a simplified form of F(SOP) for the following truth table.

$$F = A'B'C' + A'BC' + A'BC + AB'C' + ABC'$$

	A	B	C	F
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	1
5	1	0	1	0
6	1	1	0	1
7	1	1	1	0

AB\C	0	1
00	1	
01	1	1
11	1	
10	1	

ABC
010
011

$A'B$

ABC
000
010
110
100

C'

$$F = C' + A'B$$

Boolean Algebra

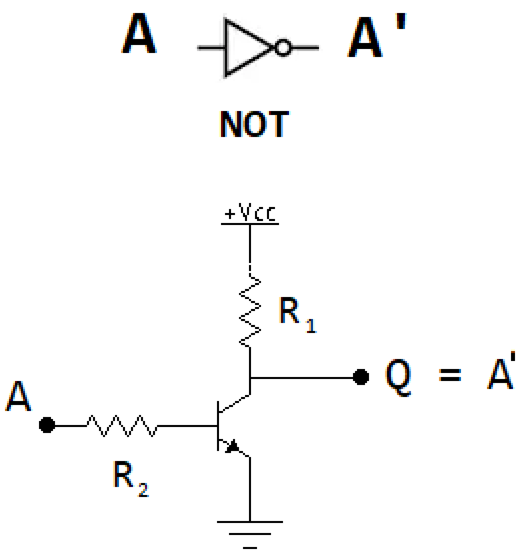
Example : Express the Boolean function F = C' + A'B as standard sum of Minterms (SOP).

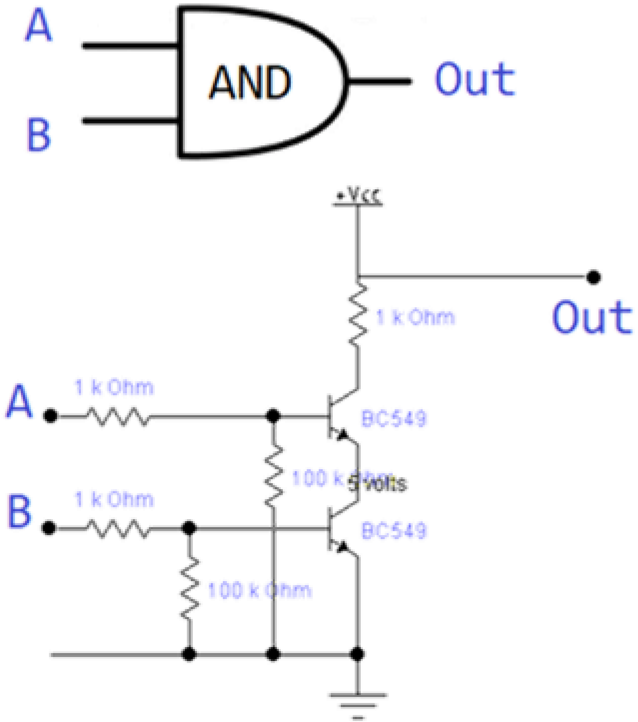
F = C' + A'B = C'.1.1 + A'.B.1 = C'(B' + B)(A' + A) + A'B(C'+C)
F = (C'B' + C'B)(A' + A) + A'BC'+A'BC
F = C'B'A' + C'B'A + C'BA' + C'BA + A'BC' + A'BC
F = A'B'C' + AB'C' + A'BC' + ABC' + ~~A'BC'~~ + A'BC

F = A'B'C' + AB'C' + A'BC' + ABC' +A'BC

Boolean Algebra

Analog to digital





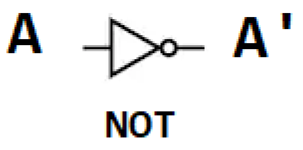
Logic gates

Building blocks in any digital circuits

NOT , AND, OR, XOR, NOR, NAND

Not gate

A	A'
0	1
1	0

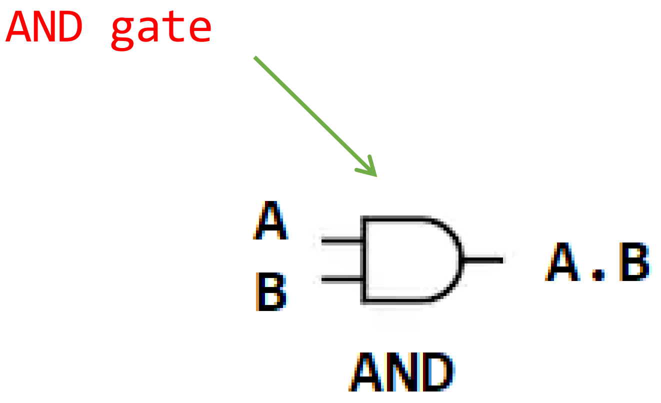


Logic gates

Building blocks in any digital circuits

NOT , AND, OR, XOR, NOR, NAND

A	B	A.B
0	0	0
0	1	0
1	0	0
1	1	1

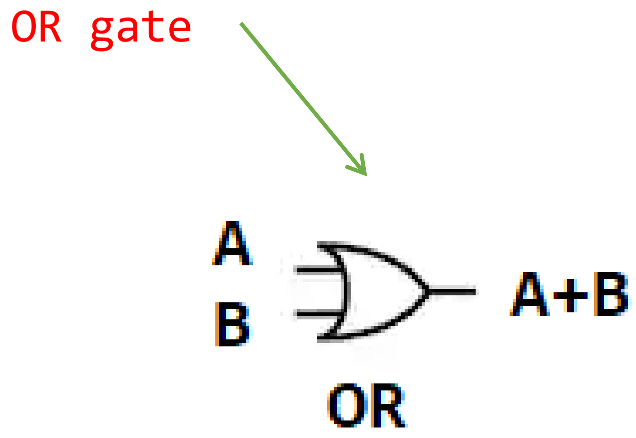


Logic gates

Building blocks in any digital circuits

NOT , AND, OR, XOR, NOR, NAND

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



Logic gates

Building blocks in any digital circuits

NOT , AND, OR, XOR, NOR, NAND

A	B	A⊕B
0	0	0
0	1	1
1	0	1
1	1	0

XOR gate

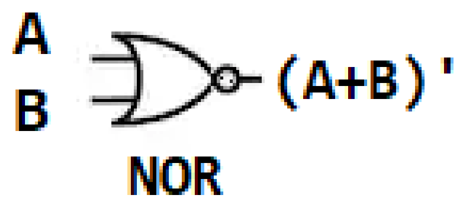
Logic gates

Building blocks in any digital circuits

NOT , AND, OR, XOR, NOR, NAND

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

NOR gate



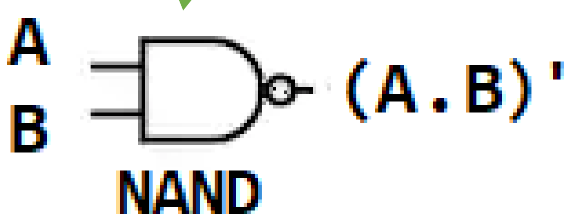
Logic gates

Building blocks in any digital circuits

NOT , AND, OR, XOR, NOR, NAND

A	B	(A.B)'
0	0	1
0	1	1
1	0	1
1	1	0

NAND gate



Logic gates

Designing rules

1. Determine the number of inputs and outputs
2. Derive the truth table for inputs and outputs
3. Consider outputs with TRUE values (1)
4. Obtain outputs in terms of inputs using **sum of products (SOP)** rule
5. Simplify the Boolean expression for each output

Logic gates

Example: Obtain the Boolean function F?

	A	B	C	F
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

Logic gates

Example: Obtain the Boolean function F?

	A	B	C	F	Minterms
0	0	0	0	1	A'B'C'
1	0	0	1	0	
2	0	1	0	0	
3	0	1	1	1	A'BC
4	1	0	0	0	
5	1	0	1	1	AB'C
6	1	1	0	1	ABC'
7	1	1	1	0	

F = A'B'C' + A'BC + AB'C + ABC'

Logic gates

Example:

AB\C	0	1
00	1	
01		1
11	1	
10		1

F = A'B'C' + A'BC + AB'C + ABC'

Logic gates

Designing

	A	B	C	F
0	0	0	0	1
1	0	0	1	0
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0

F = A'B'C' + A'BC + AB'C + ABC'