# Chapter 1

## Lecture 2

26/2/2022

Book: Sections 1.1.6 to 1.3.2

# Lecture Contents

1. Bit Operations
2. Logic circuits (digital circuit)
3. Propositional equivalences
   - Tautology
   - Contradiction
   - Contingency
4. Laws of Propositional Logic
   1. Identity laws
   2. Domination laws
   3. Idempotent laws
   4. Double negation law
   5. Commutative laws
   6. Associative laws
   7. Distributive law
   8. De Morgan’s laws
   9. Absorption laws
   10. Negation laws

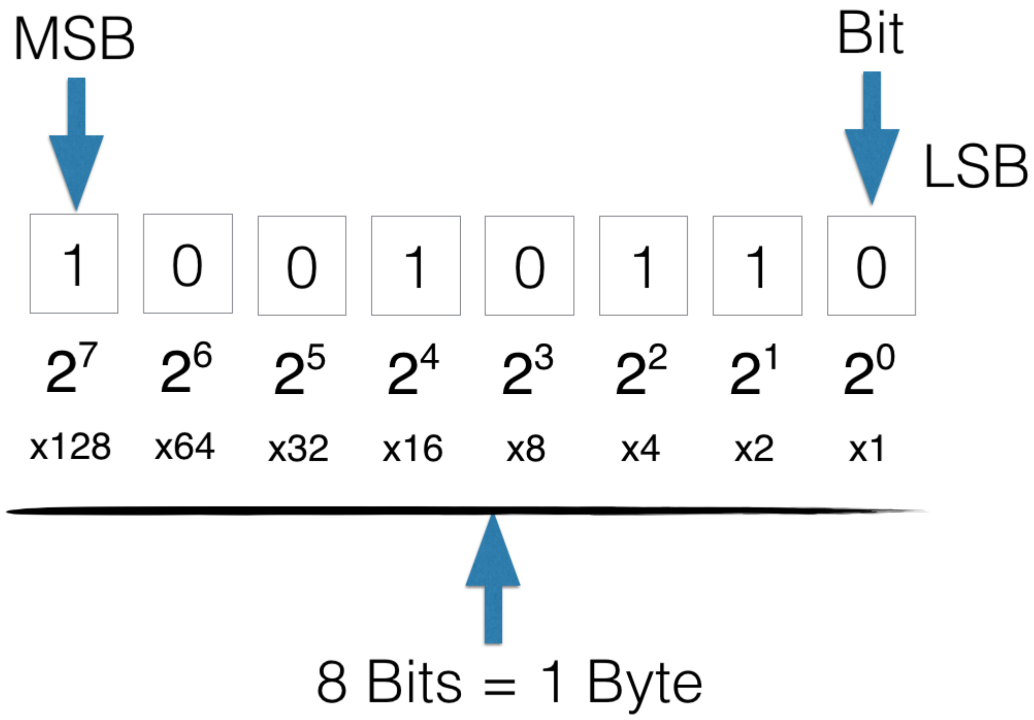
# 1.1.6. Bit Operations

- Computers represent information using bits. A bit is a symbol with two possible values: 0 (zero) and 1 (one).

  - **1** represents **T** (true)
  - **0** represents **F** (false)

- A variable is called a **Boolean variable** if its value is either **true** or **false**.

- A Boolean variable can be represented using a bit.



# 1.1.6. Bit Operations

**Bit string**

- A bit string is a sequence of zero or one bits.
- The length of bit string is the number of bits in the string.

- **Example**

**101010011** is a bit string of length nine.

# 1.1.6. Bit Operations

| *Truth Value* | *Bit* |
| --- | --- |
| T | 1 |
| F | 0 |

| $x$ | $y$ | $x \vee y$ | $x \wedge y$ | $x \oplus y$ |
| --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 | 0 |

# 1.1.6. Bit Operations

## Example

- Find the bitwise OR, bitwise AND, and bitwise XOR of the bit strings:
  01 1011 0110 - 11 0001 1101

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| AND | $\wedge$ | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| OR | $\vee$ | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 1 |
| XOR | $\oplus$ | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |

# 1.1.6. Bit Operations

- **Which of the following bits is the negation of the bits "010110"?**

a) 111001 b) 001001 $\boxed{\text{c) 101001}}$ d) 111111

- **If A is "001100" and B is "010101" then what is the value of A (Ex-or) B?**

a) 000000 b) 111111 c) 001101 $\boxed{\text{d) 011001}}$

- **Which of the following option is suitable, if A is "10110110", B is"11100000" and C is"10100000"?**

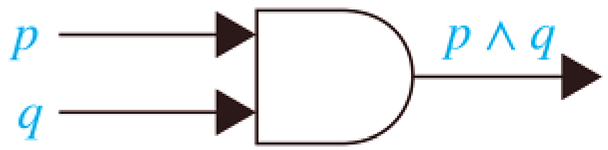
a) C=A or B b) C=¬A c) C=¬B $\boxed{\text{d) C=A and B}}$

# 1.2.6. Logic circuits (digital circuit)

- A logic circuit (or digital circuit) receives input signals $p_1, p_2, \ldots, p_n$, each a bit [either 0 (off) or 1 (on)], and produces output signals $s_1, s_2, \ldots, s_n$, each a bit.

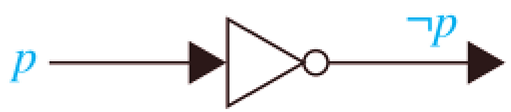
- In this course, we will restrict our attention to logic circuits with a single output signal; in general, digital circuits may have multiple outputs.

- Complicated digital circuits can be constructed from three basic circuits, called **gates**.



AND gate

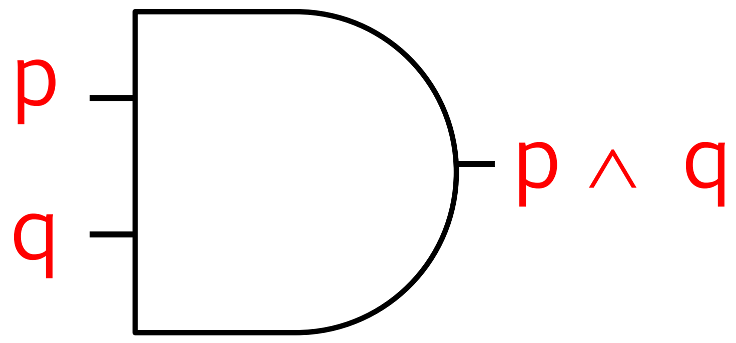
OR gate



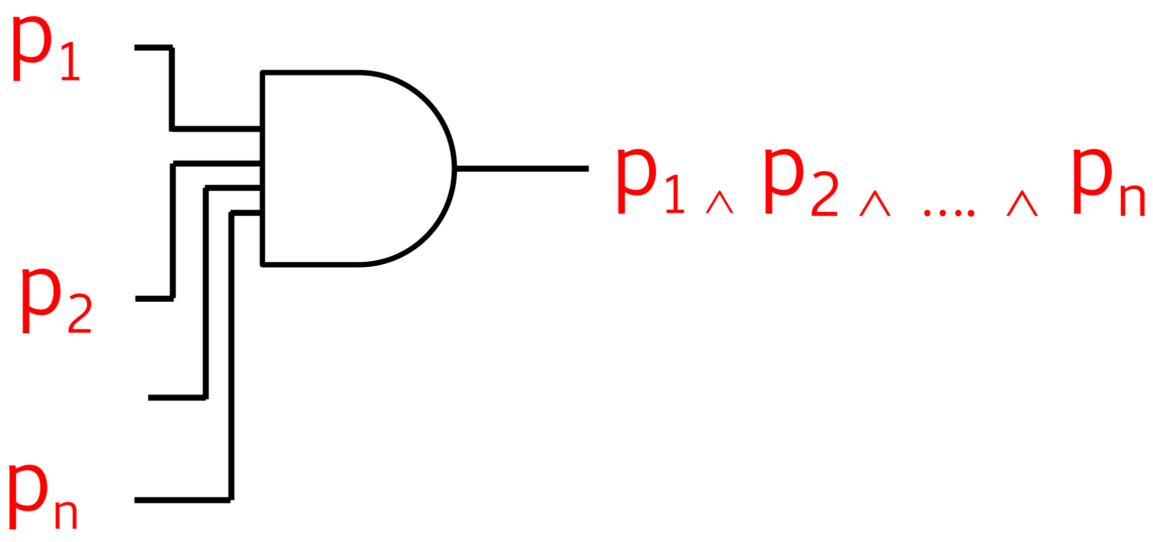
Inverter
NOT gate

# 1.2.6. Logic circuits (digital circuit)

## AND gate (Boolean Product)

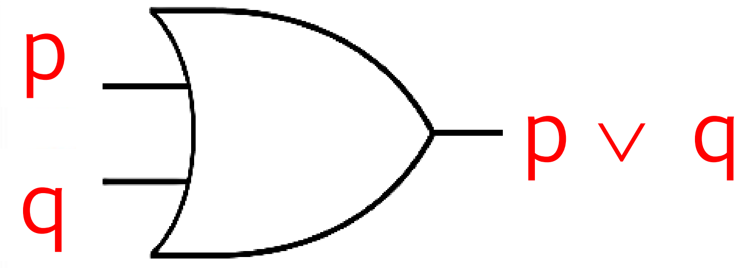



AND gates can be extended to arbitrary n inputs.

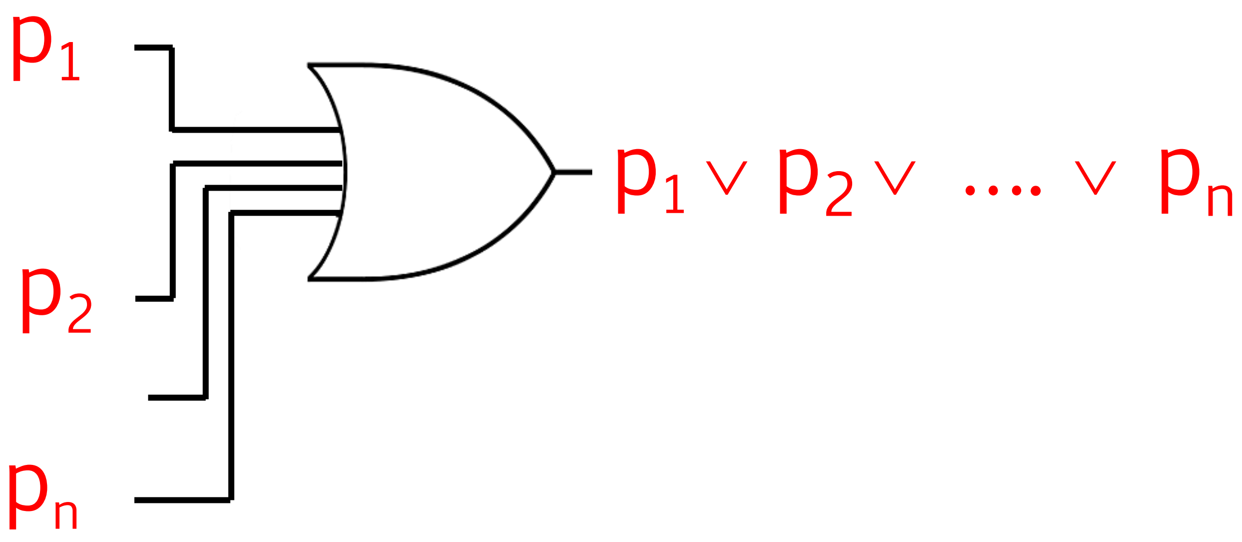


# 1.2.6. Logic circuits (digital circuit)

## OR gate (Boolean Sum)

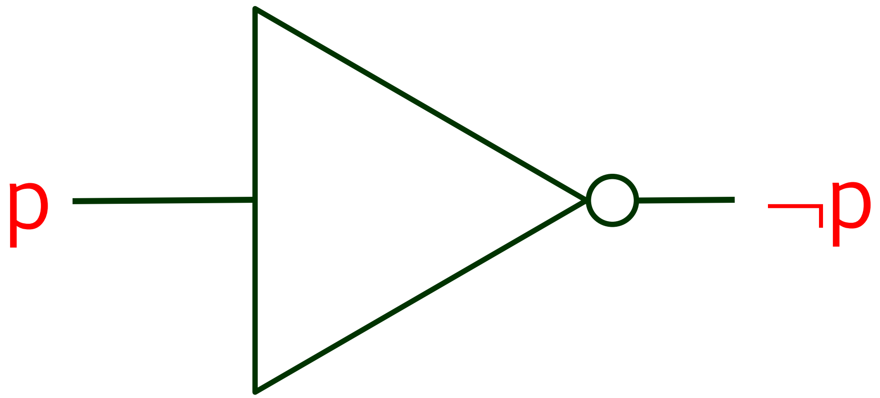



<u>OR gates can be extended to arbitrary n inputs.</u>



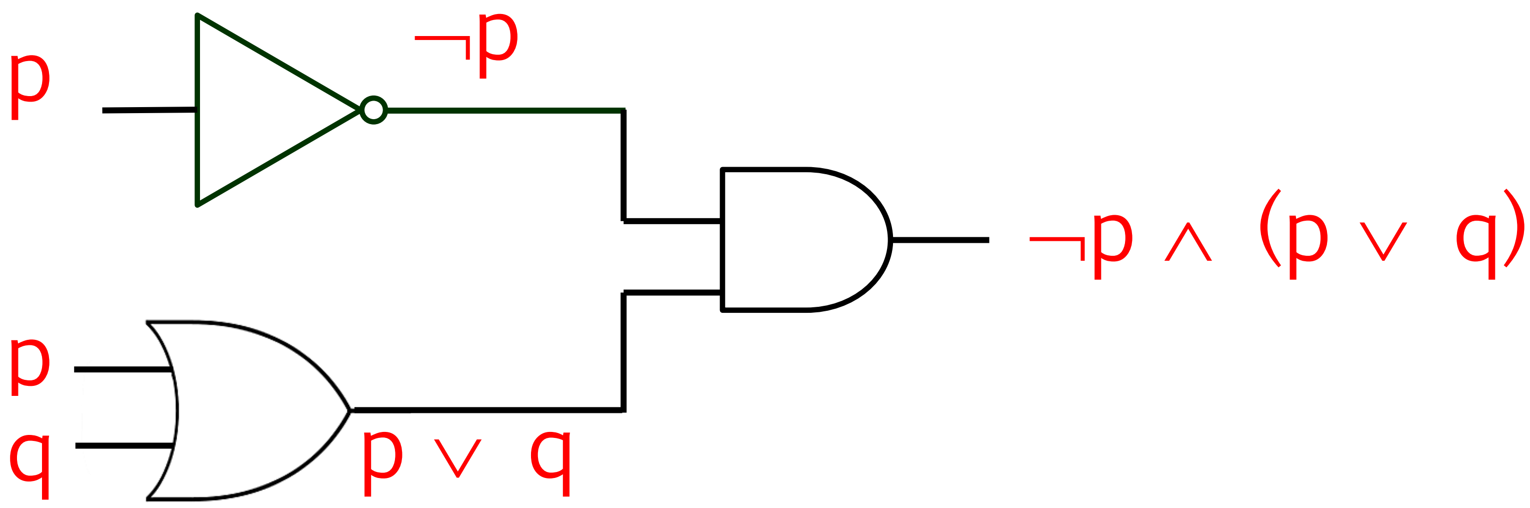
# 1.2.6. Logic circuits (digital circuit)

NOT gate (Boolean Complement)



# 1.2.6. Logic circuits (digital circuit)

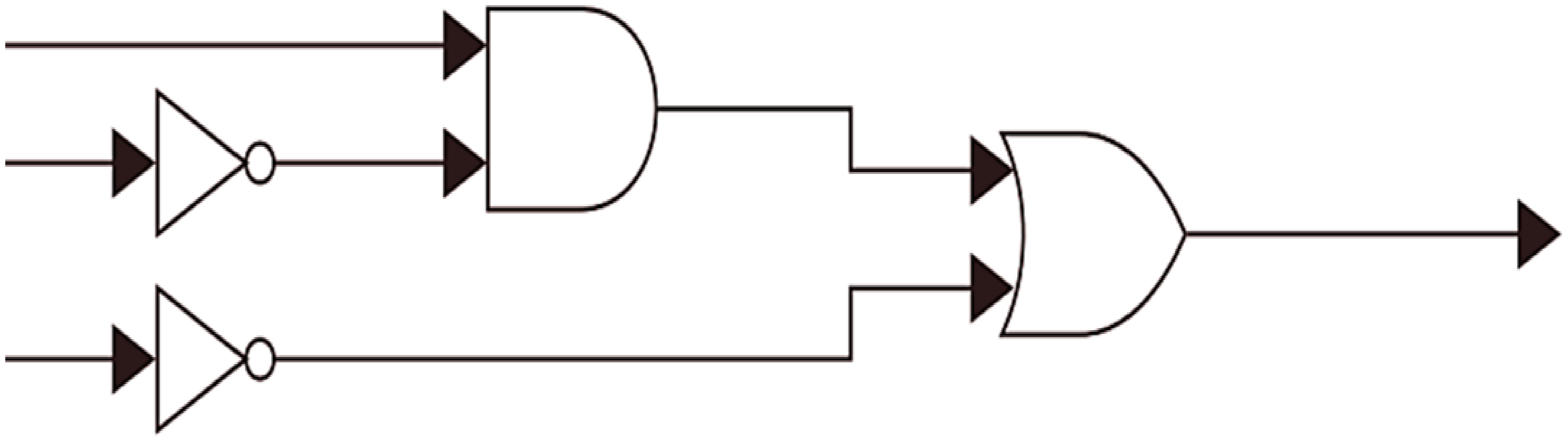
## Combination of Gates



# 1.2.6. Logic circuits (digital circuit)

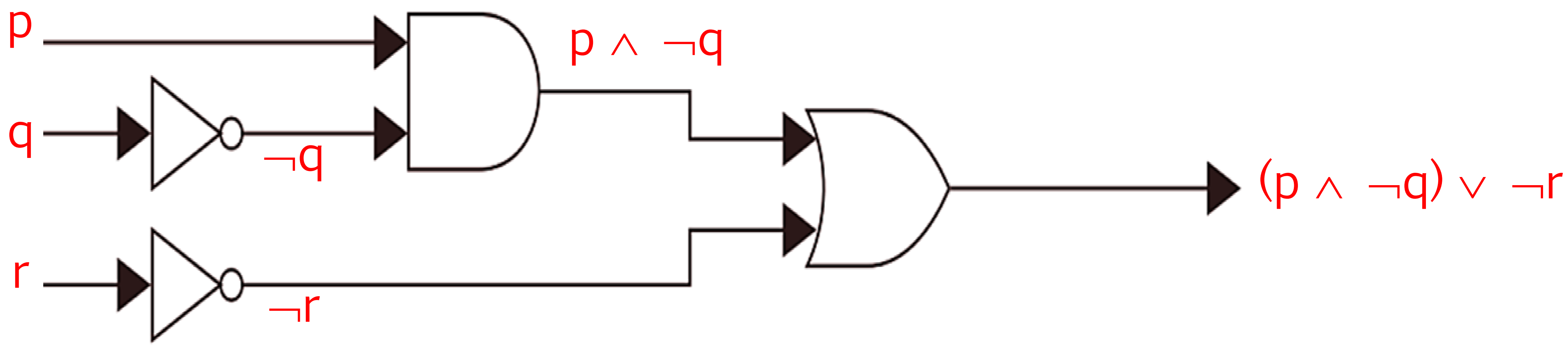
## Example (1)

- Determine the output for the combinatorial circuit in the following figure.



# 1.2.6. Logic circuits (digital circuit)

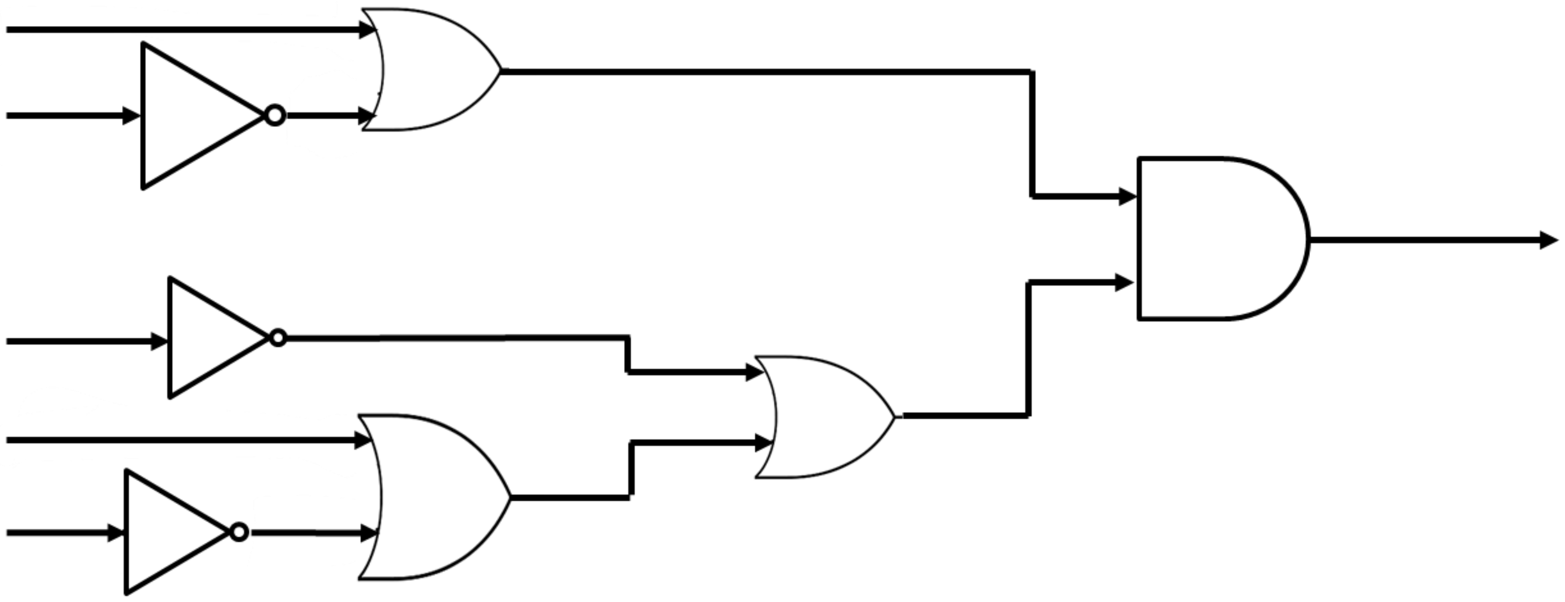
**Solution**



# 1.2.6. Logic circuits (digital circuit)

## Example (2)

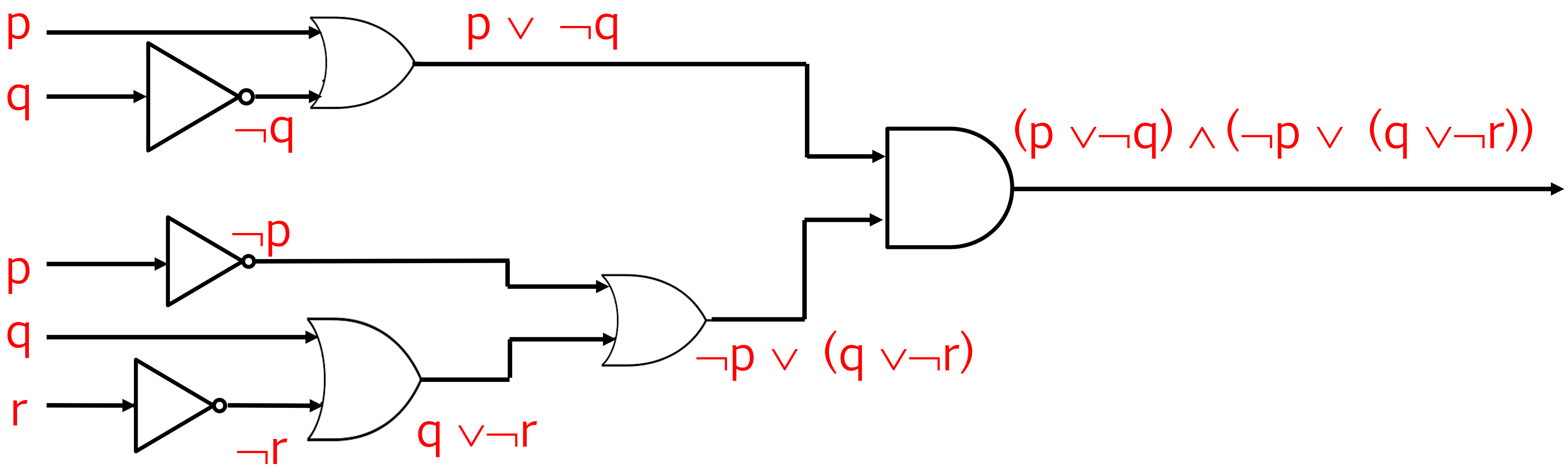
- Determine the output for the combinatorial circuit in the following figure.



# 1.2.6. Logic circuits (digital circuit)

## Example (2)

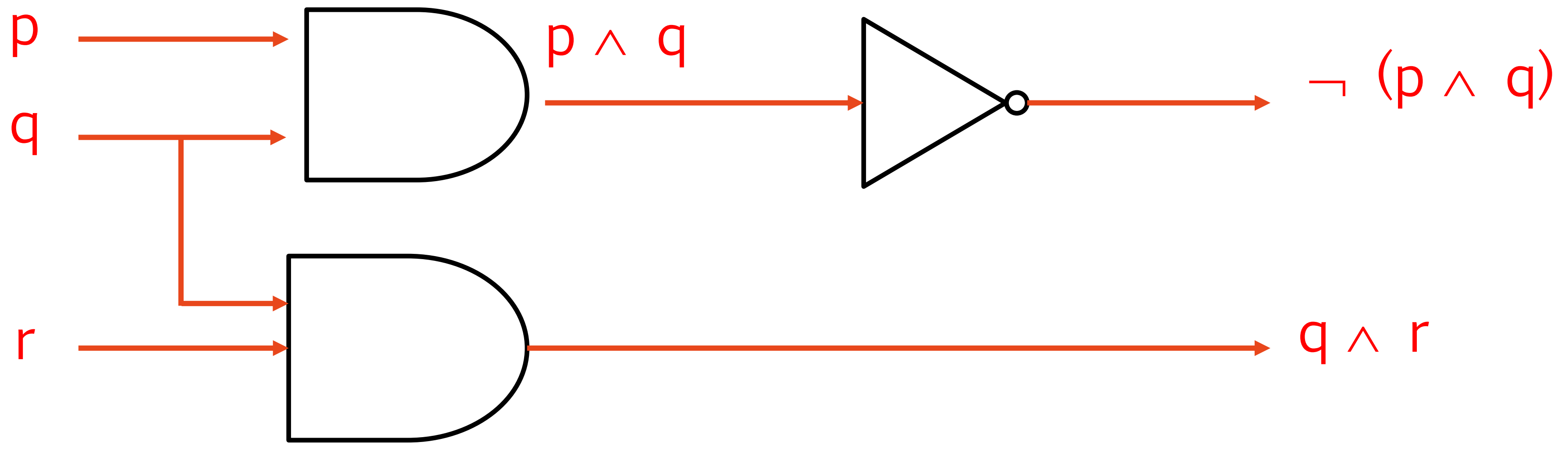
- Determine the output for the combinatorial circuit in the following figure.



# 1.2.6. Logic circuits (digital circuit)

## Example (3)

- Determine the output for the combinatorial circuit in the following figure.



# Homework (2)

يسلم في السيكشن القادم

# 1.2.6. Logic circuits (digital circuit)

## Problem (1)

- Build a digital circuit that produces the output:

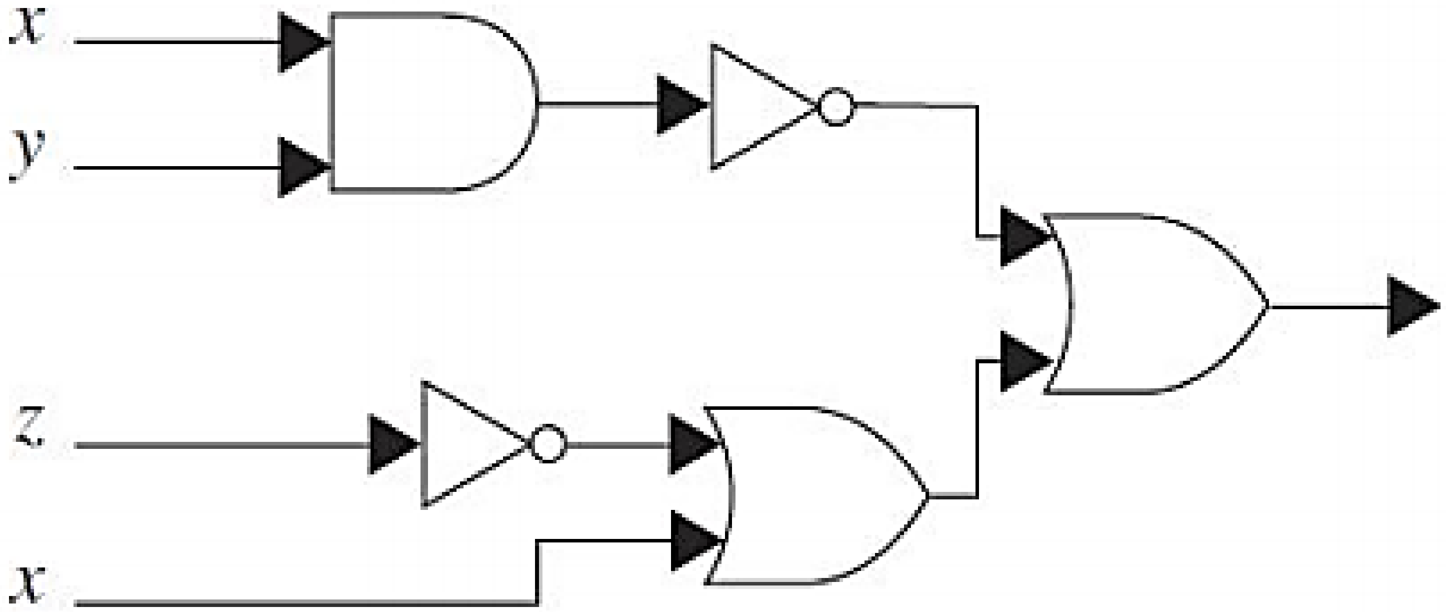
$$(p \vee \neg r) \wedge (\neg p \vee (q \vee \neg r))$$

when given input bits $p$, $q$, and $r$.

# 1.2.6. Logic circuits (digital circuit)

## Problem (2)

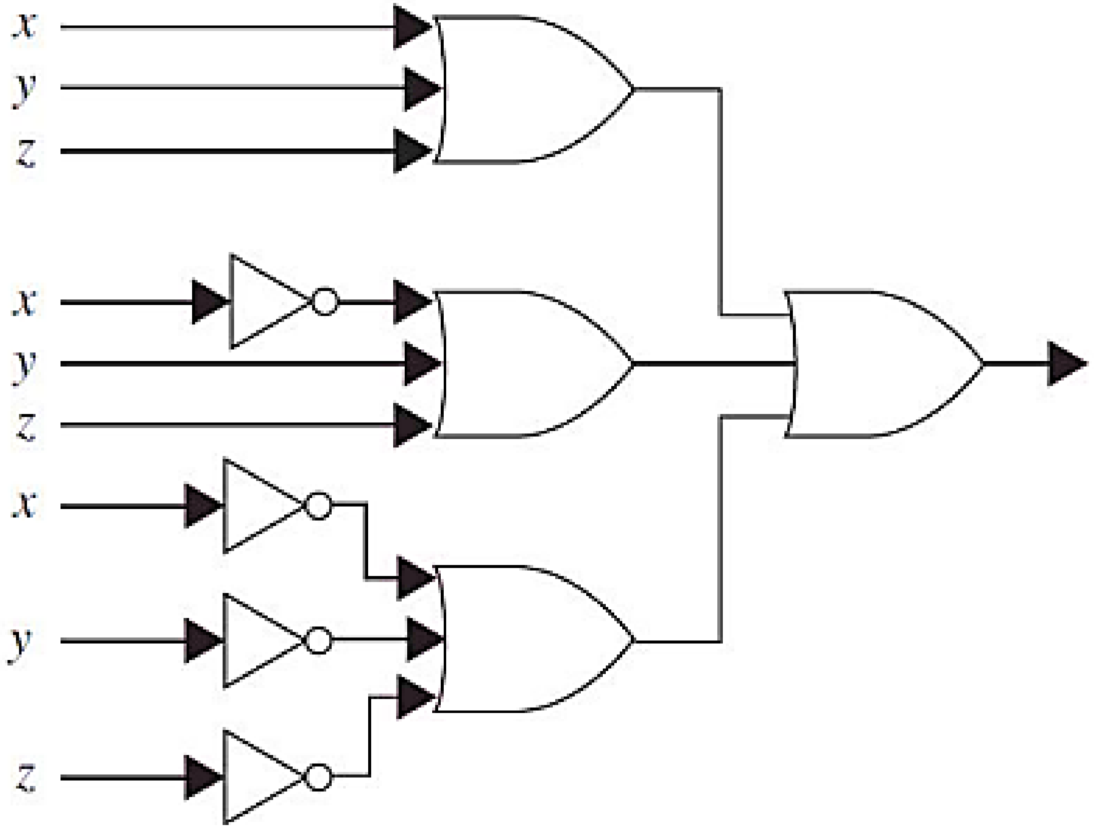
- Find the output of the given circuit.



# 1.2.6. Logic circuits (digital circuit)

## Problem (3)

- Find the output of the given circuit.



# 1.3. Propositional equivalences

The compound propositions can be classified according to their possible truth values into three types:

**Compound Propositions**

- Tautology
- Contradiction
- Contingency

# 1.3. Propositional equivalences

**Propositional Equivalences**

- **Tautology**: **A compound proposition that is always true.**
- **Contradiction**: A compound proposition that is always false.
- **Contingency**: A compound proposition that is neither a tautology nor a contradiction.

# 1.3. Propositional equivalences

## Example (1)

Show that following conditional statement is a tautology by using truth table:

- $p \vee \neg p$
- $p \wedge q \rightarrow p$

| $p$ | $\neg p$ | $p \vee \neg p$ |
|---|---|---|
| T | F | T |
| F | T | T |
| T | F | T |
| F | T | T |

| $p$ | $q$ | $p \wedge q$ | $p \wedge q \rightarrow p$ |
|---|---|---|---|
| T | T | T | T |
| F | T | F | T |
| T | F | F | T |
| F | F | F | T |

- $p \vee \neg p$ is a tautology
- $p \wedge q \rightarrow p$ is a tautology

# 1.3. Propositional equivalences

## Logical Equivalences

- Compound propositions that **<u>have the same truth values in all possible cases</u>** are called logically equivalent.
- The compound propositions p and q are called ***logically equivalent*** if $p \leftrightarrow q$ is a tautology.
- The notation $p \equiv q$ denotes that p and q are logically equivalent.

$\equiv$

$\Leftrightarrow$

# 1.3. Propositional equivalences

## Example (1)

Show that $\neg(p \vee q)$ and $\neg p \wedge \neg q$ are logically equivalent.

| $p$ | $q$ | $p \vee q$ | $\neg(p \vee q)$ | $\neg p$ | $\neg q$ | $\neg p \wedge \neg q$ |
|---|---|---|---|---|---|---|
| T | T | T | F | F | F | F |
| F | T | T | F | T | F | F |
| T | F | T | F | F | T | F |
| F | F | F | T | T | T | T |

So, $\neg(p \vee q) \leftrightarrow (\neg p \wedge \neg q)$ is logically equivalent

# 1.3. Propositional equivalences

## Example (2)

Show that $\neg p \vee q$ and $p \rightarrow q$ are logically equivalent.

| $p$ | $q$ | $\neg p$ | $\neg p \vee q$ | $p \rightarrow q$ |
|---|---|---|---|---|
| T | T | F | T | T |
| F | T | T | T | T |
| T | F | F | F | F |
| F | F | T | T | T |

So, $\neg p \vee q$ and $p \rightarrow q$ is logically equivalent

# 1.3. Propositional equivalences

## Example (3)

Show that $p \vee (q \wedge r)$ and $(p \vee q) \wedge (p \vee r)$ are logically equivalent.

| $p$ | $q$ | $r$ | $q \wedge r$ | $p \vee (q \wedge r)$ | $p \vee q$ | $p \vee r$ | $(p \vee q) \wedge (p \vee r)$ |
|---|---|---|---|---|---|---|---|
| T | T | T | T | T | T | T | T |
| F | T | T | T | T | T | T | T |
| T | F | T | F | T | T | T | T |
| F | F | T | F | F | F | T | F |
| T | T | F | F | T | T | T | T |
| F | T | F | F | F | T | F | F |
| T | F | F | F | T | T | T | T |
| F | F | F | F | F | F | F | F |

So, $p \vee (q \wedge r)$ and $(p \vee q) \wedge (p \vee r)$ is logically equivalent

# 1.3. Laws of Propositional Logic

| Logical Equivalences. | |
|---|---|
| $p \wedge \mathbf{T} \equiv p$<br>$p \vee \mathbf{F} \equiv p$ | Identity laws |
| $p \vee \mathbf{T} \equiv \mathbf{T}$<br>$p \wedge \mathbf{F} \equiv \mathbf{F}$ | Domination laws |
| $p \vee p \equiv p$<br>$p \wedge p \equiv p$ | Idempotent laws |
| $\neg(\neg p) \equiv p$ | Double negation law |
| $p \vee q \equiv q \vee p$<br>$p \wedge q \equiv q \wedge p$ | Commutative laws |

T compound proposition is always true
F compound proposition is always false

# 1.3. Propositional equivalences

| Logical Equivalences. | |
|---|---|
| $(p \vee q) \vee r \equiv p \vee (q \vee r)$<br>$(p \wedge q) \wedge r \equiv p \wedge (q \wedge r)$ | Associative laws |
| $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$<br>$p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$ | Distributive laws |
| $\neg(p \wedge q) \equiv \neg p \vee \neg q$<br>$\neg(p \vee q) \equiv \neg p \wedge \neg q$ | De Morgan's laws |
| $p \vee (p \wedge q) \equiv p$<br>$p \wedge (p \vee q) \equiv p$ | Absorption laws |
| $p \vee \neg p \equiv \mathbf{T}$<br>$p \wedge \neg p \equiv \mathbf{F}$ | Negation laws |

# Thank you