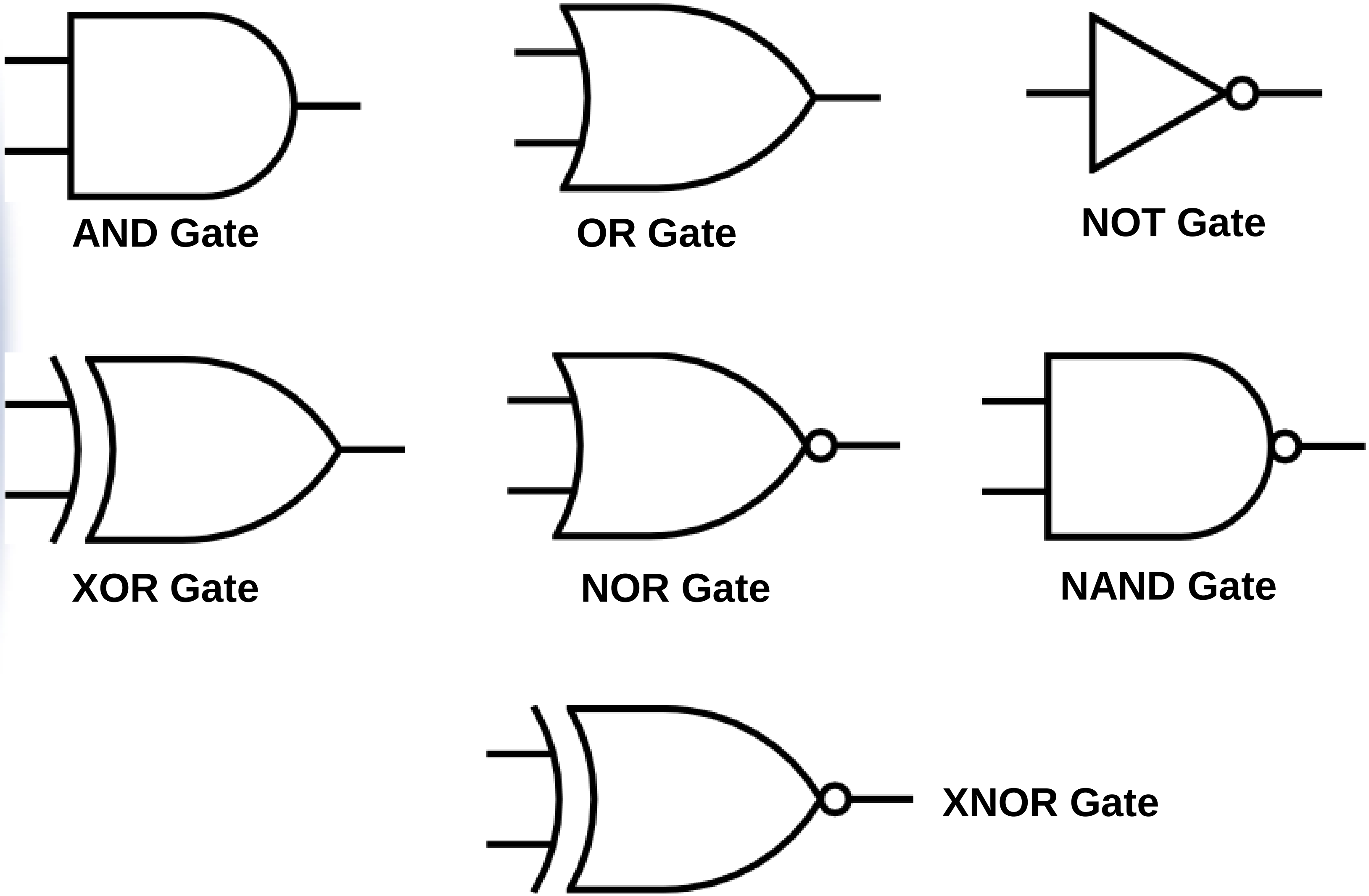
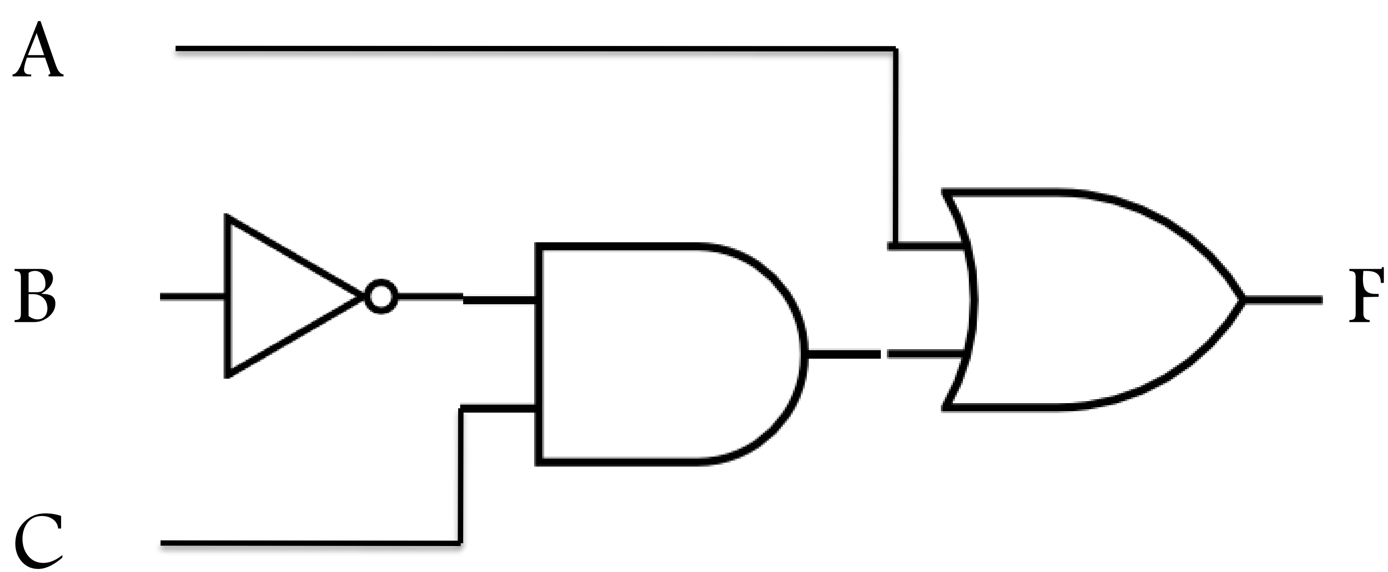
Chapter 3
Boolean Algebra, Logic Functions, and Logic Gates

Digital Logic Gates

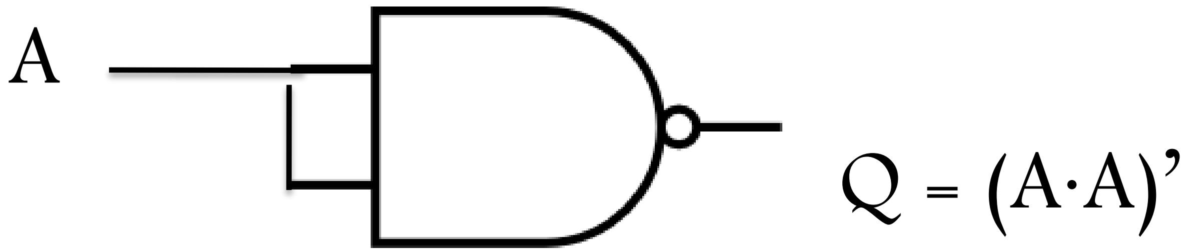


Digital Logic gates

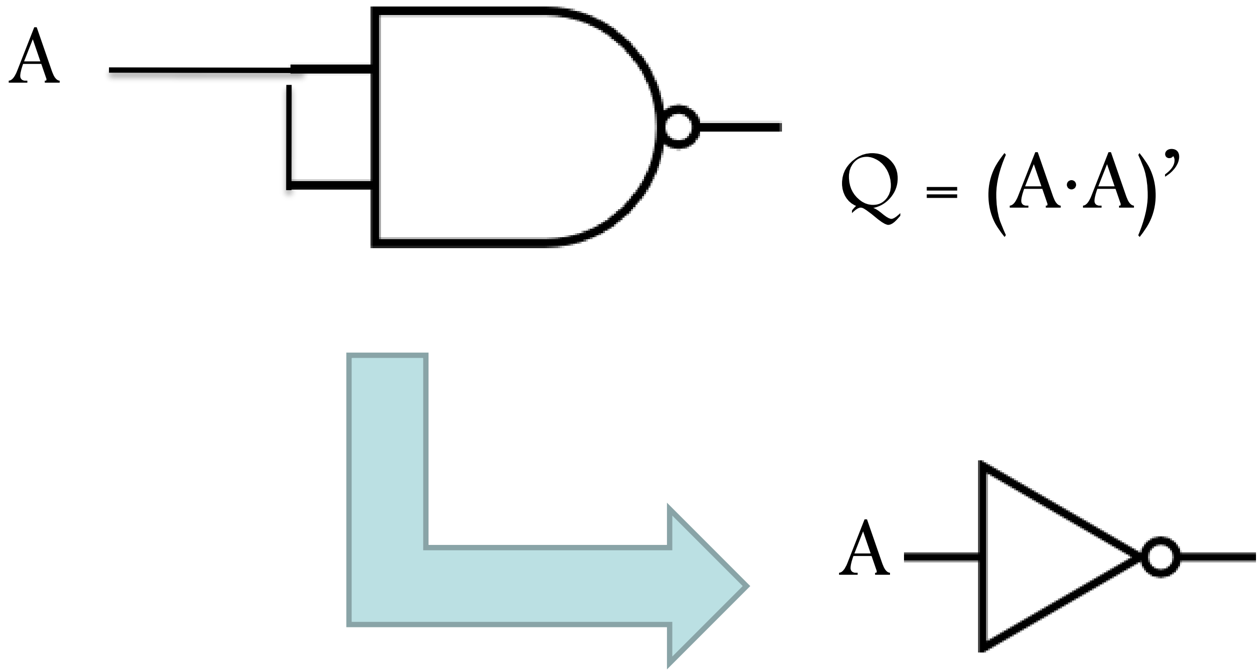
- Draw the logic diagram of the function $F = A+B'C$



Universality of a NAND gate



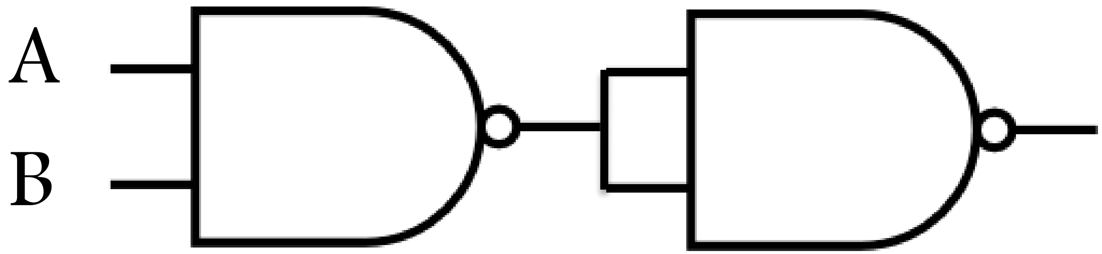
Universality of a NAND gate



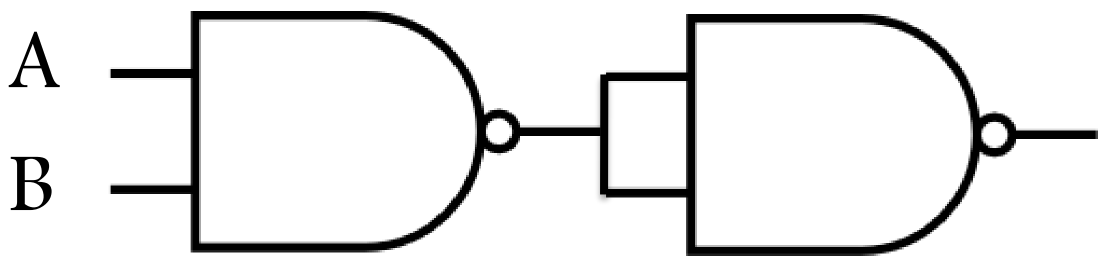
$Q = (A \cdot A)'$

$Q = A'$

Universality of a NAND gate

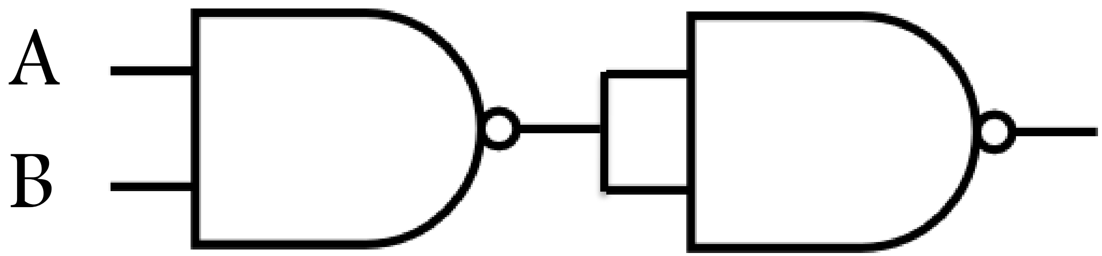


Universality of a NAND gate

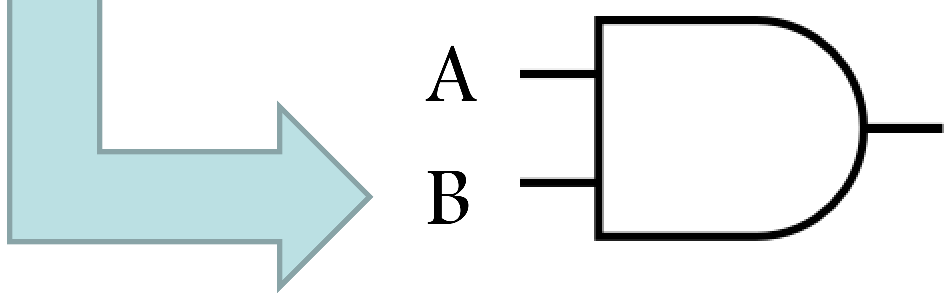


$$Q = ((AB)'(AB)')'$$

Universality of a NAND gate

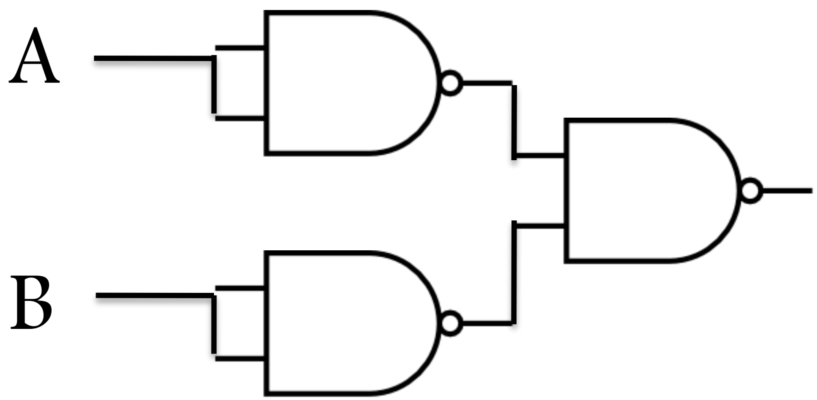


$Q = ((AB)'(AB)')'$

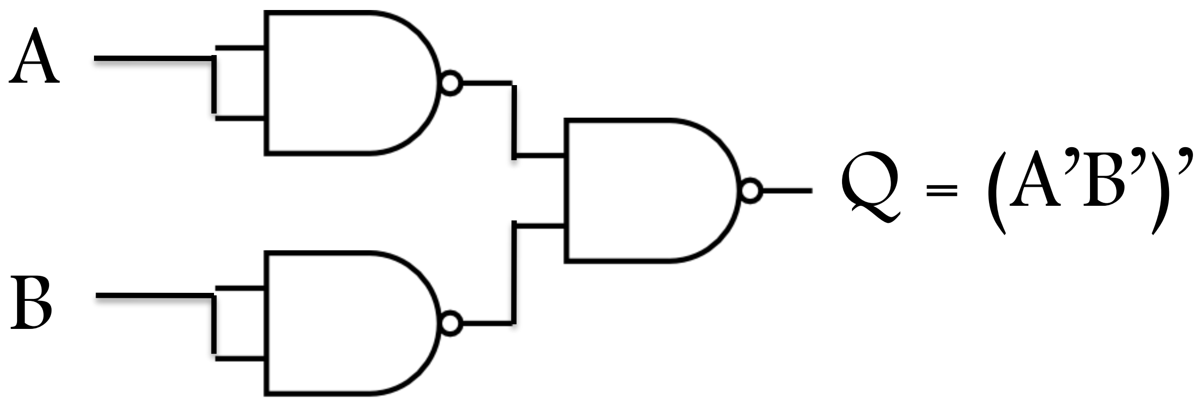


$Q = AB$

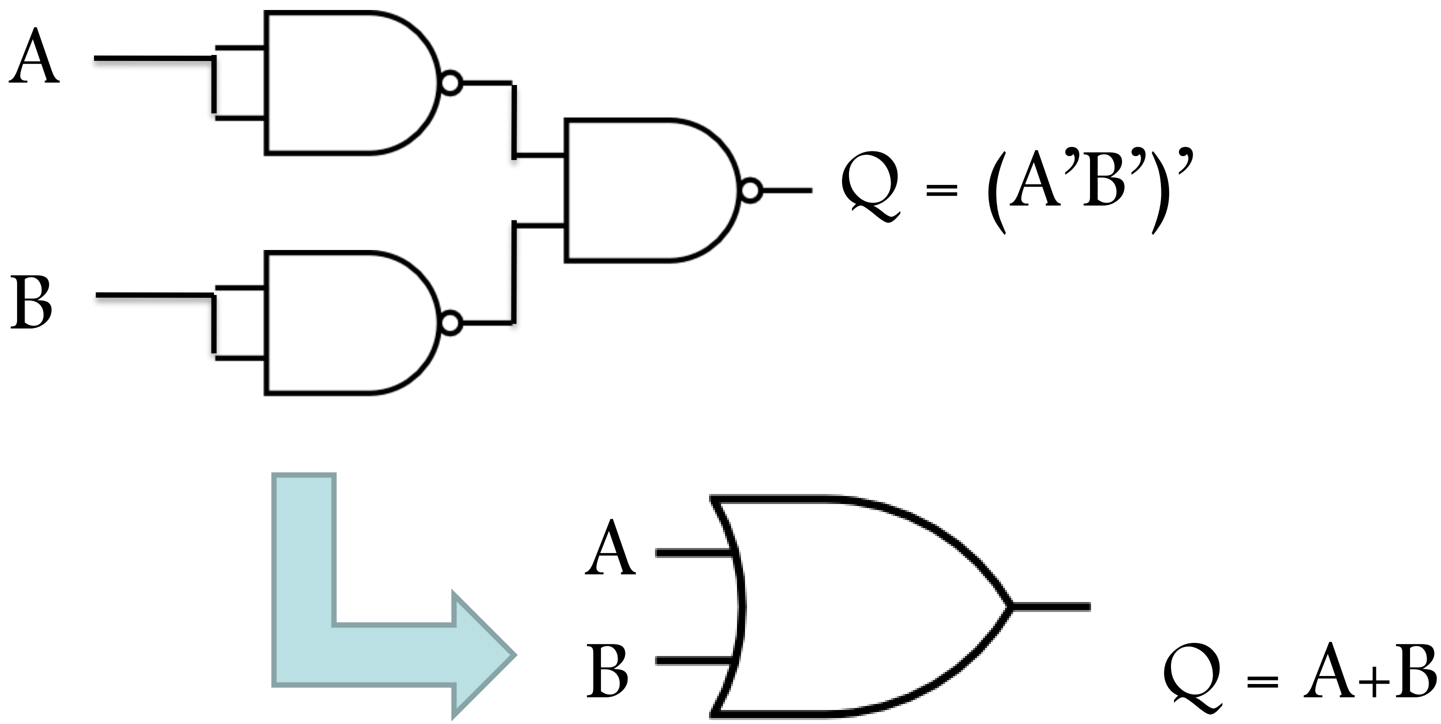
Universality of a NAND gate



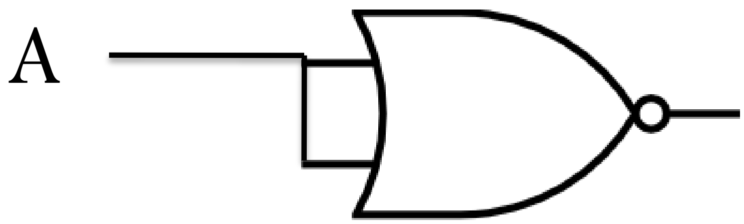
Universality of a NAND gate



Universality of a NAND gate

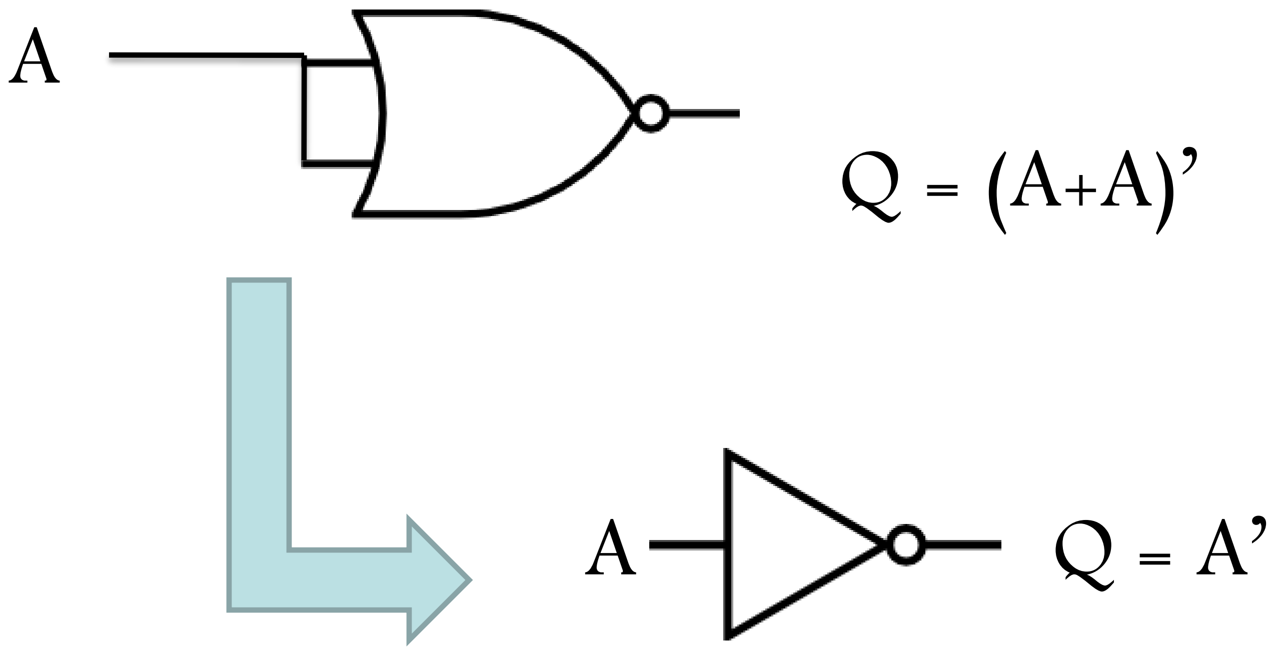


Universality of a NOR gate

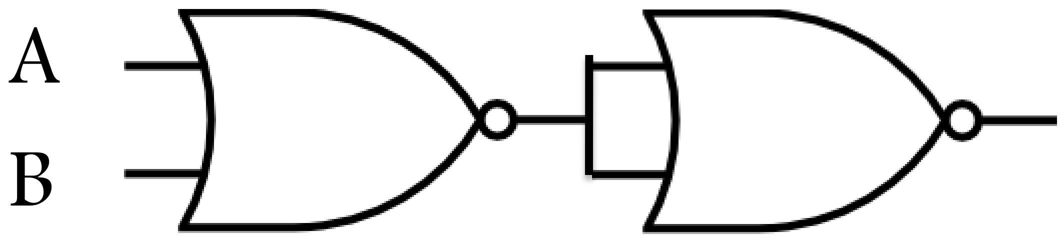


$Q = (A+A)'$

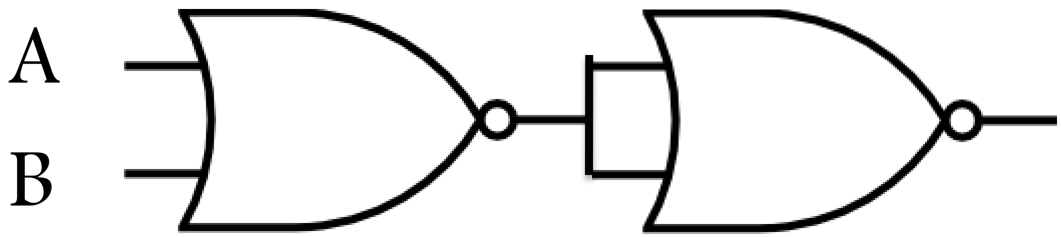
Universality of a NOR gate



Universality of a NOR gate

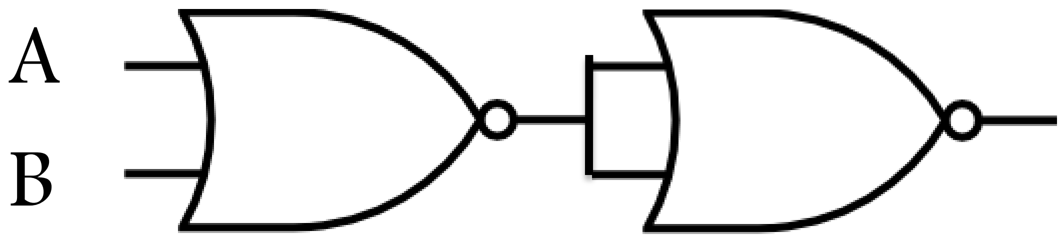


Universality of a NOR gate

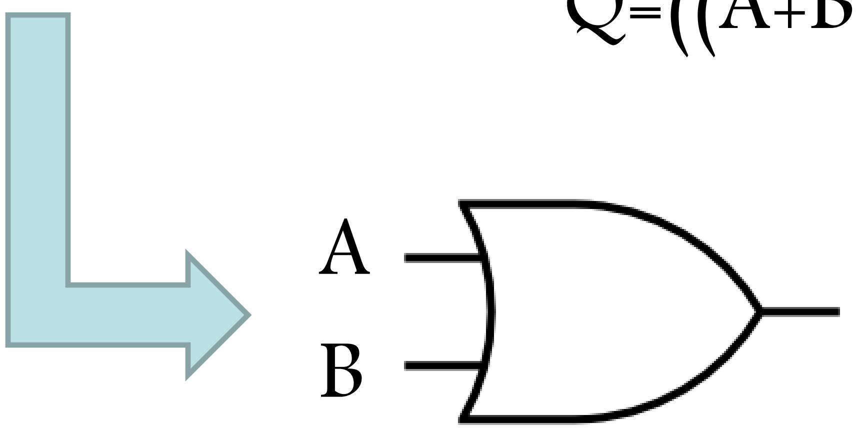


$Q=((A+B)'+(A+B)')'$

Universality of a NOR gate

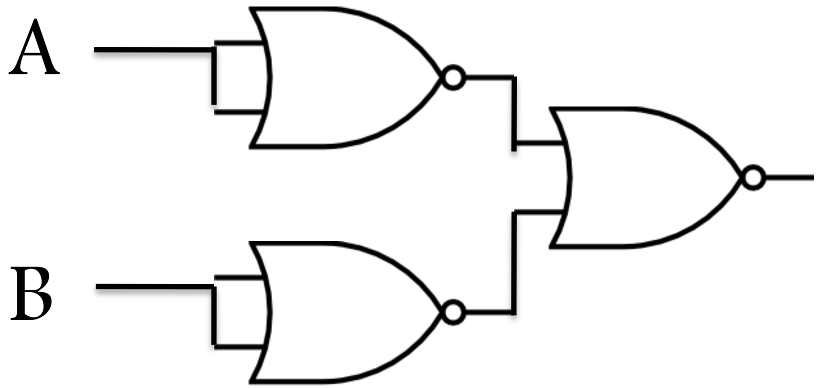


$Q=((A+B)'+(A+B)')'$

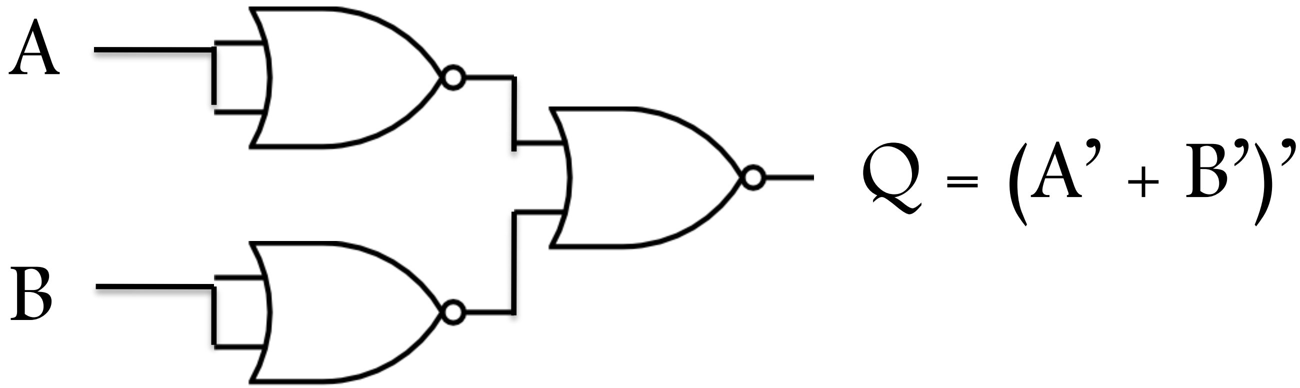


$Q = A+B$

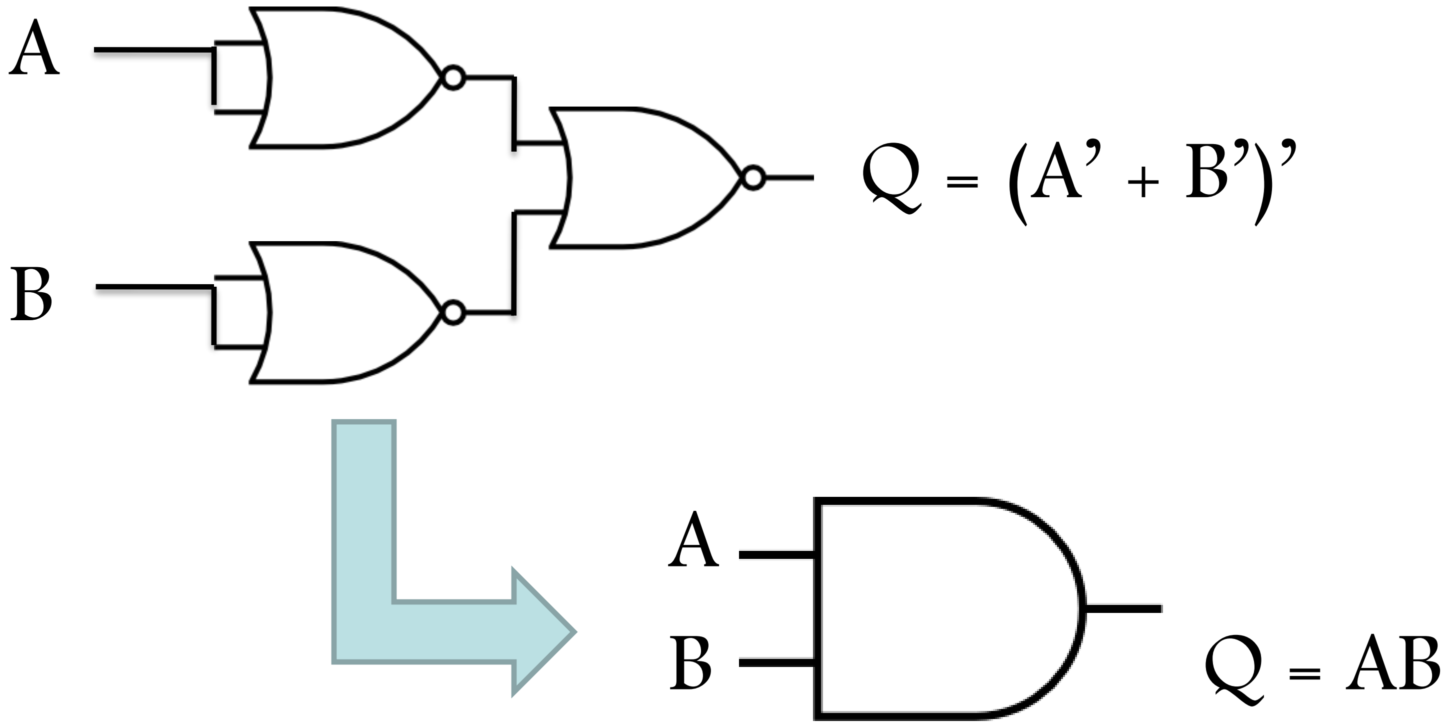
Universality of a NOR gate



Universality of a NOR gate

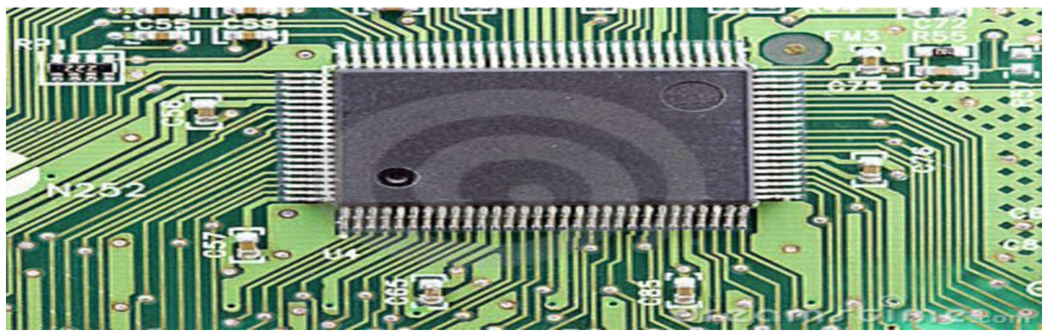


Universality of a NOR gate



Real Gates

- Logic gates are integrated form
  - Built within a solid piece of silicon called IC (Integrated Circuit)



  - Several gates are included in a single plastic moulding

IC Families

- Transistor-Transistor Logic (TTL)
- Emitter Coupled Logic (ECL)
- Complementary Metal-Oxide-Semiconductor (CMOS)

Levels of IC

- Small-scale Integration
  - ICs with 1 to 10 gates
- Medium-scale Integration
  - ICs with 10 to 100 gates
- Large-scale Integration
  - ICs with 100 to 1000s of gates
- Very large-scale Integration
  - ICs with 1000s to millions of gates

Chapter 4 (Part 1)
Simplification of Logic Circuits

Recall: Representations of Boolean Functions

- Truth Table
  - unique

Example:

x	y	F
0	0	0
0	1	1
1	0	1
1	1	1

- Algebraic expression
  - not unique
  - convenient for manipulation

Example: $F = x + y$

- Logic circuits
  - not unique
  - close to implementation

Simplification of Boolean Functions

- Simpler circuit is faster
- Simpler circuit is less expensive
- Reduce complexity of the gate level implementation
- Reduce signal propagation delays

Ways to simplify Boolean functions

- Boolean Algebra
- Graphical method (Karnaugh Map)
- Tabular method (Quine-McCluskey)

Simplification: Boolean Algebra

Simplify: **x’y’ + xyz + x’y**

Simplification: Boolean Algebra

Simplify: **x'y' + xyz + x'y**

= x'(y' + y) + xyz *Comm / Dist.*

Simplification: Boolean Algebra

Simplify: **$x'y' + xyz + x'y$**

$= x'(y' + y) + xyz$ *Comm / Dist.*

$= x' + xyz$ *Inv / Identity*

Simplification: Boolean Algebra

Simplify: $x'y' + xyz + x'y$

$= x'(y' + y) + xyz$ *Comm / Dist.*

$= x' + xyz$ *Inv / Identity*

$= (x' + x)(x' + yz)$ *Dist.*

Simplification: Boolean Algebra

Simplify: $x'y' + xyz + x'y$

$= x'(y' + y) + xyz$ — *Comm / Dist.*

$= x' + xyz$ — *Inv / Identity*

$= (x' + x)(x' + yz)$ — *Dist.*

$= x' + yz$ — *Inv / Identity*

Simplification: Boolean Algebra

Simplify: **AB + A(B+C) + B(B+C)**

Simplification: Boolean Algebra

Simplify: **AB + A(B+C) + B(B+C)**

= AB + AB + AC + BB + BC *Dist.*

Simplification: Boolean Algebra

Simplify: **AB + A(B+C) + B(B+C)**

= AB + AB + AC + BB + BC *Dist.*

= AB + AC + B + BC *Idempotency*

Simplification: Boolean Algebra

Simplify: **AB + A(B+C) + B(B+C)**

= AB + AB + AC + BB + BC *Dist.*

= AB + AC + B + BC *Idempotency*

= AB + AC + B *Absorption*

Simplification: Boolean Algebra

Simplify: **$AB + A(B+C) + B(B+C)$**

$= AB + AB + AC + BB + BC$ *Dist.*

$= AB + AC + B + BC$ *Idempotency*

$= AB + AC + B$ *Absorption*

$= B + AC$ *Absorption*

Simplification: Graphical method

- Karnaugh map (K-map)
  - alternate way of representing Boolean functions
  - a graphical tool for assisting in the general simplification procedure
  - A simpler way to handle most jobs of manipulating logic functions

General Steps of K-Map Simplification

- Express function in canonical form
- Map expression on a K-Map
- Group 1's or 0's
- Determine the minimum expression

Step 1: Function in Canonical form

- Sum of minterms
  - $F(x,y) = x'y' + xy = \Sigma(0,3)$
  - $G(a,b,c) = a'b'c + abc' + a'bc = \Sigma(1,3,6)$
- Product of maxterms
  - $H(x,y) = (x+y')(x'+y) = \Pi(1,2)$
  - $I(a,b,c) = (a+b+c)(a'+b+c') = \Pi(0,5)$

Step 2: Map Expression

X \ Y	0	1
0	m_0	m_1
1	m_2	m_3

Two-variable map

X \ YZ	00	01	11	10
0	m_0	m_1	m_3	m_2
1	m_4	m_5	m_7	m_6

Three-variable map

Step 2: Map Expression

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

Four-variable map

Mapping Example

- F = x’y’z + x’yz’ + xyz’ + xyz

X \ YZ	00	01	11	10
0		1		1
1			1	1

Mapping Example

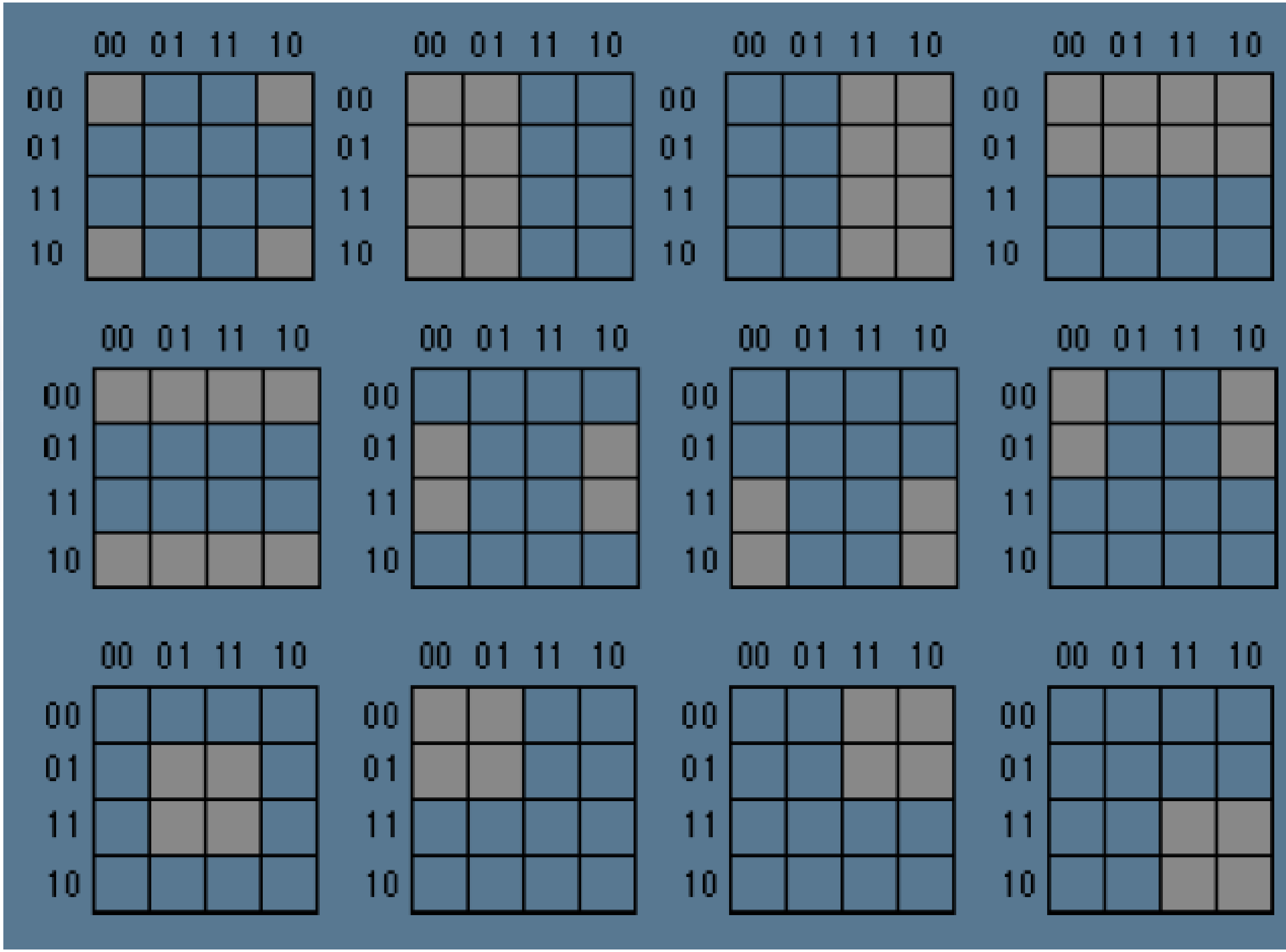
- $G = \Sigma\,(2,4,6,8,12,14,15)$

WX \ YZ	00	01	11	10
00				1
01	1			1
11	1		1	1
10	1			

Step 3: Group 1’s (or 0’s)

- Grouping Rules:
  - A group must contain either 1,2,4,8 and 16 cells.
  - Each cell in a group must be adjacent to one or more cells in that same group.
  - Always include the largest possible group in accordance with the first rule.
  - Each element of a group must be included in at least one group.

Sample Groupings: Four-variable map



Mapping Example

- F = x’y’z + x’yz’ + xyz’ + xyz

X \ YZ	00	01	11	10
0		1		1
1			1	1

Mapping Example

- $G = \Sigma\,(2,4,6,8,12,14,15)$

WX \ YZ	00	01	11	10
00				1
01	1			1
11	1		1	1
10	1			

Grouping Examples

A \ BC	00	01	11	10
0	1		1	
1			1	1

Grouping Examples

A \ BC	00	01	11	10
0	1		1	
1			1	1

Grouping Examples

A \ BC	00	01	11	10
0	1		1	
1			1	1

Grouping Examples

A \ BC	00	01	11	10
0	1		1	
1			1	1

Grouping Examples

A \ BC	00	01	11	10
0	1	1		1
1	1	1	1	

Grouping Examples

A \ BC	00	01	11	10
0	1	1		1
1	1	1	1	

Grouping Examples

A \ BC	00	01	11	10
0	1	1		1
1	1	1	1	

Grouping Examples

A \ BC	00	01	11	10
0	1	1		1
1	1	1	1	

Grouping Examples

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

Grouping Examples

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

Grouping Examples

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

Grouping Examples

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

Step 4: Determine Minimum Expression

- Determine minimum terms

# of cells	# of vars
1	3
2	2
4	1
8	Fn's value is 1

three-variable map

# of cells	# of vars
1	4
2	3
3	2
8	1
16	Fn's value is 1

four-variable map

Determining Minimum Expression Example

A \ BC	00	01	11	10
0	1		1	
1			1	1

Determining Minimum Expression Example

A'B'C'

A \ BC	00	01	11	10
0	1		1	
1			1	1

Determining Minimum Expression Example

A'B'C'

A \ BC	00	01	11	10
0	1		1	
1			1	1

BC

Determining Minimum Expression Example

A'B'C'

A \ BC	00	01	11	10
0	1		1	
1			1	1

BC

AB

Determining Minimum Expression Example

A’B’C’

A \ BC	00	01	11	10
0	1		1	
1			1	1

BC

AB

= A’B’C’ + BC + AB

Determining Minimum Expression Example

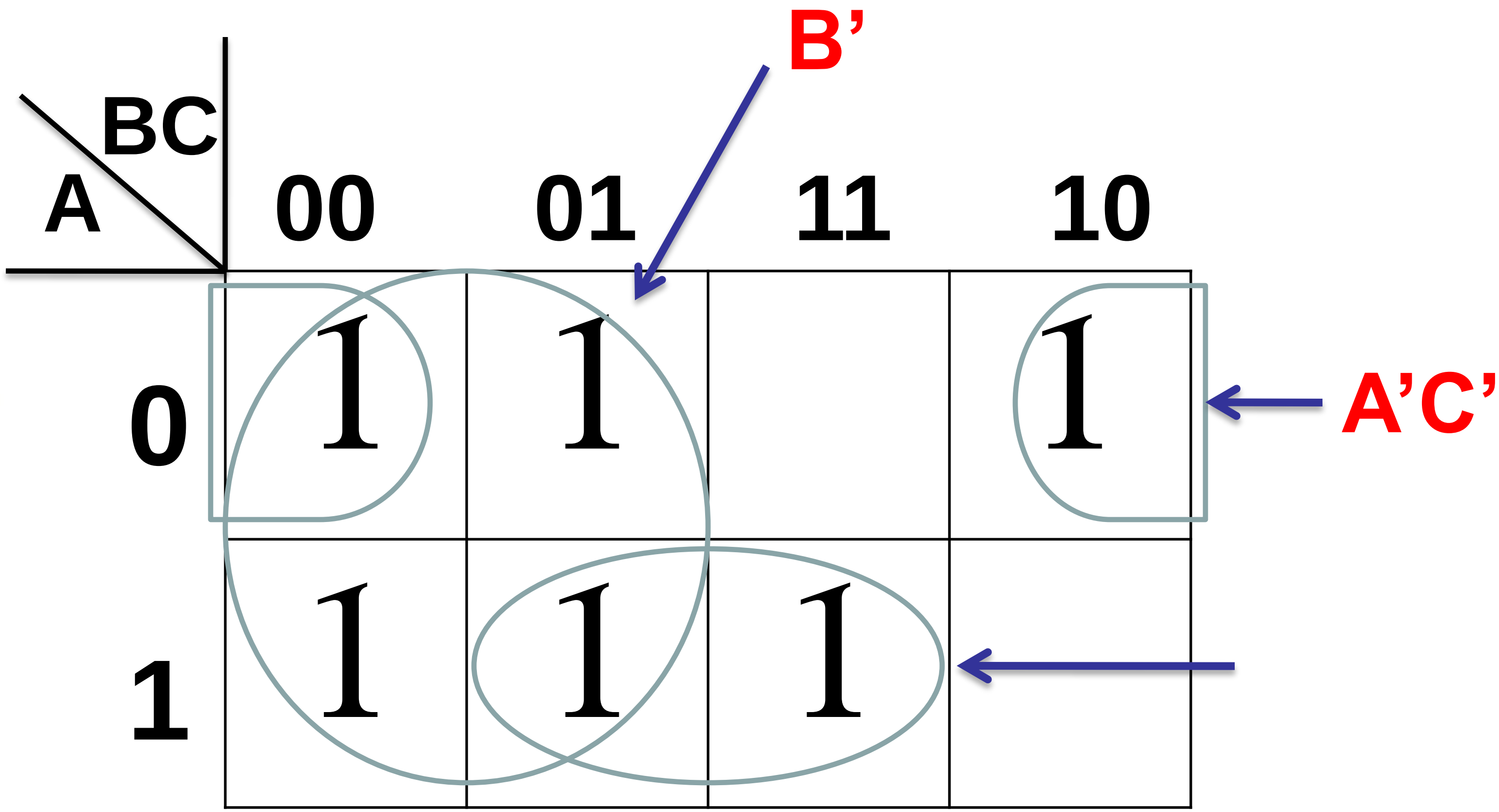
A \ BC	00	01	11	10
0	1	1		1
1	1	1	1	

Determining Minimum Expression Example

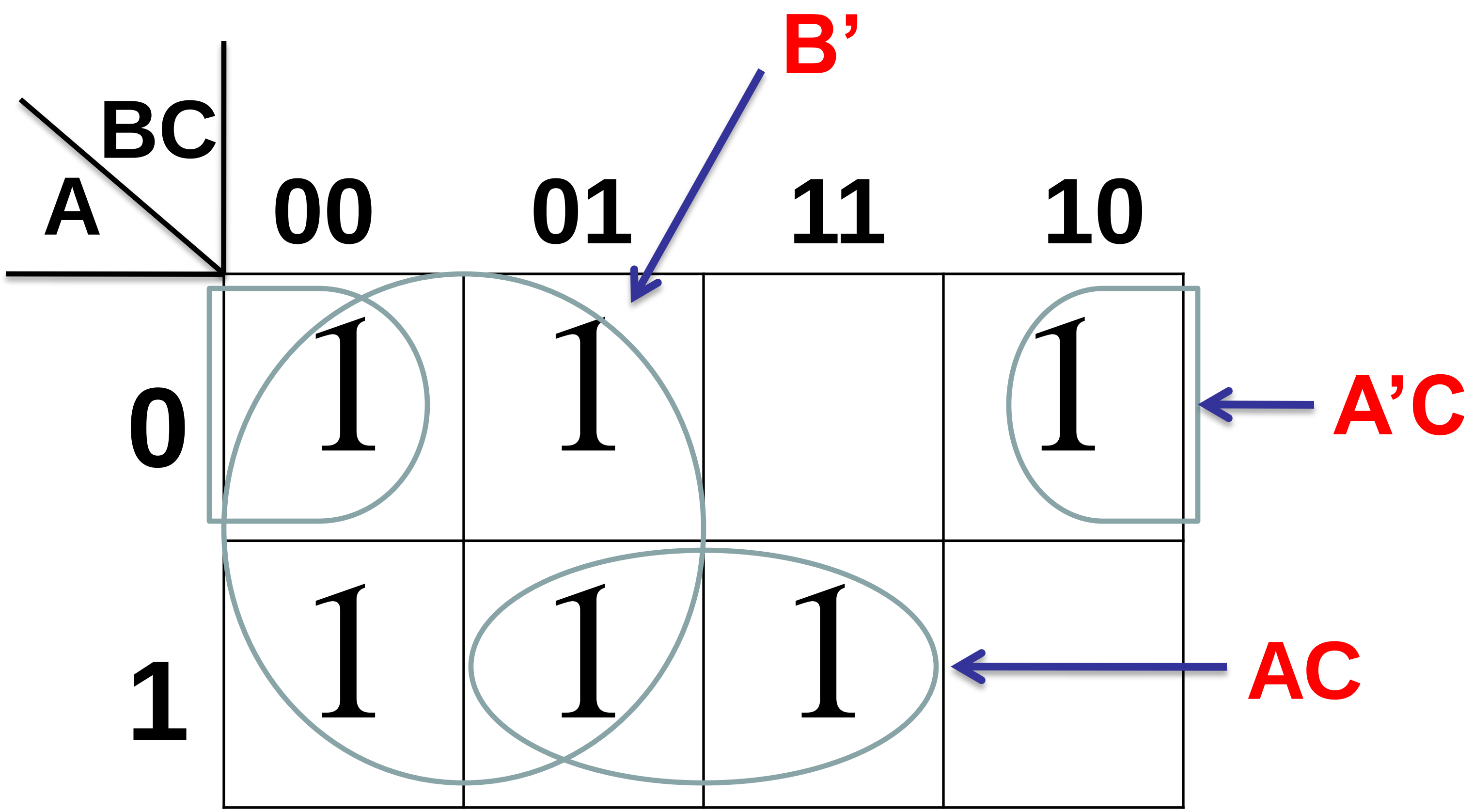
B’

A \ BC	00	01	11	10
0	1	1		1
1	1	1	1	

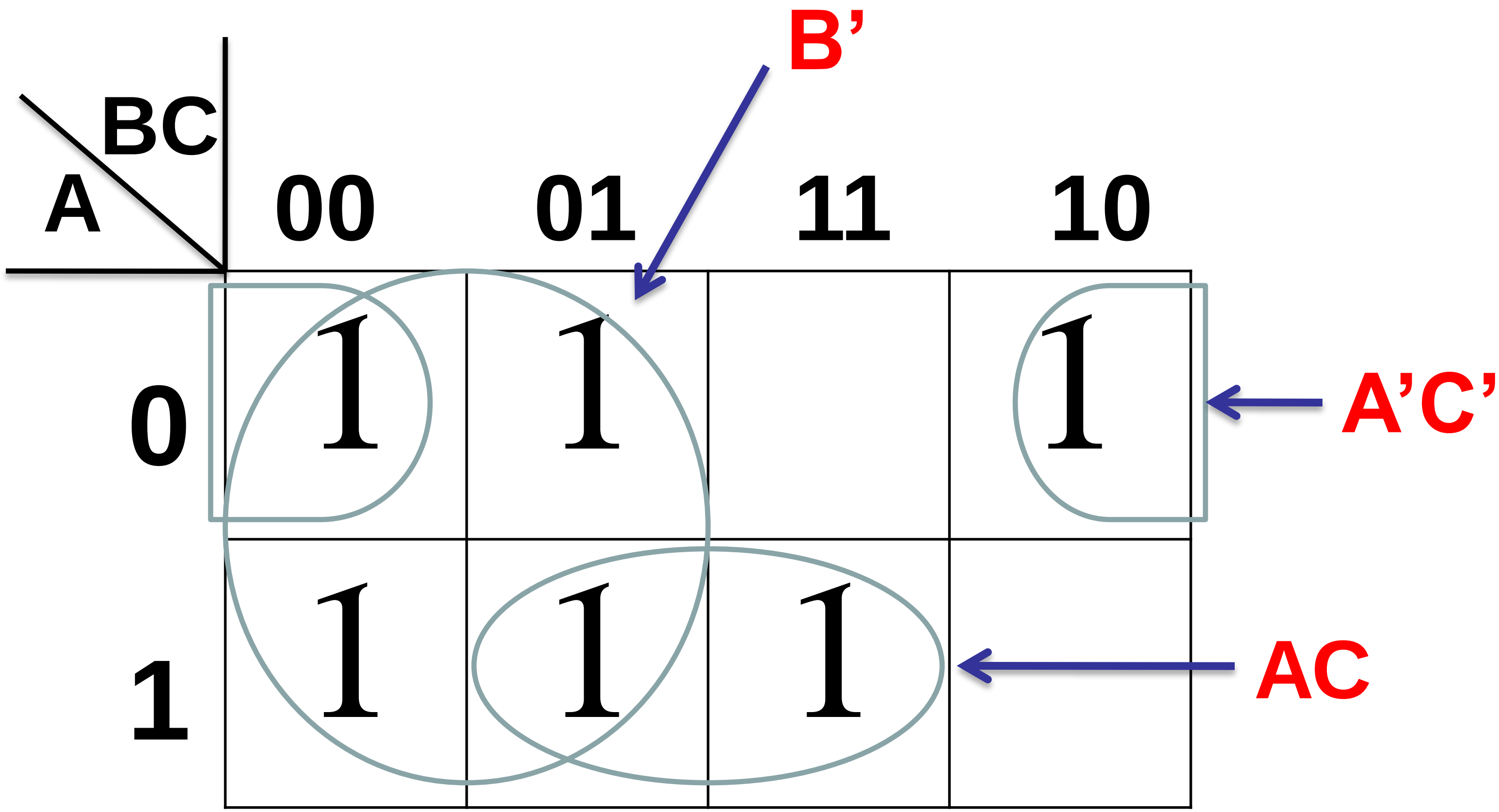
Determining Minimum Expression Example



Determining Minimum Expression Example



Determining Minimum Expression Example



= A'C' + AC + B'

Determining Minimum Expression Example

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

Determining Minimum Expression Example

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

D’

Determining Minimum Expression Example

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

D’

BC’

Determining Minimum Expression Example

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

D’

BC’

AB’C

Determining Minimum Expression Example

AB \ CD	00	01	11	10
00	1			1
01	1	1		1
11	1	1		1
10	1		1	1

D’

BC’

AB’C

= D’ + BC’ + AB’C