Digital Engineering

Assoc. Prof. Osama Elnahas, Dr. Dina Abdelhafiz

Dr. Bassant Tolba, Dr. Radwa Rady

Second Year –Information Technology Program
Fall 2025

Lecture 5

DeMorgan’s Laws

- DeMorgan's Laws:
  - $\overline{(X+Y+Z)} = \bar{X} \cdot \bar{Y} \cdot \bar{Z}$
  - $\overline{(X \cdot Y \cdot Z)} = \bar{X} + \bar{Y} + \bar{Z}$
  - Replace operators: $+ \leftrightarrow \cdot$ and replace each variable with its complement.
  - Example: $\overline{(X+\bar{Y}+Z)} = \bar{X} \cdot Y \cdot \bar{Z}$
  - Example: $\overline{(X \cdot \bar{Y}+Z)} = \overline{(X \cdot \bar{Y})} \cdot \bar{Z} = (\bar{X}+Y) \cdot \bar{Z}$

Summary

- Did you notice something ??

$X+0 = X$	$X \cdot 1 = X$
$X+1 = 1$	$X \cdot 0 = 0$
$X+X = X$	$X \cdot X = X$
$X+X' = 1$	$X \cdot X' = 0$
$X+Y = Y+X$	$X \cdot Y = Y \cdot X$
$(X+Y)+Z = X+(Y+Z) = X+Y+Z$	$(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z) = X \cdot Y \cdot Z$
$X \cdot (Y+Z) = X \cdot Y+X \cdot Z$	$X+(Y \cdot Z) = (X+Y) \cdot (X+Z)$
$(X+Y+Z)' = X' \cdot Y' \cdot Z'$	$(X \cdot Y \cdot Z)' = X'+Y'+Z'$

Duality

- Replace operators: $+ \leftrightarrow \cdot$ and replace constants $0 \leftrightarrow 1$: Then you will get new law. (Variables are not changed).

$X+0 = X$	$X \cdot 1 = X$
$X+1 = 1$	$X \cdot 0 = 0$
$X+X = X$	$X \cdot X = X$
$X+X' = 1$	$X \cdot X' = 0$
$X+Y = Y+X$	$X \cdot Y = Y \cdot X$
$(X+Y)+Z = X+(Y+Z) = X+Y+Z$	$(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z) = X \cdot Y \cdot Z$
$X \cdot (Y+Z) = X \cdot Y+X \cdot Z$	$X+(Y \cdot Z) = (X+Y) \cdot (X+Z)$
$(X+Y+Z)' = X' \cdot Y' \cdot Z'$	$(X \cdot Y \cdot Z)' = X'+Y'+Z'$

Duality

- Replace operators: $+ \leftrightarrow \cdot$ and replace constants $0 \leftrightarrow 1$: Then you will get new law. (Variables are not changed).
- Extra basic laws:

$0+0 = 0$	$1 \cdot 1 = 1$
$1+1 = 1$	$0 \cdot 0 = 0$
$0+1 = 1+0 = 1$	$1 \cdot 0 = 0 \cdot 1 = 0$

Duality

- Given a function: $F = A \cdot \bar{B} + C$
- It is required to find its dual:
  - Replace operators: $+ \leftrightarrow \cdot$ and replace constants $0 \leftrightarrow 1$: Then you will get new law. (Variables are not changed).

$$F^D = ??$$

Duality

- Given a function: $F = A \cdot \bar{B} + C$
- It is required to find its dual:
  - Replace operators: $+ \leftrightarrow \cdot$ and replace constants $0 \leftrightarrow 1$: Then you will get new law. (Variables are not changed).

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

- It is required to find its complement:
  - DeMorgans Law: Replace operators: $+ \leftrightarrow \cdot$ and replace each variable with its complement.

$$\bar{F} = \overline{(A \cdot \bar{B} + C)} = ??$$

Duality

- Given a function: $F = A \cdot \bar{B} + C$
- It is required to find its dual:
  - Replace operators: $+ \leftrightarrow \cdot$ and replace constants $0 \leftrightarrow 1$: Then you will get new law. (Variables are not changed).

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

- It is required to find its complement:
  - DeMorgans Law: Replace operators: $+ \leftrightarrow \cdot$ and replace each variable with its complement.

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

- Did you notice something?? $\bar{F} \leftrightarrow F^D$

Duality

- To change between $\bar{F} \leftrightarrow F^D$: Change each variable to its complement.

- Example: $F \ = \ A \cdot \bar{B} + C$

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

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

- Generally:

$$\bar{F}(A, B, C) = F^D(\bar{A}, \bar{B}, \bar{C})$$

Duality

$$\bar{F}(A,B,C) = F^D(\bar{A},\bar{B},\bar{C})$$

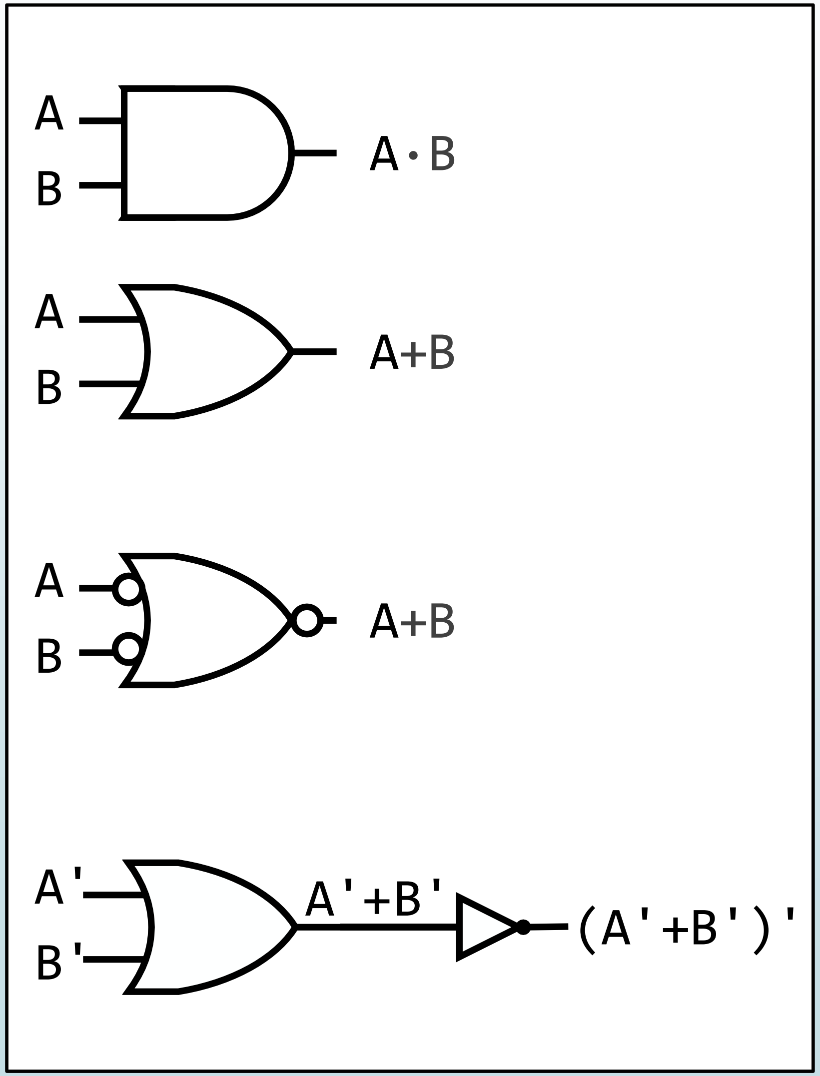
- Take the complement:

$$F(A,B,C) = \overline{(F^D(\bar{A},\bar{B},\bar{C}))}$$

- How to read this ?
  - If you have a function $F(A,B,C)$
  - Find its dual function $F^D(A,B,C)$
  - Complement all the inputs $F^D(\bar{A},\bar{B},\bar{C})$
  - Complement the output $\overline{(F^D(\bar{A},\bar{B},\bar{C}))}$
  - You will get the original function $F(A,B,C) = \overline{(F^D(\bar{A},\bar{B},\bar{C}))}$

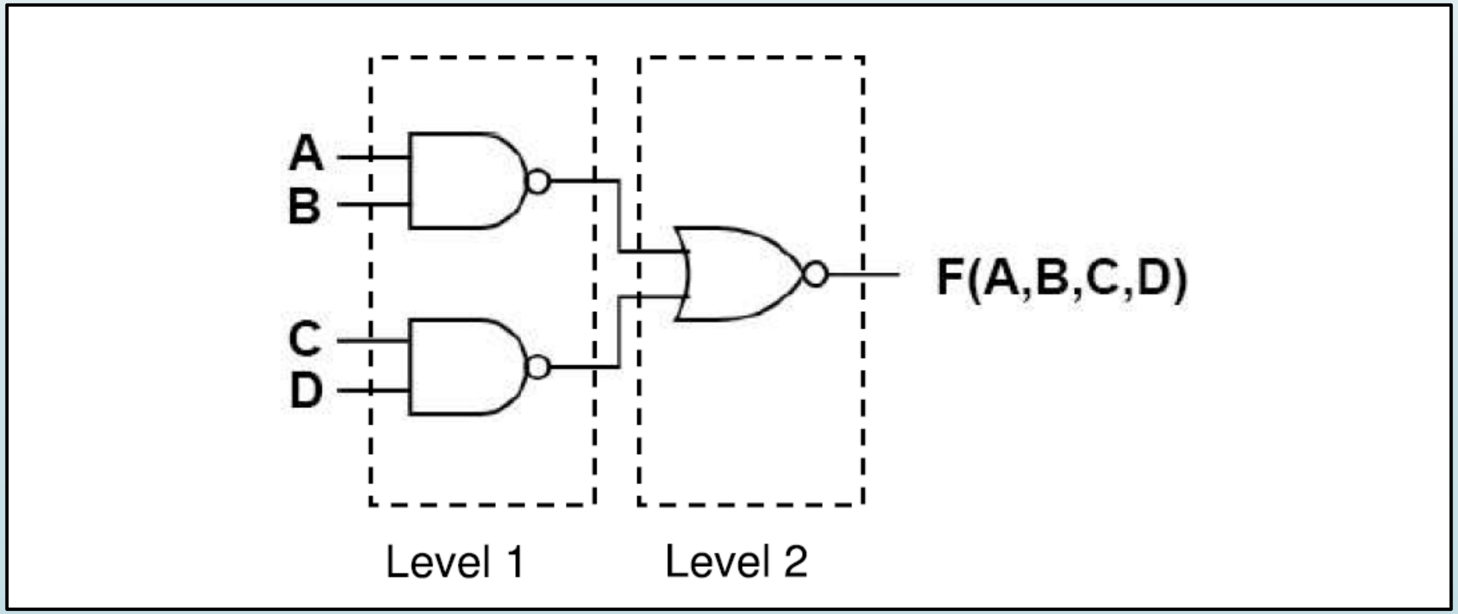
Graphical DeMorgan's Law

- Example:
  - If you have a function $F(A,B) = A \cdot B$
  - Find its dual function $F^D(A,B) = A + B$
  - Complement all the inputs $F^D(\bar{A},\bar{B}) = \bar{A} + \bar{B}$
  - Complement the output $\overline{(F^D(\bar{A},\bar{B}\ ))} = \overline{(\bar{A} + \bar{B})}$
  - You will get the original function $F(A,B) = \overline{(F^D(\bar{A},\bar{B}\ ))} = A \cdot B$



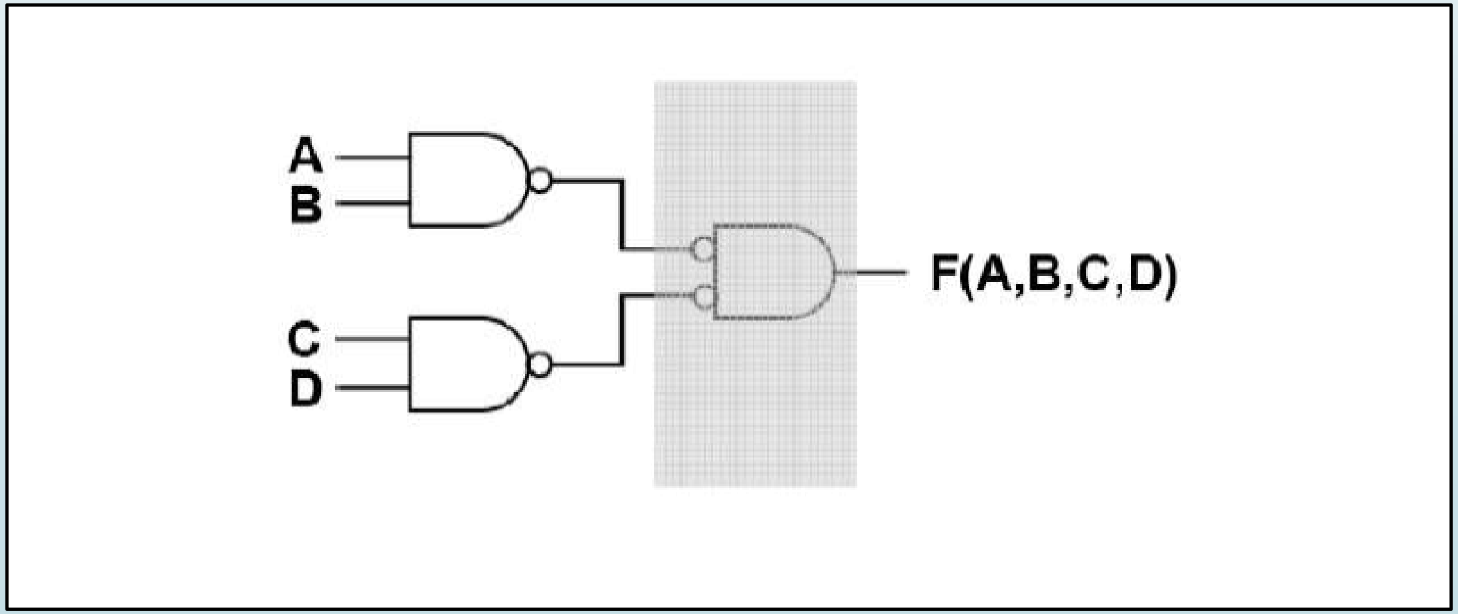
Graphical DeMorgan's Law

- Another example:
  - Given a circuit of multi-level, ignore the equation.
  - Apply Graphical DeMorgans Law on the last level.



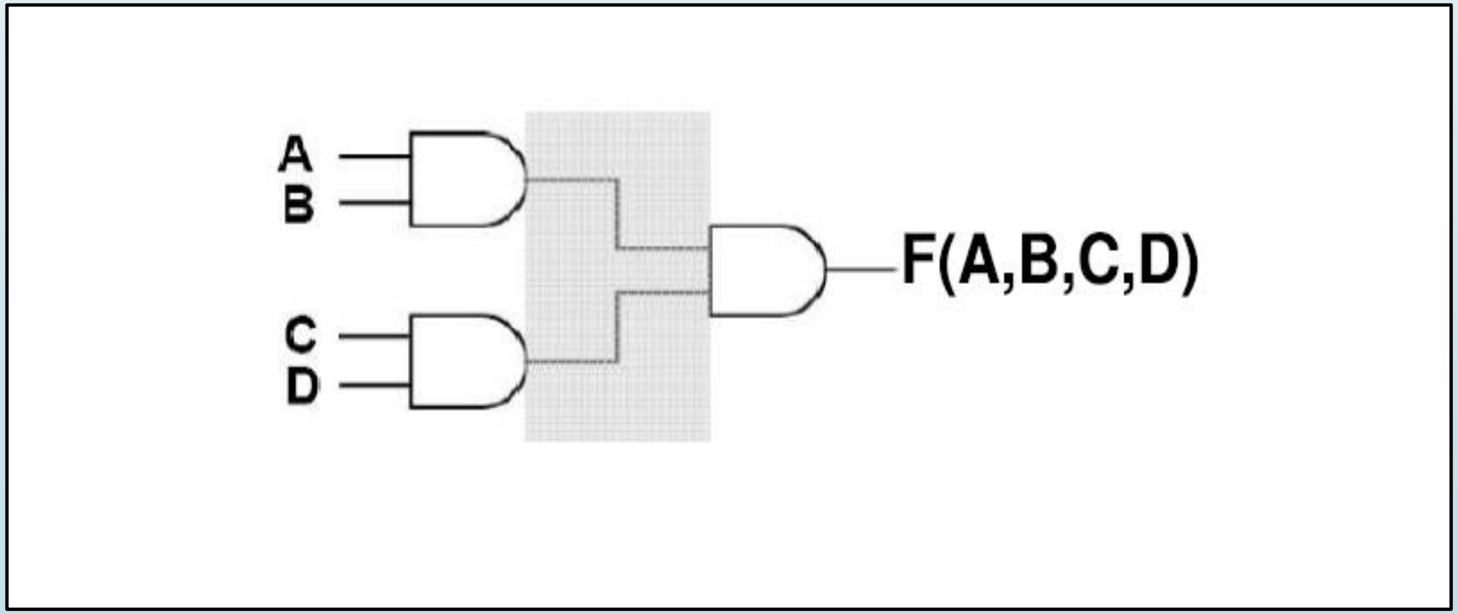
Graphical DeMorgan's Law

- Another example:
  - Flip the OR into AND
  - Add complement circles to inputs.
  - Remove complement circle from output.



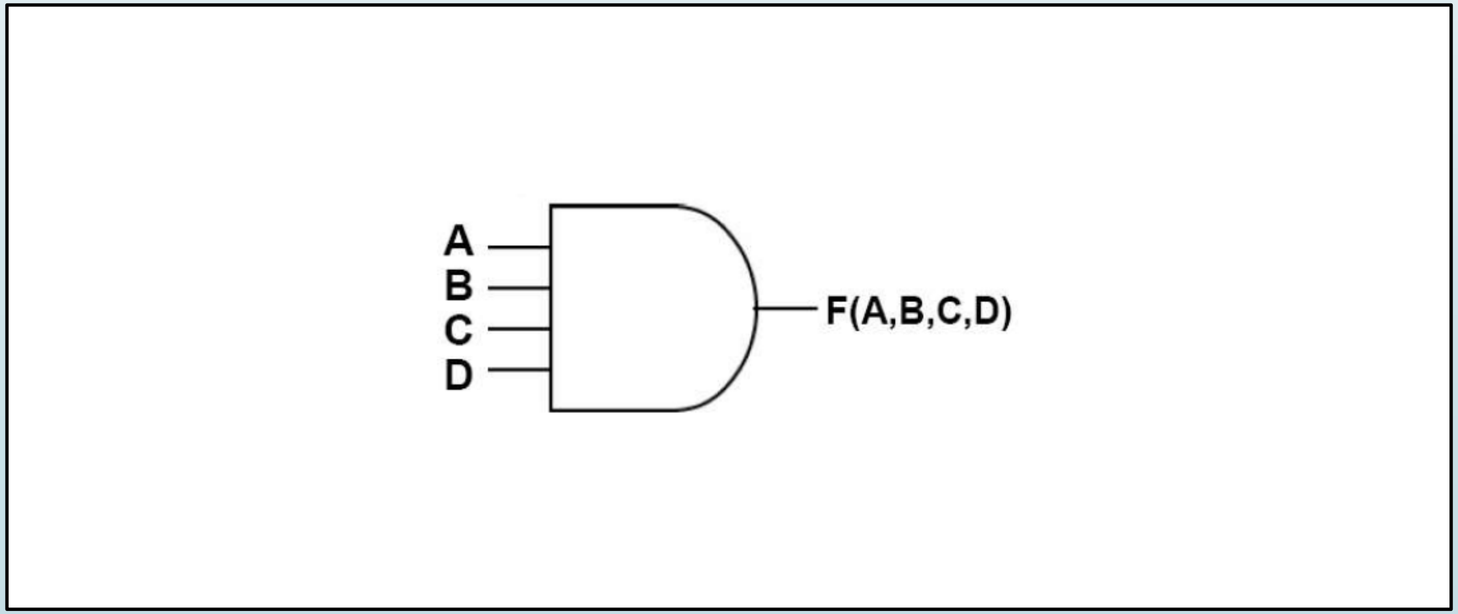
Graphical DeMorgan's Law

- Another example:
  - Flip the OR into AND
  - Add complement circles to inputs.
  - Remove complement circle from output.



Graphical DeMorgan's Law

- Another example:
  - Flip the OR into AND
  - Add complement circles to inputs.
  - Remove complement circle from output.



Graphical DeMorgan's Law

- Another example:
  - Then, this multi-level circuit is reduced to single gate.
  - Two different circuits for the same function.
  - This may reduce the cost of implementation.

Simplification Theorems

- Simplification theorem (1):

$$X + XY = X(1+Y)$$
$$= X \cdot 1$$
$$= X$$

- Apply duality: ??

$$X(X+Y) = X$$

Simplification Theorems

- Simplification theorem (2):

$$\begin{aligned} XY + XY' &= X(Y+Y') \\ &= X\cdot 1 \\ &= X \end{aligned}$$

- Apply duality:

$$\begin{aligned} (X+Y)\cdot(X+Y') &= X+(Y\cdot Y') \\ &= X+0 \\ &= X \end{aligned}$$

Simplification Theorems

- Redundancy theorem:

  XY + X'Z + YZ = XY + X'Z

  - The last term is redundant.

- Apply duality: ??

  (X+Y)·(X'+Z)·(Y+Z) = (X+Y)·(X'+Z)

Summary

- Summary part (2):

X + XY = X	X(X+Y) = X
XY + XY' = X	(X+Y)(X+Y') = X
XY + X'Z + YZ = XY + X'Z	(X+Y).(X'+Z).(Y+Z) = (X+Y).(X'+Z)

Circuit Minimization using Boolean Algebra

Circuit Minimization

- How do you think could we simplify these equations ??
- One possible solution is to use Boolean Algebra.
- Consider the sum of products equation:

$$f = A'BC' + A'BC + AB'C + ABC$$

- Search for any two similar terms except one complement ??

$$f = \mathbf{A'BC'+A'BC}+AB'C+ABC$$

$$f = A'BC'+\mathbf{A'BC}+AB'C+\mathbf{ABC}$$

$$f = A'BC'+A'BC+\mathbf{AB'C+ABC}$$

Circuit Minimization

- For each group apply the rule XY+XY' = X(Y+Y') = X.

  f = **A'BC'+A'BC**+AB'C+ABC → A'B(C'+C) = A'B

  f = A'BC'+**A'BC**+AB'C+**ABC** → (A'+A)BC = BC

  f = A'BC'+A'BC+**AB'C+ABC** → A(B'+B)C = AC

- If you found two intersecting group, duplicate the common term, applying the rule X+X = X:

  f = A'BC' + **A'BC+A'BC** + AB'C + **ABC+ABC**

- Group the similar terms:

  f = (A'BC'+A'BC) + (A'BC+ABC) + (AB'C+ABC)

Circuit Minimization

- Group the similar terms:

  $f = (A'BC'+A'BC) + (A'BC+ABC) + (AB'C+ABC)$

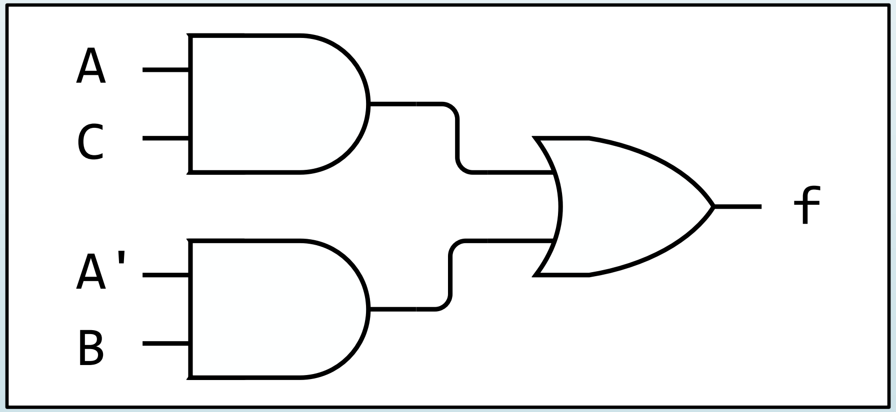
- Apply the rule $XY+XY' = X(Y+Y') = X$, then:

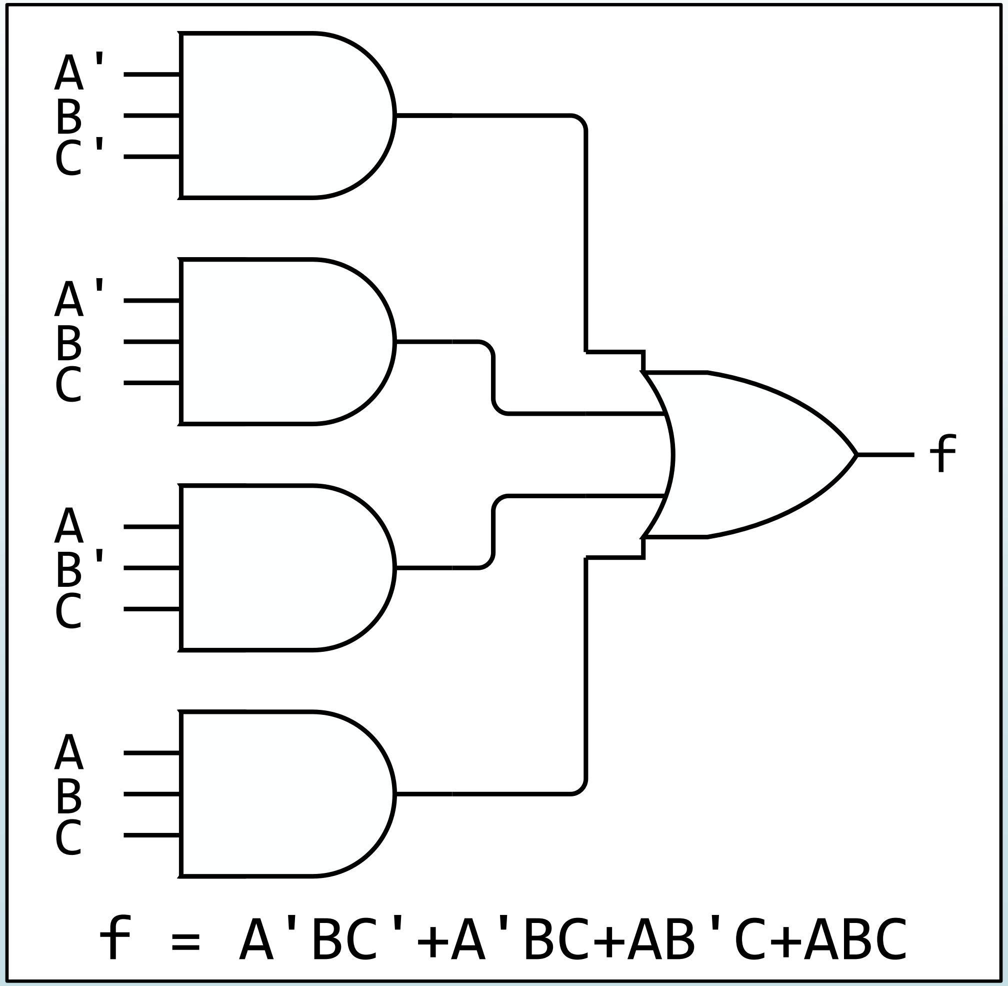
  $f = A'B(C'+C) + (A'+A)BC + A(B'+B)C$

- Then: $f = A'B + BC + AC$
- Remember the redundancy rule: $XY+X'Z+YZ = XY+X'Z$
- Then: $f = AC + A'B + BC = AC + A'B$

Circuit Minimization

- The final simplified function:
- Then: f = AC + A'B
- The circuit implementation:

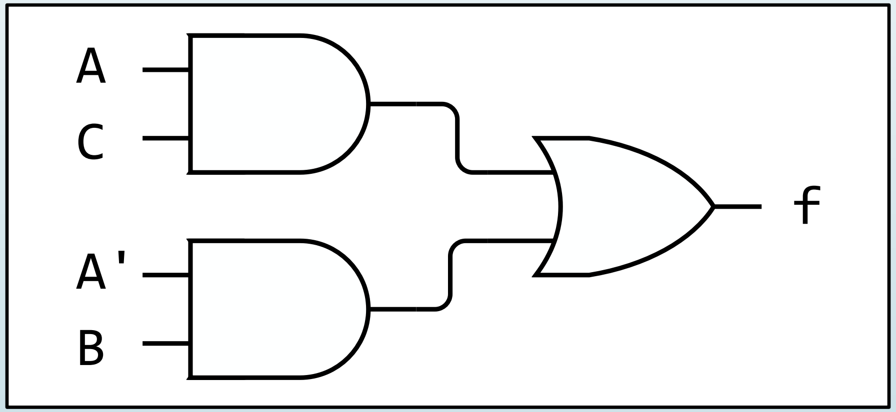




- Compare to the original one.

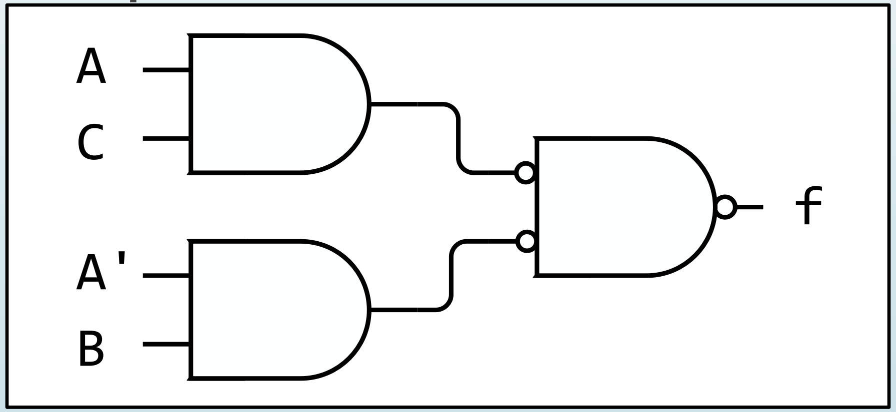
Boolean Algebra

- To further reduce the cost: use graphical DeMorgans law on the OR gate:



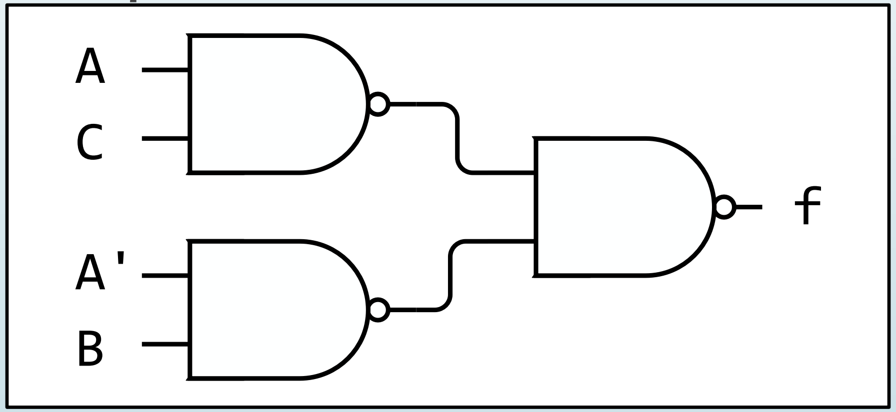
Boolean Algebra

- To further reduce the cost: use graphical DeMorgans law on the OR gate:
  - Change the OR to AND and complement all its inputs and output.



Boolean Algebra

- To further reduce the cost: use graphical DeMorgans law on the OR gate:
  - Change the OR to AND and complement all its inputs and output.



  - This is a NAND-NAND circuit that is cheaper than the AND-OR circuit, smaller and consumes less power.

Karnaugh Map

Boolean Algebra Minimization

- Recall the prime number detector designed before:

$$f = \sum_{A,B,C} m(2,3,5,7)$$

- Resolve the minterms:

```
f = A'BC' + A'BC + AB'C + ABC

f = A'B + BC + AC
```

- Search for any two similar terms except one complement.
- Apply the rule XY+XY' = X(Y+Y') = X

Karnaugh Map

- The previous example contains only 3 input variables.
- The simplification will be harder and longer with increasing the number of input variables.

- The American scientist Maurice Karnaugh invented his map for doing this grouping visually for faster simplification.

Karnaugh Map

- Assume 3-input variables. Consider the 8 minterms list.
- Assume a table of 8 cells.

A	B	C	Minterms
0	0	0	$A'B'C'=m_0$
0	0	1	$A'B'C=m_1$
0	1	0	$A'BC'=m_2$
0	1	1	$A'BC=m_3$
1	0	0	$AB'C'=m_4$
1	0	1	$AB'C=m_5$
1	1	0	$ABC'=m_6$
1	1	1	$ABC=m_7$

Karnaugh Map

- Assume that we will put m_0 in the first cell. Find the similar minterms except one complement ??

0 A'B'C'	

A	B	C	Minterms
0	0	0	$A'B'C'=m_0$
0	0	1	$A'B'C=m_1$
0	1	0	$A'BC'=m_2$
0	1	1	$A'BC=m_3$
1	0	0	$AB'C'=m_4$
1	0	1	$AB'C=m_5$
1	1	0	$ABC'=m_6$
1	1	1	$ABC=m_7$

Karnaugh Map

- You will find m_1. Put it beside m_0. Search for other similar minterm.

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

A	B	C	Minterms
0	0	0	$A'B'C'=m_0$
0	0	1	$A'B'C=m_1$
0	1	0	$A'BC'=m_2$
0	1	1	$A'BC=m_3$
1	0	0	$AB'C'=m_4$
1	0	1	$AB'C=m_5$
1	1	0	$ABC'=m_6$
1	1	1	$ABC=m_7$

Karnaugh Map

- You will find m_2. Put it beside m_0. Search for other similar minterm.

0 A'B'C'	
1 A'B'C	
2 A'BC'	

A	B	C	Minterms
0	0	0	A'B'C'=m_0
0	0	1	A'B'C=m_1
0	1	0	A'BC'=m_2
0	1	1	A'BC=m_3
1	0	0	AB'C'=m_4
1	0	1	AB'C=m_5
1	1	0	ABC'=m_6
1	1	1	ABC=m_7

Karnaugh Map

- You will find m_4. Put it beside m_0. No more minterms similar to m_0.

0 A'B'C'	4 AB'C'
1 A'B'C	
2 A'BC'	

A	B	C	Minterms
0	0	0	A'B'C'=m_0
0	0	1	A'B'C=m_1
0	1	0	A'BC'=m_2
0	1	1	A'BC=m_3
1	0	0	AB'C'=m_4
1	0	1	AB'C=m_5
1	1	0	ABC'=m_6
1	1	1	ABC=m_7

Karnaugh Map

- Similarly, fill in all the table so that each two adjacent cells are different in only one complement.
- Check them …

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

A	B	C	Minterms
0	0	0	$A'B'C'=m_0$
0	0	1	$A'B'C=m_1$
0	1	0	$A'BC'=m_2$
0	1	1	$A'BC=m_3$
1	0	0	$AB'C'=m_4$
1	0	1	$AB'C=m_5$
1	1	0	$ABC'=m_6$
1	1	1	$ABC=m_7$

Karnaugh Map

- Note the common letter in each row and column. Check it ...

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

A	B	C	Minterms
0	0	0	A'B'C'=m_0
0	0	1	A'B'C=m_1
0	1	0	A'BC'=m_2
0	1	1	A'BC=m_3
1	0	0	AB'C'=m_4
1	0	1	AB'C=m_5
1	1	0	ABC'=m_6
1	1	1	ABC=m_7

Karnaugh Map

- Note the common letter in each row and column. Check it …

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

A	B	C	Minterms
0	0	0	$A'B'C'=m_0$
0	0	1	$A'B'C=m_1$
0	1	0	$A'BC'=m_2$
0	1	1	$A'BC=m_3$
1	0	0	$AB'C'=m_4$
1	0	1	$AB'C=m_5$
1	1	0	$ABC'=m_6$
1	1	1	$ABC=m_7$

Karnaugh Map

- The adjacent cells are different in only one complement. So, we can group any two adjacent cells.

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

A	B	C	Minterms
0	0	0	A'B'C'=m_0
0	0	1	A'B'C=m_1
0	1	0	A'BC'=m_2
0	1	1	A'BC=m_3
1	0	0	AB'C'=m_4
1	0	1	AB'C=m_5
1	1	0	ABC'=m_6
1	1	1	ABC=m_7

Karnaugh Map

- Example: A'B'C+AB'C = B'C
- The result is the common letters in the group.

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

A	B	C	Minterms
0	0	0	A'B'C'=m_0
0	0	1	A'B'C=m_1
0	1	0	A'BC'=m_2
0	1	1	A'BC=m_3
1	0	0	AB'C'=m_4
1	0	1	AB'C=m_5
1	1	0	ABC'=m_6
1	1	1	ABC=m_7

Karnaugh Map

- We can also group 4 cells. Example:

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

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

A	B	C	Minterms
0	0	0	A'B'C'= m_0
0	0	1	A'B'C= m_1
0	1	0	A'BC'= m_2
0	1	1	A'BC= m_3
1	0	0	AB'C'= m_4
1	0	1	AB'C= m_5
1	1	0	ABC'= m_6
1	1	1	ABC= m_7

Karnaugh Map

- This table is the Karnaugh Map (K-map).
- We distributed the input side of truth table.

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

A	B	C	Minterms
0	0	0	A'B'C'=m_0
0	0	1	A'B'C=m_1
0	1	0	A'BC'=m_2
0	1	1	A'BC=m_3
1	0	0	AB'C'=m_4
1	0	1	AB'C=m_5
1	1	0	ABC'=m_6
1	1	1	ABC=m_7

Karnaugh Map

- Replace each input with its corresponding output.
- Note the order of ones and zeros.

		A'	A	
B'		0: 0	4: 0	C'
B'		1: 0	5: 1	C
B		3: 1	7: 1	C
B		2: 1	6: 0	C'

A	B	C	Minterms	f
0	0	0	$A'B'C'=m_0$	0
0	0	1	$A'B'C=m_1$	0
0	1	0	$A'BC'=m_2$	1
0	1	1	$A'BC=m_3$	1
1	0	0	$AB'C'=m_4$	0
1	0	1	$AB'C=m_5$	1
1	1	0	$ABC'=m_6$	0
1	1	1	$ABC=m_7$	1

Karnaugh Map

- Previously, we considered only the active minterms.
- Now, we will consider only the ones inside the map.

		A'	A	
B'		0: 0	4: 0	C'
B'		1: 0	5: 1	C
B		3: 1	7: 1	C
B		2: 1	6: 0	C'

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

Karnaugh Map

- Give the letters of this group:
- `f = ??`

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

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

Karnaugh Map

- Give the letters of this group:
- f = AC + ??

	A'	A	
B'	0: 0	4: 0	C'
B'	1: 0	5: 1	C
B	3: 1	7: 1	C
B	2: 1	6: 0	C'

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

Karnaugh Map

- This is the only left group, do not consider it. We have all 1s.
- f = AC + A'B

	A'	A	
B'	(0) 0	(4) 0	C'
B'	(1) 0	(5) 1	C
B	(3) 1	(7) 1	C
B	(2) 1	(6) 0	C'

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

Karnaugh Map

- This is the final simplified equation. Compare with before.
- f = AC + A'B

		A'	A	
B'		0: 0	4: 0	C'
B'		1: 0	5: 1	C
B		3: 1	7: 1	C
B		2: 1	6: 0	C'

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

Steps Summary

- Given a truth table, it is required to find its equation.
- Draw the Karnaugh Map for 3-inputs

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

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

Steps Summary

- Given a truth table, it is required to find its equation.
- Draw the Karnaugh Map for 3-inputs
- Put the function output inside the cells.
- Find the minimum number of the largest groups that cover all the ones.
- Groups must be 2^n elements, i.e.: 1,2,4,8
- The equations is the summation of these groups letters.
- $f = AC + A'B$

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

	A'	A	
B'	0: 0	4: 0	C'
	1: 0	5: 1	C
B	3: 1	7: 1	
	2: 1	6: 0	C'

Grouping Notes

- Consider this case. You have 2 options for grouping.
- Select the minimum number of the largest groups.
- This leads to the result without extra simplification.
- $f = AB' + AB = A$

	A'	A	
B'	(0) 0	(4) 1	C'
	(1) 0	(5) 1	C
B	(3) 0	(7) 1	
	(2) 0	(6) 1	C'

$f = A$

	A'	A	
B'	(0) 0	(4) 1	C'
	(1) 0	(5) 1	C
B	(3) 0	(7) 1	
	(2) 0	(6) 1	C'

Grouping Notes

- Consider this case. You have 2 options for grouping.
- Select the minimum number of the largest groups.
- This leads to the result without extra simplification.
- f= ?+?

		A'	A	
B'		0: 0	4: 1	C'
B'		1: 1	5: 1	C
B		3: 1	7: 1	C
B		2: 0	6: 1	C'

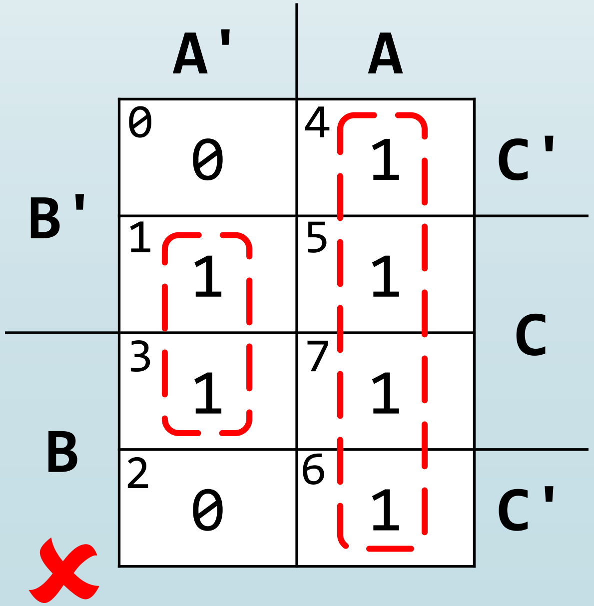
✗

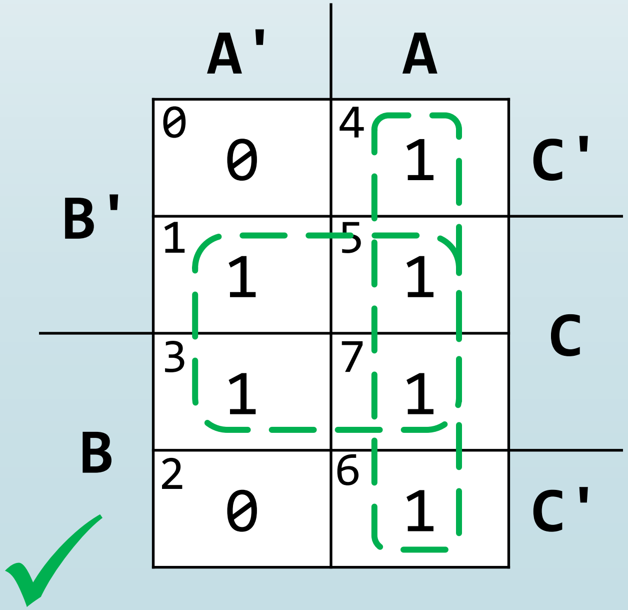
		A'	A	
B'		0: 0	4: 1	C'
B'		1: 1	5: 1	C
B		3: 1	7: 1	C
B		2: 0	6: 1	C'

✓

Grouping Notes

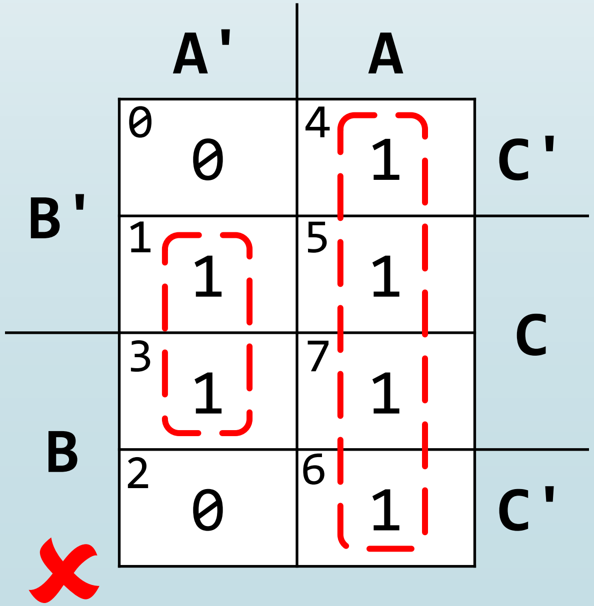
- Consider this case. You have 2 options for grouping.
- Select the minimum number of the largest groups.
- This leads to the result without extra simplification.
- f= A+A'C   f = ?+?

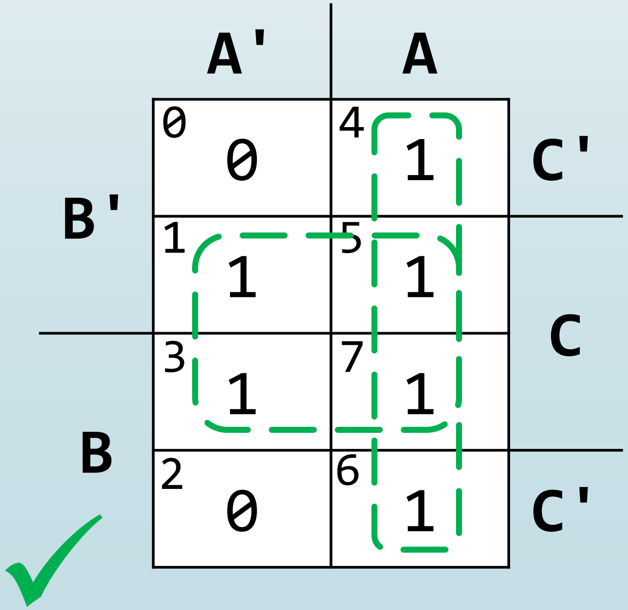




Grouping Notes

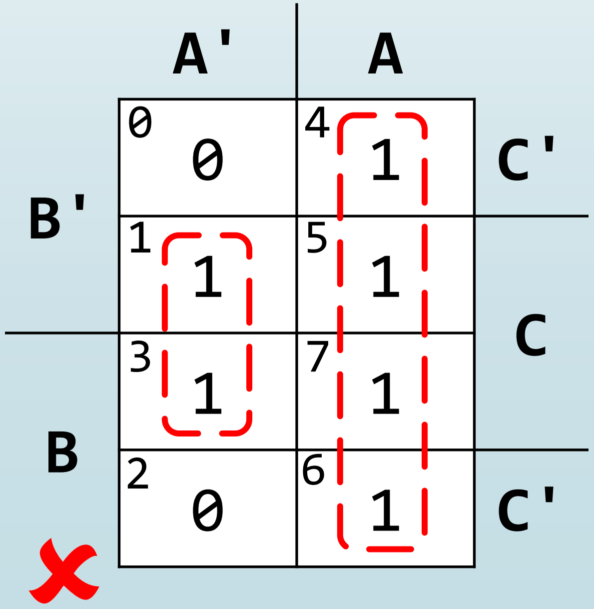
- Consider this case. You have 2 options for grouping.
- Select the minimum number of the largest groups.
- This leads to the result without extra simplification.
- f= A+A'C f = A+C

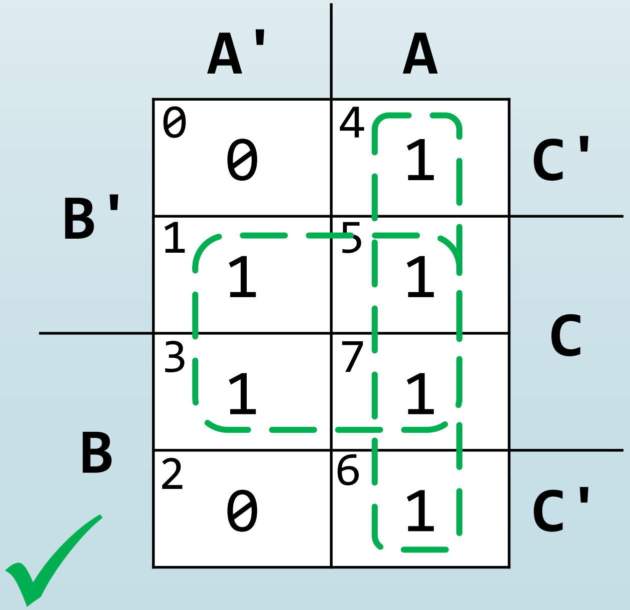




Grouping Notes

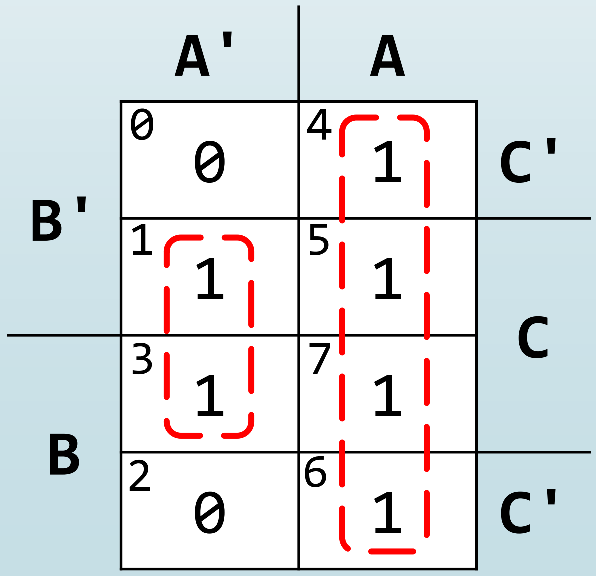
- Consider this case. You have 2 options for grouping.
- Select the minimum number of the largest groups.
- This leads to the result without extra simplification.
- f= A+A'C =(A+A')(A+C)=A+C   f = A+C

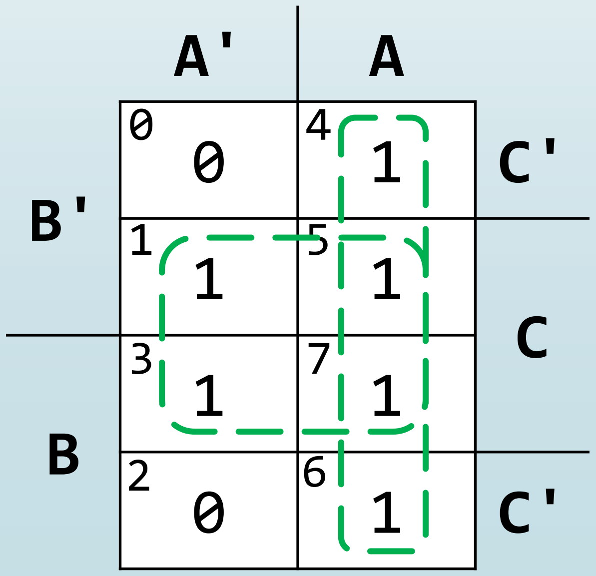




Grouping Notes

- Consider this case. You have 2 options for grouping.
- Select the minimum number of the largest groups.
- This leads to the result without extra simplification.
- f= A+A'C =(A+A')(A+C)=A+C   f = A+C





Two Inputs

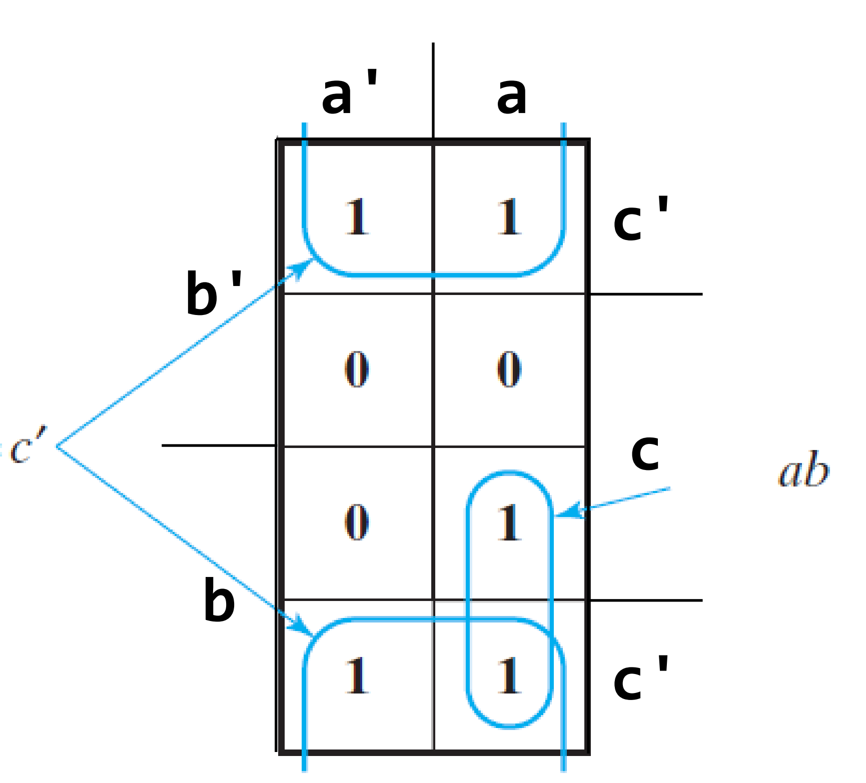
- In case of 2- input functions. It might be easier to resolve it without Karnaugh map.
- Consider the given truth table.
- Using minterms only: f= A'B'+A'B = A'
- However, you could use the Karnaugh map in this way:
  - Draw the 2-inputs K-map like this:
  - f = A'

A	B	f
0	0	1
0	1	1
1	0	0
1	1	0

	A'	A
B'	0 1	2 0
B	1 1	3 0

Examples from Textbook

- f = ab + c'



Examples from Textbook

- The clear cells are zeros by default.
- f = xy+x'z+yz = xy+x'z   f= x'z+xy