

AI for STEM Competition

Asia Pacific STEAM_AI Technology
Innovation Challenge

04 Car movement

P

CocoRobo

O

P

W

R

E

0





Chapter 1
Meet the
Mecanum Wheel
Robot



Chapter 2
Mecanum Wheel
Control



Chapter 3
Understanding
Bluetooth



Chapter 4
Manual Control of
Mecanum wheels for
the robot

目錄

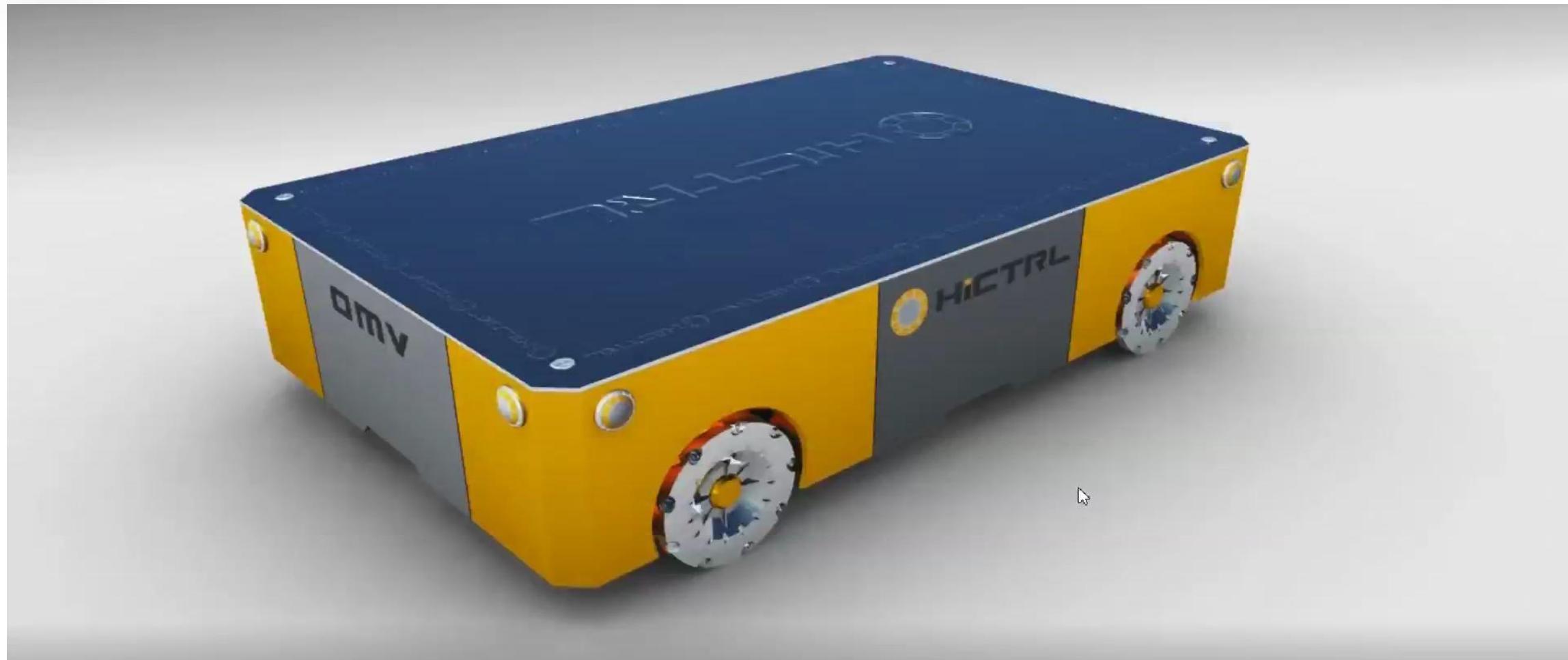
CONTENTS



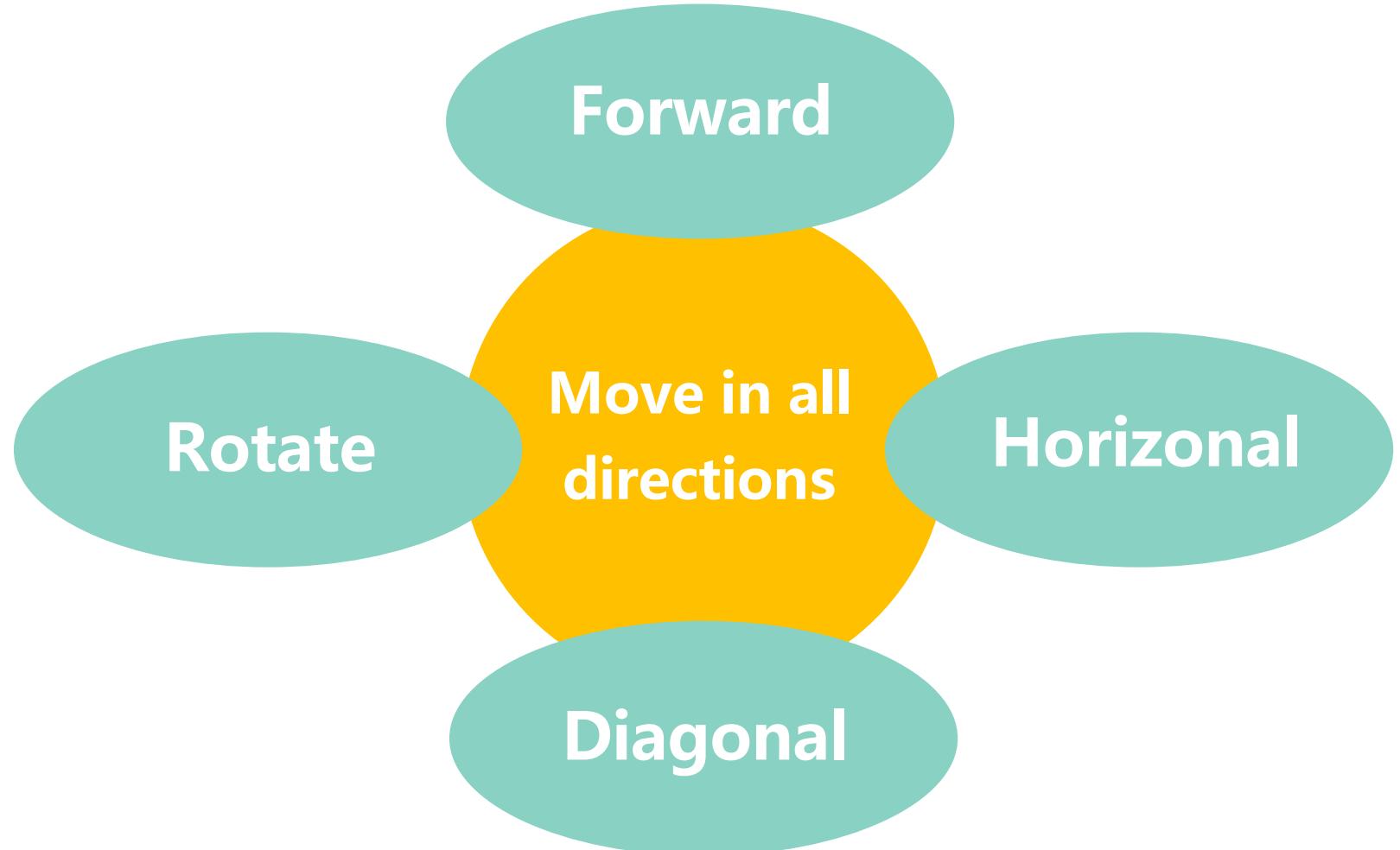
ONE.

Meet the
Mecanum Wheel
Robot

● Meet the Mecanum Wheel Robot



● Meet the Mecanum Wheel Robot



● Meet the Mecanum Wheel Robot



Activity one

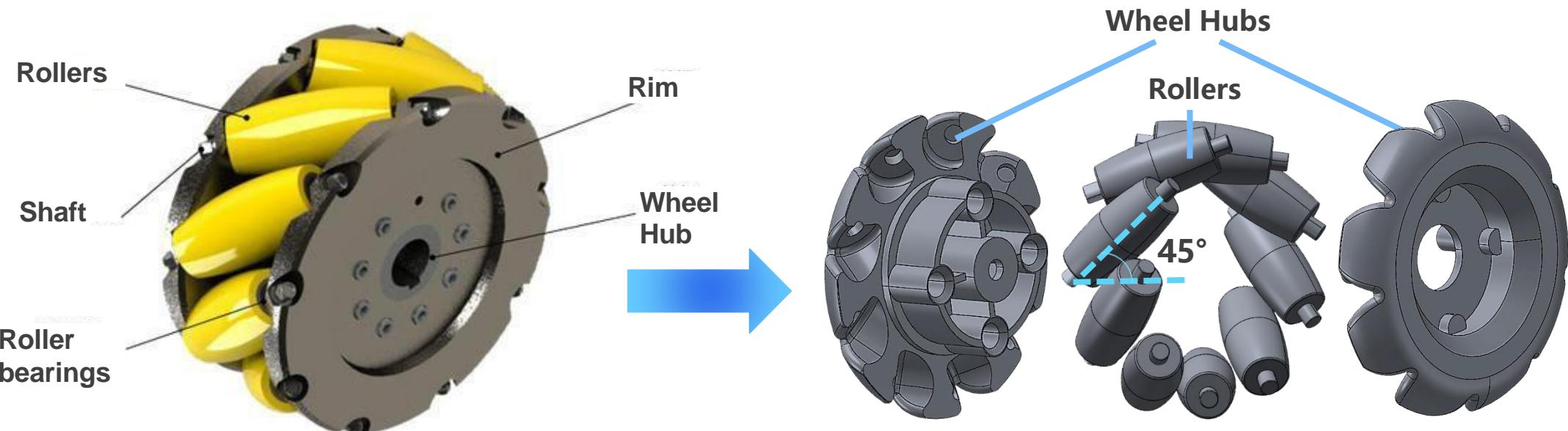
- Take out the Mecanum wheel and observe;
- Tell me, how is it different from a normal wheel?
- Discuss what do you think they do?

● Meet the Mecanum Wheel Robot



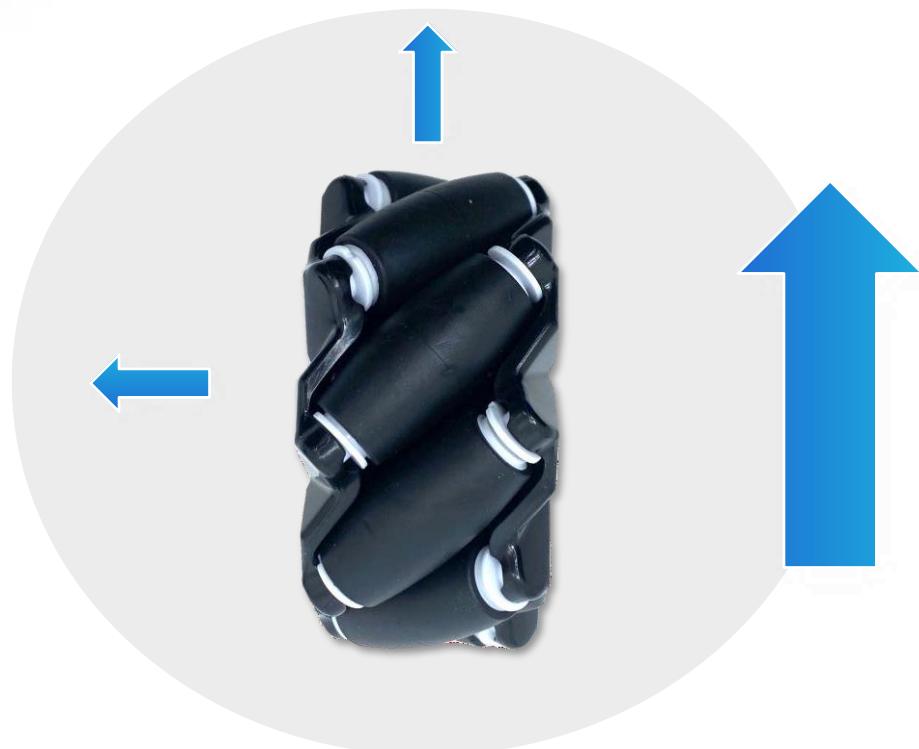
The Mecanum Wheel is an **all-direction** enabler wheel structure invented by the Swedish Mecanum Company.

Meet the Mecanum Wheel Robot

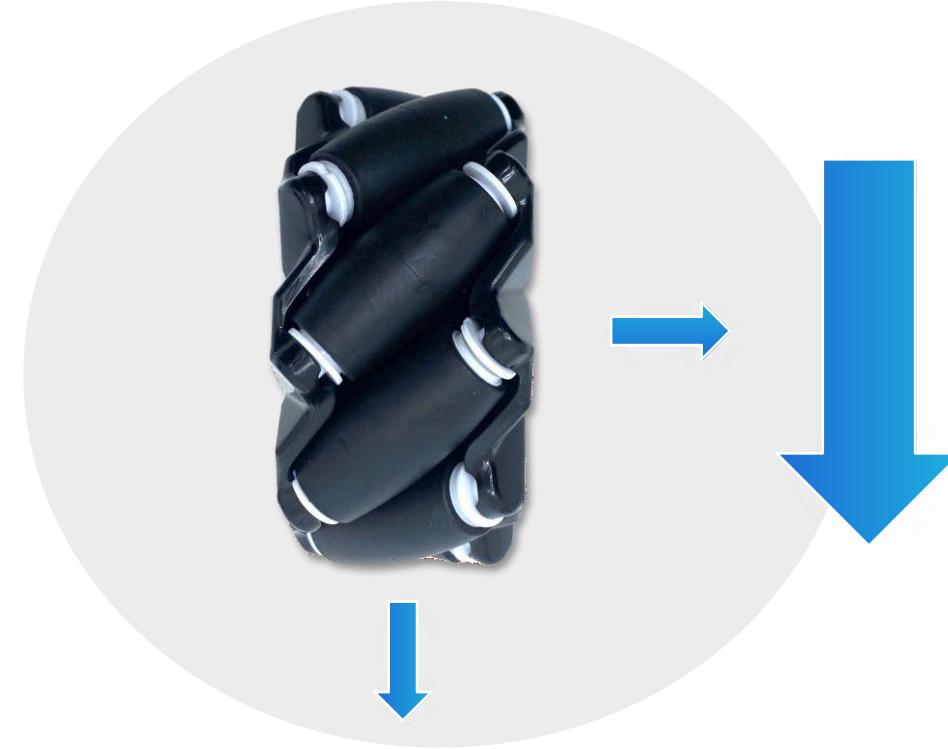


The **mecanum wheel** are made up of a hub and rollers around it, the hub keeps the rollers at an axis angle of **45°**.

The basic composition of the Mecanum Wheel



Looking at the forces when moving in the positive direction, the forces can be broken down into left and forward



Looking at the forces when moving in the negative direction, the forces can be broken down into right and backward

● The basic composition of the Mecanum Wheel

Universal

Make it easier to change the direction of the transport robot to meet different needs

flexibility

The flexibility of the base movements makes it easier for the completion of the intelligent transportation

Stability

At low speeds, the stability of the mecanum wheel is beneficial.

Low Efficiency

In motion, it is always necessary to cancel out the velocity component in the unwanted direction, hence it is less effective than ordinary wheels

Poor ground contact performance

Requires the surface its driving on to be much more flat. For the four wheels to reliably touch the ground, an elastic suspension mechanism is required. This makes the design more complex.

● Observe the Mecanum Wheel

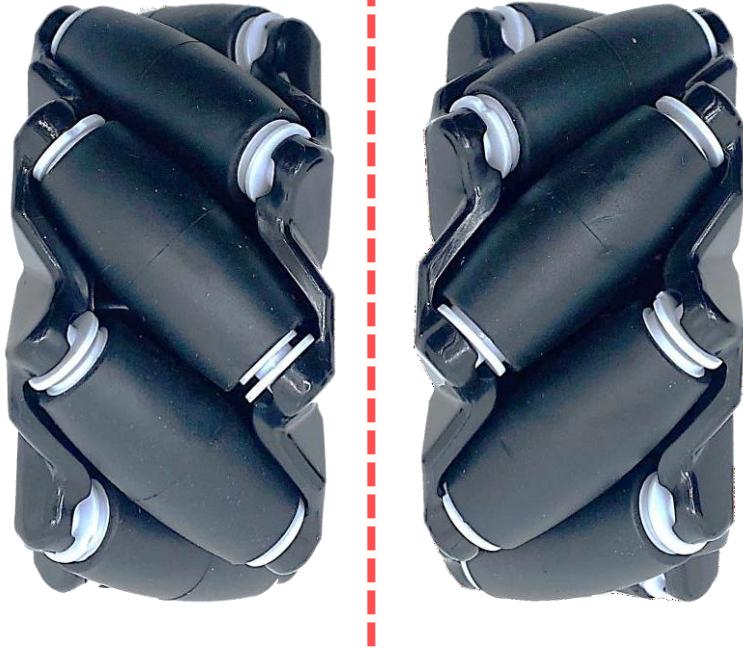
Activity 2



Looking at the diagram below,
what is the difference
between these two wheels
and what is the connection
between them?



● Observe the Mecanum Wheel



The left and right mecanum wheels are mirror images of each other

We refer to them as (A wheel) and (B wheel)

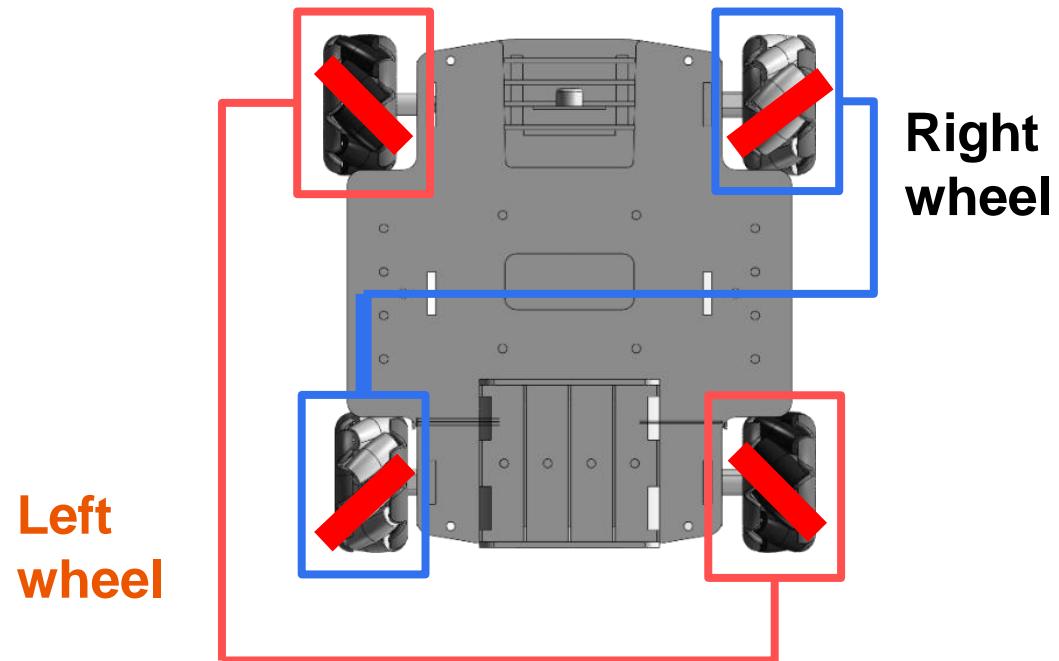


**Left wheel
(A wheel)**



**Right wheel
(B round)**

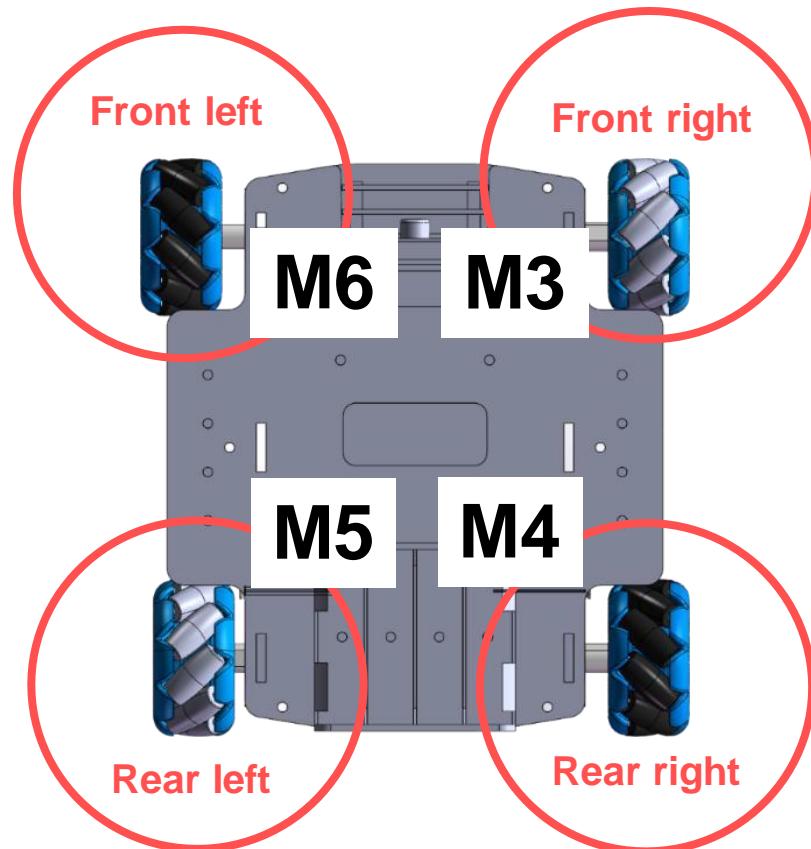
Introduction to Mecanum Wheel



When installing the wheels, the direction of the wheel rollers must be the same diagonally.

The four mecanum wheels mounted are not identical, they are made up of two pairs of mecanum wheels - the left wheel and the right wheel.

Mecanum Wheel Installation



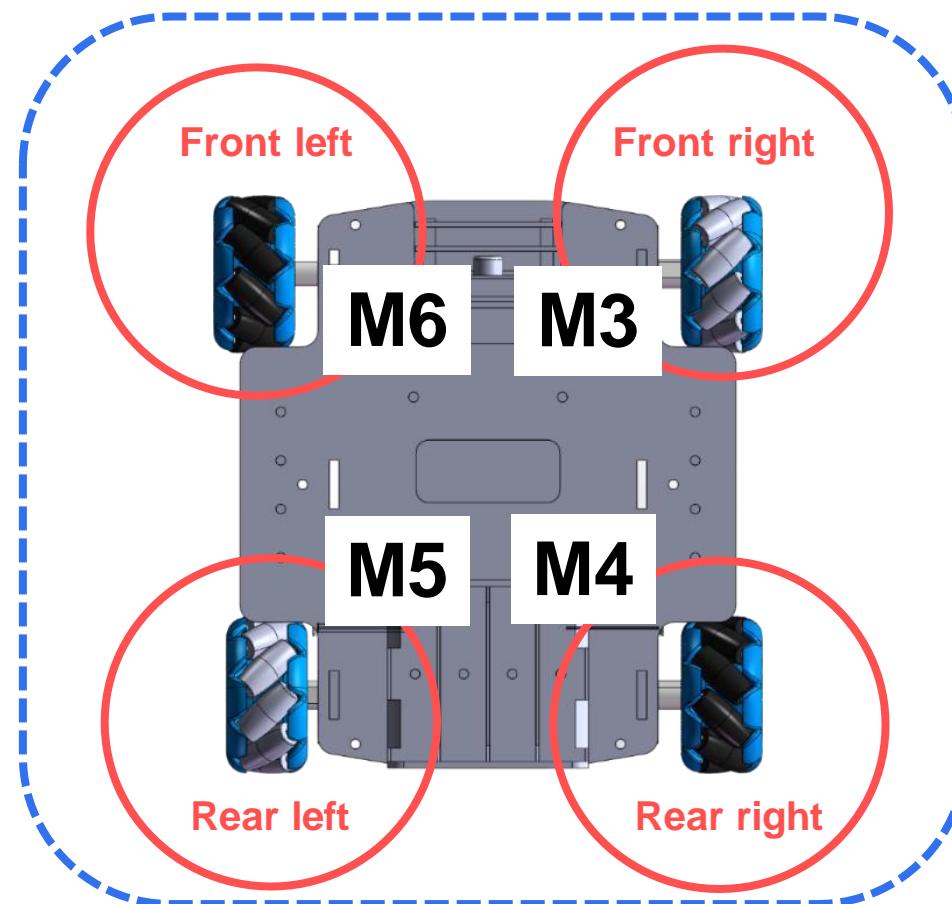
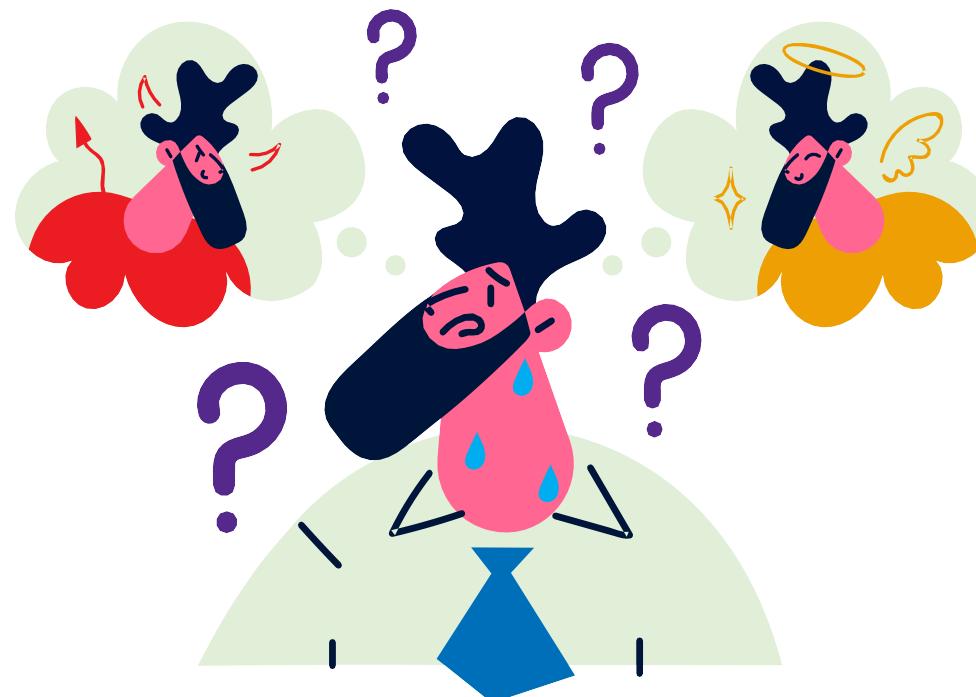
Control the rotation of the motor, observe whether the rotation direction of the corresponding wheel is consistent with the program setting, and record whether the wheel is clockwise or counterclockwise in the program when the wheel rotates clockwise.



● Principle Motion of the Mecanum Wheels



Why did you choose this way to assemble the wheels?



TWO.

Mecanum
wheel control
control

● Exploration of the Mecanum Wheel Movements



Small experiment

1. Place a paper card on the table, write a program to rotate any wheel of the robot clockwise, power up the robot;
2. Hold the robot in your hand, contact the paper card with a single wheel. Observe the movement direction of the paper.

● Exploration of the Mecanum Wheel Movements



Small experiment

1. Place a paper card on the table, write a program to rotate any wheel of the robot counter-clockwise, power up the robot;
2. Hold the robot in your hand, contact the paper card with a single wheel. Observe the movement direction of the paper. Is there any difference?

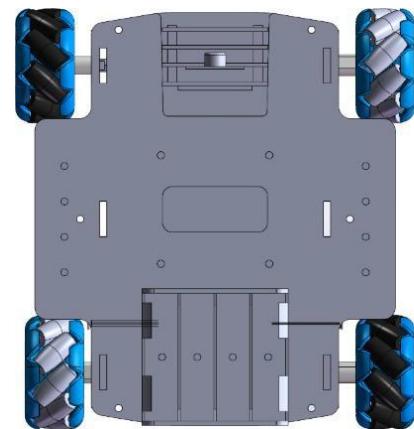
● Exploration of the Mecanum Wheel Movements



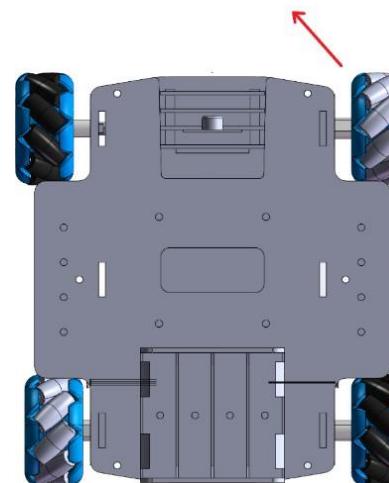
Principle of Moving Mecanum Wheel Robot

Summary of the experiment

- When the wheel of the mecanum wheel rotates clockwise, the paper piece moves diagonally to the right rear with respect to the mecanum wheel, and the mecanum wheel moves diagonally to the left front with respect to the paper piece.
- When the wheel of the mecanum wheel rotates counter-clockwise, the paper moves diagonally to the left front with respect to the mecanum wheel, and the mecanum wheel moves diagonally to the right rear with respect to the paper piece.



The wheels turn clockwise in the direction of the paper



The wheels turn counterclockwise in the direction of the paper piece

● Principle of Moving Mecanum Wheel Robot



The change in the position of an object with respect to other motions is called **relative motion**.



When a person runs on a treadmill, the person runs forward and the tracks of the treadmill move backwards. In the same way, if the piece of paper moves backwards, the cart moves forward.

● Principle of Moving Mecanum Wheel Robot

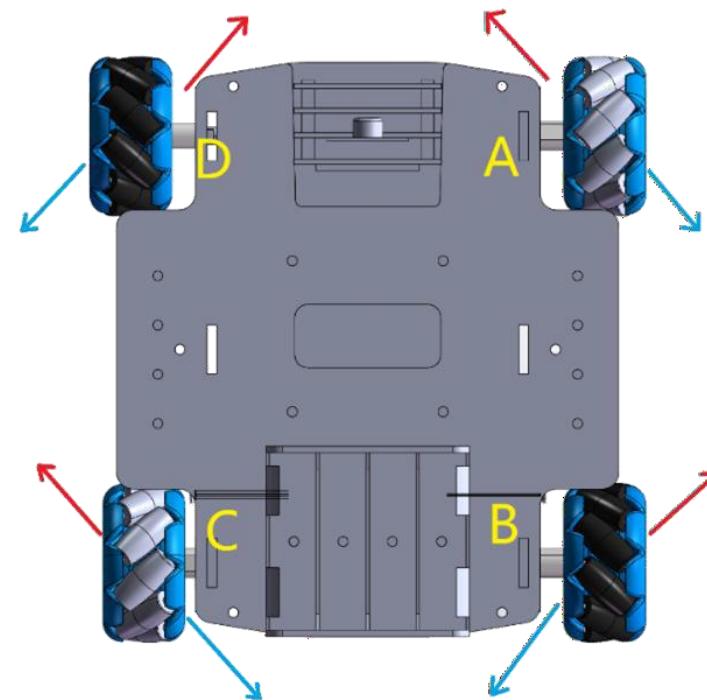


Red arrows

When the wheels rotates clockwise, it corresponds to the direction of movement of the mecanum wheel relative to the ground;

Blue arrows

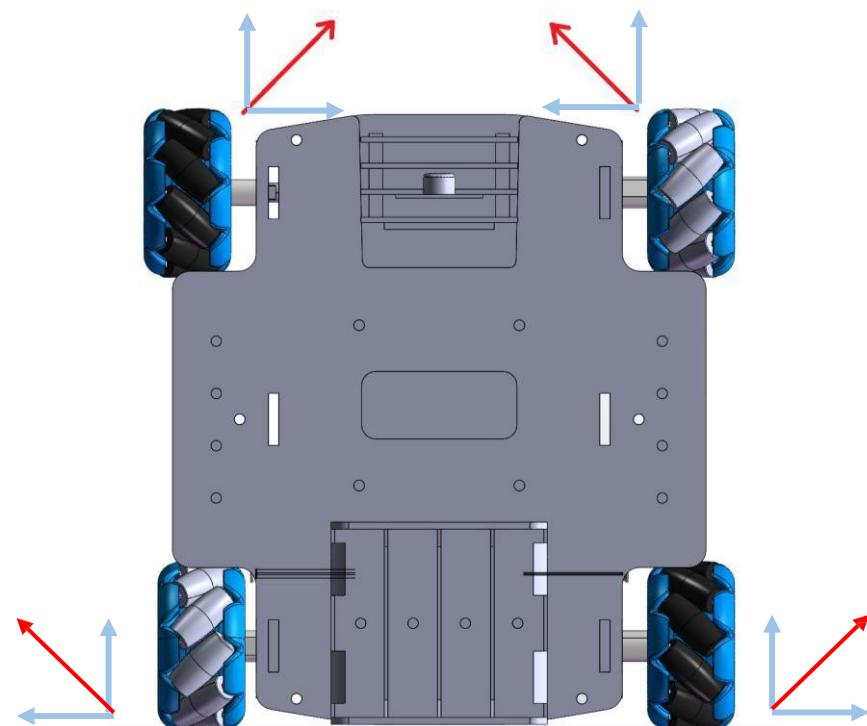
When the wheels rotates counterclockwise, it corresponds to the direction of movement of the mecanum wheel relative to the ground.



● Principle of Moving Mecanum Wheel Robot



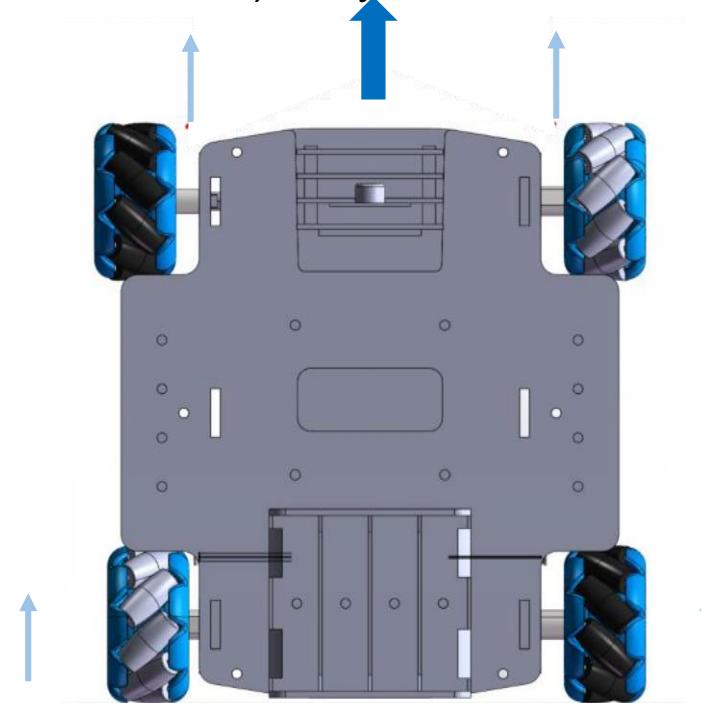
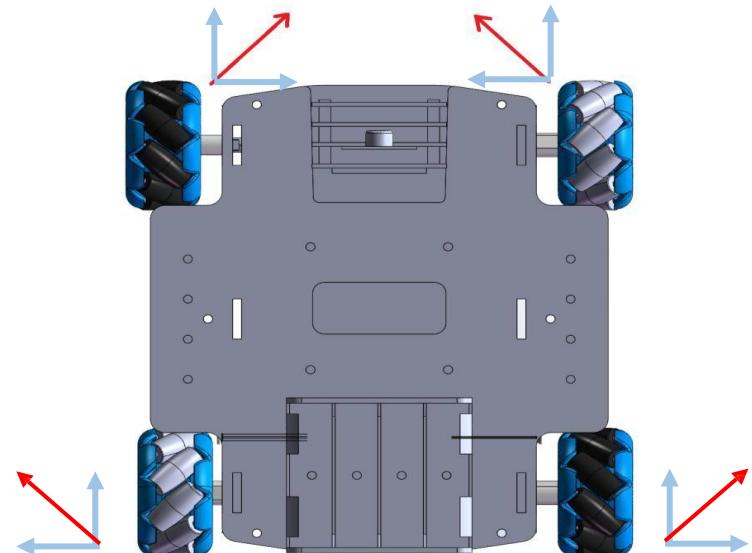
When the wheels **turn clockwise at the same time**, the velocity direction of the wheel is as follows (each wheel velocity can be divided into horizontal and vertical velocity):



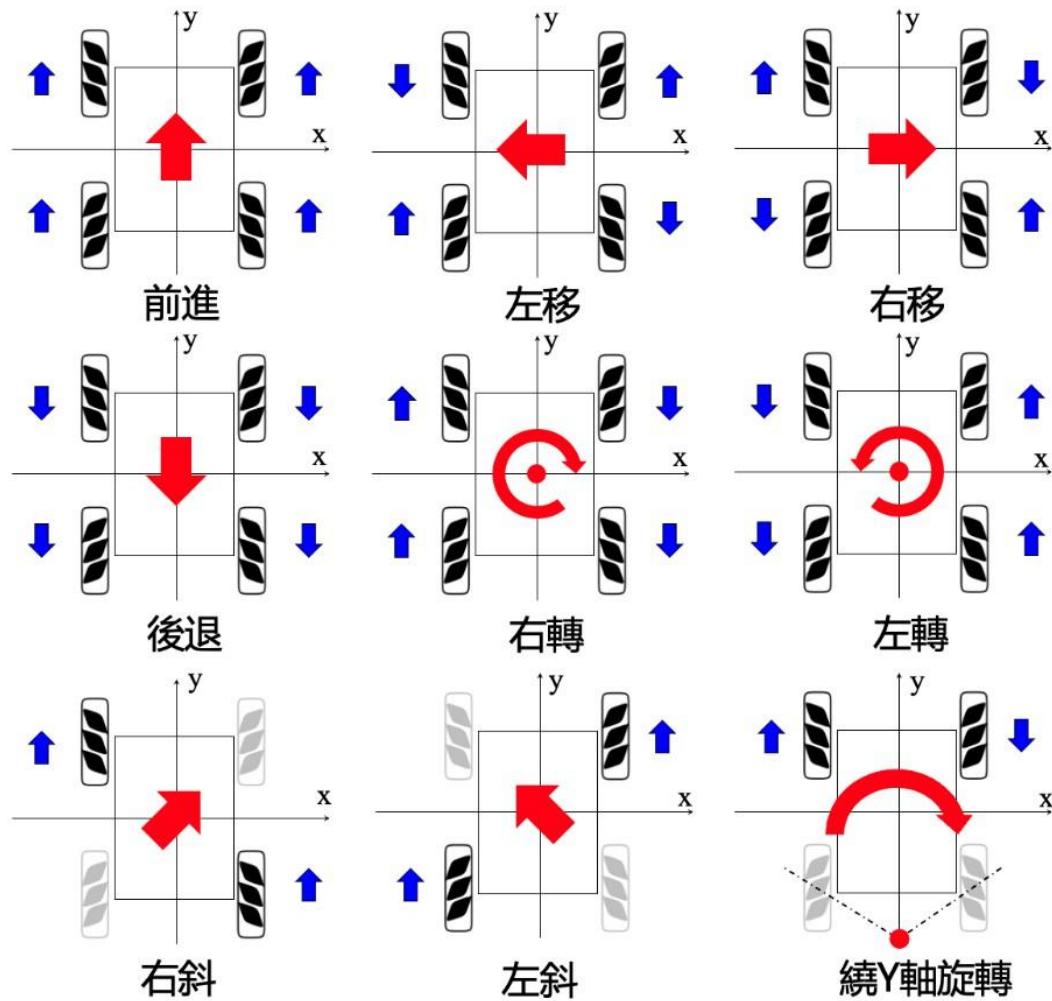
● Principle of Moving Mecanum Wheel Robot



If the wheels have the same velocity, their combined velocities are in the following direction (in the diagram, the horizontal velocities are equal in magnitude and opposite in direction, thus canceling out the horizontal velocities.) Only the vertical velocity is left to be accumulated):



● Principle of Moving Mecanum Wheel Robot



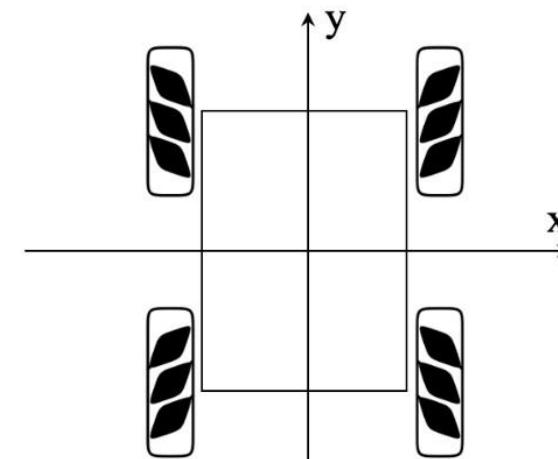
The overall direction of movement of the mecanum wheel robot

The direction in which the wheels turn

Motion Control of the Robot

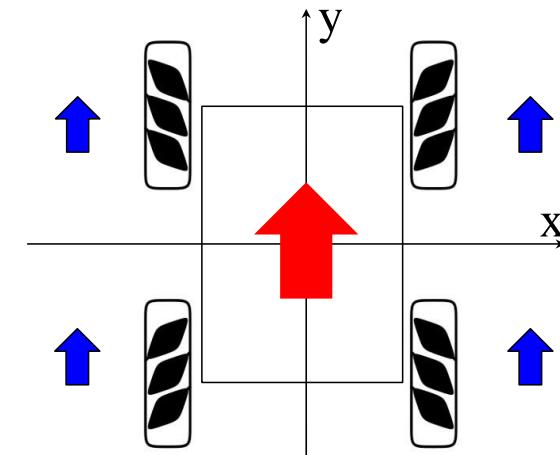
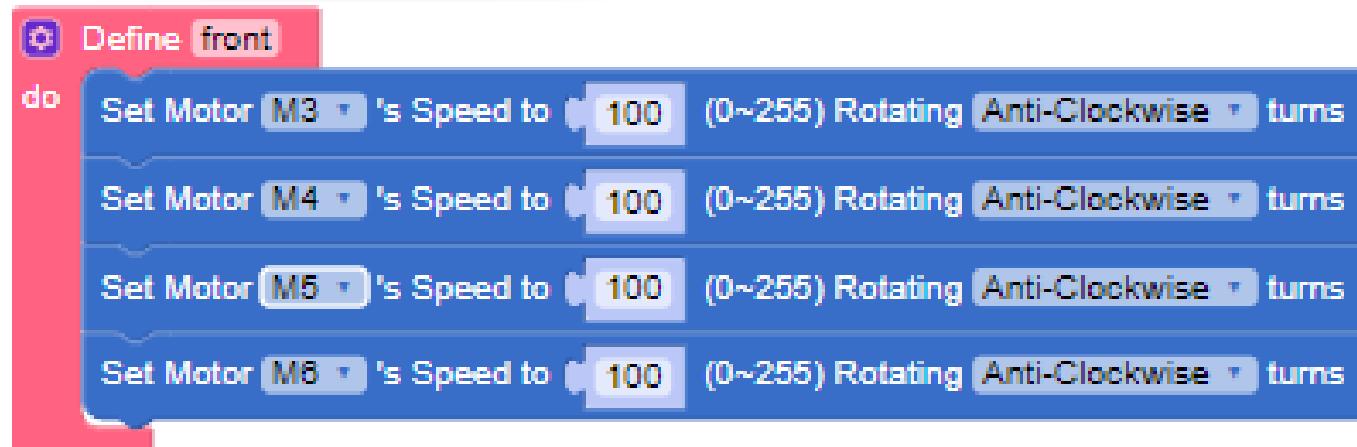
s t o p

```
Define stop
do
  Set Motor M3's Speed to 0 (0~255) Rotating Clockwise turns
  Set Motor M4's Speed to 0 (0~255) Rotating Clockwise turns
  Set Motor M5's Speed to 0 (0~255) Rotating Clockwise turns
  Set Motor M6's Speed to 0 (0~255) Rotating Clockwise turns
```



Motion Control of the Robot

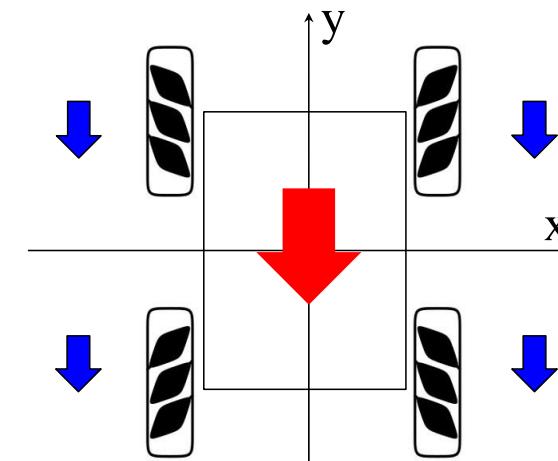
a d v a n c e



Motion Control of the Robot

Reverse

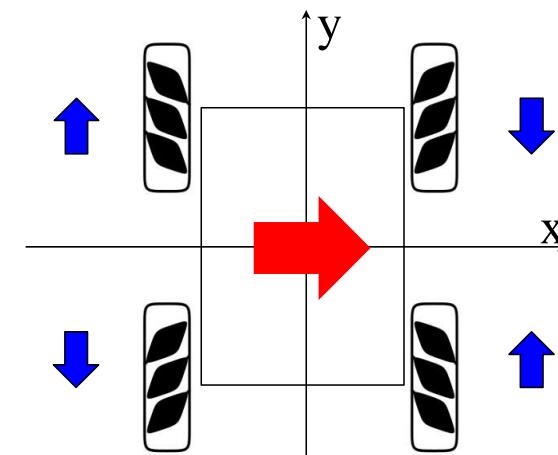
```
Define back
do
  Set Motor M3's Speed to 100 (0~255) Rotating Clockwise turns
  Set Motor M4's Speed to 100 (0~255) Rotating Clockwise turns
  Set Motor M5's Speed to 100 (0~255) Rotating Clockwise turns
  Set Motor M6's Speed to 100 (0~255) Rotating Clockwise turns
```



Motion Control of the Robot

Shift Right

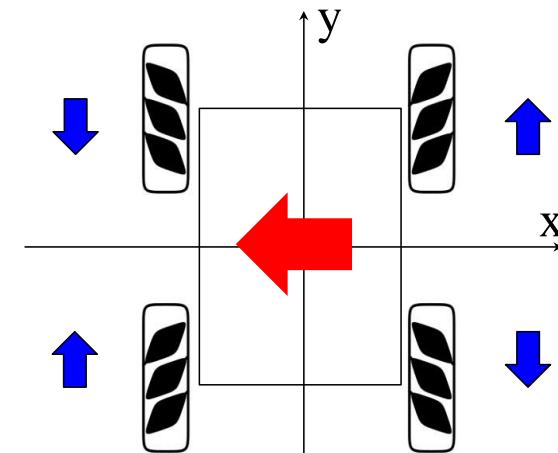
```
Define right
do
  Set Motor [M3] 's Speed to [100] (0~255) Rotating [Clockwise] turns
  Set Motor [M4] 's Speed to [100] (0~255) Rotating [Anti-Clockwise] turns
  Set Motor [M5] 's Speed to [100] (0~255) Rotating [Clockwise] turns
  Set Motor [M6] 's Speed to [100] (0~255) Rotating [Anti-Clockwise] turns
```



Motion Control of the Robot

Shift left

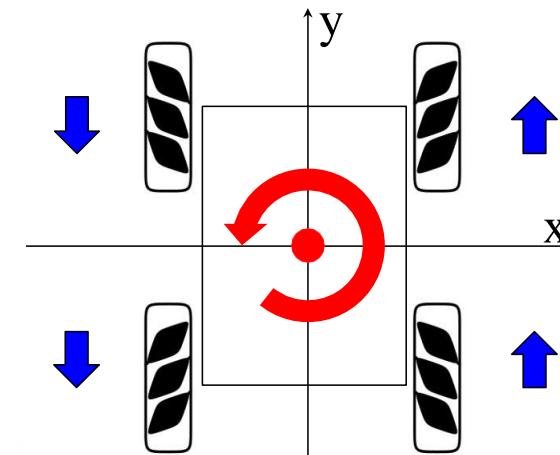
```
Define [left]
do
  Set Motor [M3] 's Speed to [100] (0-255) Rotating [Anti-Clockwise] turns
  Set Motor [M4] 's Speed to [100] (0-255) Rotating [Clockwise] turns
  Set Motor [M5] 's Speed to [100] (0-255) Rotating [Anti-Clockwise] turns
  Set Motor [M6] 's Speed to [100] (0-255) Rotating [Clockwise] turns
```



Motion Control of the Robot

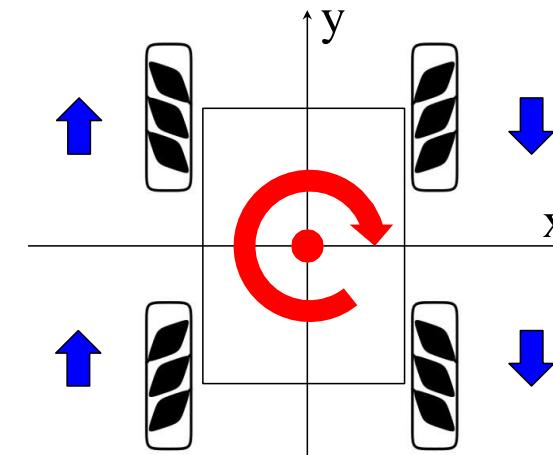
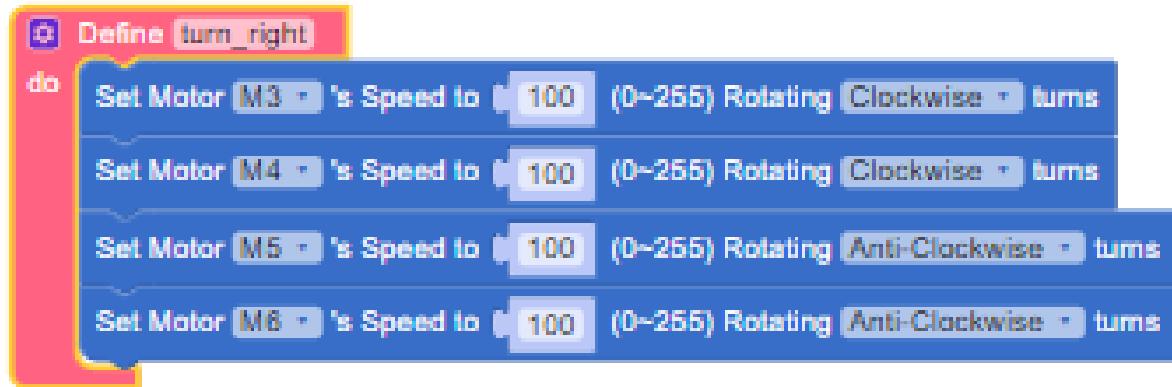
Turn left

```
Define [turn_left]
do
  Set Motor [M3] 's Speed to [100] (0~255) Rotating [Anti-Clockwise] turns
  Set Motor [M4] 's Speed to [100] (0~255) Rotating [Anti-Clockwise] turns
  Set Motor [M5] 's Speed to [100] (0~255) Rotating [Clockwise] turns
  Set Motor [M6] 's Speed to [100] (0~255) Rotating [Clockwise] turns
```



Motion Control of the Robot

Turn right



O

W

E

—

THREE.

Understand Bluetooth

Bluetooth

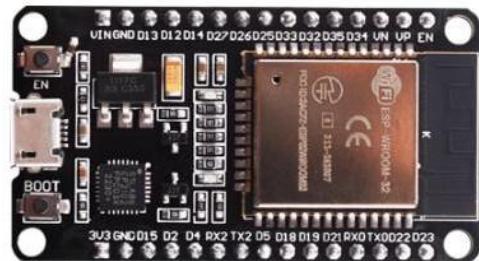
Bluetooth



Bluetooth technology is an open global specification for wireless data and voice communication, and it is based on low cost **Short Wireless** connectivity, a special type of short-range wireless technology connection that establishes a communication environment for fixed and mobile devices.



Bluetooth receiver module



**Bluetooth receiver
module**



Bluetooth sending handset

For usage of the Bluetooth module. the **robot must be powered on first**. Subsequently, after the Bluetooth receiving module is powered on, press the hand switch **[P3]** button to connect, when the 4 LEDs above the hand switch flash for 5 seconds and remain flashing, a successful connection was made. If the LED flashes for a few seconds and then turns off, it means that the connection fails.



Bluetooth receiver module

CocoPi



Small car



Bluetooth
receiver module



Game
Switches



[Hand switch] through **Bluetooth**. Send the key status to the [Bluetooth Receiving Module], and the [Bluetooth Receiving Module] will pass through **Serial port communication**. Send the button status to the [CocoPi Module], and finally the [CocoPi Module] controls the movement of the [Car] according to the pressing of the button.

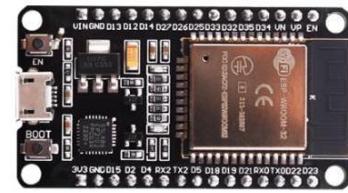
Bluetooth receiver module

-  **Task 1:** Manually press the key, and then display the keystroke data sent by the manual in the serial port data window.

CocoPi



Serial port communication



Bluetooth
receiver module

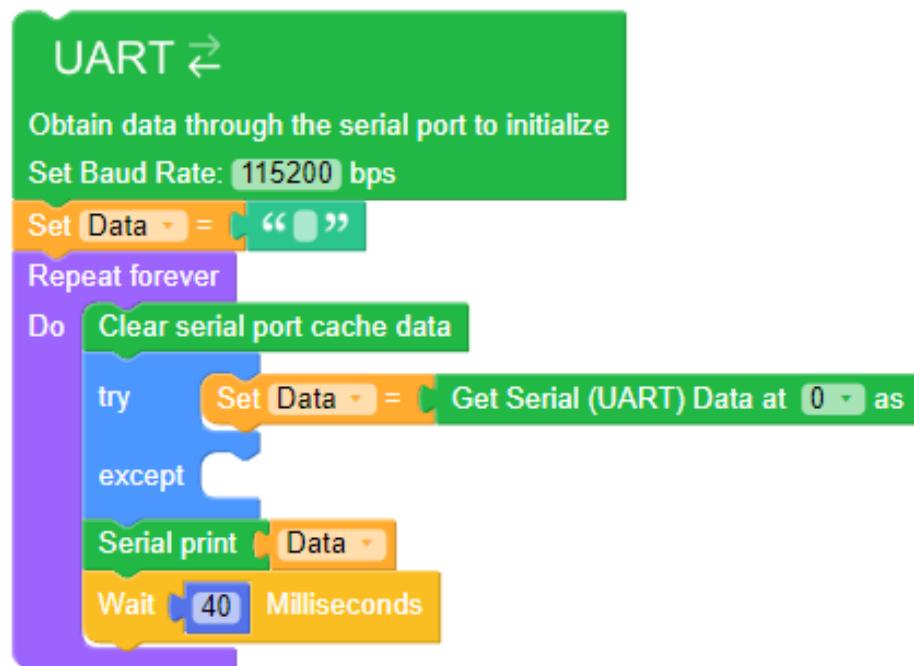


Bluetooth
handset

Bluetooth receiver module

 **Task 1:** Manually press the key, and then display the keystroke data sent by the manual in the serial port data window.

Reference

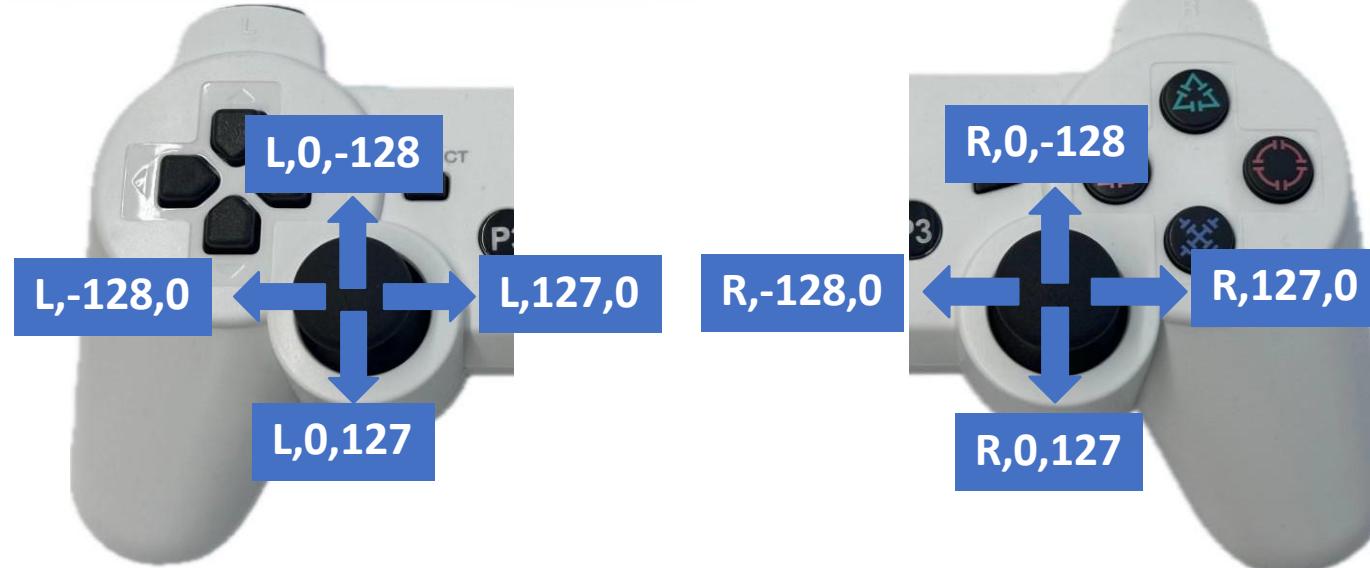


After 10 seconds of data sending experience, the serial port printing is delayed, and you can click "Run" again to experience the program

Bluetooth receiver module

-  **Task 1:** Manually press the key, and then display the keystroke data sent by the manual in the serial port data window.

The results are displayed



	UP
	DOWN
	LEFT
	RIGHT
	TRIANGLE
	CROSS
	SQUARE
	CIRCLE

● Key data parsing

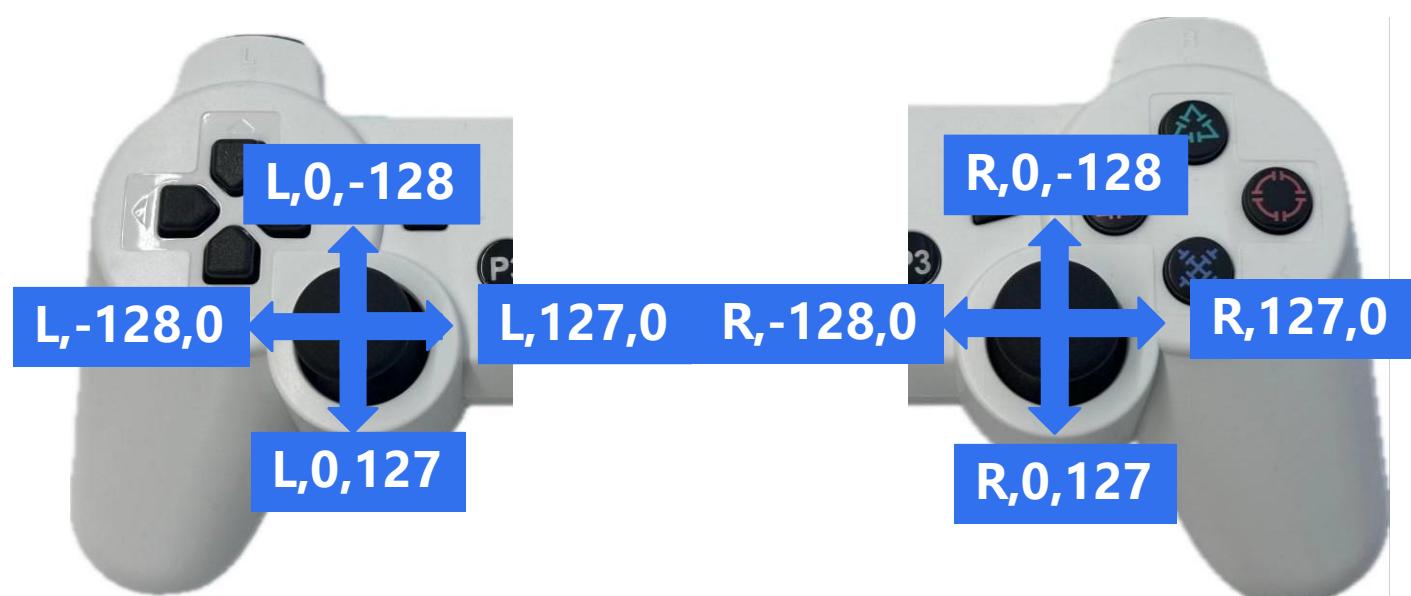
-  **Task 2:** On the basis of task 1, the data received by CocoPi is processed, and the specific keys are displayed in the sequence port window.



Key data parsing

Task 2:

On the basis of task 1, the data received by CocoPi is processed, and the specific keys are displayed in the sequence port window.



- The L and R stand for Left Joystick and Right Joystick, respectively.
- The second number indicates the range of movement of the stick's X axis, with values ranging from -128 to 127, with -128 being the leftmost and 127 being the rightmost.
- The third number indicates the range of Y-axis movement of the joystick, with values ranging from -128 to 127, with -128 being the top and 127 being the bottom.

Key data parsing



First, the received key data is divided into comma ',' and a list is generated, and then the first item in the list is determined to determine which joystick it is, the second item is determined to be left or right, and the third item is determined to be up or down.

L, 0, -128



Split string to a list by delimiter: Split and generate a list

['L', '0', '-128']

Left joystick

Go up

Shake rod data analysis

```

Set control = Split string to a list [Data] by delimiter: [Split and generate a list]
Do If [List [control] # 0 item = "L"]
  Do If [int [List [control] # 1 item > 100]
    Do Serial print ["Left: Right"]
  Else if [int [List [control] # 1 item < -100]
    Do Serial print ["Left: Left"]
  Else if [int [List [control] # 2 item > 100]
    Do Serial print ["Left: Down"]
  Else if [int [List [control] # 2 item < -100]
    Do Serial print ["Left: Up"]
  Else if [List [control] # 0 item = "R"]
    Do If [int [List [control] # 1 item > 100]
      Do Serial print ["Left: Right"]
    Else if [int [List [control] # 1 item < -100]
      Do Serial print ["Left: Left"]
    Else if [int [List [control] # 2 item > 100]
      Do Serial print ["Left: Down"]
    Else if [int [List [control] # 2 item < -100]
      Do Serial print ["Left: Up"]
  ]

```

Key data parsing

```

If [Data] = "UP"
  Do Serial print ["UP"]
Else if [Data] = "DOWN"
  Do Serial print ["DOWN"]
Else if [Data] = "LEFT"
  Do Serial print ["LEFT"]
Else if [Data] = "RIGHT"
  Do Serial print ["RIGHT"]
Else if [Data] = "TRIANGLE"
  Do Serial print ["TRIANGLE"]
Else if [Data] = "CROSS"
  Do Serial print ["CROSS"]
Else if [Data] = "SQUARE"
  Do Serial print ["SQUARE"]
Else if [Data] = "CIRCLE"
  Do Serial print ["CIRCLE"]

```

Key data parsing

Mission Two:

On the basis of task 1, the data received by CocoPi is processed, and the specific keys are displayed in the sequence port window.

Reference

```

UART ←
Obtain data through the serial port to initialize
Set Baud Rate: 115200 bps
Set Data = “ ”
Repeat forever
Do Clear serial port cache data
try Set Data = Get Serial (UART) Data at 0 as
except
Set control = Split string to a list “Data” by delimiter: Split and get
If List [control] # 0 item = “L”
Do If int [List [control] # 1 item] > 100
Do Serial print “Left: Right”
Else if int [List [control] # 1 item] < -100
Do Serial print “Left: Left”
Else if int [List [control] # 2 item] > 100
Do Serial print “Left: Down”
Else if int [List [control] # 2 item] < -100
Do Serial print “Left: Up”
Do If Data = “UP”
Do Serial print “UP”
Else if Data = “DOWN”
Do Serial print “DOWN”
Do Serial print “Left: Up”

```

```

Else if List [control] # 0 item = “R”
Do If int [List [control] # 1 item] > 100
Do Serial print “Left: Right”
Else if int [List [control] # 1 item] < -100
Do Serial print “Left: Left”
Else if int [List [control] # 2 item] > 100
Do Serial print “Left: Down”
Else if int [List [control] # 2 item] < -100
Do Serial print “Left: Up”
Do If Data = “UP”
Do Serial print “UP”
Else if Data = “DOWN”
Do Serial print “DOWN”

```

```

Else if Data = “LEFT”
Do Serial print “LEFT”
Else if Data = “RIGHT”
Do Serial print “RIGHT”
Else if Data = “TRIANGLE”
Do Serial print “TRIANGLE”
Else if Data = “CROSS”
Do Serial print “CROSS”
Else if Data = “SQUARE”
Do Serial print “SQUARE”
Else if Data = “CIRCLE”
Do Serial print “CIRCLE”
Wait 40 Milliseconds

```

FOUR.

Manual Control
of Mecanum
wheels for the
robot



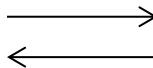
● Manual Control of the Mecanum Wheel Robot



Task 3: Use the hand button to control the mecanum wheel robot to move forward, backward, left and right; The right joystick controls the mecanum wheel robot to rotate left and right.



 Bluetooth®



Sending end



Receiver

Manual Control of the Mecanum Wheel Robot

-  **Task 3:** Use the hand button to control the mecanum wheel robot to move forward, backward, left and right; The right joystick controls the mecanum wheel robot to rotate left and right.

Step analysis

Initialize the settings

Step1

Perform the corresponding tasks based on the received data

Step2

Judge the data received

Step3

Manual Control of the Mecanum Wheel Robot



Task 3: Use the hand button to control the mecanum wheel robot to move forward, backward, left and right; The right joystick controls the mecanum wheel robot to rotate left and right.

Reference

```

Motor Driver Setup
Set Motor [M3] 's Speed to 0 (0~255) Rotating Clockwise turns
Set Motor [M4] 's Speed to 0 (0~255) Rotating Clockwise turns
Set Motor [M5] 's Speed to 0 (0~255) Rotating Clockwise turns
Set Motor [M6] 's Speed to 0 (0~255) Rotating Clockwise turns

UART →
Obtain data through the serial port to initialize
Set Baud Rate: 115200 bps
Set [Data] = " "
Repeat forever
Do Clear serial port cache data
try Set [Data] = Get Serial (UART) Data at 0 as
except
Set [control] = Split string to a list [Data] by delimiter: Split and generate a list
If List [control] # 0 item = "L"
Do If int List [control] # 1 item > 100
Do right
Else if int List [control] # 1 item < -100
Do left

```

```

Else if int List [control] # 2 item > 100
Do back
Else if int List [control] # 2 item < -100
Do front
Else if List [control] # 0 item = "R"
Do If int List [control] # 1 item > 100
Do turn_right
Else if int List [control] # 1 item < -100
Do turn_left

Define front
do Set Motor [M3] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M4] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M5] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M6] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns

Define back
do Set Motor [M3] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M4] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M5] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M6] 's Speed to 100 (0~255) Rotating Clockwise turns

Define right
do Set Motor [M3] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M4] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M5] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M6] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns

Define left
do Set Motor [M3] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M4] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M5] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M6] 's Speed to 100 (0~255) Rotating Clockwise turns

Define turn_right
do Set Motor [M3] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M4] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M5] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M6] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns

Define turn_left
do Set Motor [M3] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M4] 's Speed to 100 (0~255) Rotating Anti-Clockwise turns
Set Motor [M5] 's Speed to 100 (0~255) Rotating Clockwise turns
Set Motor [M6] 's Speed to 100 (0~255) Rotating Clockwise turns

```

Manual Control of the Mecanum Wheel Robot

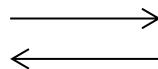
-  **Task 3:** Use the hand button to control the mecanum wheel robot to move forward, backward, left and right; The right joystick controls the mecanum wheel robot to rotate left and right.

Small Car Preparation Trial

Use the hand to remotely control the car and see if you can control it normally



 **Bluetooth®**



發送端



接收端

Manual Control of the Mecanum Wheel Robot

problem solving

The motor needs to give a stop command to stop rotating, so even if the robot stops the button after a movement, the robot will still move according to the previous motion mode, so we need to control the robot to stop moving when the button is not pressed.

The controller will continue to send when the button is not pressed '**NONE**' The information comes.

A Scratch script for controlling a Mecanum wheel robot. It uses a list variable 'control' and a data variable 'Data'. The script starts with an 'If' condition checking the value of 'Data'. If 'Data' is equal to 'NONE', it performs a 'stop' action. Otherwise, it enters a 'Do' loop. Inside the loop, it checks the value of 'control' at index 0. If it's 'L', it performs a 'right' motion; if it's 'R', it performs a 'turn_right' motion. If neither, it checks index 1. If greater than 100, it performs a 'left' motion; if less than -100, it performs a 'turn_left' motion. If none of these conditions are met, it performs a 'back' motion. After the 'Do' loop, it checks index 2 of 'control'. If greater than 100, it performs a 'front' motion; if less than -100, it performs a 'turn_left' motion. Finally, it checks index 0 again. If it's 'L', it performs a 'right' motion; if 'R', it performs a 'turn_right' motion. If none of these conditions are met, it performs a 'left' motion.

```
if (Data) = "NONE" then
    stop
else
    do
        if (List [control v] [0] item) = "L" then
            right
        else if (List [control v] [0] item) = "R" then
            turn-right
        else if (List [control v] [0] item) > 100 then
            left
        else if (List [control v] [0] item) < -100 then
            turn-left
        else if (List [control v] [0] item) > 100 then
            back
        else if (List [control v] [0] item) < -100 then
            front
        else if (List [control v] [0] item) = "L" then
            right
        else if (List [control v] [0] item) = "R" then
            turn-right
        else
            left
    end
end
```

Manual Control of the Mecanum Wheel Robot



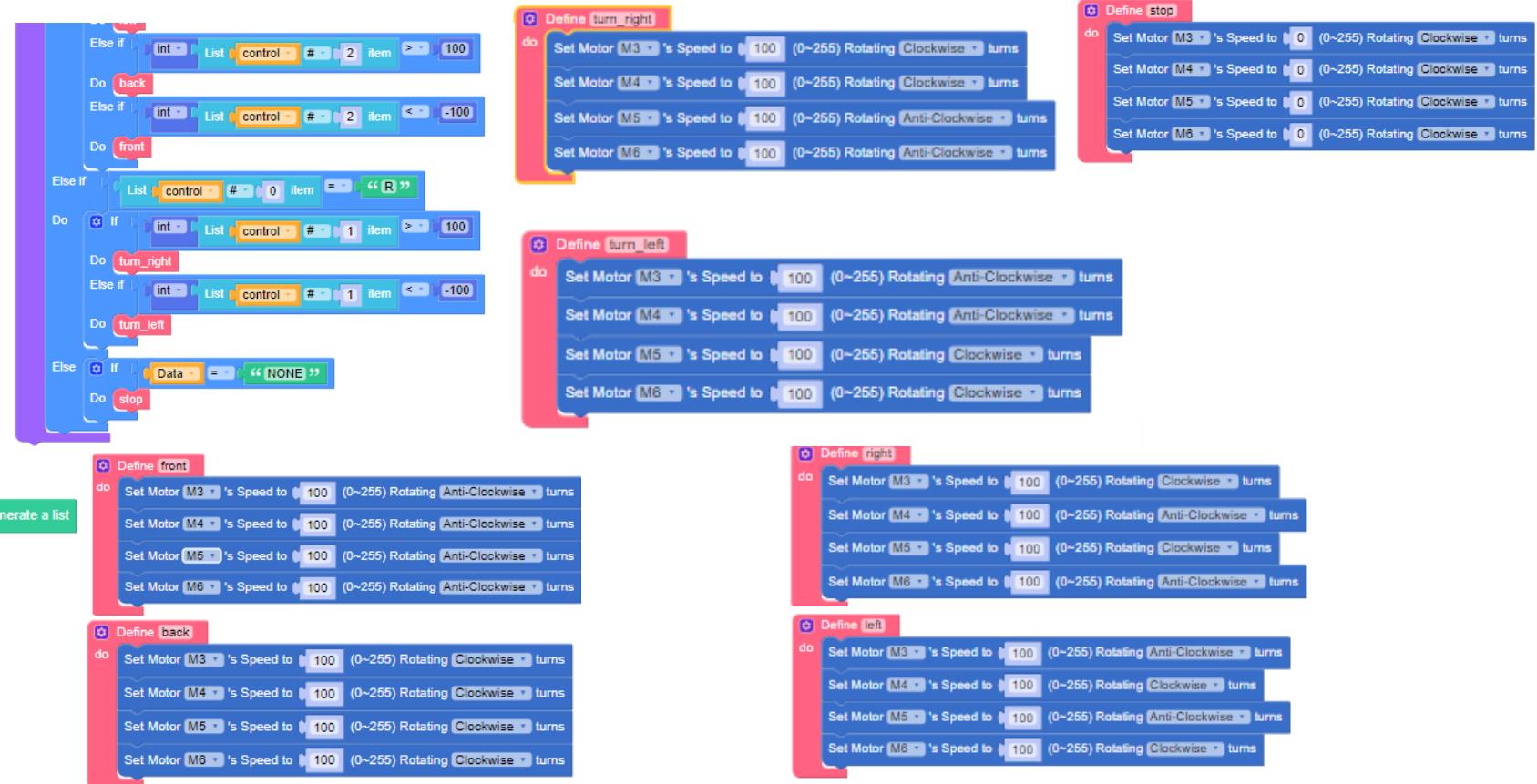
Reference

```

Motor Driver Setup
Set Motor M3 's Speed to 0 (0~255) Rotating Clockwise turns
Set Motor M4 's Speed to 0 (0~255) Rotating Clockwise turns
Set Motor M5 's Speed to 0 (0~255) Rotating Clockwise turns
Set Motor M6 's Speed to 0 (0~255) Rotating Clockwise turns

UART ←
Obtain data through the serial port to initialize
Set Baud Rate: 115200 bps
Set Data = " "
Repeat forever
Do Clear serial port cache data
try Set Data = Get Serial (UART) Data at 0 as
except
Set control = Split string to a list Data by delimiter: Split and generate a list
If List control # 0 item = "L"
Do If int List control # 1 item > 100
Do right
Else if int List control # 1 item < -100
Do left
Else if int List control # 2 item > 100
Do back
Else if int List control # 2 item < -100
Do front
Else if List control # 0 item = "R"
Do If int List control # 1 item > 100
Do turn_right
Else if int List control # 1 item < -100
Do turn_left
Else If Data = "NONE"
Do stop

```



● Manual Control of the Mecanum Wheel Robot

Precautions



1. If the controller is not used for more than 5 minutes, it will be disconnected.
2. When the Bluetooth receiver module is turned off, the controller will be automatically disconnected.
3. You need to turn on the robot to power on the Bluetooth receiving module before running the program, and then use the hand button to press [P3] to connect.
4. The controller shows successful connection but the LED light goes out very quickly. It may be due to insufficient power, which requires charging to be performed.

See you in the
next session!

T H A N K S

J U S T L E A V E P R E S E N T A T I O N O T T O R I C H I T E C T U R E