# Digital Engineering


**Assoc. Prof. Osama Elnahas, Dr. Dina Abdelhafiz**

**Dr. Bassant Tolba, Dr. Radwa Rady**


Second Year –Information Technology Program

Fall 2025

# Lecture 4

# Minterms and Maxterms

# Minterms and Maxterms

- Remember, we are now designing a circuit. It is required to find its circuit diagram.
- Previously, we got the truth table which is the first step in designing a logic circuit.
- The next step is to convert this truth table into an equation, so that we could implement it using the basic gates.
- This is done using minterms and maxterms.
- The minterms and maxterms are ways for expressing an input case as Boolean function.

# Minterms and Maxterms

- Given a truth table. It is required to find the output equation.
- Example: The prime number detector circuit.

| 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 |

# Minterms and Maxterms

- Ignore the output side.

| 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 |

# Minterms and Maxterms

- How to make an input results in 1 on AND gate?

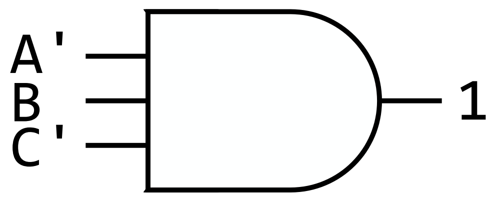
| A | B | C |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 0 | 1 |
| 0 | 1 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 0 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |
| 1 | 1 | 1 |

# Minterms and Maxterms

- How to make an input results in 1 on AND gate?
- Example: When A=0, B=1, C=0, we need the AND output =1

| 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 |

A'BC'



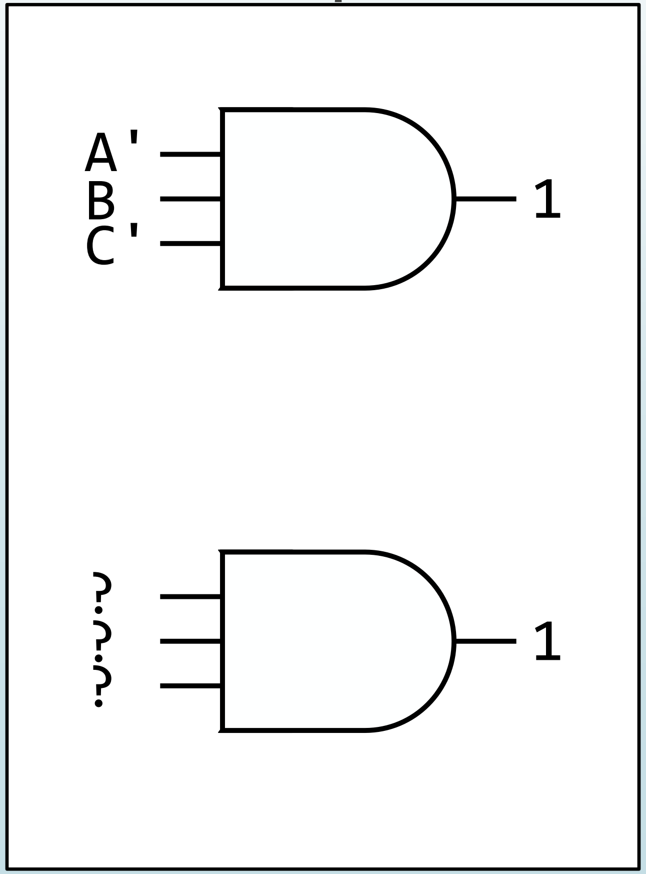
# Minterms and Maxterms

- How to make an input results in 1 on AND gate?
- Example: When A=1, B=0, C=1, we need the AND output =1

| 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 |

A'BC'

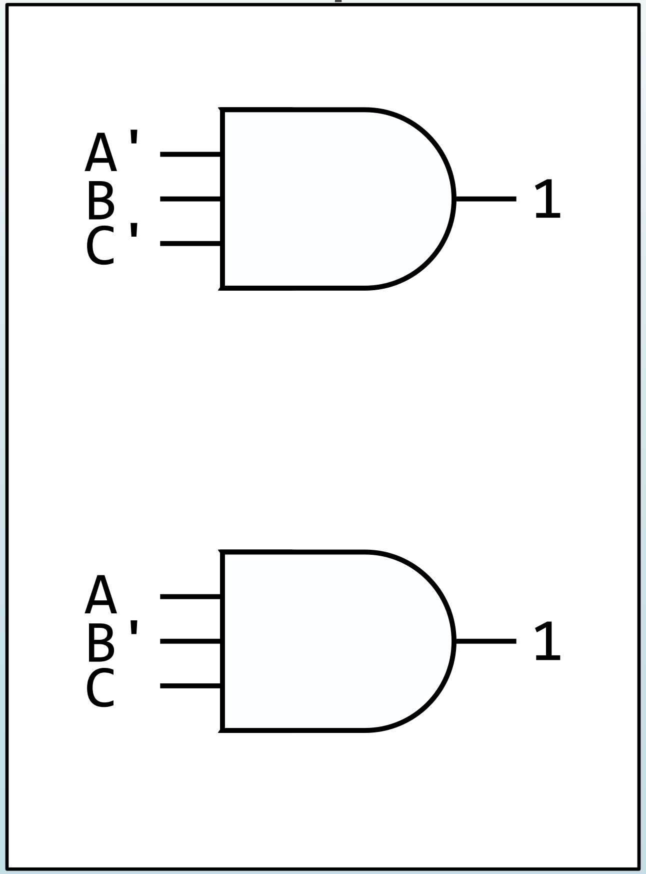
??



# Minterms and Maxterms

- How to make an input results in 1 on AND gate?
- Example: When A=1, B=0, C=1, we need the AND output =1

| A | B | C |  |
|---|---|---|---|
| 0 | 0 | 0 |  |
| 0 | 0 | 1 |  |
| 0 | 1 | 0 | A'BC' |
| 0 | 1 | 1 |  |
| 1 | 0 | 0 |  |
| 1 | 0 | 1 | AB'C |
| 1 | 1 | 0 |  |
| 1 | 1 | 1 |  |



# Minterms and Maxterms

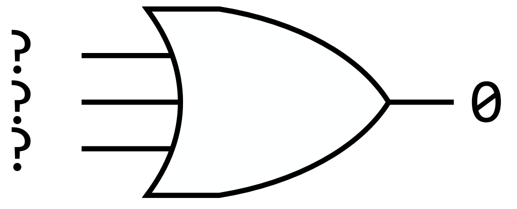
- How to make an input results in 1 on AND gate?
- Repeat for all the input combinations.

| A | B | C | |
|---|---|---|---|
| 0 | 0 | 0 | A'B'C' |
| 0 | 0 | 1 | A'B'C |
| 0 | 1 | 0 | A'BC' |
| 0 | 1 | 1 | A'BC |
| 1 | 0 | 0 | AB'C' |
| 1 | 0 | 1 | AB'C |
| 1 | 1 | 0 | ABC' |
| 1 | 1 | 1 | ABC |

# Minterms and Maxterms

- How to make an input results in 0 on OR gate?
- Example: When A=0, B=0, C=1, we need the OR output =1

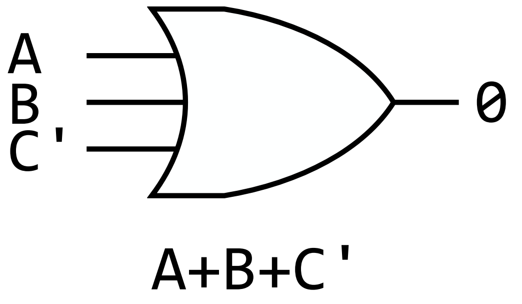
| A | B | C | |
|---|---|---|---|
| 0 | 0 | 0 | A'B'C' |
| 0 | 0 | 1 | A'B'C |
| 0 | 1 | 0 | A'BC' |
| 0 | 1 | 1 | A'BC |
| 1 | 0 | 0 | AB'C' |
| 1 | 0 | 1 | AB'C |
| 1 | 1 | 0 | ABC' |
| 1 | 1 | 1 | ABC |



# Minterms and Maxterms

- How to make an input results in 0 on OR gate?
- Example: When A=0, B=0, C=1, we need the OR output =1

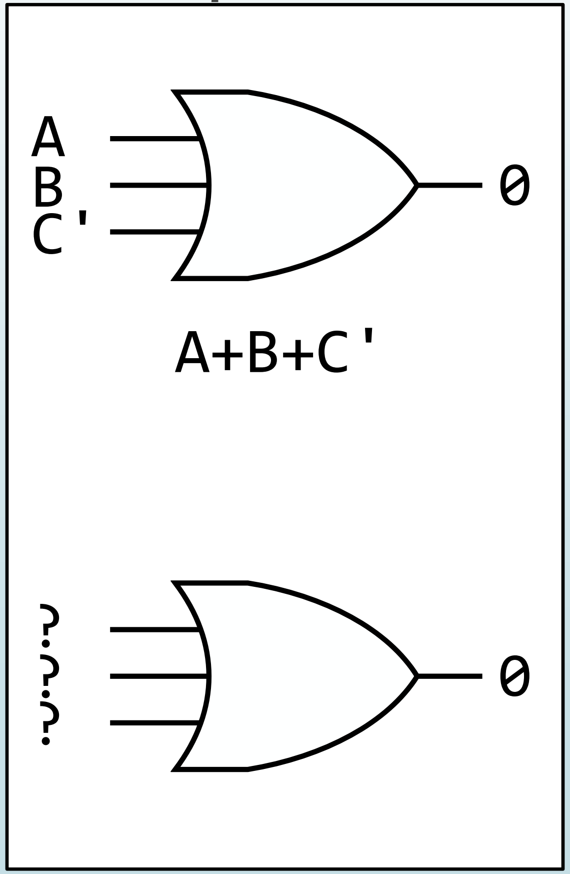
| A | B | C | |
|---|---|---|---|
| 0 | 0 | 0 | A'B'C' |
| 0 | 0 | 1 | A'B'C |
| 0 | 1 | 0 | A'BC' |
| 0 | 1 | 1 | A'BC |
| 1 | 0 | 0 | AB'C' |
| 1 | 0 | 1 | AB'C |
| 1 | 1 | 0 | ABC' |
| 1 | 1 | 1 | ABC |



# Minterms and Maxterms

- How to make an input results in 0 on OR gate?
- Example: When A=1, B=1, C=0, we need the OR output =1

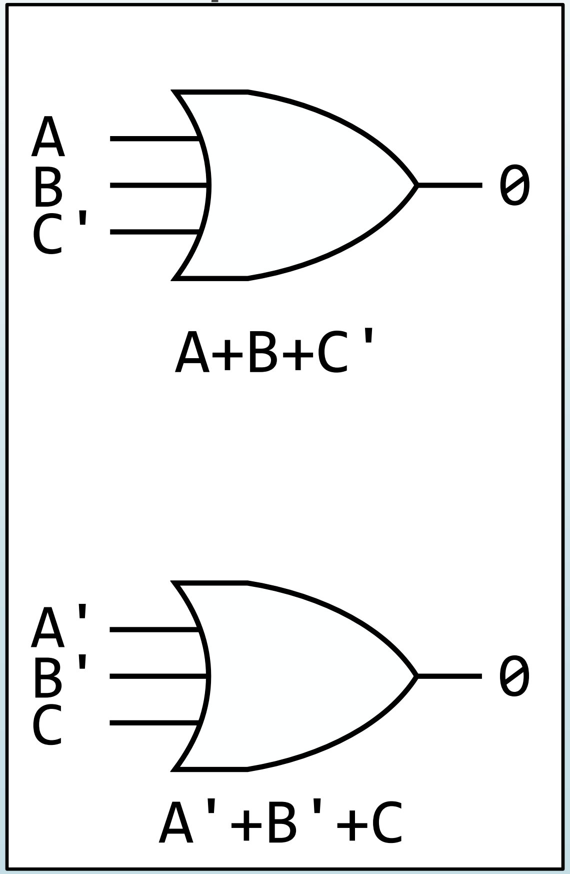
| A | B | C | |
|---|---|---|---|
| 0 | 0 | 0 | A'B'C' |
| 0 | 0 | 1 | A'B'C |
| 0 | 1 | 0 | A'BC' |
| 0 | 1 | 1 | A'BC |
| 1 | 0 | 0 | AB'C' |
| 1 | 0 | 1 | AB'C |
| 1 | 1 | 0 | ABC' |
| 1 | 1 | 1 | ABC |



# Minterms and Maxterms

- How to make an input results in 0 on OR gate?
- Example: When A=1, B=1, C=0, we need the OR output =1

| A | B | C | |
|---|---|---|---|
| 0 | 0 | 0 | A'B'C' |
| 0 | 0 | 1 | A'B'C |
| 0 | 1 | 0 | A'BC' |
| 0 | 1 | 1 | A'BC |
| 1 | 0 | 0 | AB'C' |
| 1 | 0 | 1 | AB'C |
| 1 | 1 | 0 | ABC' |
| 1 | 1 | 1 | ABC |



# Minterms and Maxterms

- How to make an input results in 0 on OR gate?
- Repeat for all the input combinations.

| A | B | C | | |
|---|---|---|---|---|
| 0 | 0 | 0 | A'B'C' | A+B+C |
| 0 | 0 | 1 | A'B'C | A+B+C' |
| 0 | 1 | 0 | A'BC' | A+B'+C |
| 0 | 1 | 1 | A'BC | A+B'+C' |
| 1 | 0 | 0 | AB'C' | A'+B+C |
| 1 | 0 | 1 | AB'C | A'+B+C' |
| 1 | 1 | 0 | ABC' | A'+B'+C |
| 1 | 1 | 1 | ABC | A'+B'+C' |

# Minterms and Maxterms

- Names of terms:

| A | B | C | | |
|---|---|---|---|---|
| 0 | 0 | 0 | A'B'C' | A+B+C |
| 0 | 0 | 1 | A'B'C | A+B+C' |
| 0 | 1 | 0 | A'BC' | A+B'+C |
| 0 | 1 | 1 | A'BC | A+B'+C' |
| 1 | 0 | 0 | AB'C' | A'+B+C |
| 1 | 0 | 1 | AB'C | A'+B+C' |
| 1 | 1 | 0 | ABC' | A'+B'+C |
| 1 | 1 | 1 | ABC | A'+B'+C' |

# Minterms and Maxterms

- The product terms are called Minterms.
- The sum terms are called Maxterms.

| A | B | C | Minterms | Maxterms |
|---|---|---|---|---|
| 0 | 0 | 0 | A'B'C' | A+B+C |
| 0 | 0 | 1 | A'B'C | A+B+C' |
| 0 | 1 | 0 | A'BC' | A+B'+C |
| 0 | 1 | 1 | A'BC | A+B'+C' |
| 1 | 0 | 0 | AB'C' | A'+B+C |
| 1 | 0 | 1 | AB'C | A'+B+C' |
| 1 | 1 | 0 | ABC' | A'+B'+C |
| 1 | 1 | 1 | ABC | A'+B'+C' |

# Minterms and Maxterms

- The product terms are called Minterms (m).
- The sum terms are called Maxterms (M).

| A | B | C | Minterms | Maxterms |
|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ |

# Minterms

- Till now we just give an expression to describe the input.
- Now, it is required to find the equation of the circuit output.

| A | B | C | Minterms | Maxterms |
|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ |

# Minterms

- Now, consider the complete truth table.
- Example: The prime number detector circuit.

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Minterms

- Look at all the input cases that results in 1. The output f will be 1 in case of ***the 2nd input OR the 3rd OR the 5th OR 7th .***

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Minterms

- Convert this sentence into equation:

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

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Minterms

- Convert this sentence into equation:

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

- Note that we used the minterm because it result in 1, while the maxterm result in 0.
- Then, the output f will be 1 in case of *the* $2^{nd}$ *input OR the* $3^{rd}$ *OR the* $5^{th}$ *OR* $7^{th}$ *.*
- This is the required expression.
- This is called **Sum of Products** form.
- Each of the 4 minterms is called **product term.** It is the logical product (AND) of many variables.

# Sum of Products

- Try all the inputs (prime numbers) and check the output.

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

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Sum of Products

- Try all the inputs (prime numbers) and check the output.

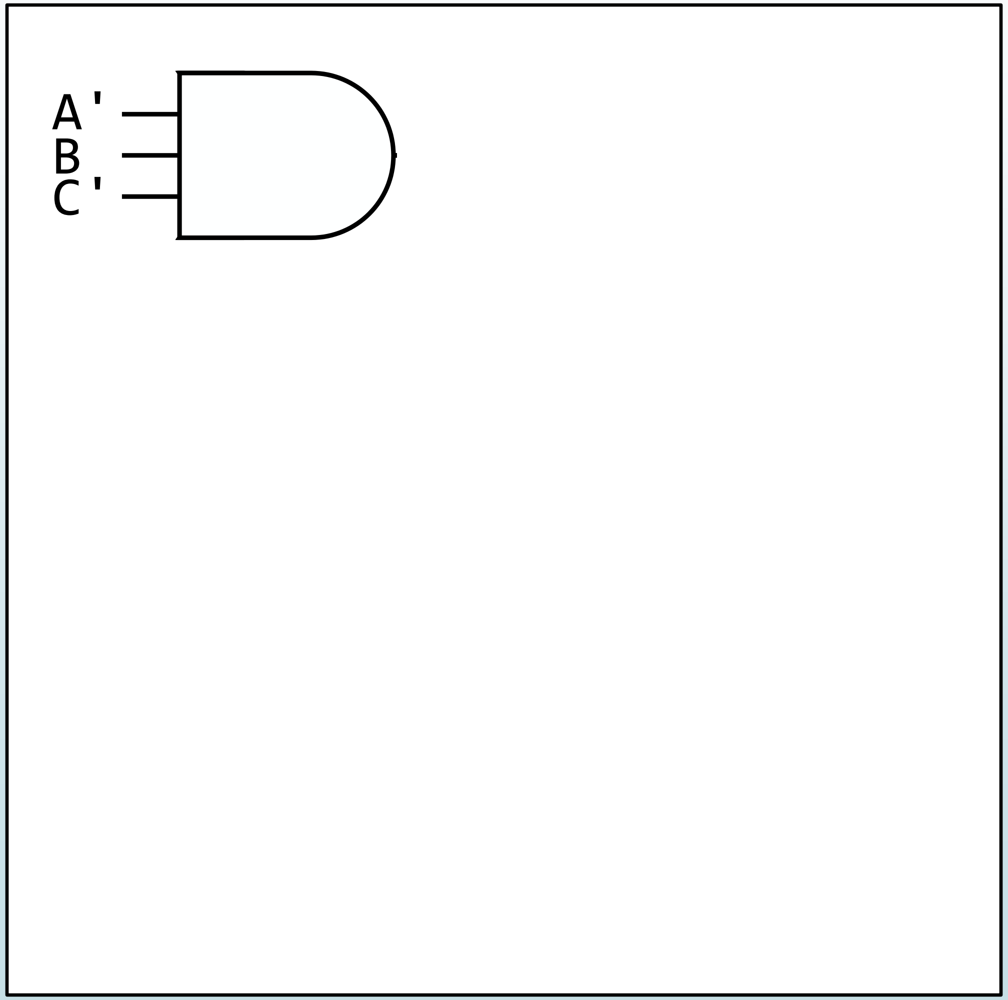
$$f = (A'BC')+(A'BC)+(AB'C)+(ABC)$$

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Logic Circuit

- Convert this equation into circuit:

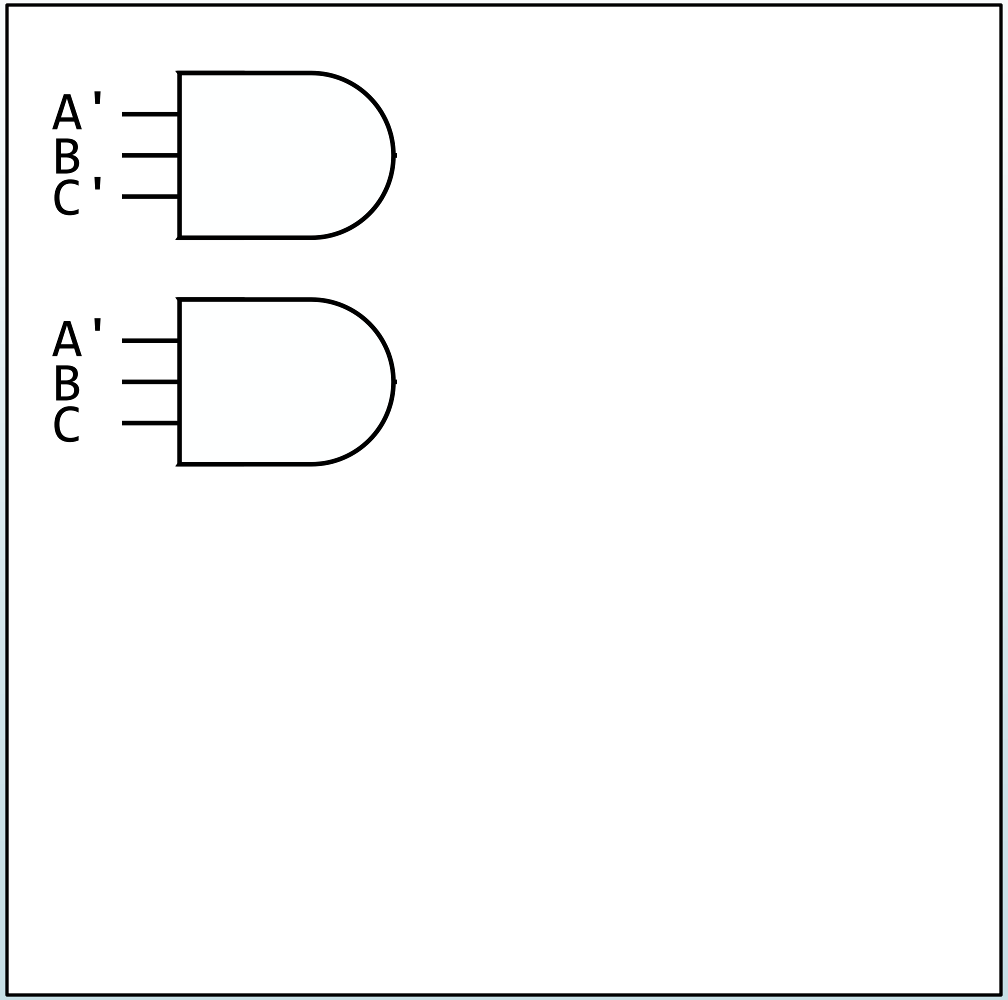
  f = **A'BC'**+A'BC+AB'C+ABC



# Logic Circuit

- Convert this equation into circuit:

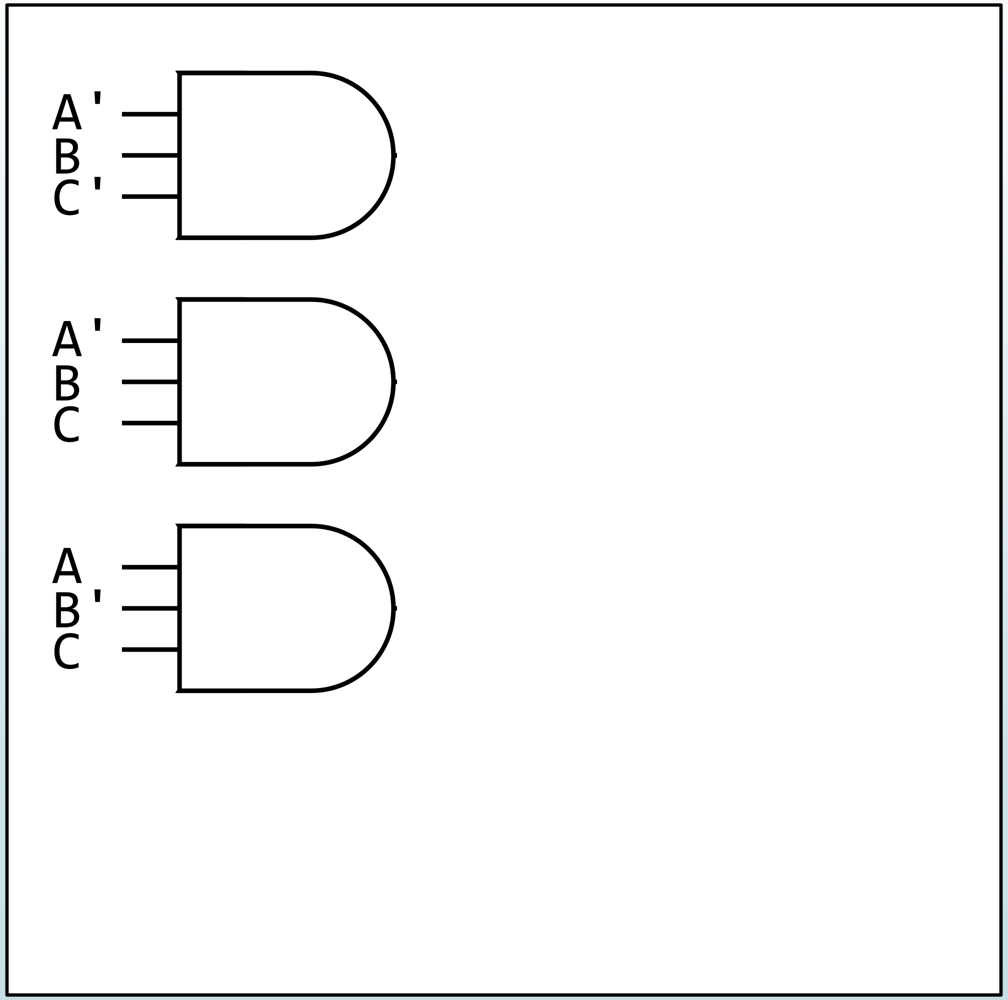
  f = A'BC'+**A'BC**+AB'C+ABC



# Logic Circuit

- Convert this equation into circuit:

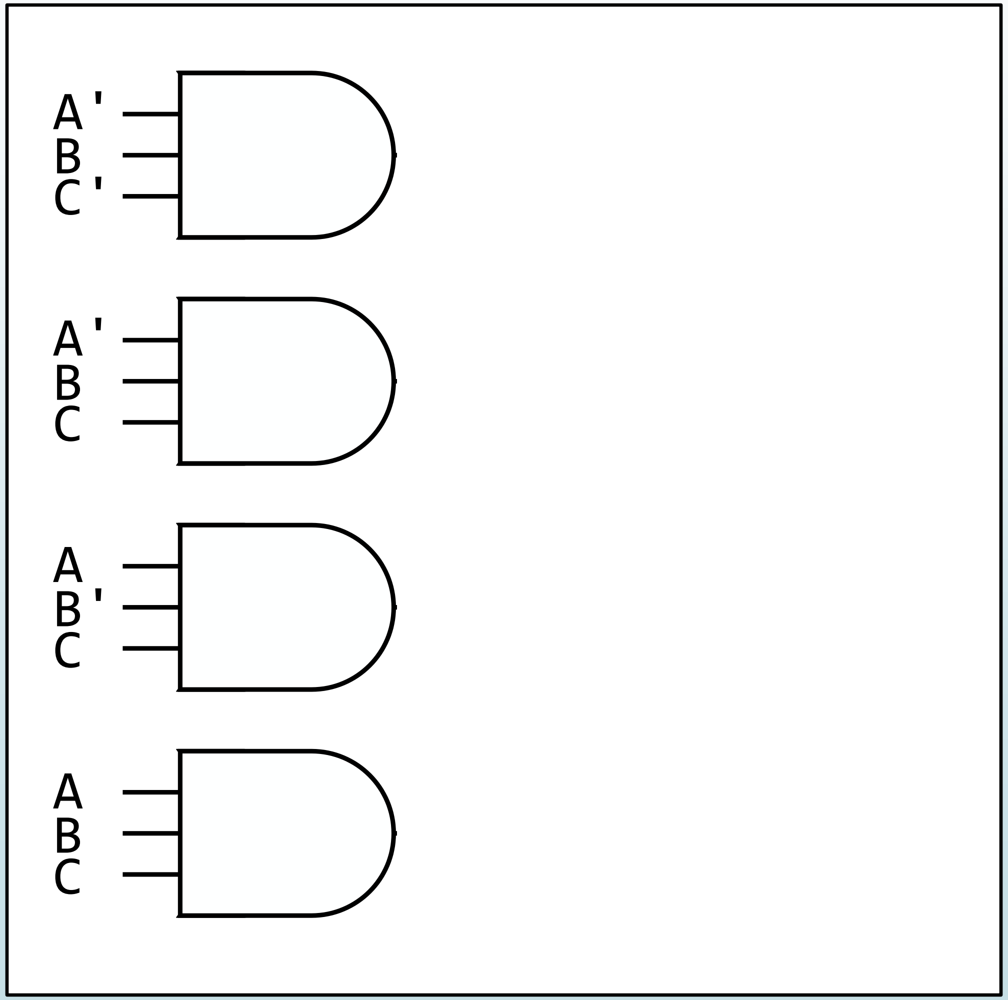
  f = A'BC'+A'BC+**AB'C**+ABC



# Logic Circuit

- Convert this equation into circuit:

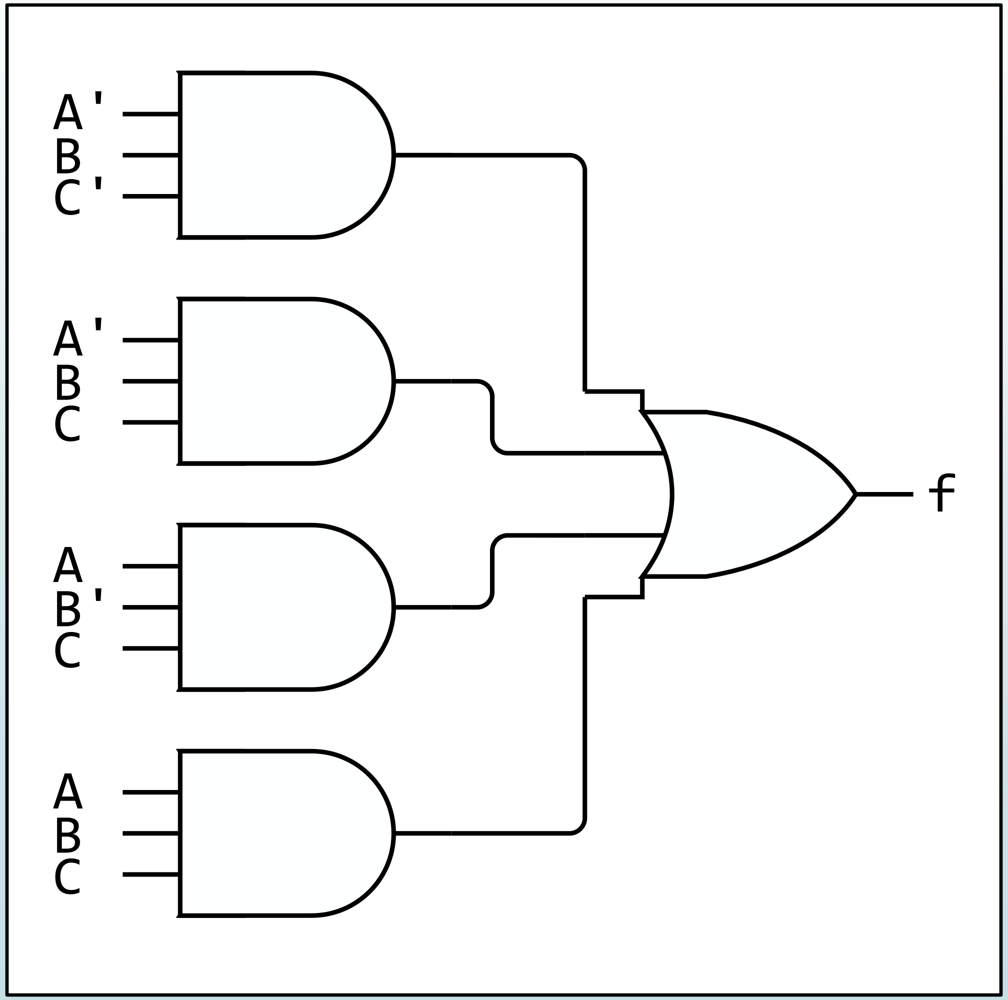
  f = A'BC'+A'BC+AB'C+**ABC**



# Logic Circuit

- Convert this equation into circuit:

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

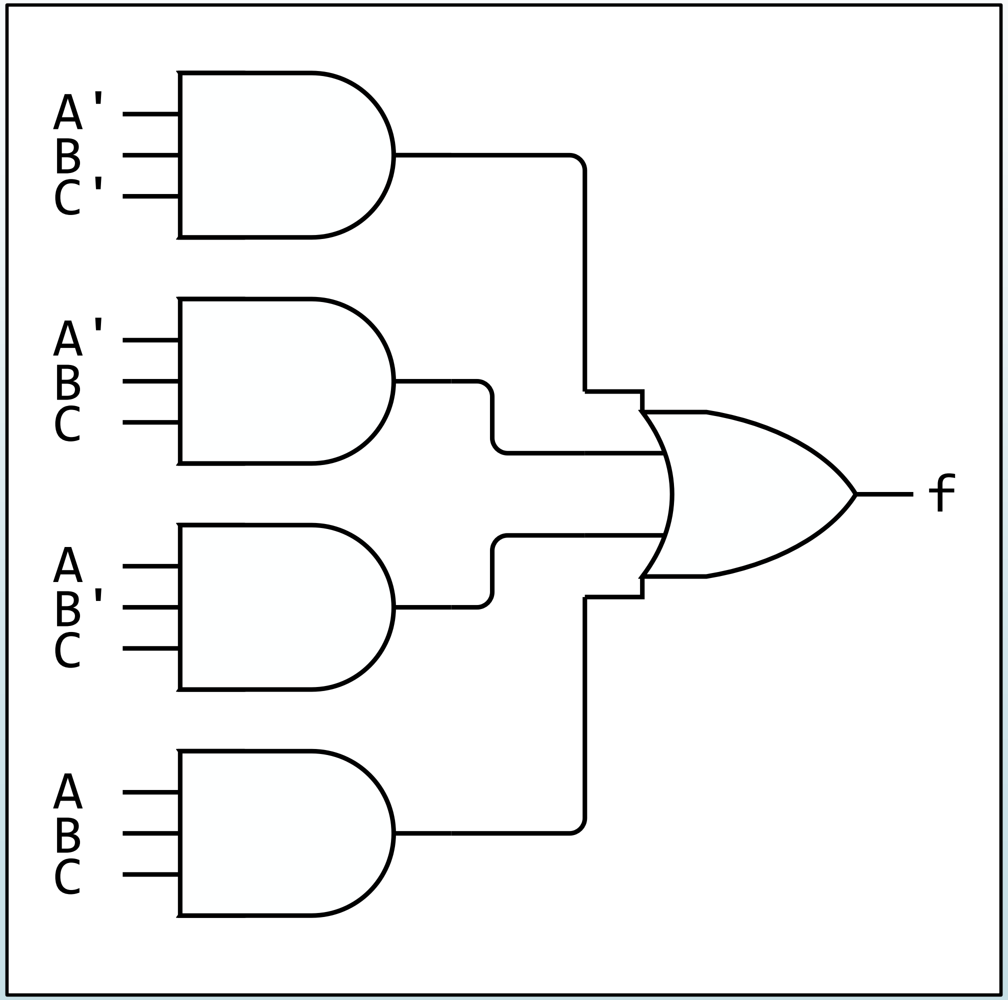


# Logic Circuit

- Convert this equation into circuit:

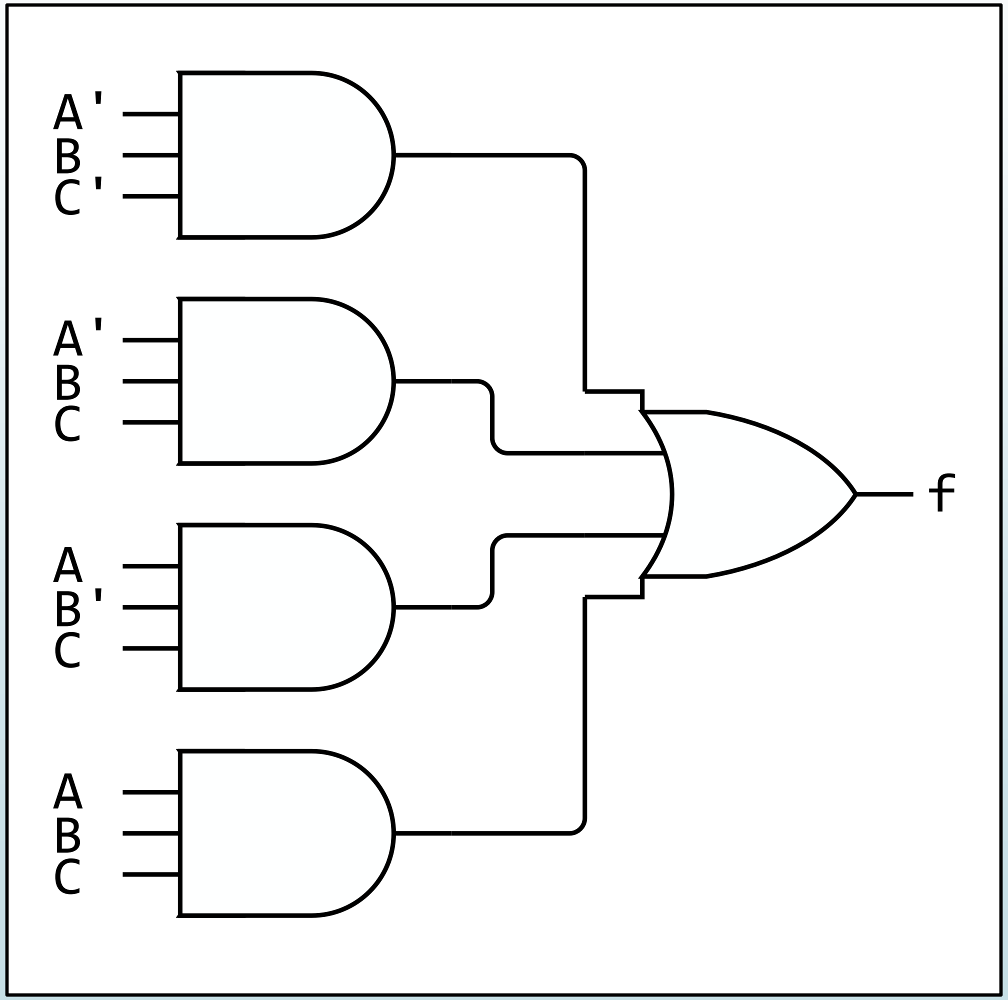
  f = A'BC'+A'BC+AB'C+ABC

- Now, we designed a circuit for the prime number detector.
- However, the design is not perfect. It is too long and complex. It costs many gates.
- We will discuss how to minimize it later.



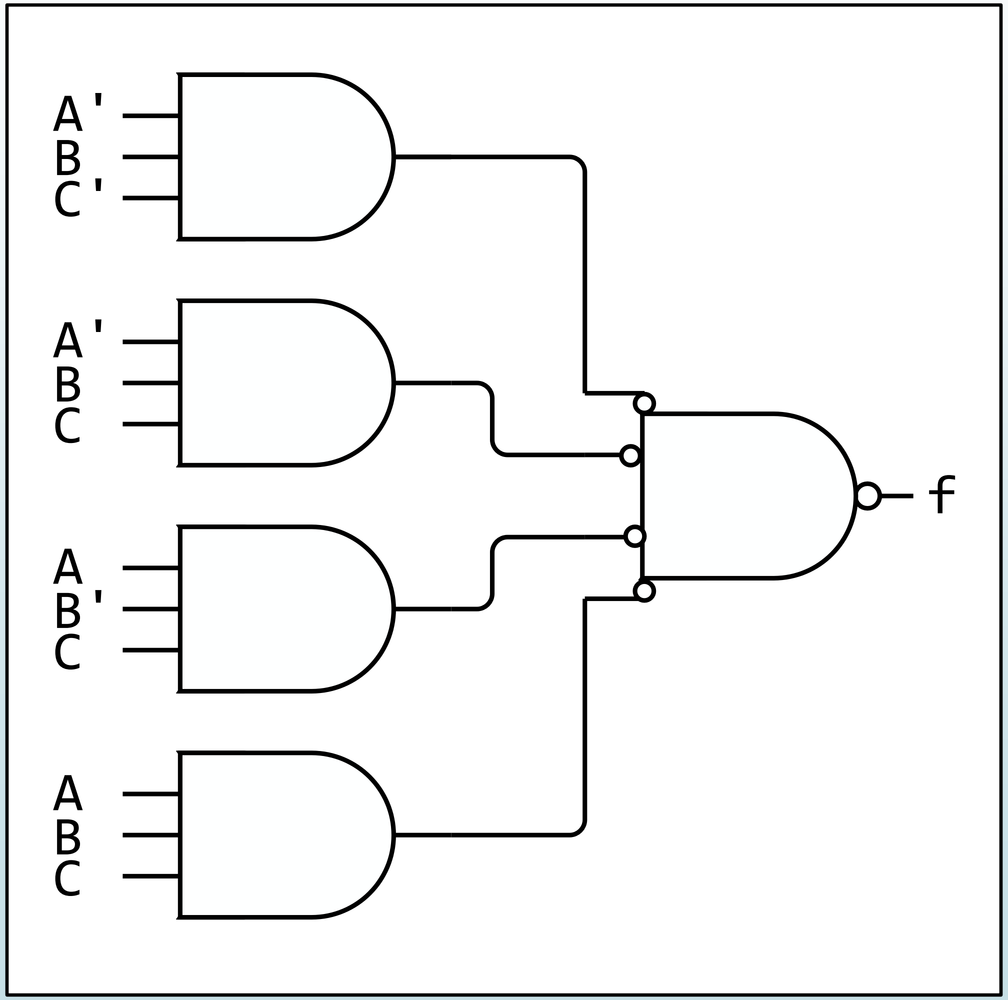
# Logic Circuit

- This is an AND-OR circuit.
- The implementation of sum of product equation.



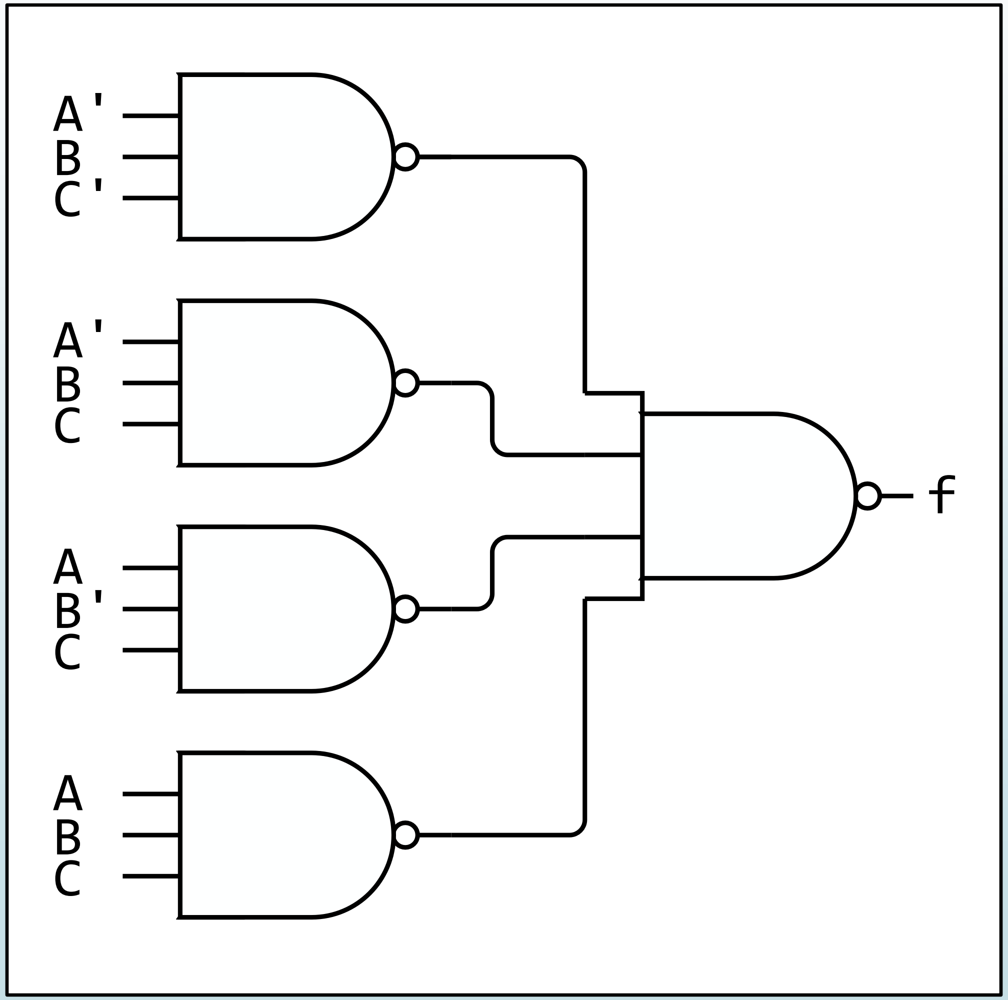
# Logic Circuit

- This is an AND-OR circuit.
- The implementation of sum of product equation.
- One cost reduction step is to apply the graphical DeMorgans law on the OR gate.
  - Change the OR to AND.
  - Complement all its inputs and output.



# Logic Circuit

- This is a NAND-NAND circuit equivalent to the same function.
  - Due to the gate components, the NAND gate is usually cheaper than the AND and OR gates for the same number of inputs.
  - Due to the gate components, the NAND gate is usually faster, smaller and consumes less power for the same number of inputs.



# Maxterms

- Another way to find the expression is to consider the input cases that makes f=0.

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Maxterms

- If any input of them happened, its maxterm is 0, then f=0.

$$f = (A+B+C)\cdot(A+B+C')\cdot(A'+B+C)\cdot(A'+B'+C)$$

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Maxterms

- Try all the inputs (prime number) and check the output.

$$f = (A+B+C)\cdot(A+B+C')\cdot(A'+B+C)\cdot(A'+B'+C)$$

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Maxterms

- Try all the inputs (prime number) and check the output.

$$f = (A+B+C)\cdot(A+B+C')\cdot(A'+B+C)\cdot(A'+B'+C)$$

| A | B | C | Minterms | Maxterms | f |
|---|---|---|---|---|---|
| 0 | 0 | 0 | $A'B'C'=m_0$ | $A+B+C=M_0$ | 0 |
| 0 | 0 | 1 | $A'B'C=m_1$ | $A+B+C'=M_1$ | 0 |
| 0 | 1 | 0 | $A'BC'=m_2$ | $A+B'+C=M_2$ | 1 |
| 0 | 1 | 1 | $A'BC=m_3$ | $A+B'+C'=M_3$ | 1 |
| 1 | 0 | 0 | $AB'C'=m_4$ | $A'+B+C=M_4$ | 0 |
| 1 | 0 | 1 | $AB'C=m_5$ | $A'+B+C'=M_5$ | 1 |
| 1 | 1 | 0 | $ABC'=m_6$ | $A'+B'+C=M_6$ | 0 |
| 1 | 1 | 1 | $ABC=m_7$ | $A'+B'+C'=M_7$ | 1 |

# Maxterms

- Try all the inputs (prime number) and check the output.

$$f = (A+B+C)\cdot(A+B+C')\cdot(A'+B+C)\cdot(A'+B'+C)$$

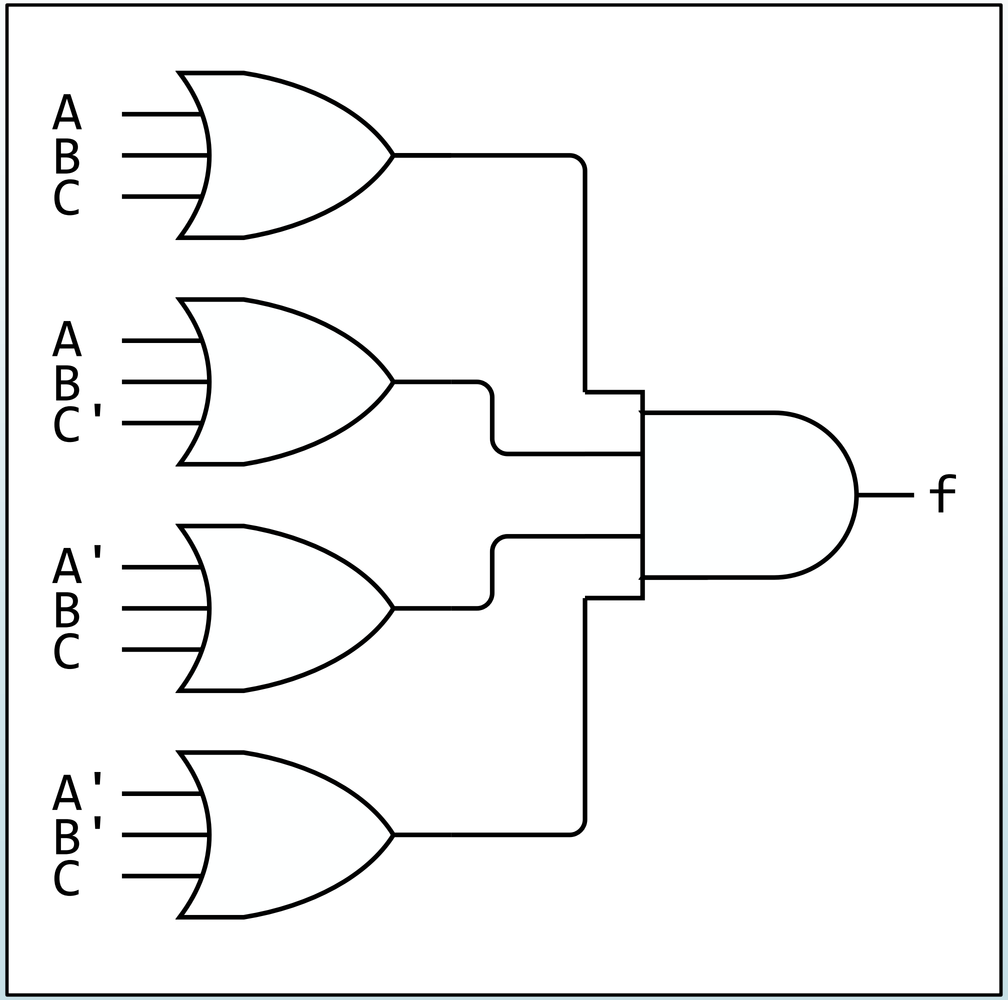
- The expression must be using the AND, because if there is one input of the AND = 0, its output will be 0.
- This is another expression, for the same logic function (detection of prime number).
- It results also in different circuit for the same function.
- This is called the **Product of Sum** form.
- Each of the 4 maxterms is called **sum term**. It is the logical sum (OR) of many variables.

# Logic Circuit

- Convert this equation to circuit:

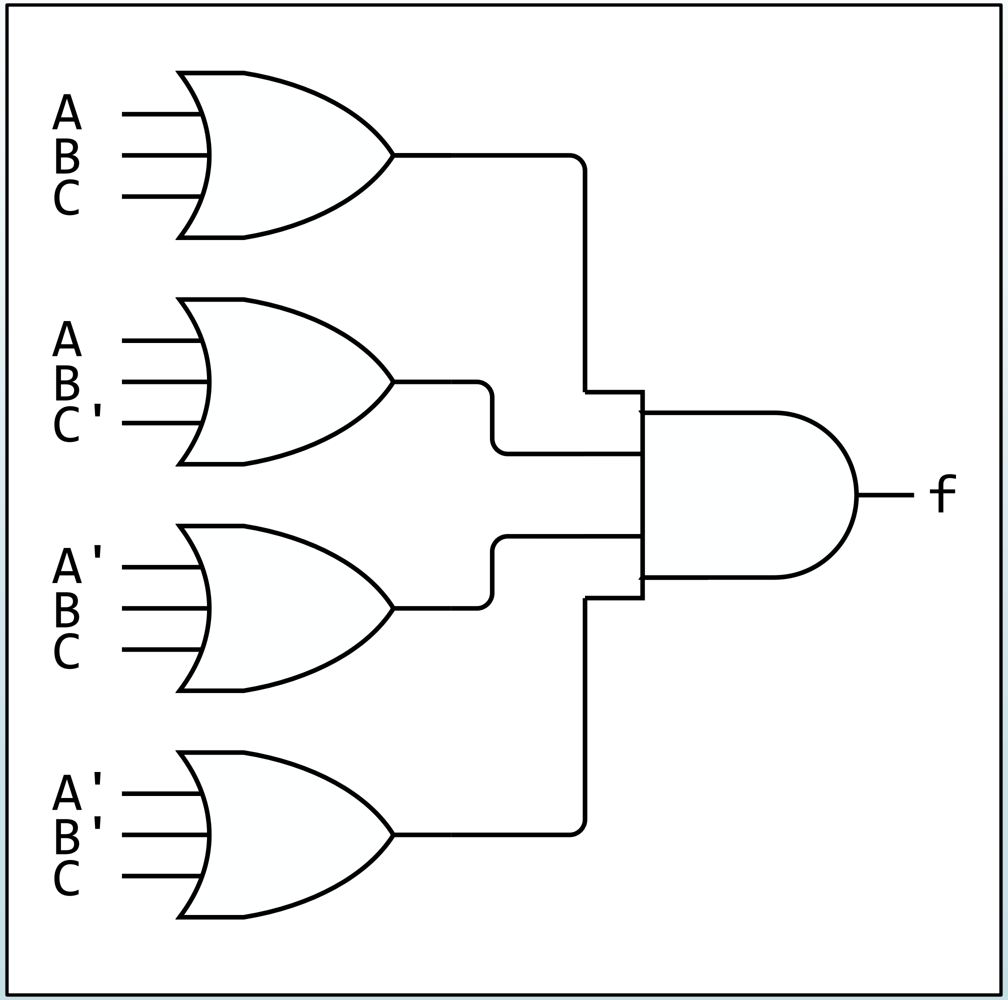
$$f = (A+B+C)\cdot(A+B+C')\cdot(A'+B+C)\cdot(A'+B'+C)$$

- Now, we designed another circuit for the prime number detector.
- However, the design is not perfect. It is too long and complex. It costs many gates.



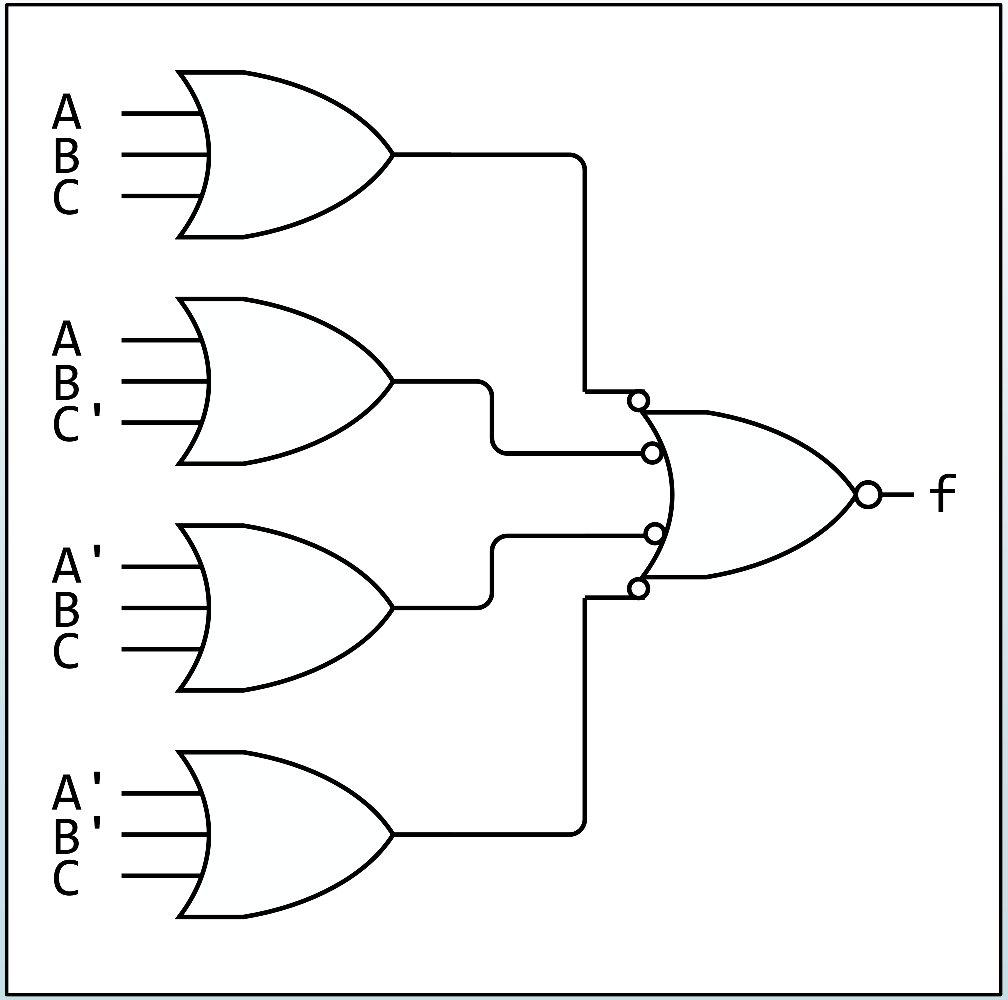
# Logic Circuit

- This is an OR-AND circuit.



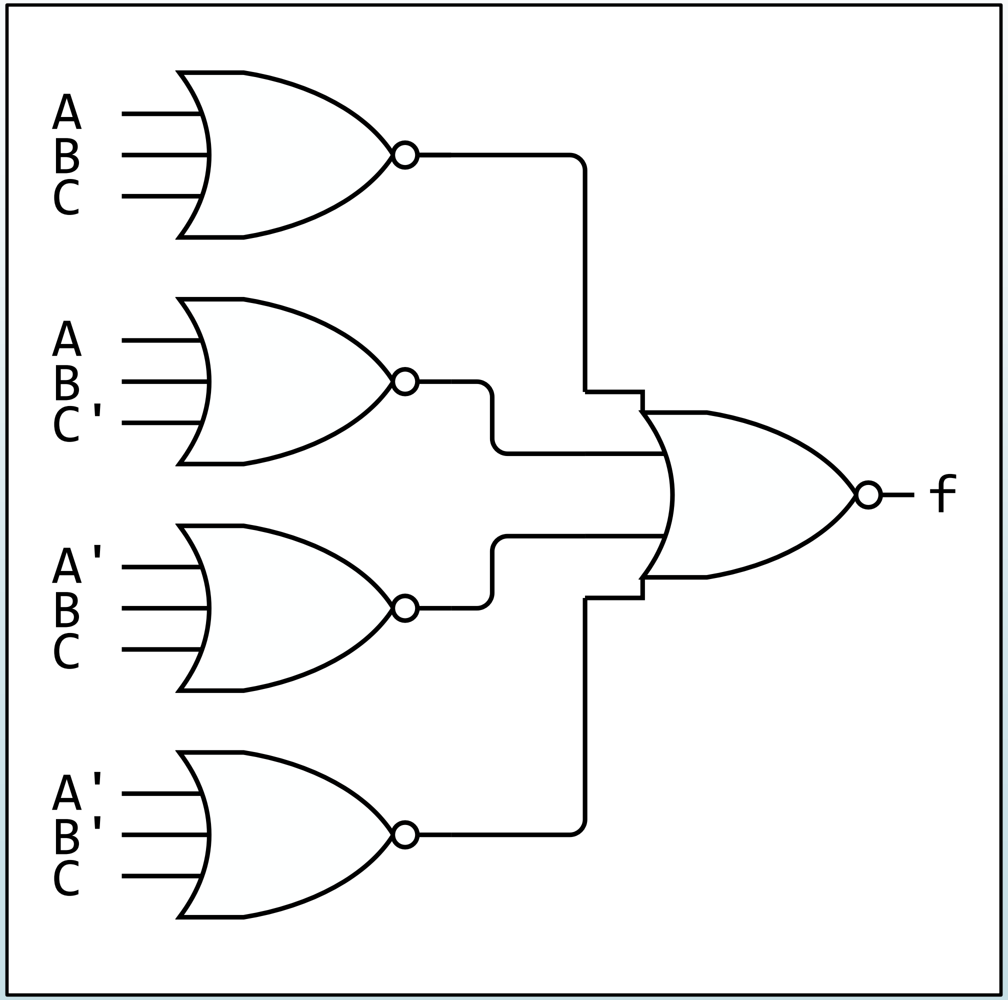
# Logic Circuit

- This is an OR-AND circuit.
- One cost reduction step is to apply the graphical DeMorgans law on the OR gate.
  - Change the AND to OR.
  - Complement all its inputs and output.



# Logic Circuit

- This is a NOR-NOR circuit equivalent to the same function.
  - Due to the gate components, the NOR gate is usually cheaper than the AND and OR gates for the same number of inputs.
  - Due to the gate components, the NOR gate is usually faster, smaller and consumes less power for the same number of inputs.



# Definitions

- **Product term:** a term consists of the logical product of many variable, either complemented or not. Example: `A'B'CD`
- **Sum of Products form:** Any logical sum of product terms.
  - Example: `AB' + BC`
- **Canonical Sum of products of a circuit:** The sum of product form of the circuit function, where output = `1`.
  - Example: `f = A'BC' + A'BC + AB'C + ABC`
- **On-set:** the minterms list the turns on the output.
  - Example: $f = \sum_{A,B,C} m(2,3,5,7)$

# Definitions

- **Sum term:** a term consists of the logical sum of many variable, either complemented or not. Example: `A'+B'+C+D`
- **Product of Sums form:** Any logical product of sum terms.
  - Example: `(A+B')·(B+C)`
- **Canonical product of sums of a circuit:** The product of sums form of the circuit function, where output = `0`.
  - Example: `f = (A+B+C)·(A+B+C')·(A'+B+C)·(A'+B'+C)`
- **Off-set:** the maxterms list the turns off the output.
  - Example: $f = \prod_{A,B,C} M(0,1,4,6)$

# Logic Circuit Description

- Many forms to represent a logical circuit:
  - Truth table.
  - Canonical sum of products: f = A'BC'+A'BC+AB'C+ABC
  - On-set: $f = \sum_{A,B,C} m(2,3,5,7)$
  - Canonical product of sums: f = (A+B+C)·(A+B+C')·(A'+B+C)·(A'+B'+C)
  - Off-set: $f = \prod_{A,B,C} M(0,1,4,6)$
  - Circuit Diagram.

| 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 |

# Question ??

- Give the following expression in the standard sum of products form.

$$f(C,B,A) = \sum m(0,3,6)$$

- Answer (Go back to the table):
  - f = m0 + m3 + m6
  - f = C'B'A' + m3 + m6

m0 = m000 → C=0, B=0, A=0

m3 = ??

# Question ??

- Give the following expression in the standard sum of products form.

$$f(C,B,A) = \sum m(0,3,6)$$

- Answer (Go back to the table):
  - f = m0 + m3 + m6 — m0 = m000 → C=0, B=0, A=0
  - f = C'B'A' + m3 + m6 — m3 = m011 → C=0, B=1, A=1
  - f = C'B'A' + C'BA + m6 — m6 = ??

# Question ??

- Give the following expression in the standard sum of products form.

$$f(C,B,A) = \sum m(0,3,6)$$

- Answer (Go back to the table):
  - f = m0 + m3 + m6
  - f = C'B'A' + m3 + m6
  - f = C'B'A' + C'BA + m6
  - f = C'B'A' + C'BA + CBA'

m0 = m000 → C=0, B=0, A=0

m3 = m011 → C=0, B=1, A=1

m6 = m110 → C=1, B=1, A=0