DFROBOT
DRIVE THE FUTURE

Maqueen Plus V2

Getting Started Tutorial

MakeCode

content

Chapter 1

Introduction to Maqueen Plus V2

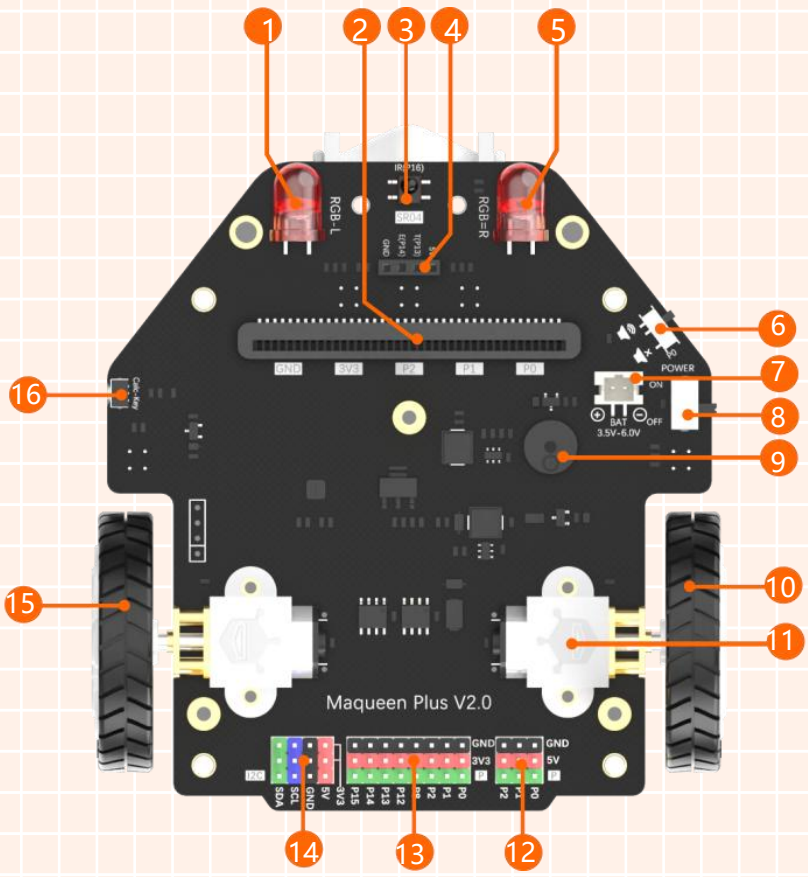
Introduction

What is Maqueen Plus V2?

Maqueen Plus V2 is the upgraded version of the STEAM Maqueen Plus educational robot, offering a variety of enhanced features to strengthen its capabilities, more flexible and easy to use. It supports graphical coding platforms like MakeCode and Mind+, on which we can program it to make various interesting projects by simply dragging and snapping blocks. Follow Maqueen Plus V2 to explore the world of robotics, while learning coding in a fun way!

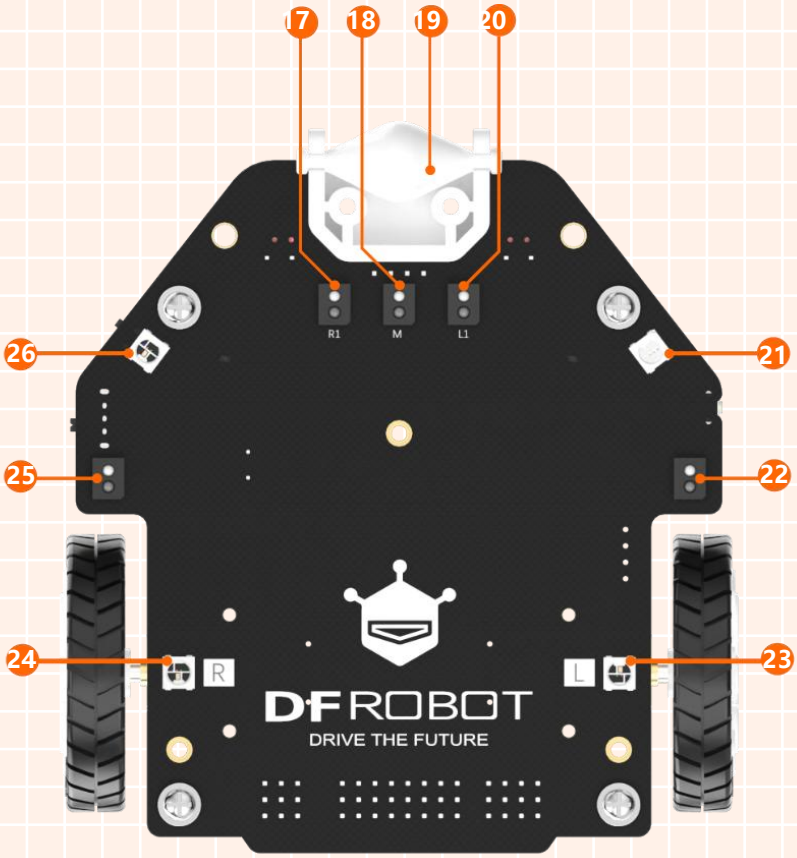
Before we get started, let's see what Maqueen Plus V2 has got there first.

Front view for Maqueen Plus V2 main-board



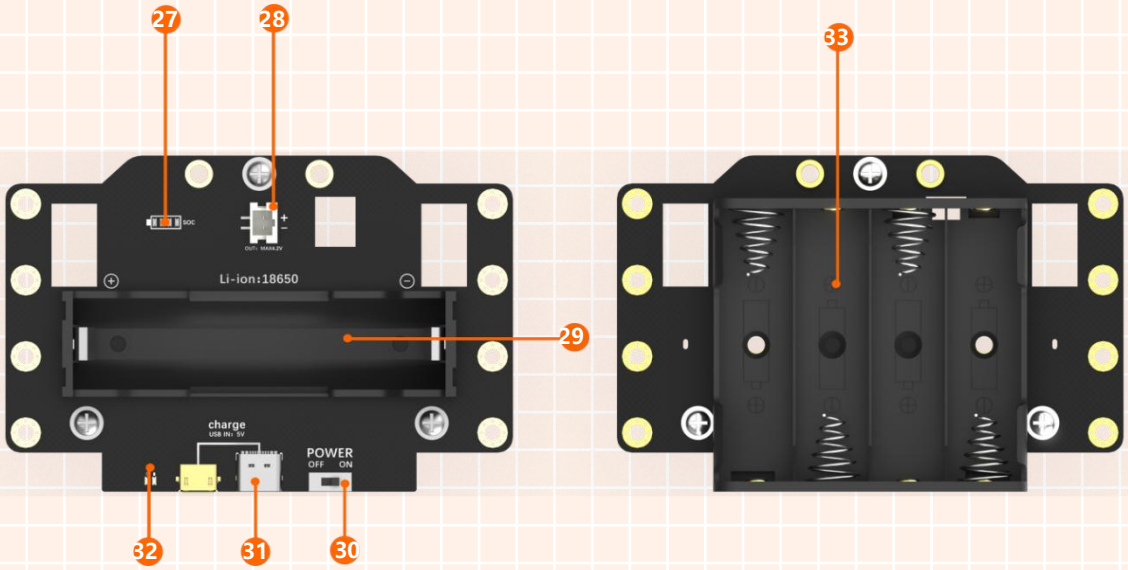
1	RGB-LED-L	2	micro:bit socket	3	Infrared receiver	4	Ultrasonic sensor port	5	RGB-LED-R	6	Buzzer switch
7	Power supply port	8	Power supply switch	9	Buzzer	10	Right wheel	11	Metal motor	12	Servo port
13	I/O port	14	I2C port	15	Left wheel	16	Line-tracking reset key				

Back view for Maqueen Plus V2 main-board



17	R1 line-tracking sensor	18	M line-tracking sensor	19	Support wheel	20	L1 line-tracking sensor	21	RGB LED 0
22	L2 line-tracking sensor	23	RGB LED 1	24	RGB LED 2	25	R2 line-tracking sensor	26	RGB LED 3

Overview for 18650 & AA Battery Case



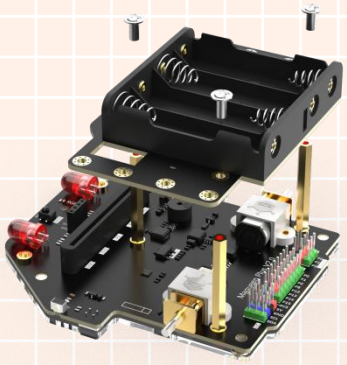
27 Power supply indicator
28 Power supply port
29 18650 battery holder
30 Power supply switch
31 USB charging port
32 Charging indicator
33 AA battery holder

Maqueen Plus V2 has been equipped with so many functions, and now we can’ t wait to try them. OK, here we go!

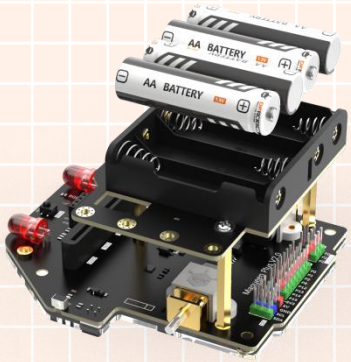
Assembly

Maqueen Plus V2 - AA Battery

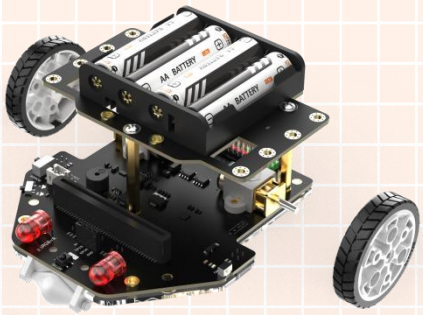
Assembly Diagram for Maqueen Plus V2 - AA Battery



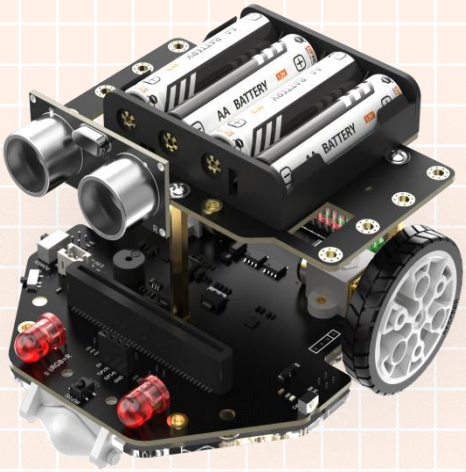
1 Install AA battery



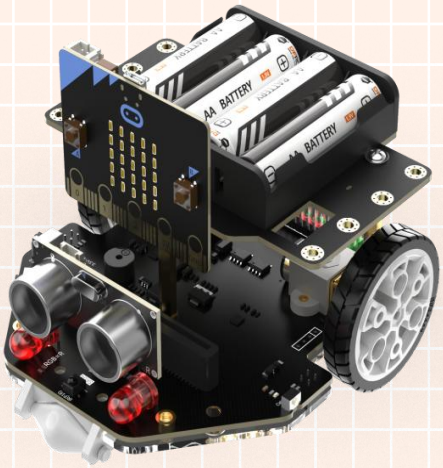
2 Install 4 AA batteries



3 Install wheels



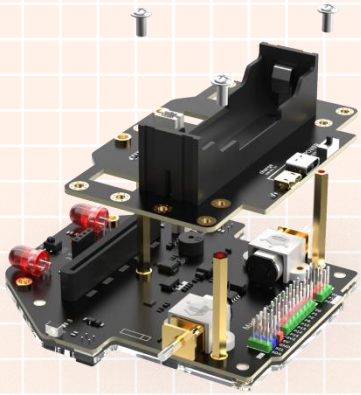
4 Install ultrasonic sensor



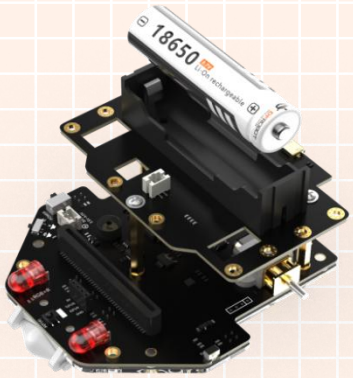
5 Plug in micro:bit board

Maqueen Plus V2 -18650 Battery

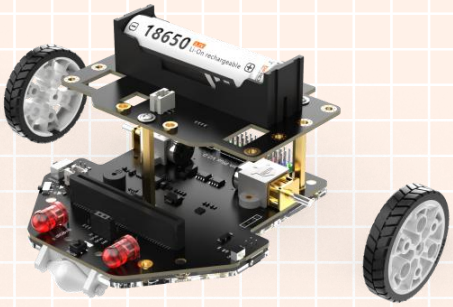
Assembly Diagram for Maqueen Plus V2 -18650 Battery



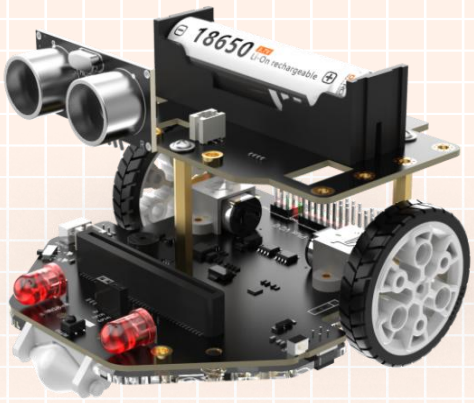
1 Install 18650 battery case



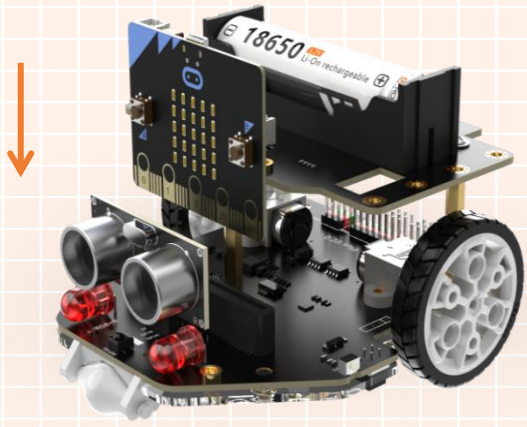
2 Install 18650 battery



3 Install wheels



4 Install ultrasonic sensor



5 Plug in micro:bit board

6 Battery indicator

7 Charging port

Note: when the 18650 battery is fully charged, all LEDs will be on. The LEDs will be off one by one as the power gradually decreases. If all lights go out, the battery needs to be recharged.

After we assembled Maqueen Plus V2, put it aside because first, we need to get familiar with its most important controller device – micro:bit. Just like the CPU in a computer, micro:bit is Maqueen Plus V2's "brain" for storing and processing data, which also is the key to making Maqueen Plus V2 "alive". (Maqueen Plus V2 - AA battery version is used in this tutorial.)

Introduction to micro:bit

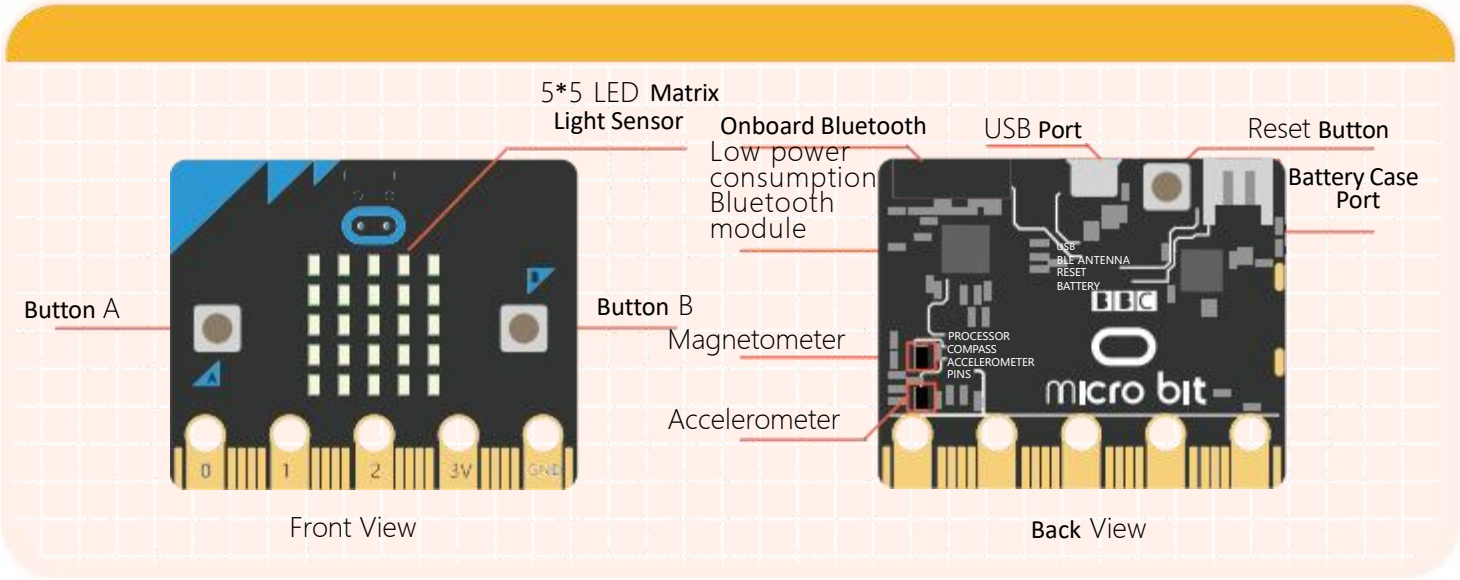
What can micro:bit do?

The micro:bit can be programmed to do various interesting things. For example, it can interconnect with automobiles via onboard Bluetooth, thus the automobile can send commands to control the MCU. Multiple micro:bit boards can communicate with each other via radio to complete the tasks requiring remote control. Also, it can be a navigation device as it features an on-board compass to track the direction of the wearer. And there are 5 × 5 LED matrix on the board that can display funny patterns and animation, and two programmable buttons that can be used to control game-play or pause/play a track.

In addition, micro:bit is equipped with commonly-used sensors like light sensor and temperature sensor. So it can be widely used in light-related projects and applications requiring temperature detection, and so on.

micro:bit Function

On the credit card size board (5cm*4cm/1.97*1.57") there are 5 × 5 programmable LED matrix, two programmable buttons, light sensor, accelerometer, compass, temperature sensor, Bluetooth module and other electric modules.



5×5 Programmable LED Matrix	There are 25 programmable LED lights on micro:bit which can display patterns, words, and numbers.
2 Programmable Buttons	Used separately or together to make things happen. For example, press down A to display a heart pattern.
Light Sensor	The 25 LEDs can act as light sensors to measure how much light is falling on the micro:bit.
Accelerometer and Compass	Measure the gestures or forces in 3 dimensions, such as shaking, tilting, free fall.
Temperature Sensor	Detect the temperature in the current environment.
Bluetooth & Radio	Your micro:bit can communicate with other micro:bits by radio, and with other devices using Bluetooth.

micro:bit Programming

We can directly program micro:bit online without installing any software. There is a huge community of people making tools for programming and interacting with the micro:bit which means you can program

your micro:bit in Python, C++ and other languages, including all kinds of block editors: MakeCode, Blocks, Scratch, Mind+, etc.In this tutorial, we will use MakeCode to program, click the link https://makecode.microbit.org/ to enter MakeCodeonline programming environment.

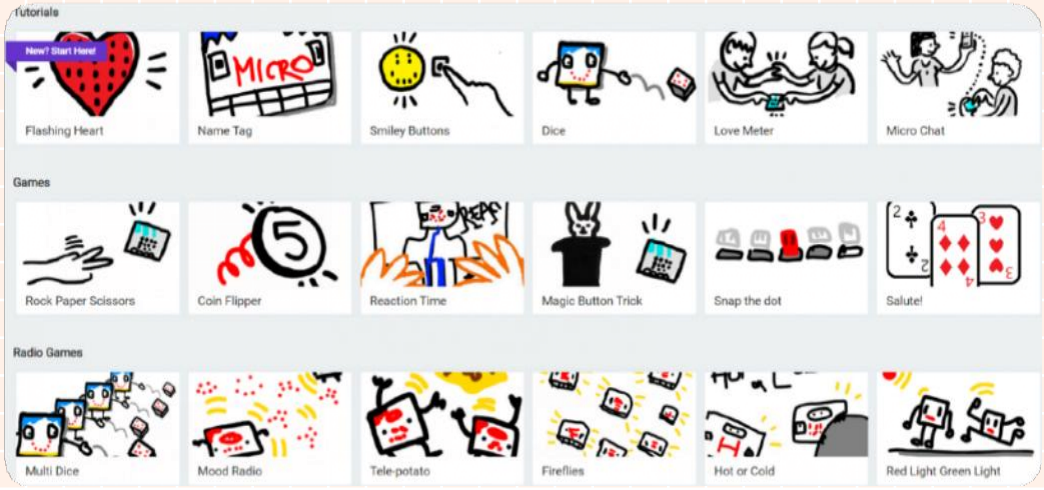
MakeCode Online Programming

MakeCode Introduction

Microsoft MakeCode is a framework for creating interactive and engaging programming experiences for those new to the world of programming. The primary goal of MakeCode is to introduce programming in a way that is approachable and inviting. MakeCode uses the blocks programming model to let the users learn coding concepts in a more tangible fashion.

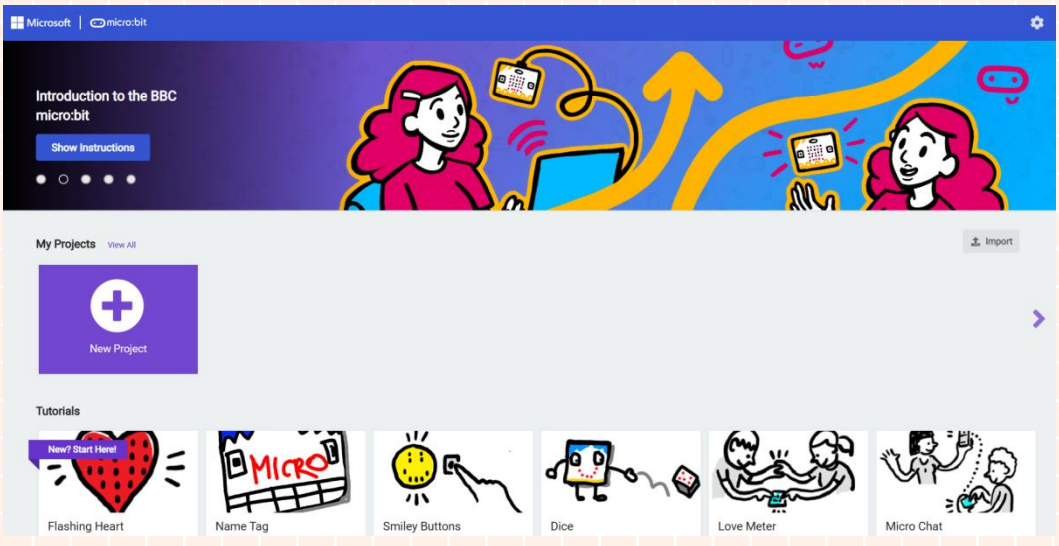
Once the user becomes comfortable with the coding elements and structure like the simulator, block editor and text editor in MakeCode, they can progress to create more complex programs.

Projects in MakeCode

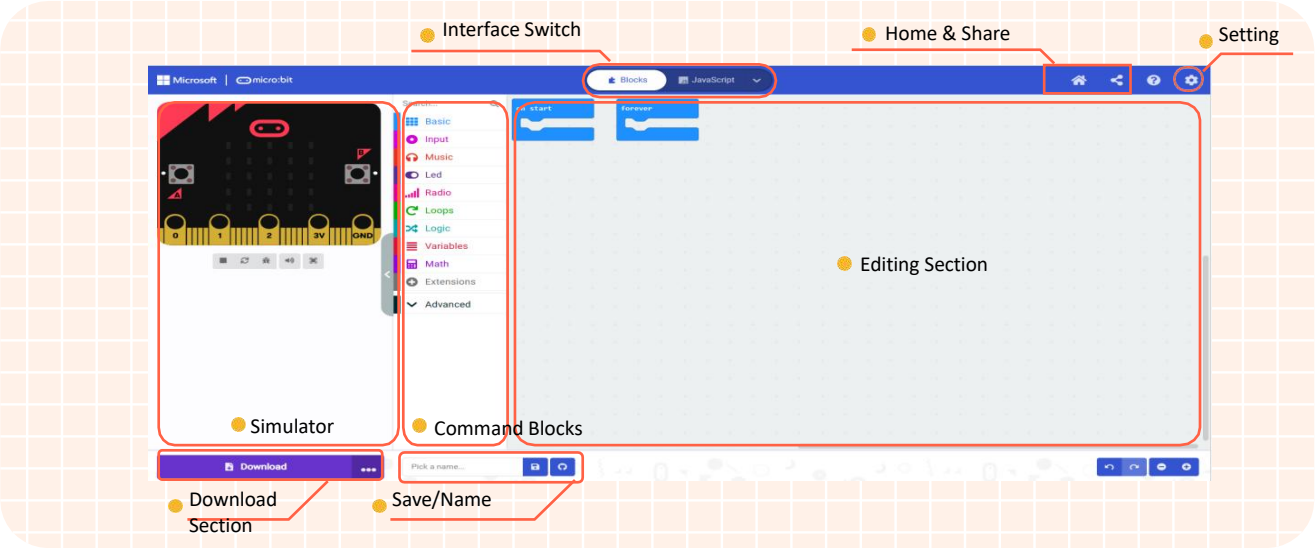


Build Up Programming Environment

1. Input https://makecode.microbit.org/ in your browser to enter the MakeCode programming environment.
   Note: it should be operated on a computer with a good Internet connection. If it cannot be loaded properly, please try it again using the Google browser



2. Open MakeCode editor and create a new project to enter MakeCode programming interface.

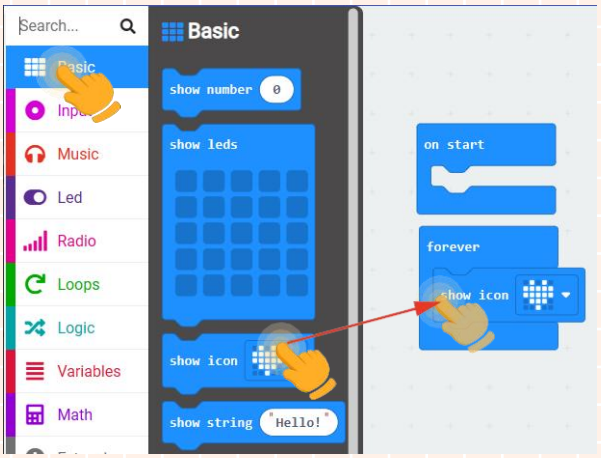


Name	Function
Home & Share	Home: create a new project, open tutorial, game, music, fashion, etc. Share: publish your project to share it or embed it in other web pages.
Interface Switch	Blocks: graphical programming, suitable for primary and middle school students. JavaScript: code in JavaScript, suitable for high school, college students, and above.
Setting	Setting: project setting, extensions, delete project, language, reset, MakeCode version number, etc.
Simulator	Simulator: test the result of your program before downloading it into micro:bit. Preview Control: start, stop and restart the simulator, slow motion, mute audio, full-screen, etc.
Command Blocks	Blocks: 17 categories of programming blocks and more extended blocks.
Editing Section	Programming Editor: construct your program by dragging and snapping colorful blocks
Download Section	Download: download the program you edited into micro:bit.
Save/Name	Name: name your project. Save: save your project. Note: when you click "save", the program will be downloaded at the same time.

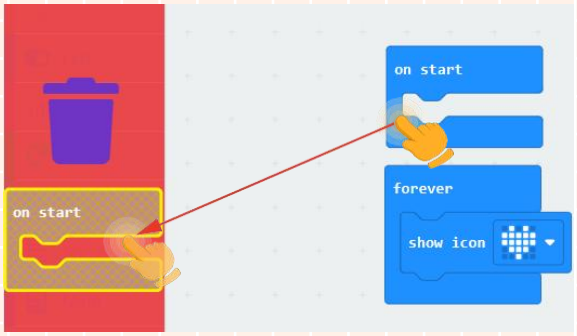
After we got a general understanding of MakeCode, let's step on our journey to code! In the first example, we will learn how to write and download a program.

1. Write a program

Step 1: drag the block we need to the editing section.



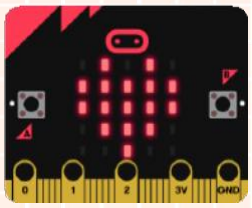
Step 2: to remove a block, drag it from the editing section to the command section, or right-click to delete.



Knowledge Expansion

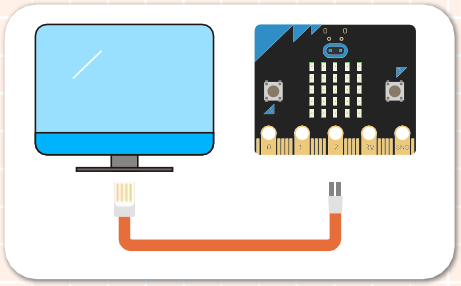
Hover your mouse pointer over the block, then the pointer will become" "; left-click to select the block, the pointer becomes " "; when you drag the block to the command section, it will become " " , then release the mouse to remove the block.

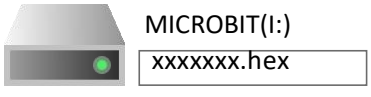
Step 3: after we completed a program, check its effect via the simulator.



2. Prepare to download

Connect the micro:bit board to your computer by a USB cable. There will be a hard disk named micro:bit appearing in your computer when the connection is successful.

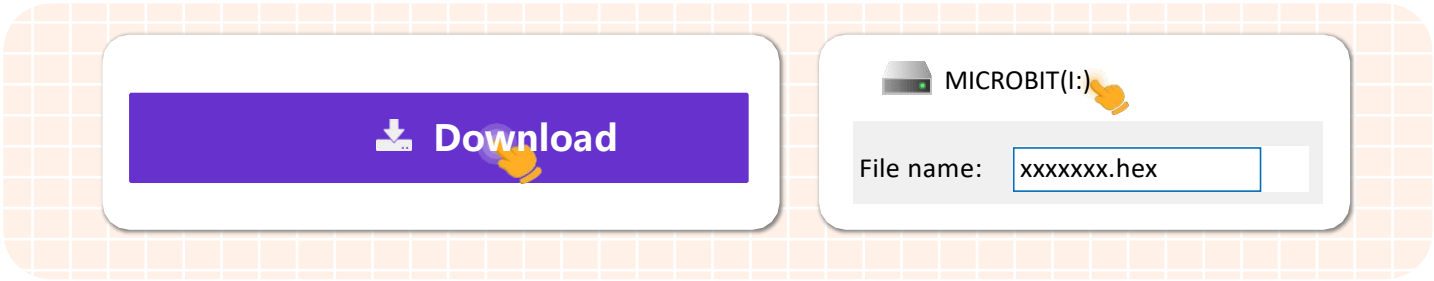




3. Download

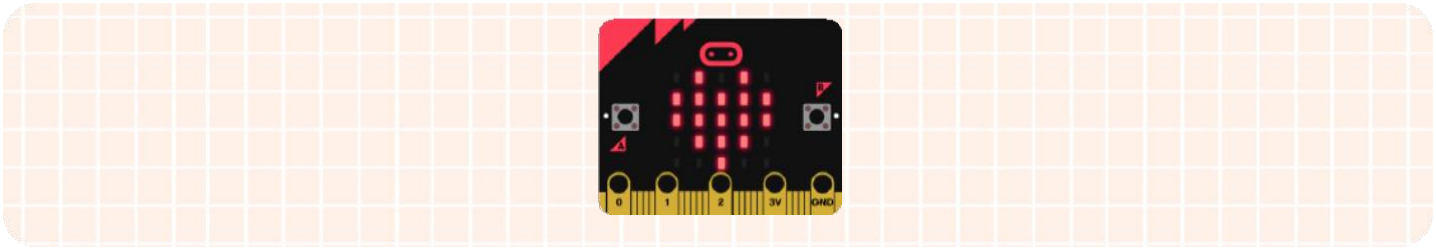
Click "Download" to download the program into your micro:bit.

Note: when downloading a program, the micro:bit power indicator will keep flashing, and please do not disconnect the USB cable.

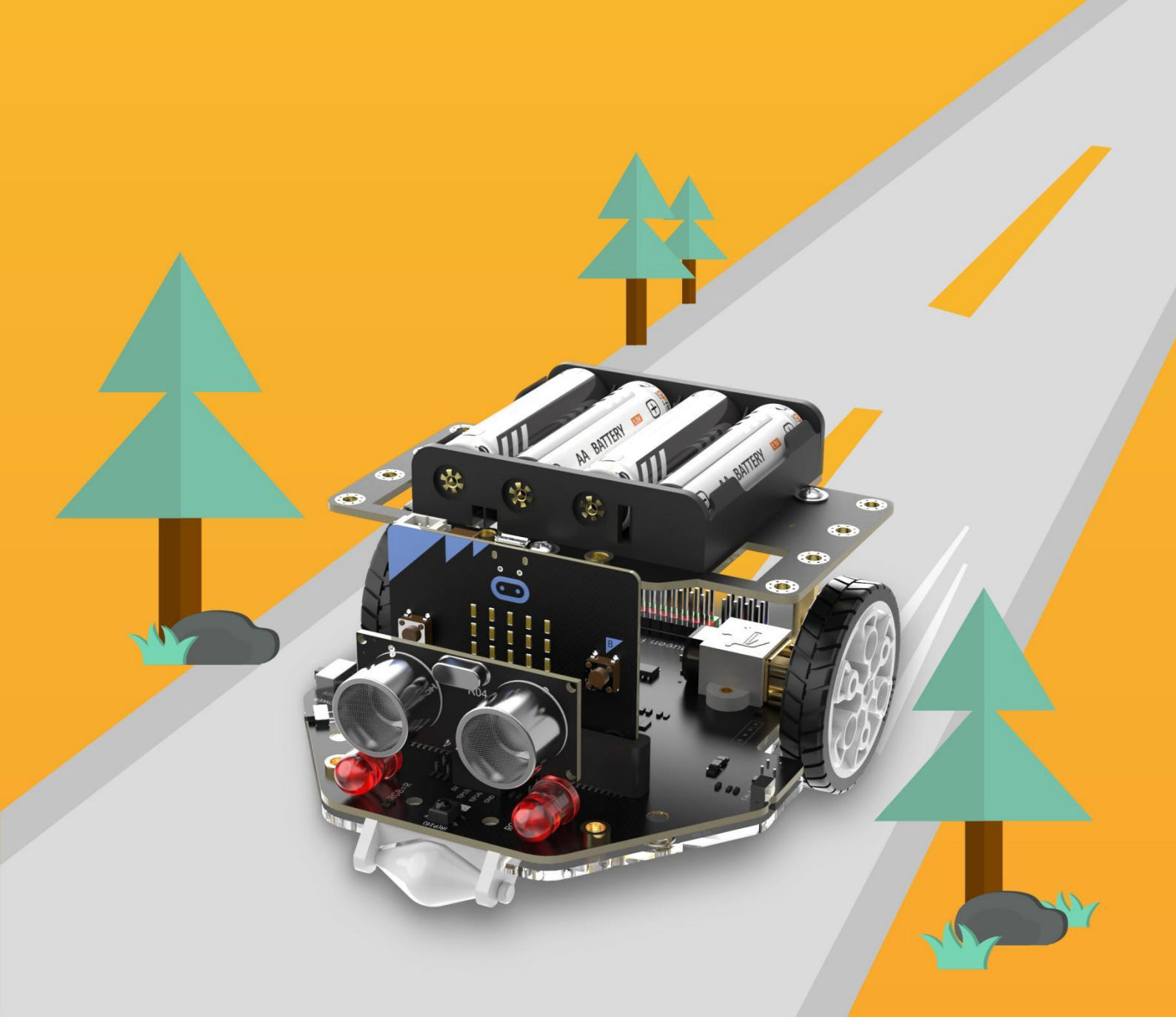


4. Download completes

Once the downloading is completed, the micro:bit LED screen will show a heart pattern.



Since we have learned the basics about Maqueen Plus V2, micro:bit and MakeCode programming, so for the next chapter, we are going to write a program to let micro:bit drive Maqueen Plus V2.



Chapter 2

Let's move, Maqueen!

Here is our Maqueen Plus V2, look at this cool guy! You must be wanna play with him right now. OK, let's get started.

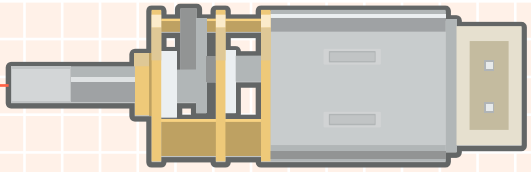
Goal

Learn how to drive a motor.

Electronic Component

Motor Brief

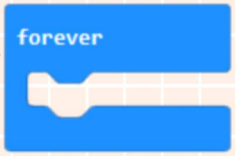
Motor



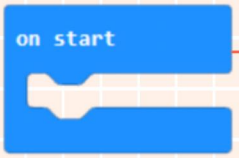
Motors can be used to drive Maqueen Plus V2 to move left, right, forward, and backward.

Command Learning

Forever



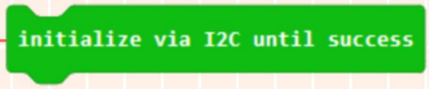
Run and keep repeating the code embedded into the block.



On start

Run the program inside it when the program starts

Initialization



Restore Maqueen Plus V2 to the default state.

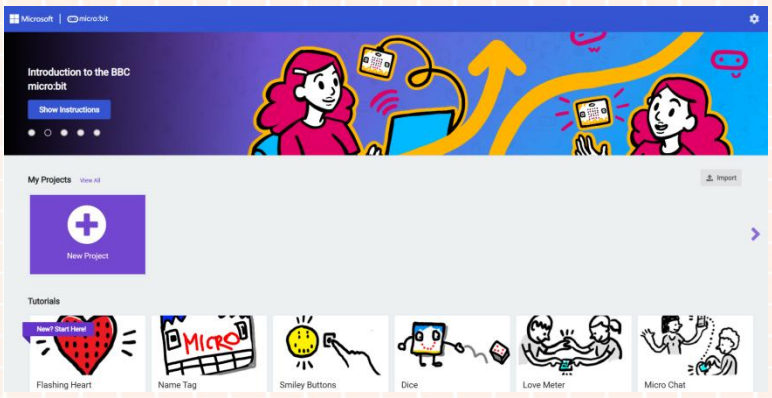
Motor Controlling

Control the speed and direction of the motor.

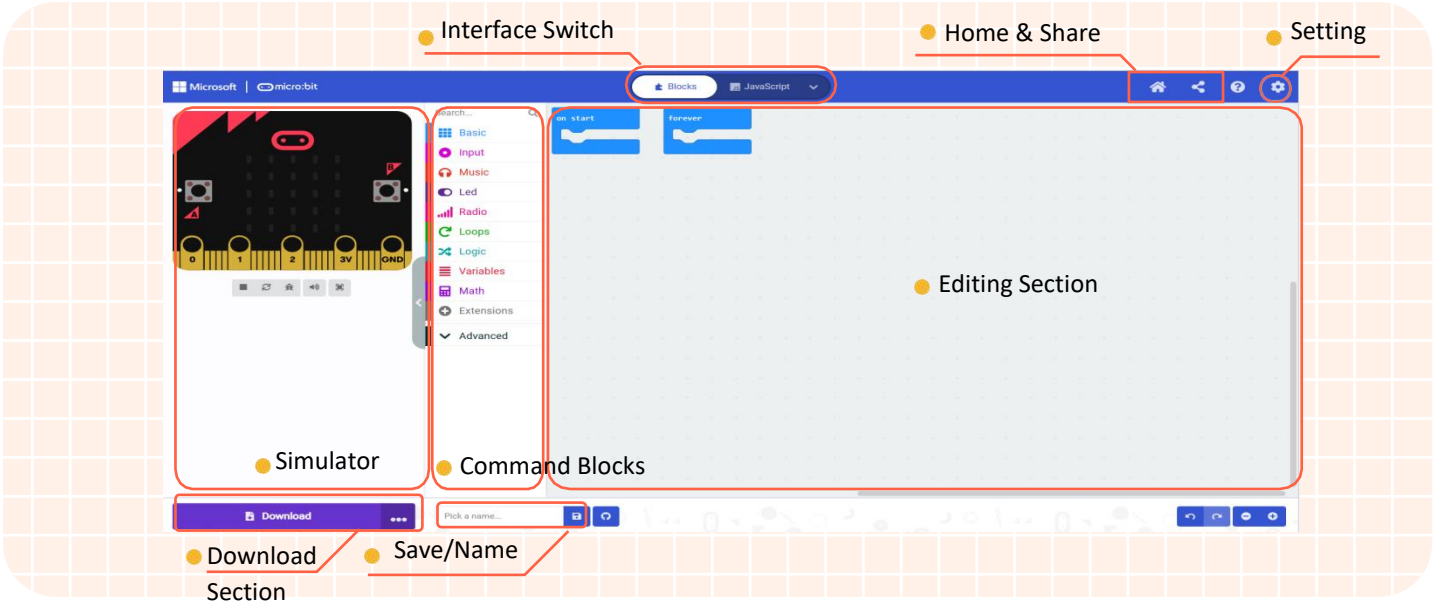
Hands-on Practice

Step 1 Create a new project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.

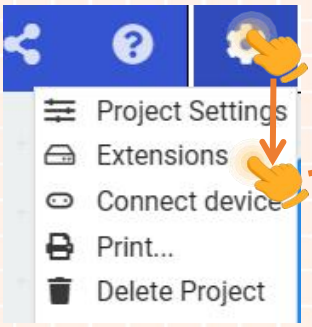


2. Click "new project" to enter the MakeCode programming interface.

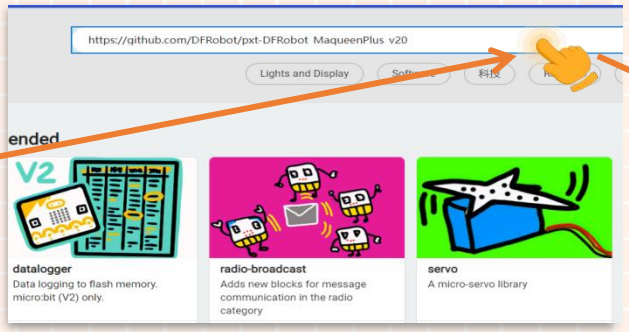


Step 2 Add the Maqueen Plus V2 library

1. To program Maqueen Plus V2, we have to find the related blocks in MakeCode Extensions. Maqueen Plus V2 Library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20



1.Click Setting ->Extensions



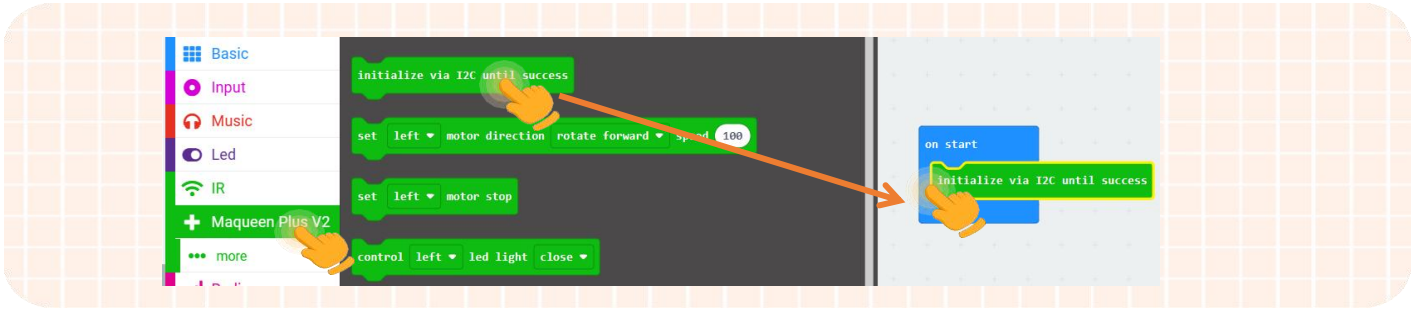
2.Input https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20 into the search bar to search and add Maqueen Plus V2 library.

3.Click DFRobot_MaqueenPlus_V2.0

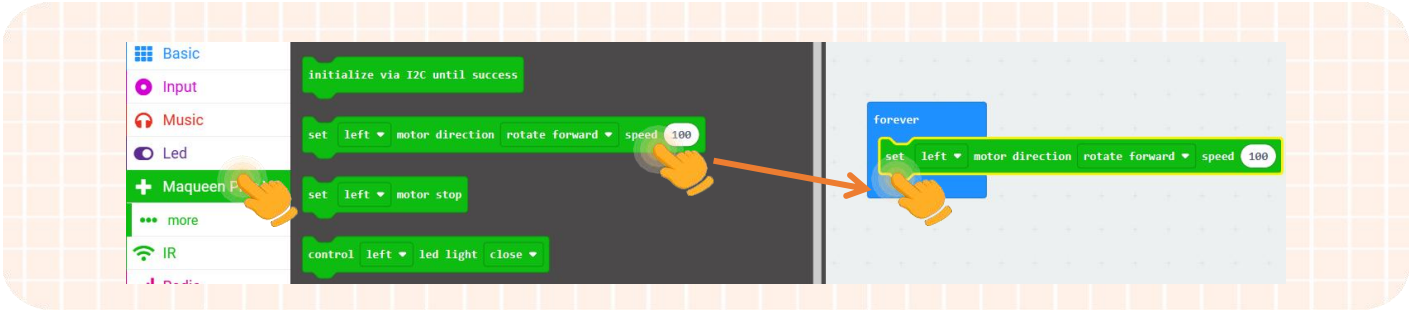
2. When the Maqueen Plus V2 library is loaded successfully, there will be an icon "**+ Maqueen Plus V2**" appearing in the command block section. Click the icon then you will see all the related blocks. Refer to appendix 1 to check the detailed description of these blocks.

Step 3 Programming

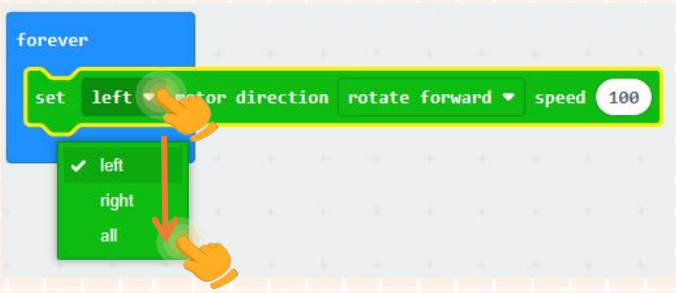
1. Embed the "initialize Maqueen Plus V2" block into the "on start" block.



2. Embed the motor control block into the "forever" block.



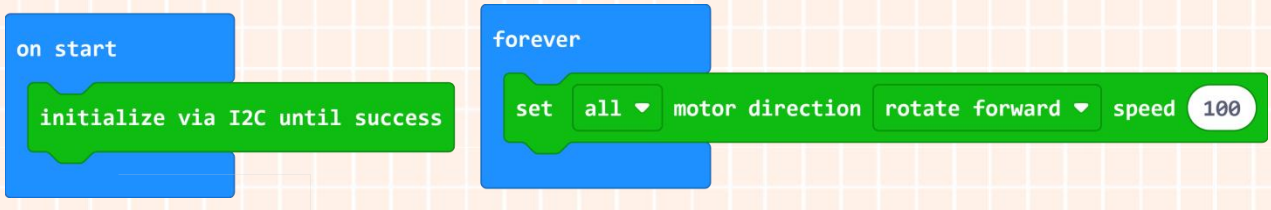
3. Change the "left" to "all" in the motor control block to let Maqueen Plus V2's two wheels move at the speed of 100.



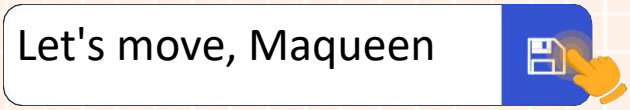
Knowledge Expansion

Most blocks can be used repeatedly, and the block with " ▾ " has multiple options for users to choose from. Besides, you can change the number in the " 200 " by typing or dragging the slider.

4. The complete program is shown below.



5. Save and name the project as "Let's move, Maqueen".

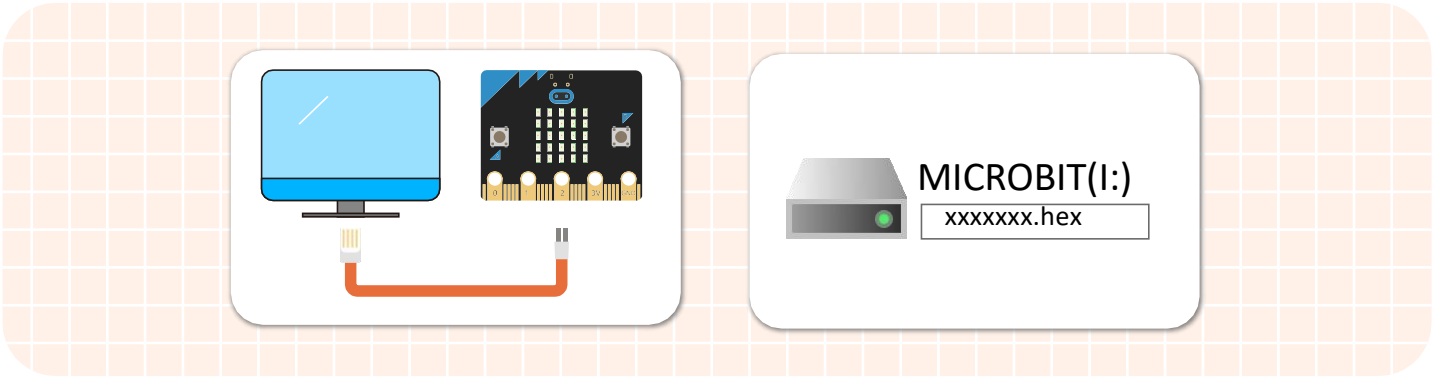


Knowledge Expansion

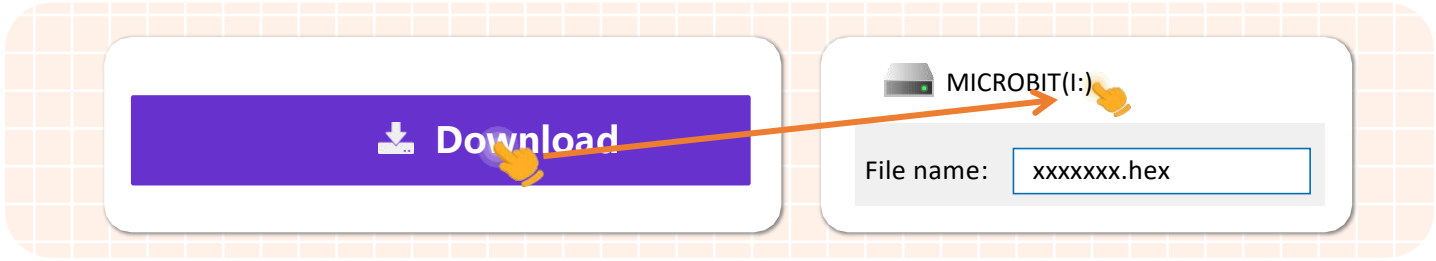
Name your project and click save.
Make it a habit to name your project!

Step 4 Download a Program

1. Connect to a computer: Connect the micro:bit to your computer with a USB cable before downloading. There will be a hard disk named "micro:bit" appearing in the computer when the connection is successful.



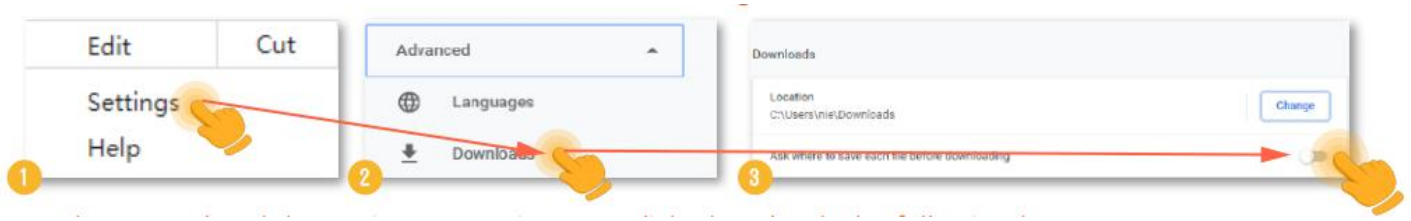
2. Download the program: Click to download your project into the micro:bit hard disk.



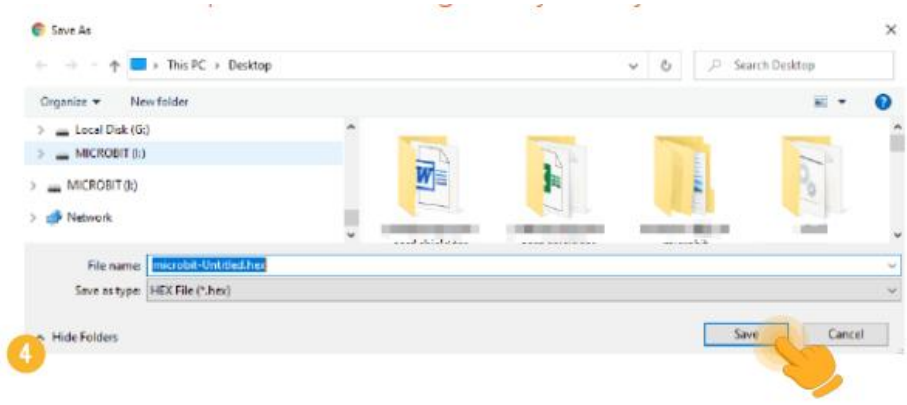
Knowledge Expansion

Note: there will be no box popping out in the Google browser since your files are directly downloaded into the default download folder. You have to change it in the settings of Google browser.

Step1: enter the setting interface in the Google browser, scroll down to find the download setting part, and enable "Ask where to save each file before downloading".

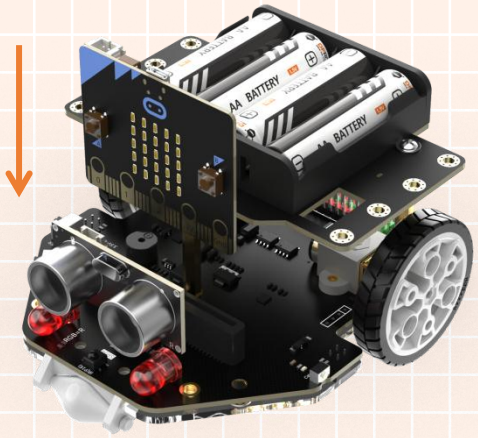


After completing the setting, every time when you click "download" the following box pops out.



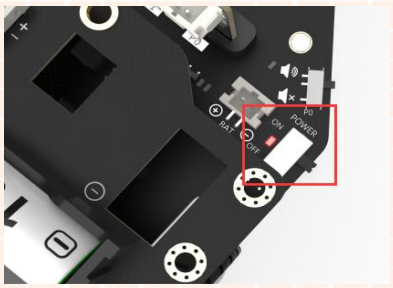
When you write data, the power indicator on the back of micro:bit board keeps flashing. Please do not disconnect the USB during downloading.

3. Install the micro:bit board: After downloading the program, unplug the USB cable and plug the micro:bit board into Maqueen Plus V2.



Step 5 Effect Display

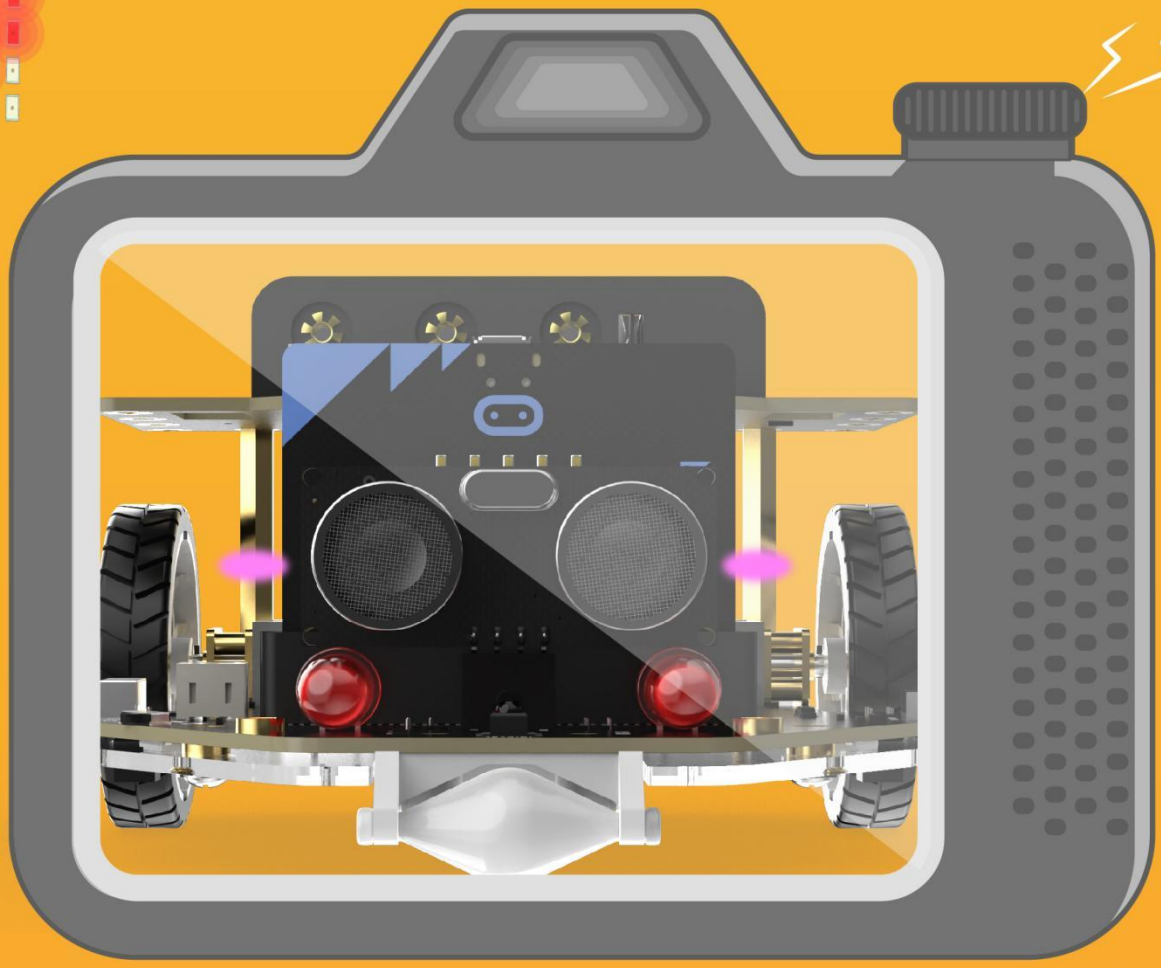
When you completed all the above steps, connect the power cable of the AA battery panel to the BAT of Maqueen Plus V2 and turn on the power switch, then Maqueen Plus V2 starts running!



Think & Explore

We have learned the program to make Maqueen Plus V2 move forward, but in our daily life, a car is also able to move backward, do you know how to realize that on Maqueen Plus V2? Can you program Maqueen Plus V2 to let it go backward at the speed of 100? Give it a try!

Tip: Just do a little change to the motor control block!



Chapter 3

Walking Emoji

Emojis are now considered to be a large part of popular culture these days. Maqueen Plus V2 also has his emojis. In this chapter, let's control Maqueen Plus V2 to walk along a circle while displaying emojis on its LED screen.

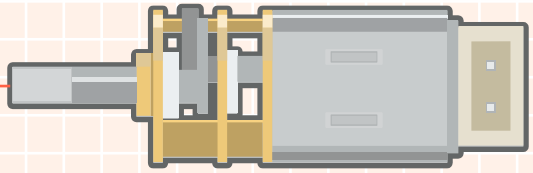
Goal

1. Learn the differential steering principle
2. Get to know the function of the "pause" block

Electronic Component

Motor Brief

Motor



Motors can be used to drive Maqueen Plus V2 to move left, right, forward, and backward.

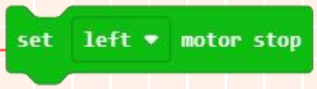
Command Learning

Block Brief

Show Icon

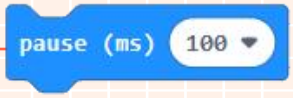
Display the selected icon on the micro:bit LED matrix

Motor Controlling



Make Maqueen Plus V2 stop

Pause



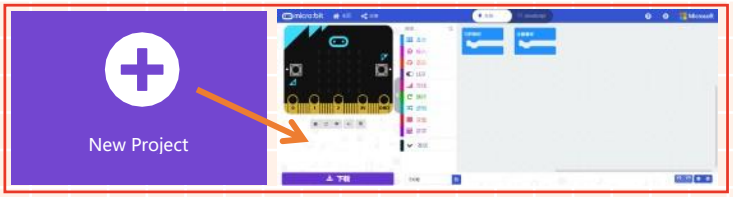
Pause for the specified time in millisecond

Hands-on Practice

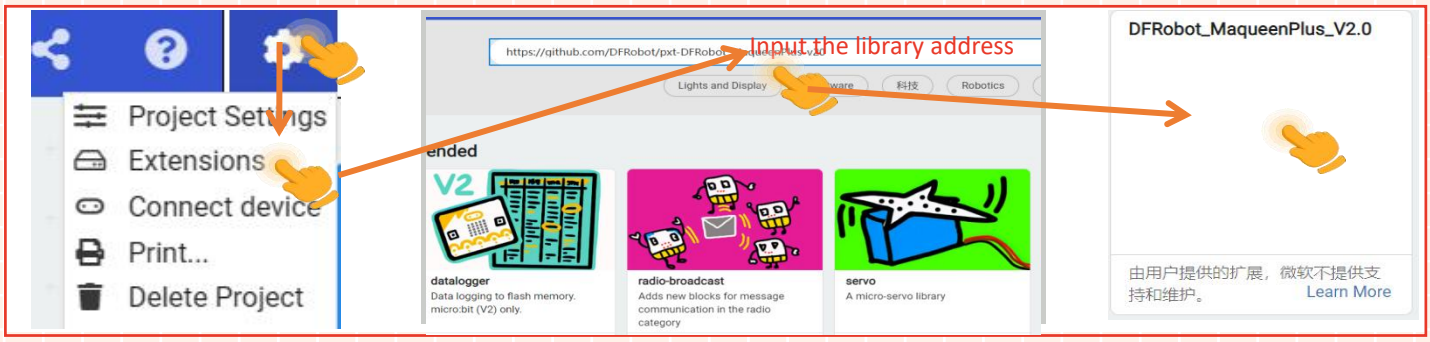
Step 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



2. Enter programming interface

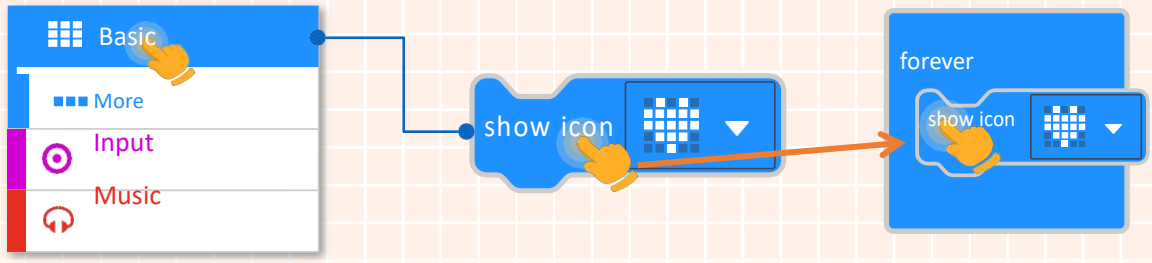


3. Add the extension library

Step 2 Programming

1. Display emojis

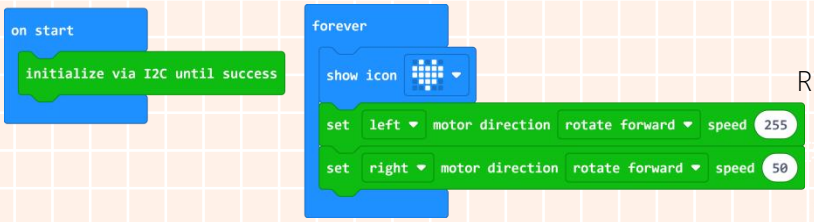
Drag the "show icon" block into a "forever" block, then a heart pattern will be displayed on the micro:bit LED matrix.

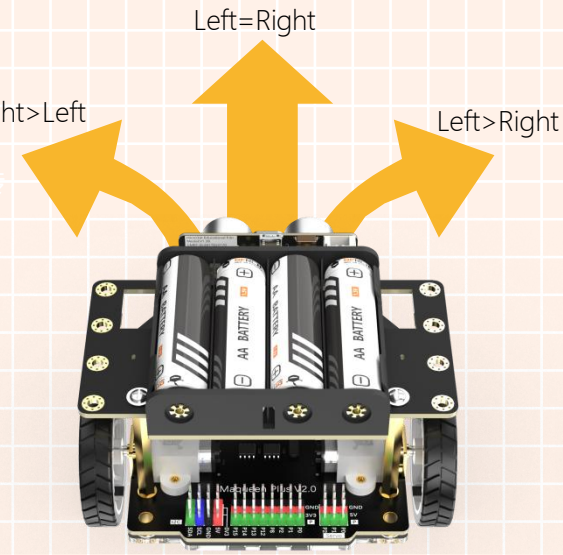


Note: click the drop-down arrow to select other patterns.

2. Maqueen Plus V2 moves clockwise

Program Maqueen Plus V2 to make it move clockwise along a circle. According to the differential steering principle, make the left motor move forward at the speed of 255, and the right motor move forward at 50.





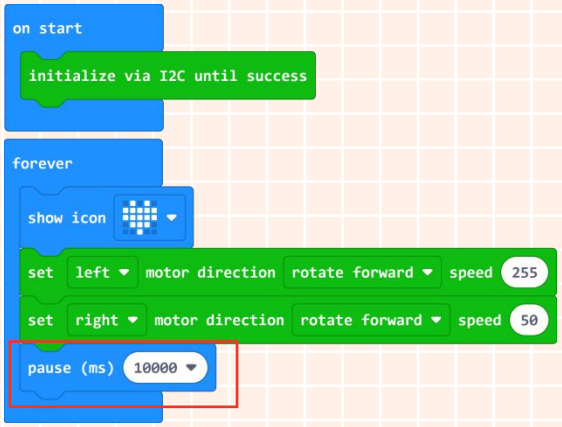
Knowledge Expansion

Differential Steering Principle:

When the speed and direction of the left and right wheels are the same, the robot car will move forward or backward. If the two wheels rotate in the same direction at different speeds, it will turn left or right.

3. Maqueen Plus V2 moves along a circle

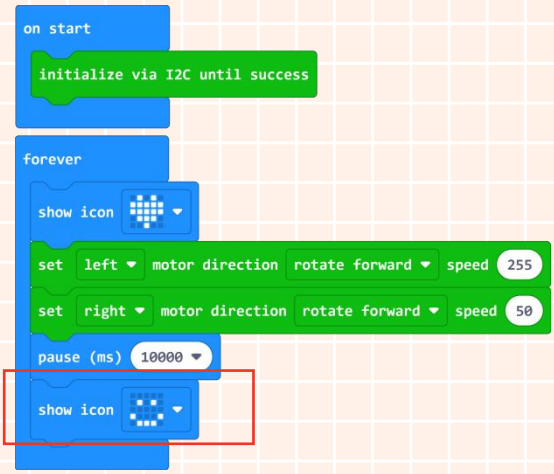
Let Maqueen Plus V2 move along a circle. Set the pause time via the pause module to make it move along a perfect circle.



Knowledge Expansion

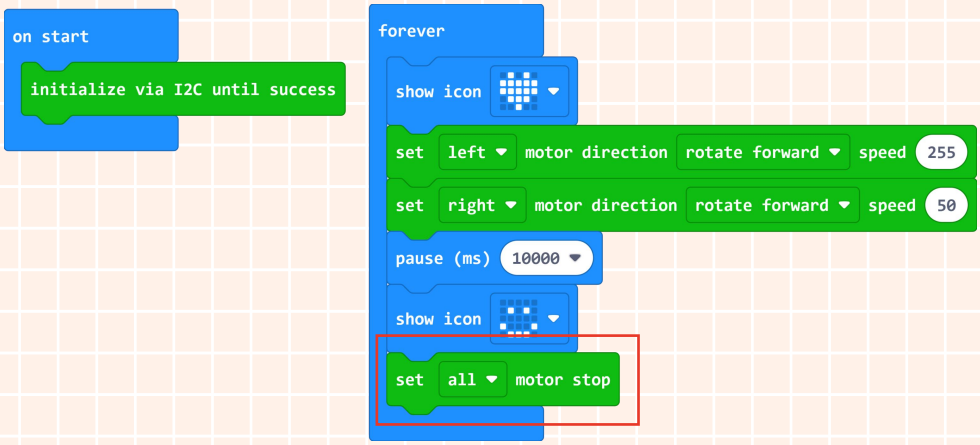
The length of time it takes for Maqueen Plus V2 to drive a complete circle is related to factors such as friction on the ground, battery power and so on. So you may have to do some adjustments according to the actual situation!

When Maqueen Plus V2 has walked a complete circle, display a smiley face on micro:bit LED matrix.

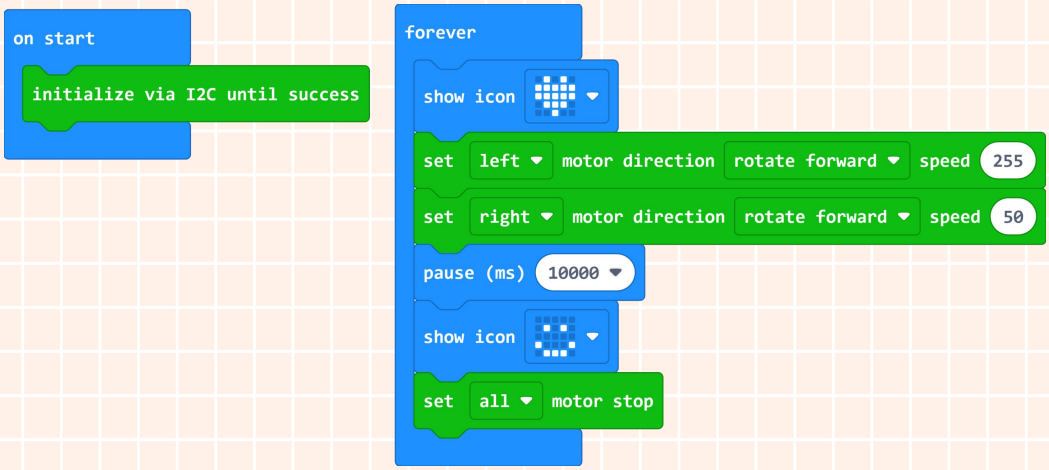


4. Maqueen Plus V2 stops

Maqueen Plus V2 will move along a circle repeatedly. Now we can use another motor control block to make it stop. As shown below, change the "left" to "all", then both motors will stop rotating.



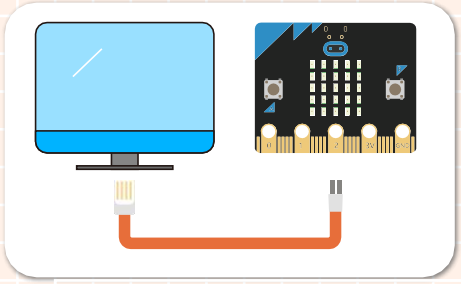
5. Complete program



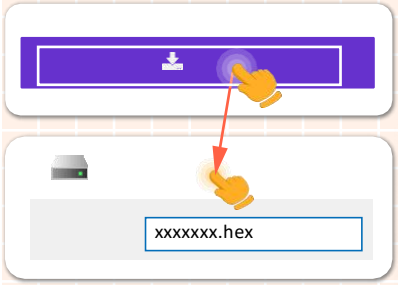
6. Name your project as "Walking Emoji" and save it.

Step 3 Download Program

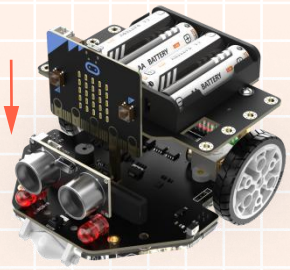
1. Connect to a computer: Connect the micro:bit to your computer with a USB cable before downloading.There will be a hard disk named "micro:bit" appearing in the computer when the connection is successful.
2. Download the program: Click "Download" to download your project into the micro:bit hard disk.
3. Plug in the micro:bit board: After downloading the program, unplug the USB cable and plug the micro:bit board into Maqueen Plus V2.



1. Connect to computer



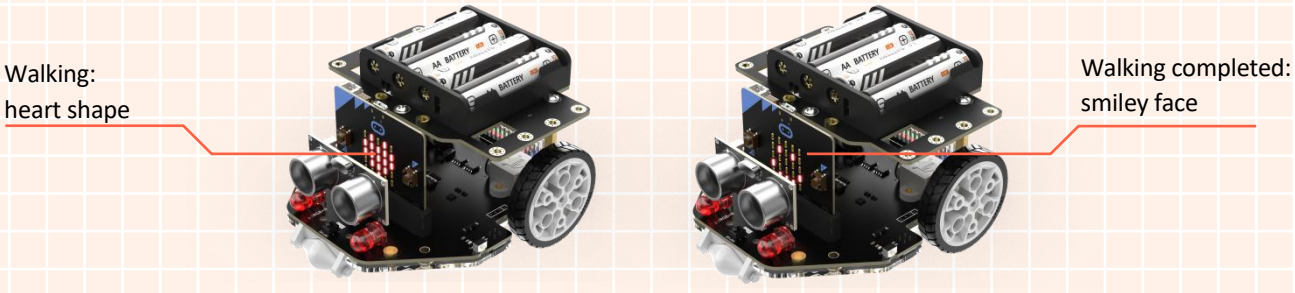
2. Download program



3. Plug in micro:bit

Step 4 Effect Display

Turn on the power switch, then Maqueen Plus V2 will start to run along a circle while showing a heart pattern on the micro:bit LED matrix. When he stops, a smiley pattern will be displayed on the screen.



Think & Explore

How is the movement state when the two motors are rotating at different speeds and directions? Program Maqueen Plus V2 to explore, and complete the form below. Tip: Just need to change the direction and speed of the motor control block.

Group	Left Motor		Right Motor		Movement
	Speed	Direction	Speed	Direction	
1	200	Forward	200	Forward	Move forward
2	200	Forward	50	Forward	
3	50	Forward	200	Forward	
4	200	Backward	200	Backward	
5	200	Backward	200	Forward	

Chapter 4

City Defender - A Police Car

There are so many city defender heroes in movies, and have you ever considered being one of them? Now let's turn this Maqueen Plus V2 into a city defender - a police car to make your dream come true!

Goal

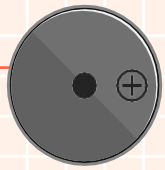
1. Learn how to use the buzzer module
2. Learn how to use the RGB light module

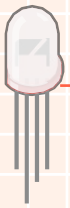
Electronic Component

Buzzer and RGB LED Brief

Buzzer

Work as an output module, can be used to play sounds.





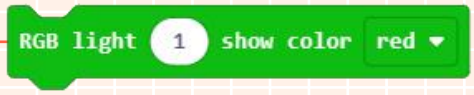
RGB Light

Display various colorful lighting effects

Command Learning

Block Brief

Control RGB LED



Control the color of the RGB LED

Play sound

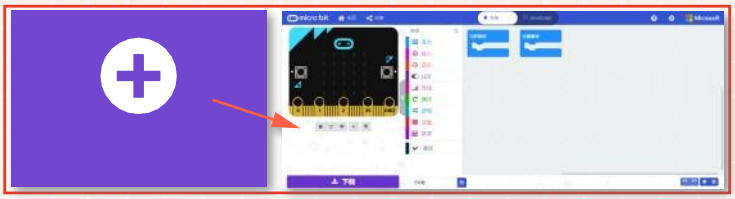
Play MakeCode built-in music

Hands-on Practice

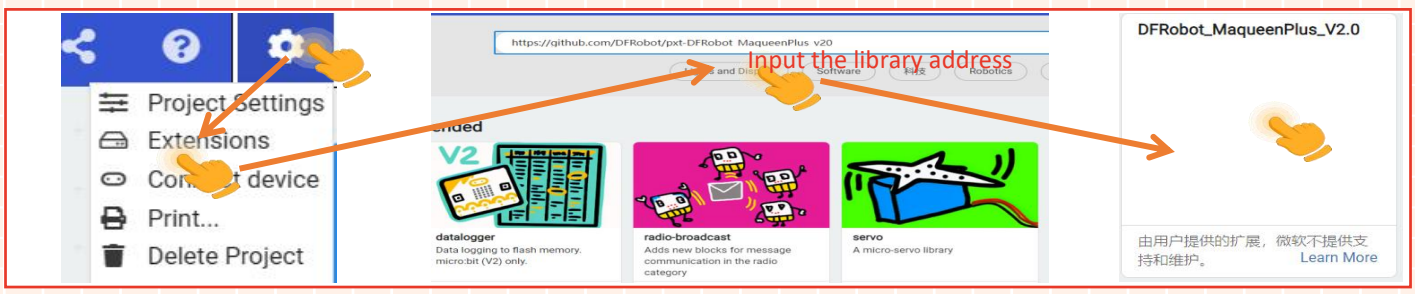
Step 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



2. Enter programming interface

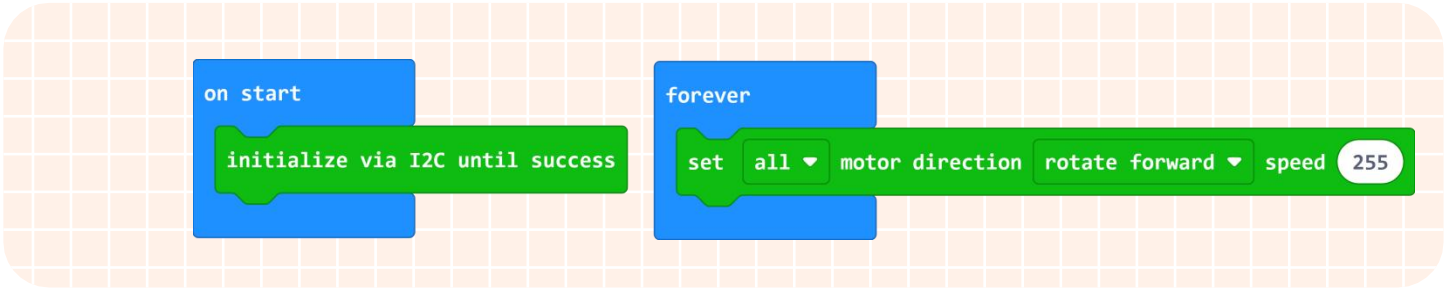


3. Add the extension library

Step 2 Programming

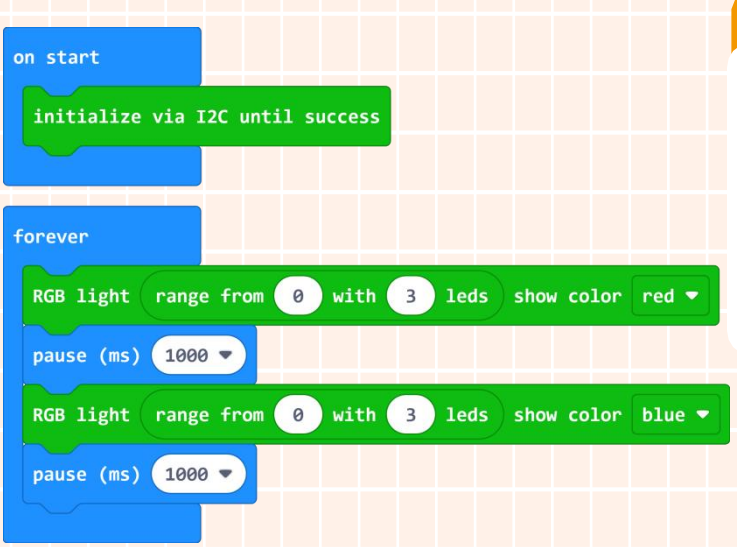
1. Program the car to make it drive

Both motors move forward at the speed of 255.



2. Program the lighting effect

Since the lights on a police car emit red light and blue light alternatively, the next step here is to exchange the lighting color every 1s.



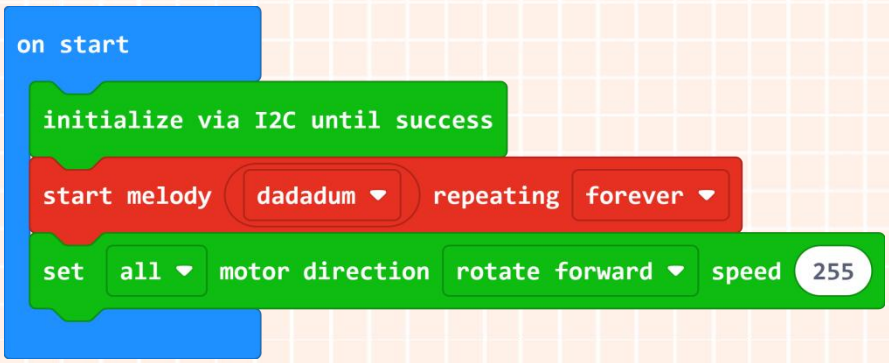
Knowledge Expansion

Why are the lights on police car red and blue?

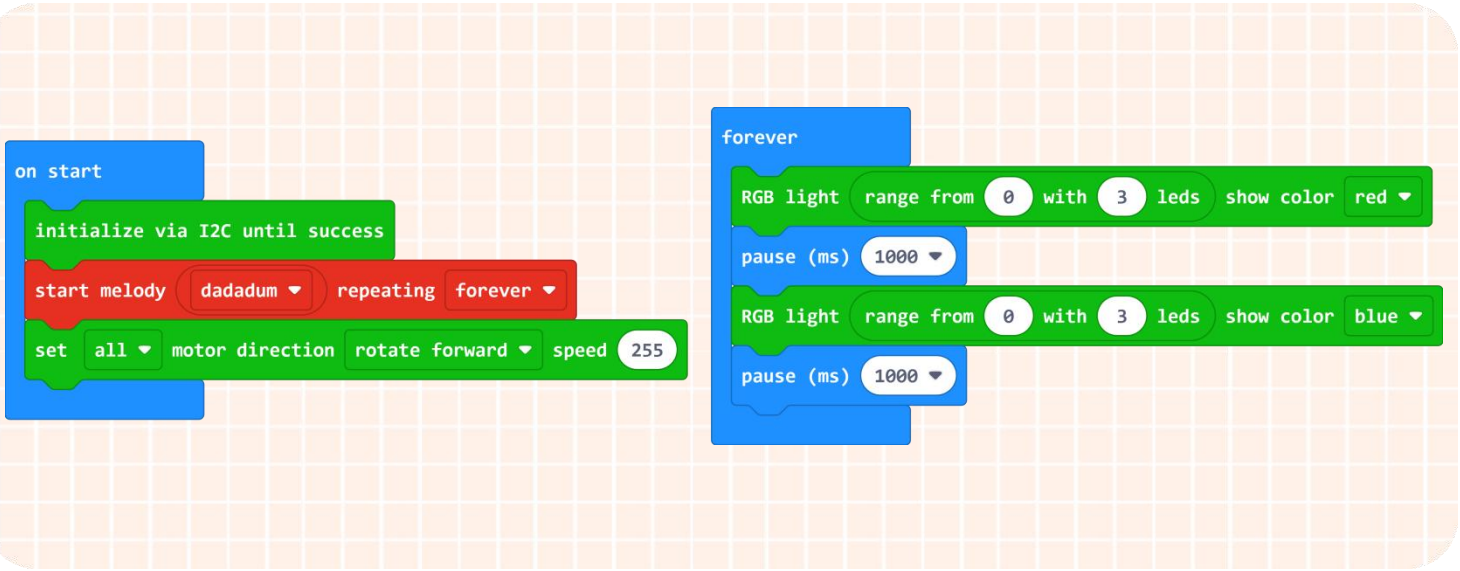
1. Alert other drivers of its presence, so that they can maneuver out the way.
2. The color red is associated with stop and warning, but most tail lights are also red, so blue lights really stand out and help to alert others in these situations.

3. Program the siren

Play the built-in sound "dadadum" repeatedly to simulate the sound of a siren.



4. The whole program is shown below.



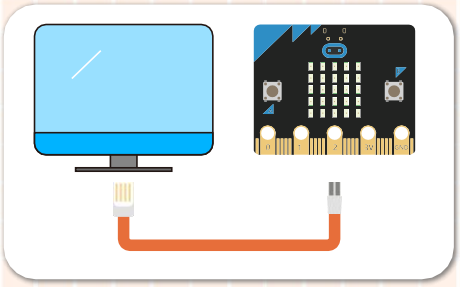
5. Name your project as "City Defender-A Police Car" and save it.

Step 3 Download Program

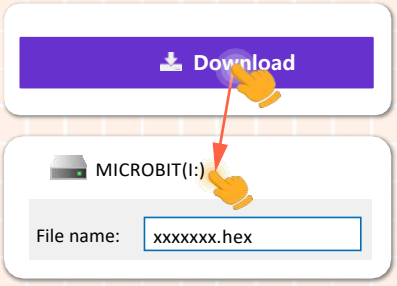
1. Connect to a computer: Connect the micro:bit to your computer with a USB cable before downloading. There will be a hard disk named "micro:bit" appearing in the computer when the connection is successful.

2. Download the program: Click "Download" to download your project into the micro:bit hard disk.

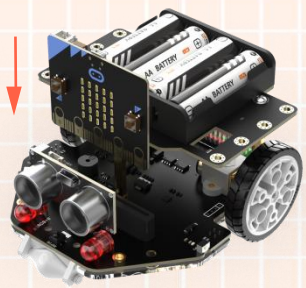
3. Plug in the micro:bit board: After downloading the program, unplug the USB cable and plug the micro:bit board into Maqueen Plus V2.



1. Connect to computer



2. Download program



3. Plug in micro:bit

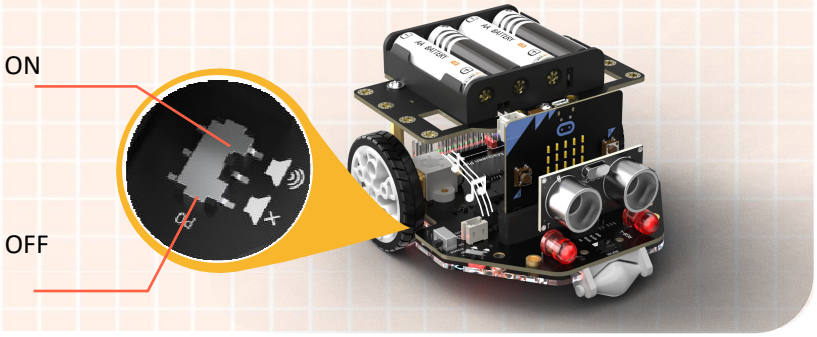
Step 4 Effect Display

Turn on the power switch, then Maqueen Plus V2 moves forward with siren wailing and lights flashing, just like a police car.

Note:

1. There is a buzzer switch on the right side of Maqueen Plus V2, and you need to turn it on when using the buzzer.

2. The buzzer and P0 are shared, so when you need to use P0 as input/output port, turn off the buzzer.



Think & Explore

We have learned how to play the built-in music in MakeCode. Do you want to make a piece of your music? Try it with Maqueen Plus V2. The block shown below is used to play notes.

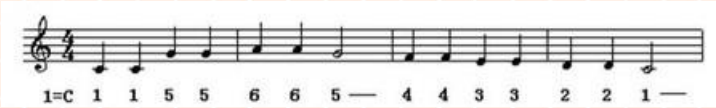
Click here to display keyboard, then select note.

Select beat for the note.

The letter "C" represents the note "do", so middle C is middle do. "1" is the beat of the note, the basic time unit, also specified by the time signature. The relation between note and notation is shown below:

C	D		E	F	G	A	B
1	2		3	4	5	6	7

Create a piece of music according to the numbered musical notation below!



Chapter 5

Light Sensing Robot

Without light, there would be no sight. We are able to see because light from an object can move through space and reach our eyes. But human eyes are very sensitive to light, both too strong and weak lights are harmful to our eyes. How do we know the changes in light brightness? Maqueen Plus V2 can help us achieve that.

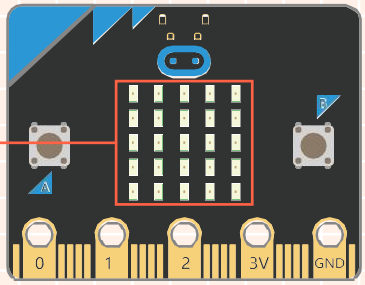
Goal

1. Learn the light sensor
2. Learn how to use the "show number" block

Electronic Component

Light Sensor Brief

Light sensor



The micro:bit LEDs can be used to estimate the amount of ambient light, and output the light level as electric signal.

Command Learning

Block Brief

Read light level

Reads the light level applied to the LED screen in a range from 0 to 255

Show number

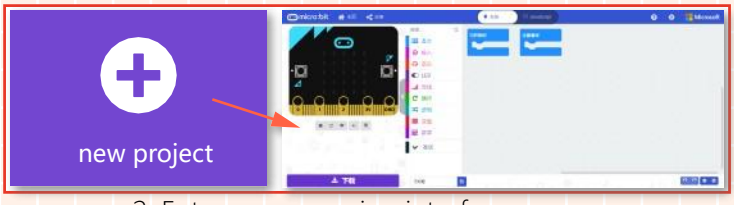
Scroll a number the micro:bit LED matrix

Hands-on Practice

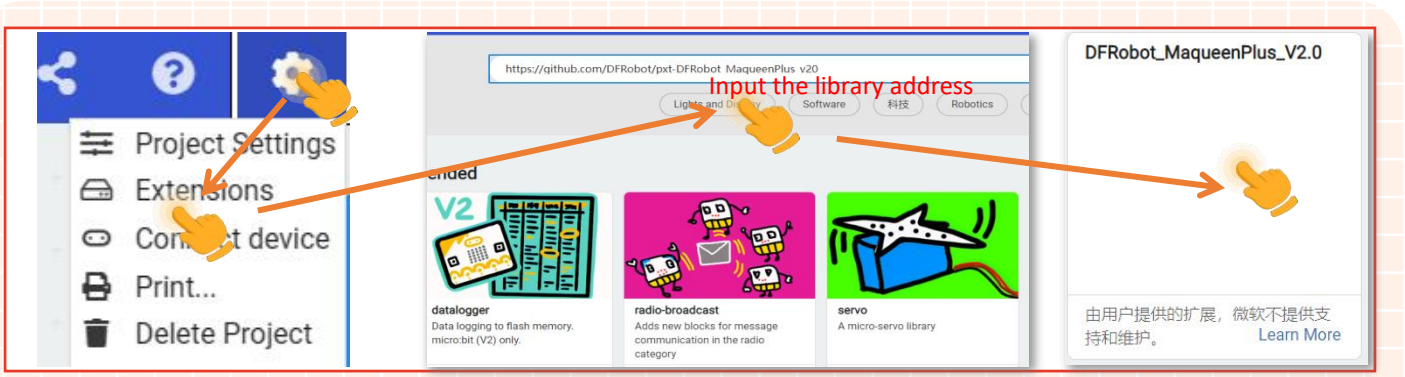
STEP 1 Create a new project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



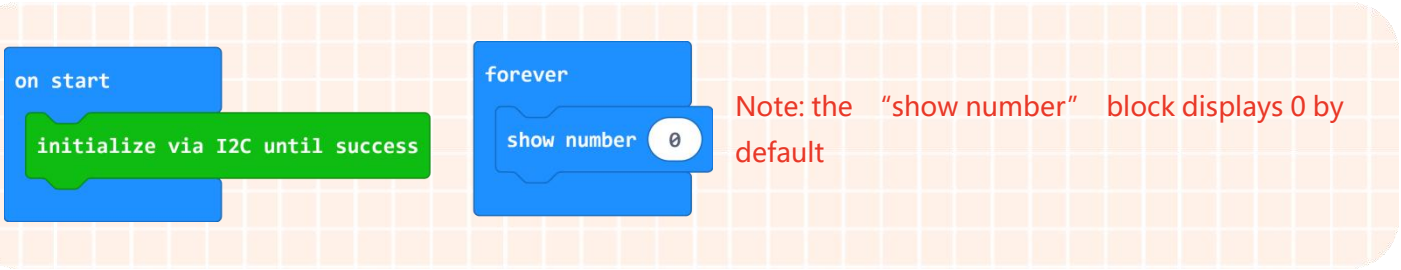
2. Enter programming interface



3. Add the extension library

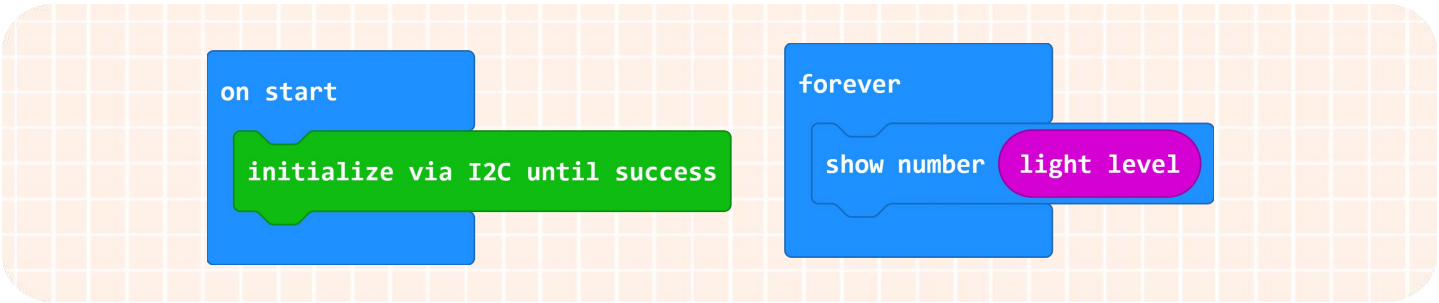
STEP 2 Programming

1. Put the "show number" block into the "forever" block, then the robot eyes will read the light level constantly.

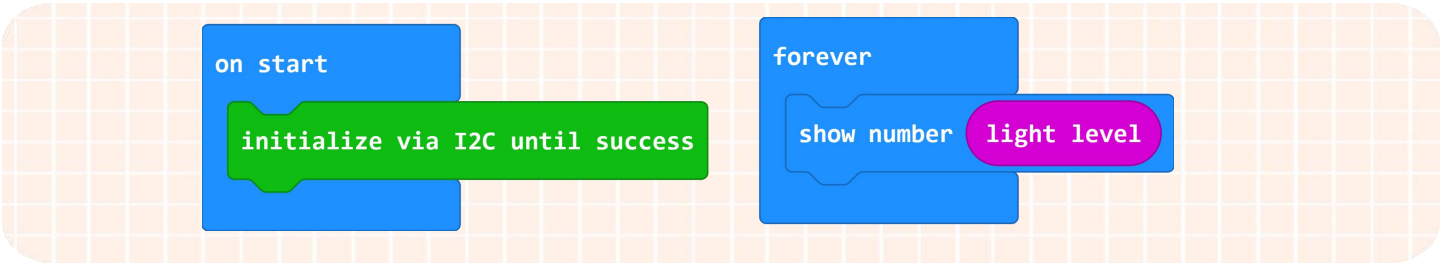


Note: the "show number" block displays 0 by default

2. To display the ambient light level on the micro:bit LED screen in real-time, we have to put the "light level" block into the "show number" block.



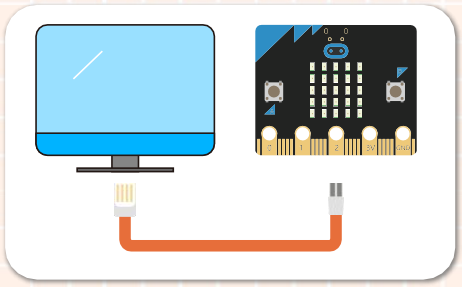
3. The complete program is shown below.



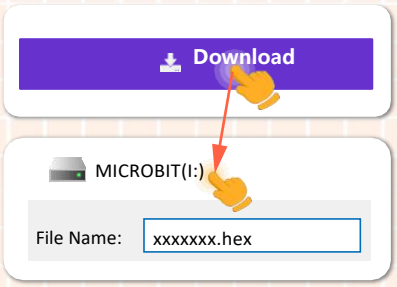
4. Name your project as "Light sensing robot" and save it.

STEP 3 Download Program

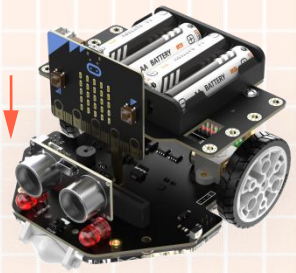
1. Connect to PC: connect the micro:bit to your computer with a USB cable before downloading. There will be a hard disk named micro:bit appearing in the computer when the connection is successful.
2. Download the program: download your project into the micro:bit hard disk.
3. Plug in the micro:bit board: after downloading the program, plug the micro:bit board into Maqueen Plus V2.



1. Connect to PC



2. Download program



3. Plug in micro:bit

STEP 4 Effect Display

Turn on the power switch of Maqueen Plus V2, the LED screen will constantly display the current light level! The following are the light levels measured at the office and photo studio.

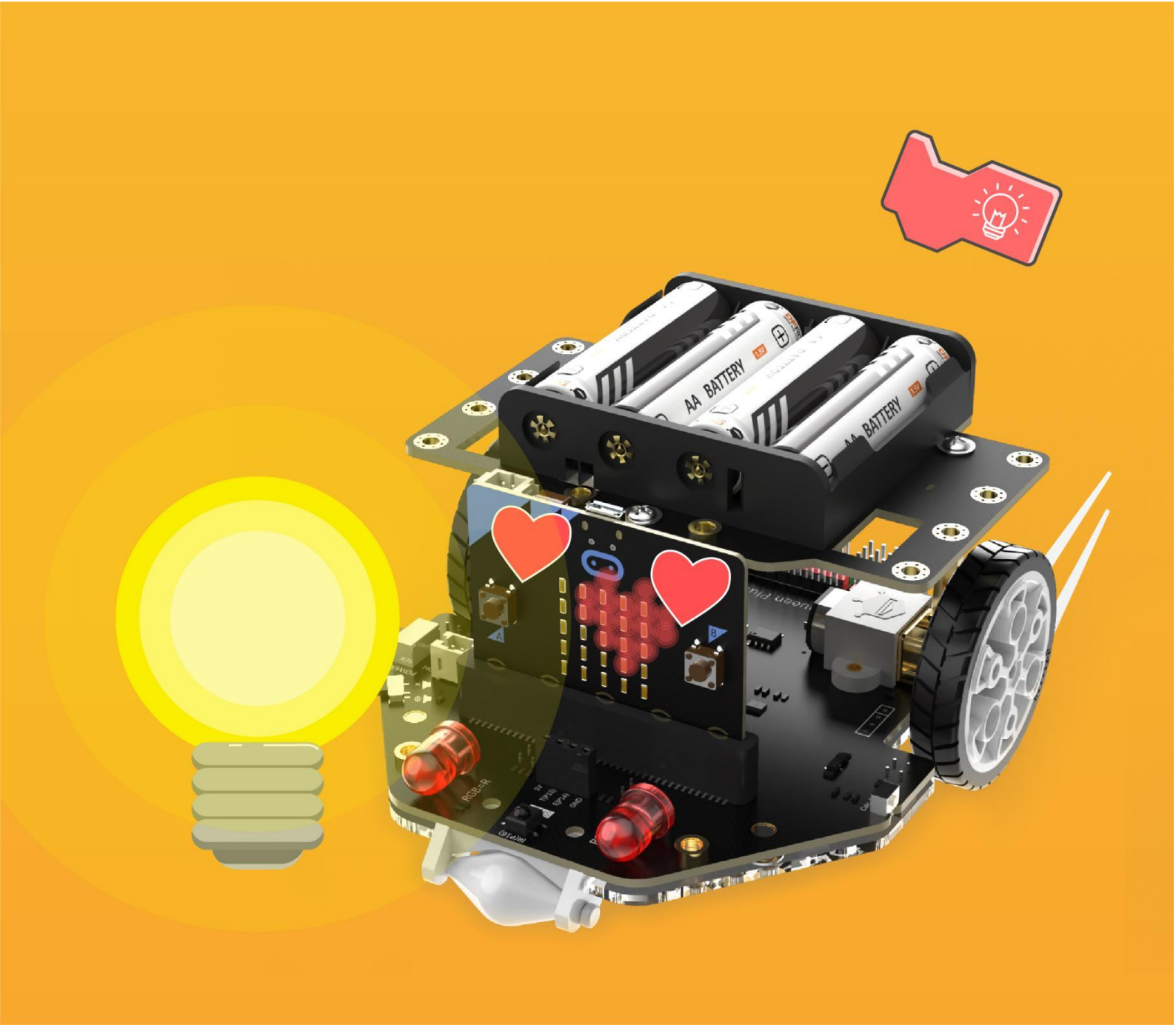
Place	Light Level
Office	125
Photo studio	255

Note: data in the experiments are for reference only.

Think & Explore

Light level varies from places, let our Maqueen Plus explore! After that, please think if the light level of each place is reasonably designed.

Place	Light Level
Kitchen	255
living-room	
Bedroom	
Study room	



Chapter 6

Moth Robot

You must have seen that at summer night, plenty of moths fly around the streetlight, flame, and any places with bright light. Why are moths attracted to flame? One idea is that moths are able to find their way partly by using light as a compass. You know what, Maqueen Plus V2 can turn into a moth robot because it has a pair of light-sensitive eyes.

Goal

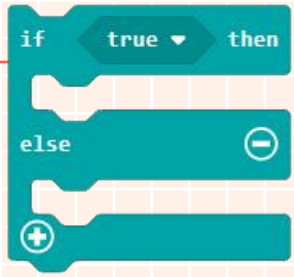
1. Learn condition block
2. Design a flowchart to help program

Command Learning

Block Brief

if...then..else

If a value is true, then do the first block of statements. Otherwise, do the second block of statements.



Comparison operator

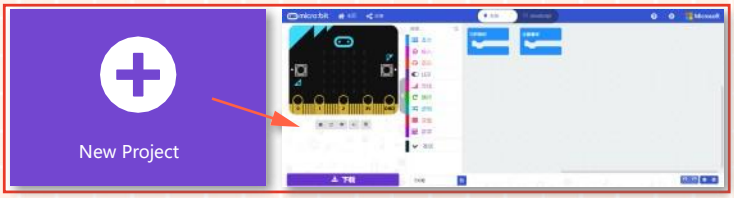
Return true if the first input is greater than the second input.

Hand-on Practice

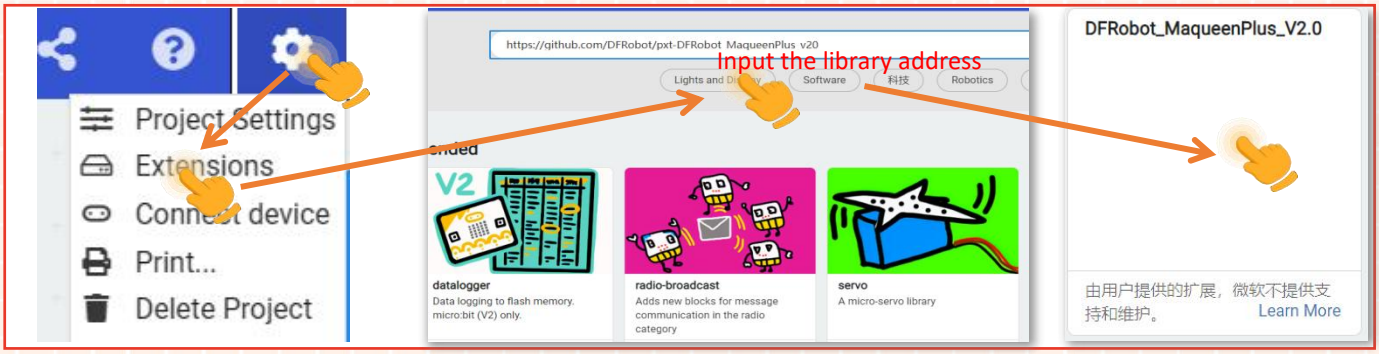
STEP 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



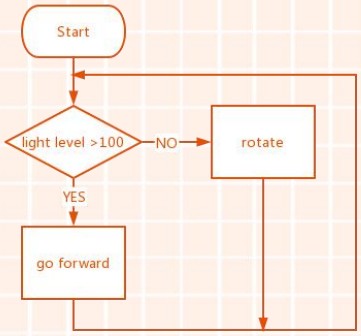
2. Enter programming interface



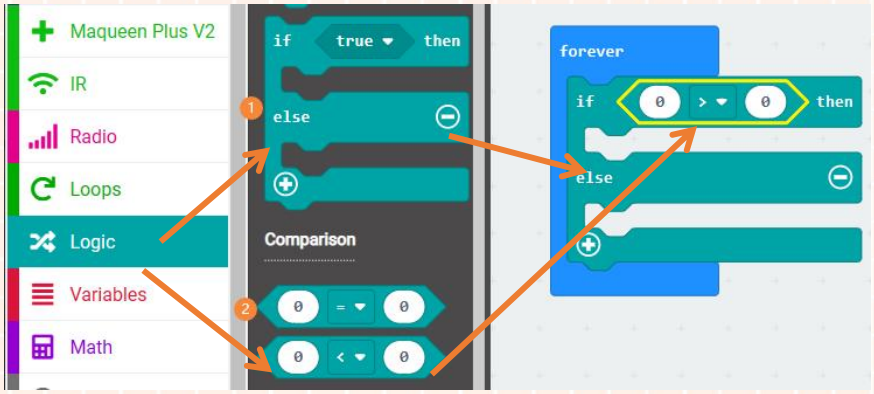
3. Add the extension library

STEP 2 Programming

1. When the light level is more than the given value (100 in the example), the moth robot moves towards the light source; when less than that value, the robot revolves around its center. Drawing a corresponding flowchart according to the above functions is gonna help us a lot with programming!



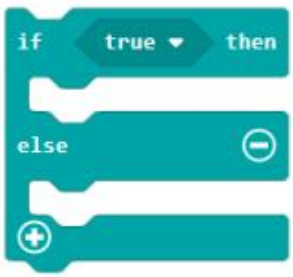
2. Put the "if...then...else" block inside the "forever" block, and embed the comparison operator block into it.



3. The key point of the whole program is the condition statement "light level >100". Different operations will be executed according to the result of the condition block.

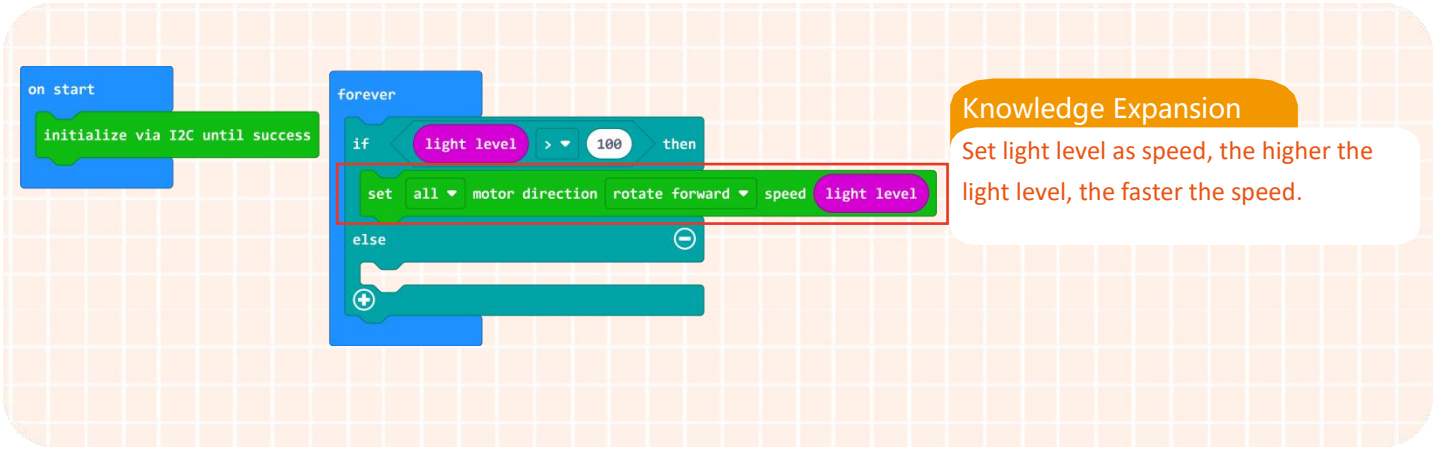
Knowledge Expansion

Click the " ⊕ " in the condition block to add a condition, click " ⊖ " to delete a condition

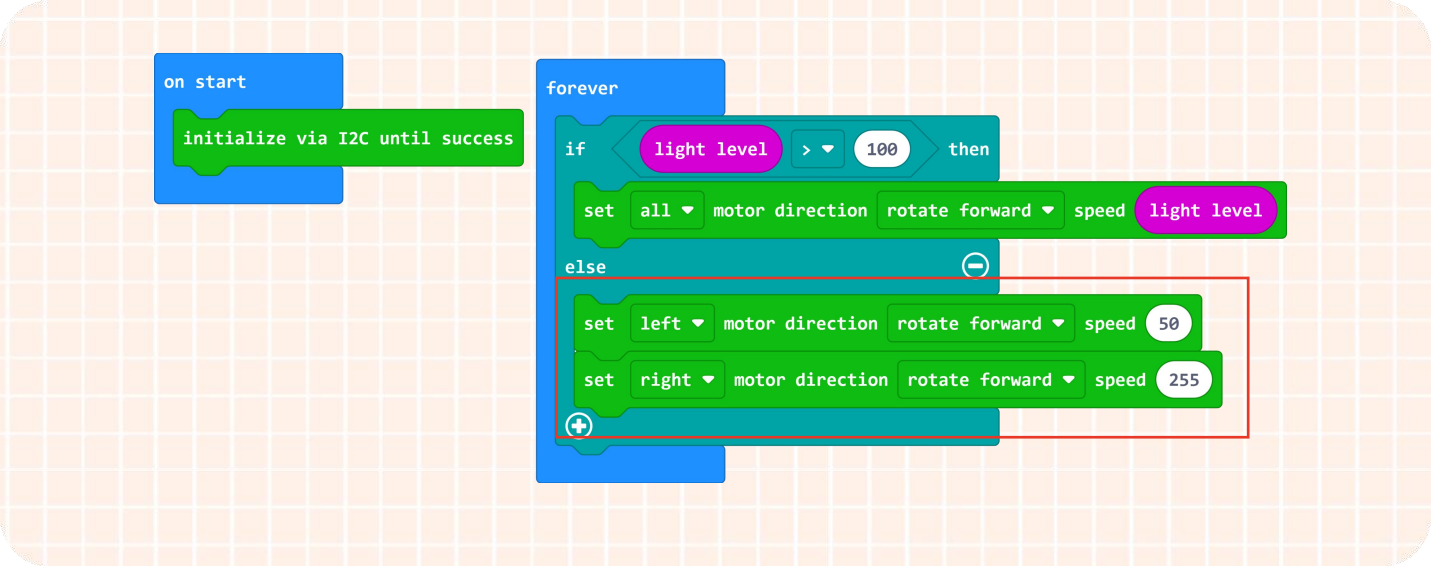


Click the" ▾ " in the comparison block to select different operators.

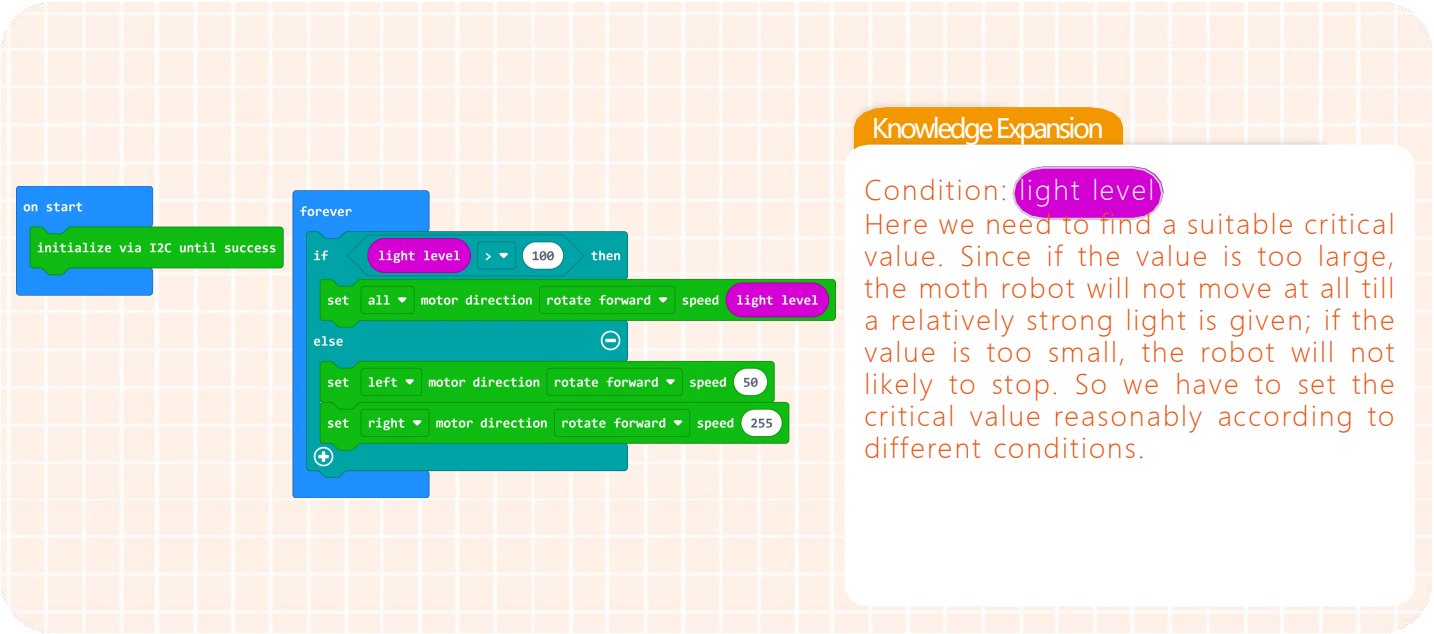
4. When the condition "Light level>100" is true, the robot car moves forward at the speed of the current light level.



5. When the condition "Light level>100" is false, the Maqueen Plus V2 rotates around its center.



6. The complete program is shown below.



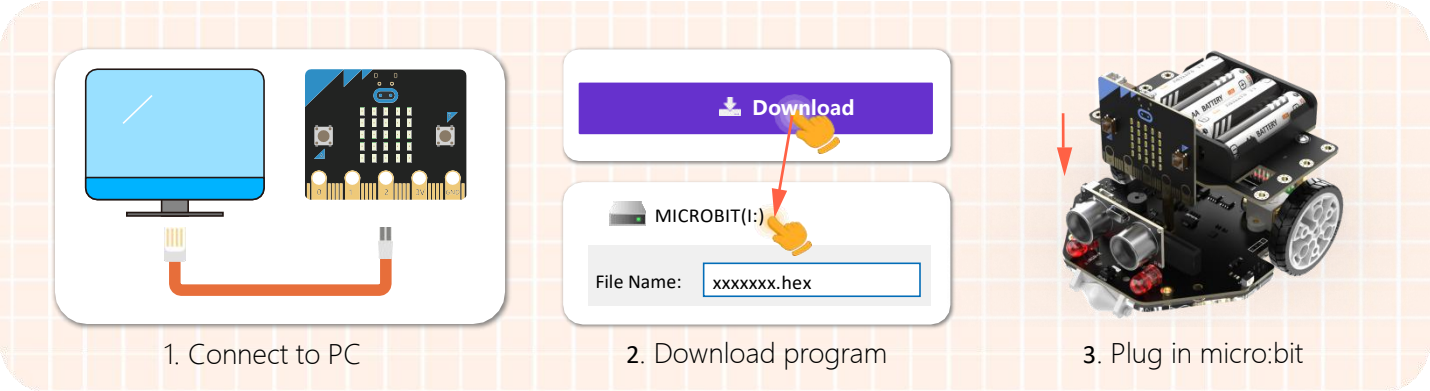
7. Name your project as "Moth robot" and save it.

STEP 3 Download Program

1.Connect to PC: connect the micro:bit to your computer with a USB cable before downloading. There will be a hard disk named micro:bit appearing in the computer when the connection is successful.

2.Download the program: download your project into the micro:bit hard disk.

3.Plug in the micro:bit board: after downloading the program, plug the micro:bit board into Maqueen Plus V2.



1. Connect to PC　　2. Download program　　3. Plug in micro:bit

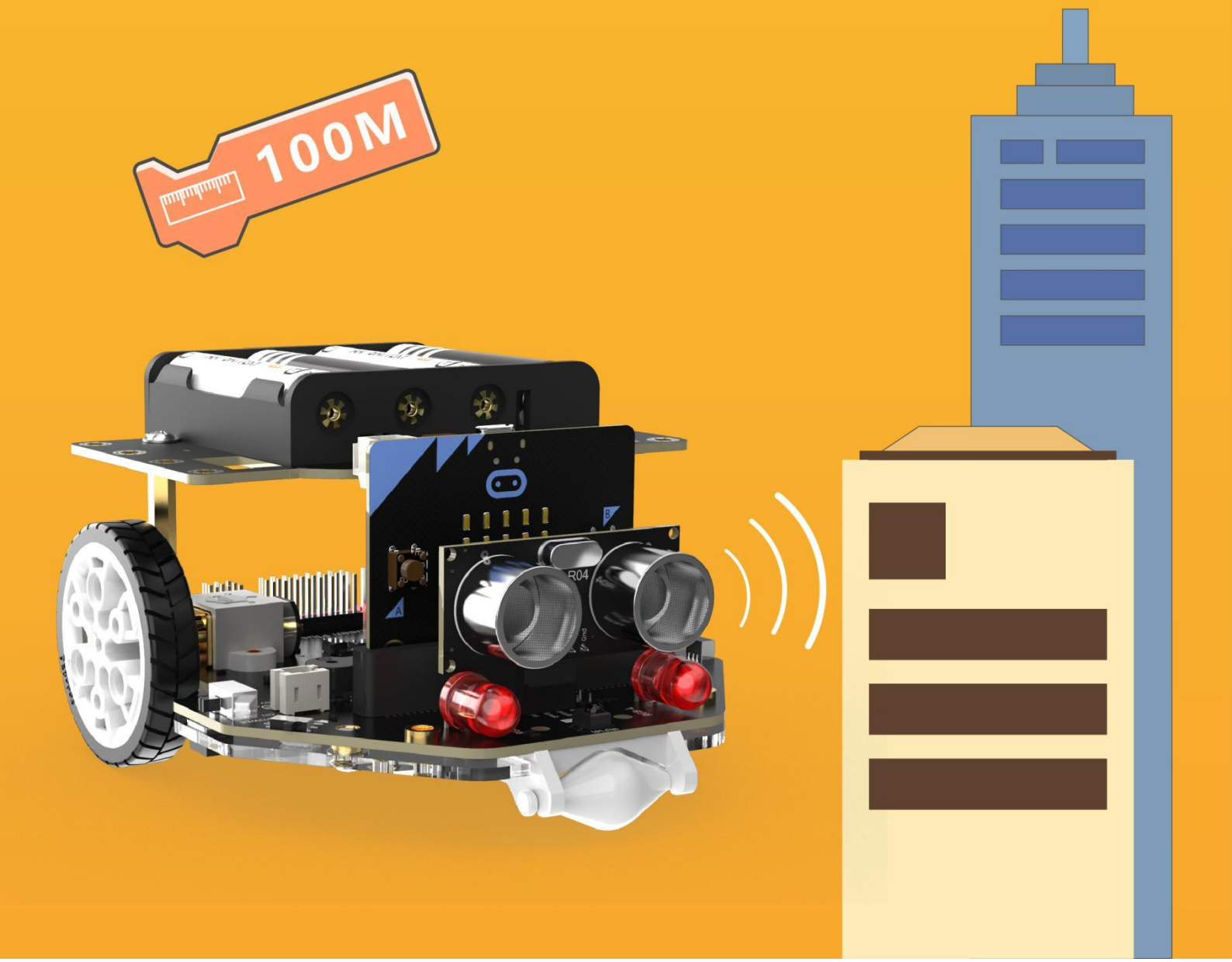
STEP 4 Effect Display

Turn on the power switch, then Maqueen Plus V2 will turn into a moth robot. When the light level is over 100, our moth robot moves towards the light, the brighter the light is, the faster Maqueen Plus V2 runs. When the light is less than 100, the robot will lose its way and rotate around. So funny, right? Come to play with this moth robot!

Think & Explore

Let’ s do a robot running competition! Use a flashlight to lead Maqueen Plus V2 to run forward, the one who uses the least time to finish the game will be the winner. Remember, do not cross the line. Invite your friends to join the game!

Tips: maintaining speed within a reasonable range holds the key to success.



Chapter 7

Little Ranging Expert

We have known that Maqueen Plus V2 is such a changeable robot with various functions, but more surprisingly, he can measure distance using his ultrasound eyes. With this buddy, you can say goodbye to your measuring tool.

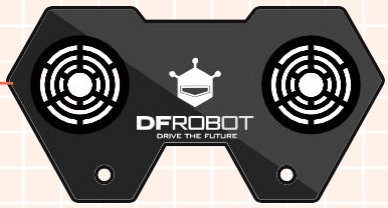
Goal

1. Get to know ultrasound
2. Learn the principle of ultrasound
3. Measure distance using ultrasound

Electronic Component

Ultrasound Brief

Ultrasonic sensor



The transmitter sends out ultrasound, and when hitting the object, the ultrasound reflects as echo and will be sensed by the receiver.

Command Learning

Block Brief

Read ultrasonic sensor

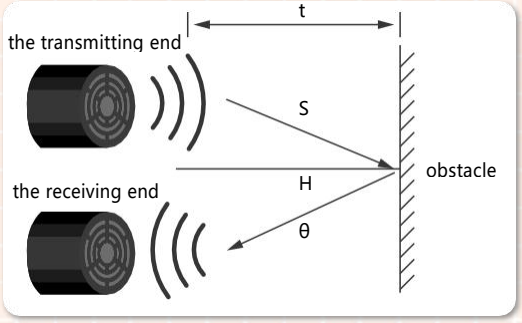
Read the distance value detected by the ultrasonic sensor, unit: cm. Preset the transmitter (TRIG) and receiver (ECHO) on the sensor.

What is ultrasound?

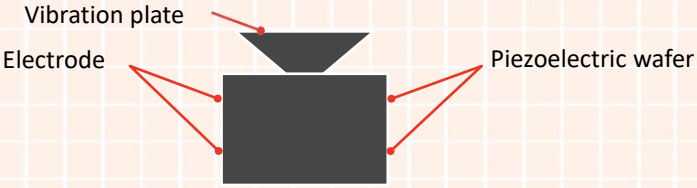
As we know, when vibrating, objects produce sound waves. Some of them can be heard by human ears, while others cannot. Scientists named the vibrating times per second as the sound frequency with its unit named as Hertz. Almost all human beings are able to hear the sound frequencies ranging from 20 to 20000Hertz. Sound frequencies out of that range are inaudible to humans. For those sound, the scientists name it as "Ultrasound".

How does an ultrasonic sensor measure distance?

Ultrasonic sensors work by sending out a sound wave at a frequency above the range of human hearing. The sensors determine the distance to a target by measuring time lapses between the sending and receiving of the ultrasonic pulse.



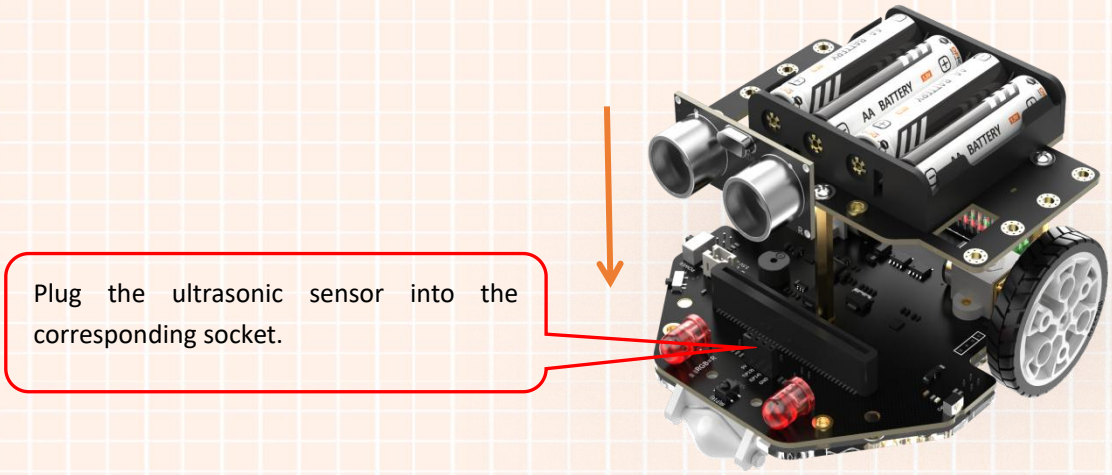
Ultrasonic sensors will convert the reflected sound into an electrical signal. A commonly used sensor consists of a piezoelectric wafer that can emit ultrasonic waves. The internal structure is shown below.



Hands-on Practice

Hardware Connection:

The pin connection is shown below:



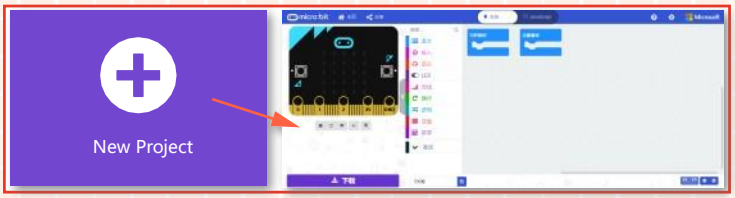
Step 1 Create a New Project

1.Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.

2.Click "new project" to enter the MakeCode programming interface.

3.Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



2. Enter programming interface

3. Add the extension library

Step 2 Programming

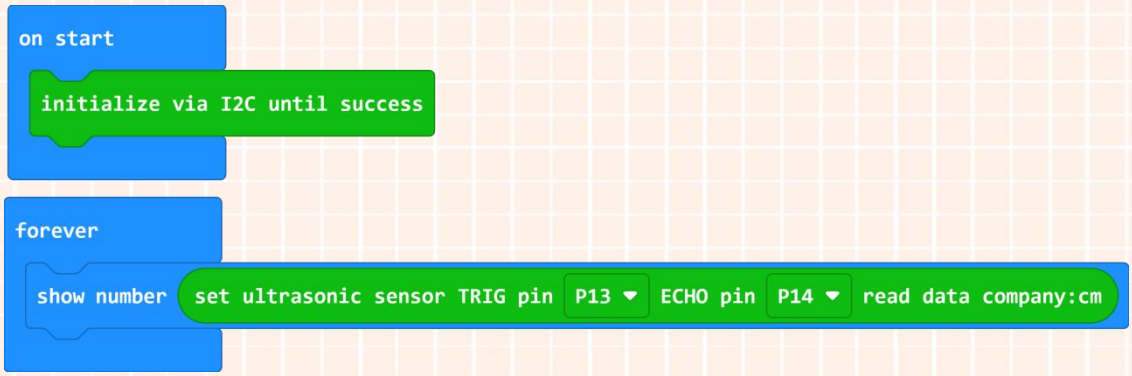
1. Place the "show number" block into the "forever" block, shown as below.

2. Put the distance value detected by the ultrasonic sensor inside the "show number" block to display the current distance on the micro:bit LED matrix in real time.

Knowledge Expansion

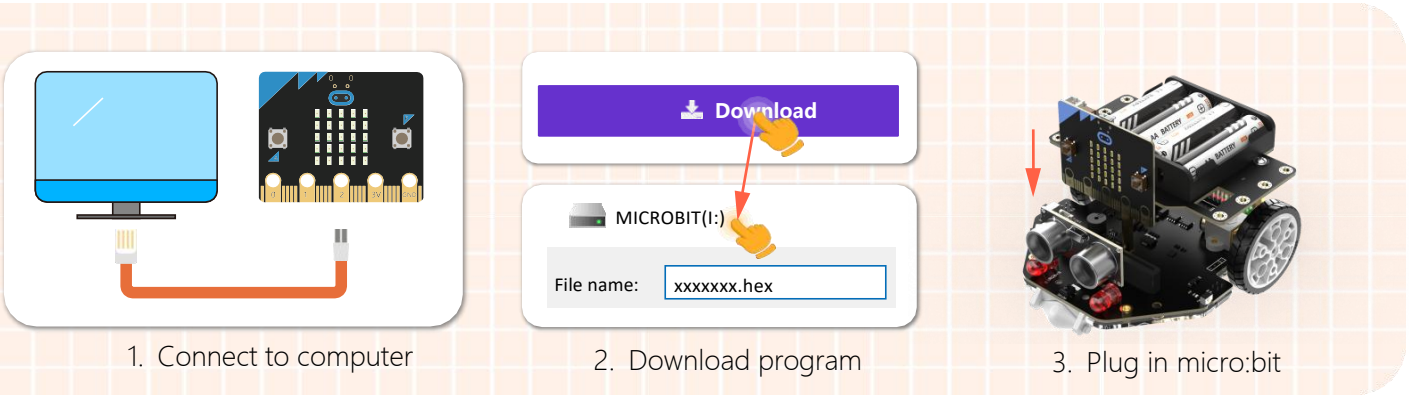
"T" is the transmitting end, corresponding to the "trig" on the ultrasonic sensor; "E" is the receiving end, corresponding to the "echo". On Maqueen Plus V2, trig is the P13, echo is the P14, so we need to set the pins in the ultrasonic block to P13, P14.

3. The entire program is shown below.



4. Name your project as "Little ranging expert" and save it.

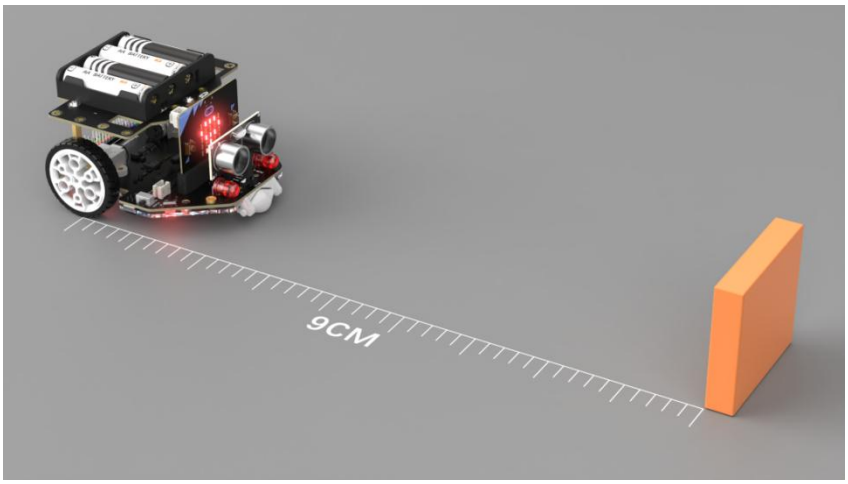
Step 3 Download Program



1. Connect to computer　　2. Download program　　3. Plug in micro:bit

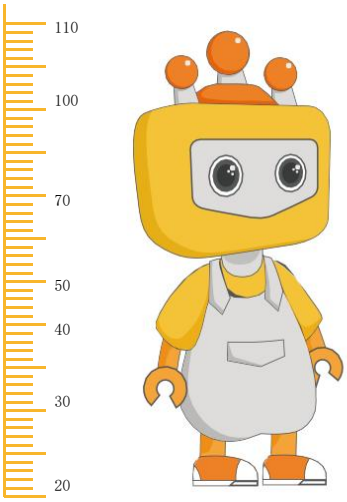
Step 4 Effect Display

Turn on the power switch, then we can use Maqueen Plus V2 to measure distance. The detected distance will be displayed on micro:bit.



Think & Explore

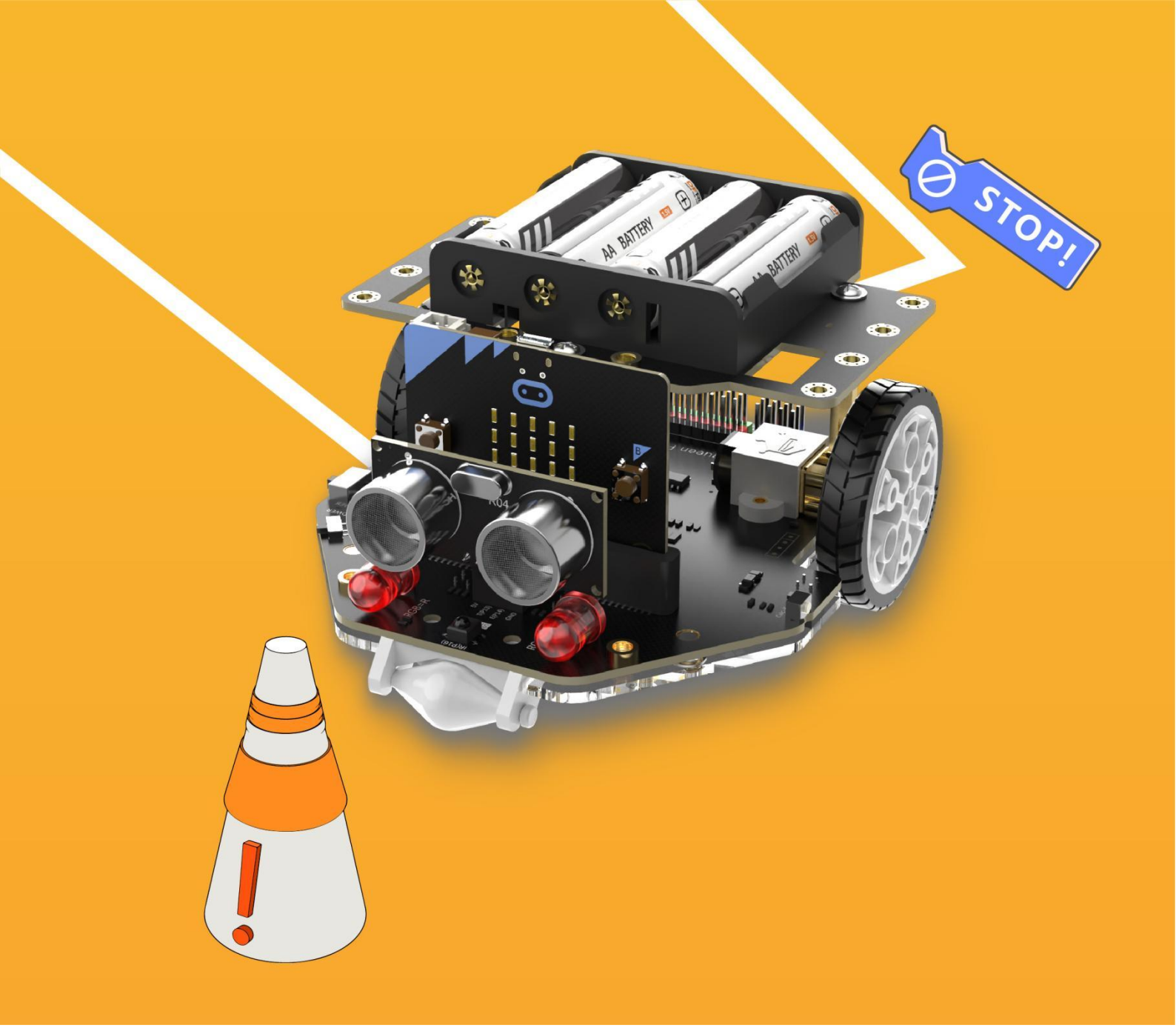
How do we measure the height of a person with an ultrasonic sensor?



To make the measurement more accurate, you have to:

1. Pay attention to the position and direction of the ultrasound sensor.

2. Calibrate the sensor within 10cm to prevent large error.



Chapter 8

Auto-braking Robot

The most frequent cause of vehicle accidents is the unexpected existence of barriers while driving. An automated braking system will assist and minimize such collisions, and ensure driver safety and comfort. Based on the ultrasonic measurement principle, the auto-braking system can detect the distance between the car and the obstacle, then the robot car will brake automatically when very near the obstacle, which could be helpful for new drivers. Now let's turn Maqueen Plus V2 into an auto-braking robot.

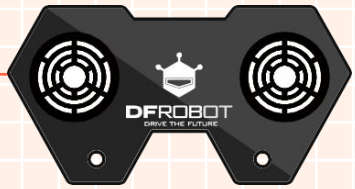
Goal

1. Learn how to use variables.
2. Learn how to embed a condition block inside another one.

Electronic Component

Ultrasonic Sensor Brief

Ultrasonic Sensor



The transmitter sends out ultrasound, and when hitting the object, the ultrasound reflects as echo and will be sensed by the receiver.

Command Learning

Block Brief

Variable

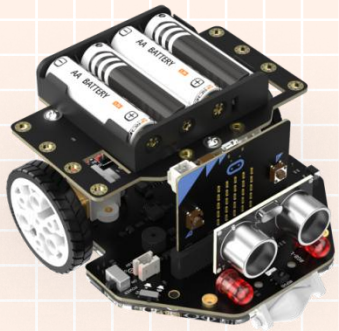
Changeable value: number, character string, list

Assign a value to variable

Assign a value to the variable distance.
Default setting: distance=0

Hands-on Practice

Install Ultrasonic Sensor

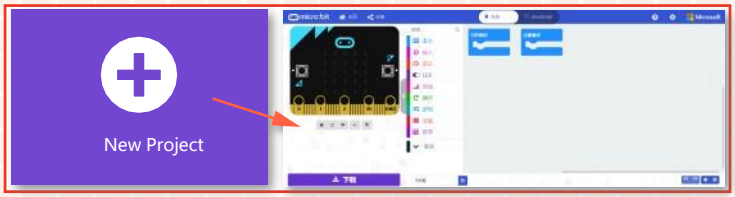


Our auto-braking robot needs to measure the distance between it and the obstacle ahead, so an ultrasonic sensor is necessary here.

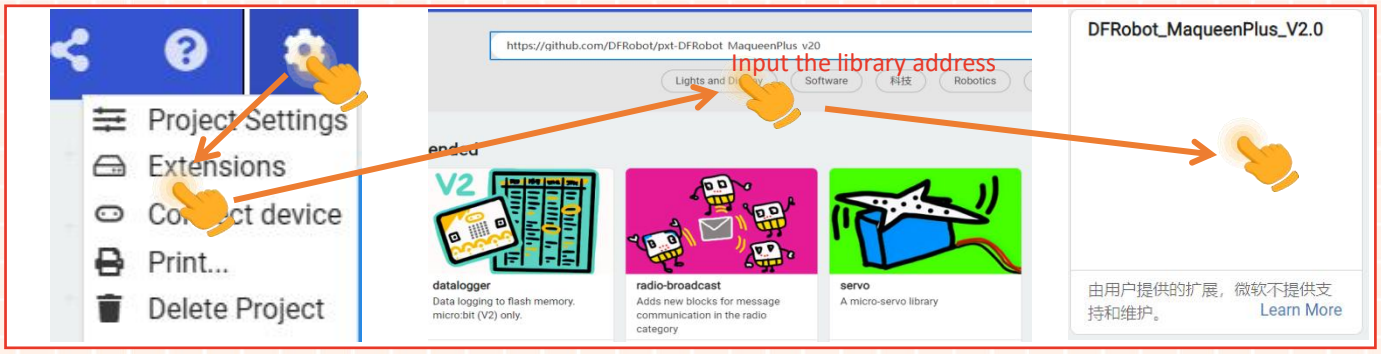
Step 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter MakeCode editor.
2. Click "new project" to enter MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



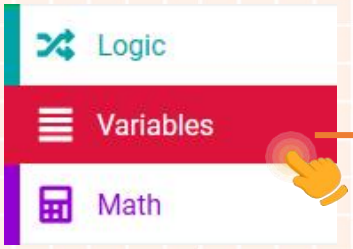
2. Enter programming interface



3. Add the extension library

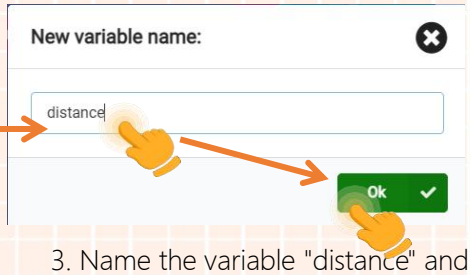
Step 2 Programming

1. Create a variable and name it "distance".



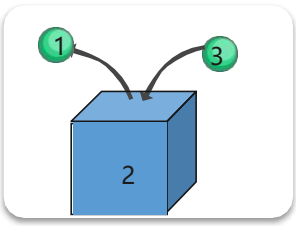
1. Left-click "Variable" in the command block section

2. Click "Make a variable"



3. Name the variable "distance" and click "OK".

Knowledge Expansion



What is a variable?

We may think of a variable as a container or box where we can store data that we need to use later, and each box can only hold one value (number, text and Boolean data) at a time. For example, use it to store an integer, after we put 1 into it, we put 2 into it, then we can only get 2 from this box. A new value will replace the previous one. The name of the box is the variable's name and the value of the variable is placed inside the box.

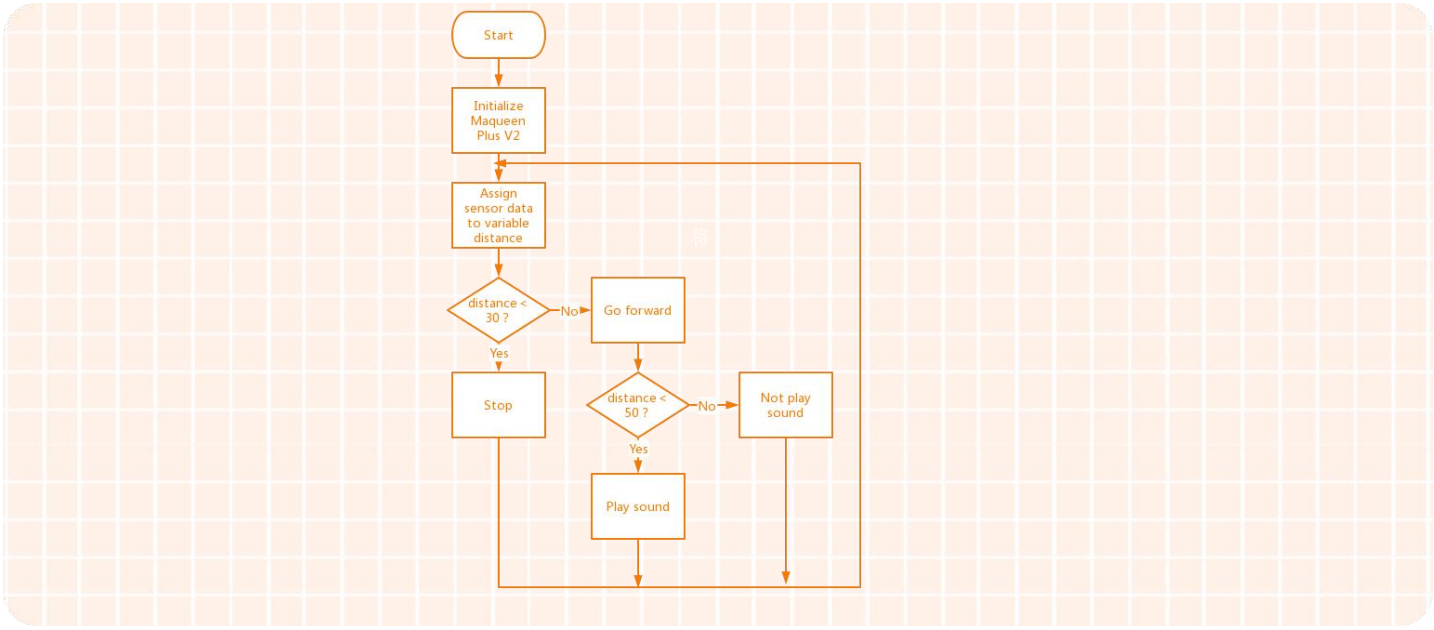
2. In this project, we need to monitor the distance value the sensor detected in real-time, so we have to assign the value to the variable "distance".

Knowledge Expansion

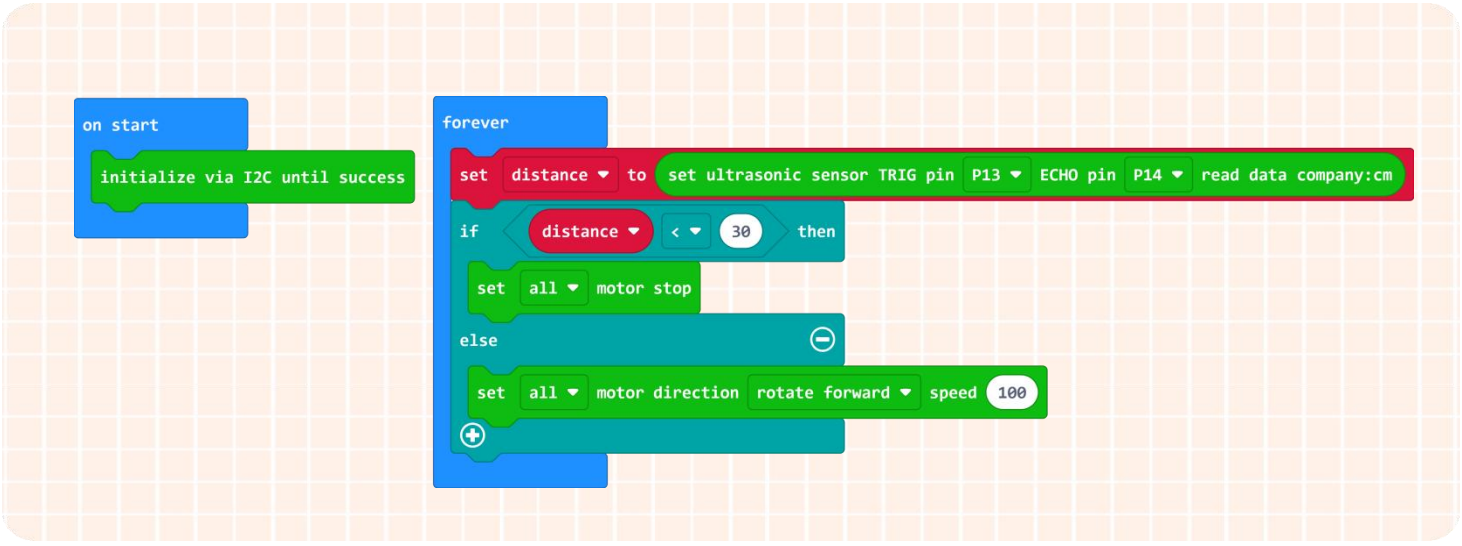
Then we can directly use the variable "distance" to call the distance value later.

3. When the robot car is near the obstacle (distance<50cm in the example), it keeps going and the buzzer keeps beeping to give an alarm. Once the distance between them is smaller than the preset value (distance<30cm in the example), the auto-braking robot stops.

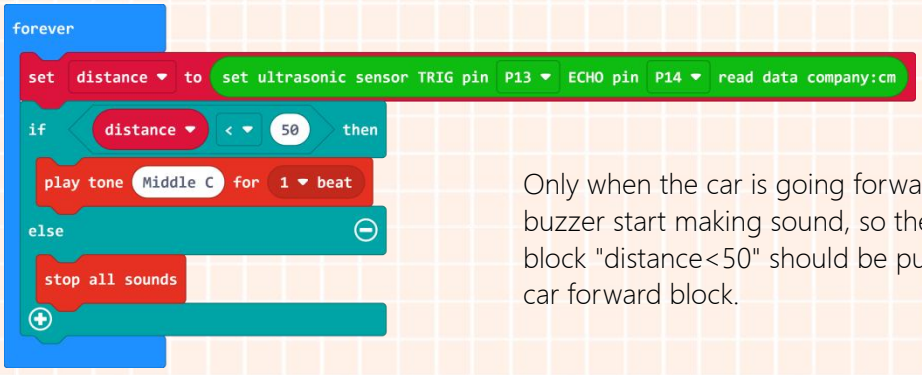
How to achieve that? Drawing a flowchart is gonna help us understand the program.



4. The robot car stops if the condition "distance < 30" is true and moves forwards if it's false.

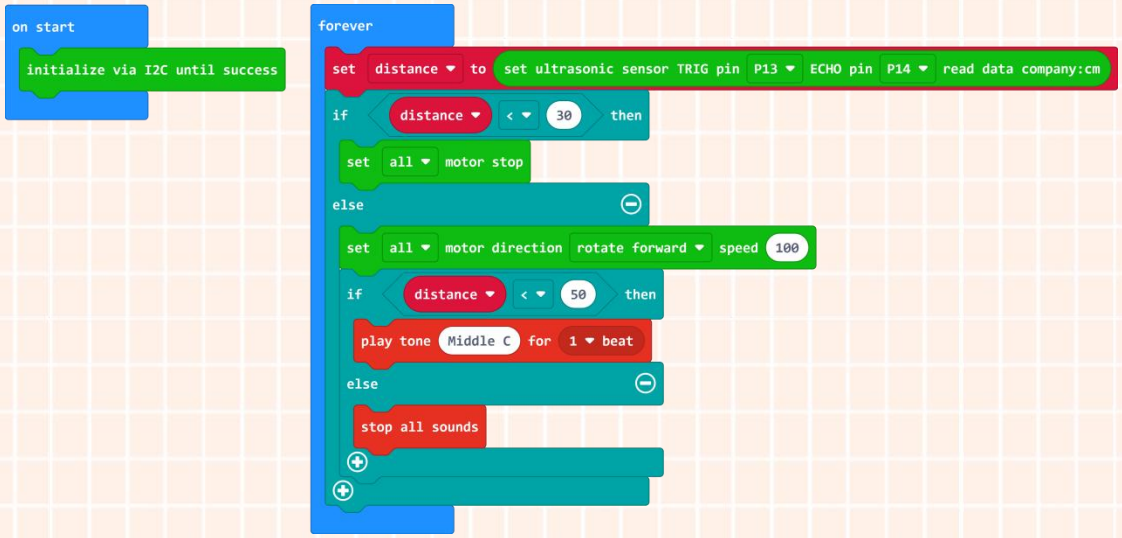


5. The buzzer keeps beeping if the condition "distance < 50" is true, otherwise, it stops making sound.

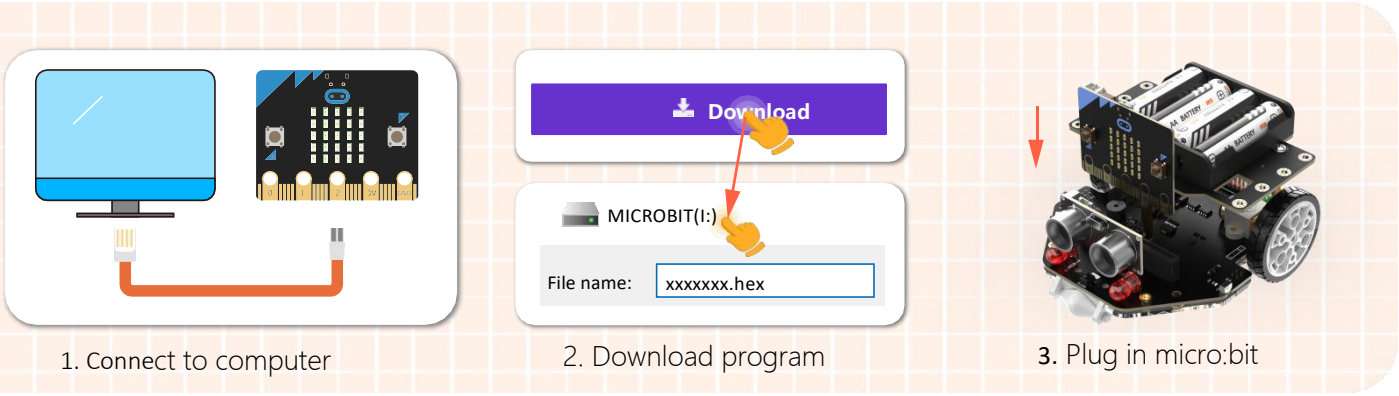


Only when the car is going forward will the buzzer start making sound, so the condition block "distance<50" should be put under the car forward block.

6. The complete program is shown below.



Step 3 Download Program



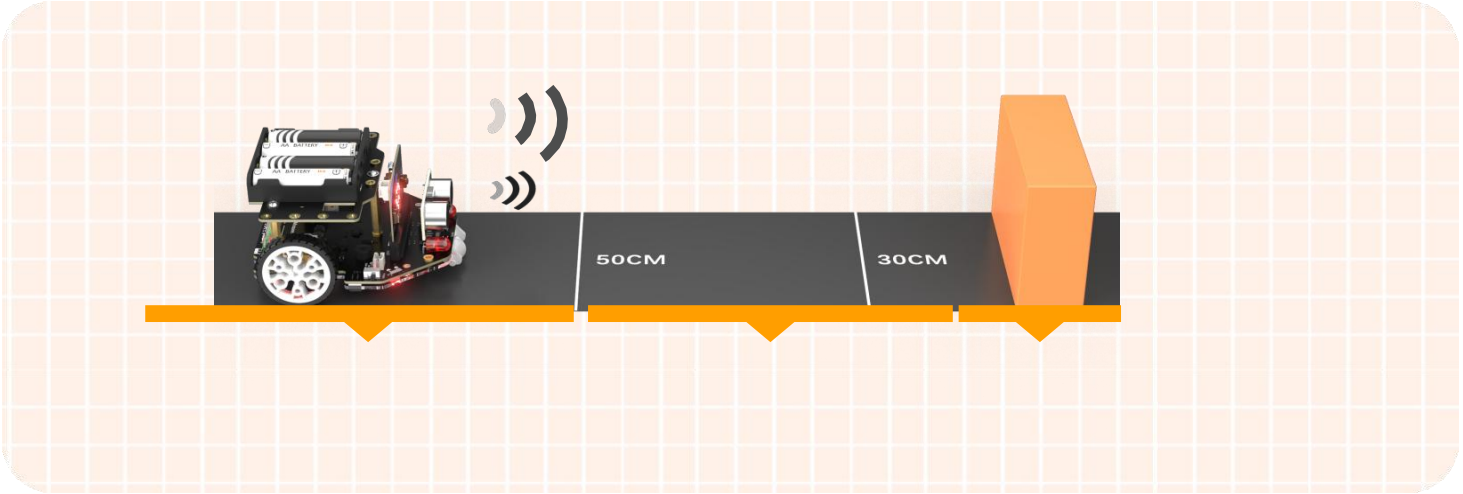
1. Connect to computer

2. Download program

3. Plug in micro:bit

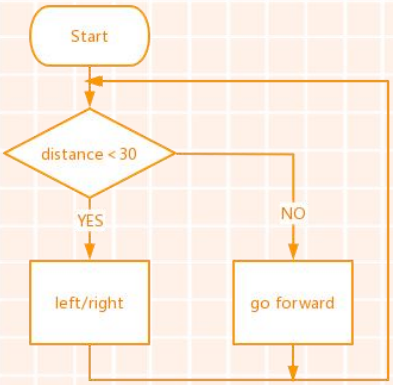
Step 4 Effect Display

After we completed all the above steps, put an obstacle in front of the Maqueen Plus V2 car, and turn on the power switch. When the detected distance is smaller than 50, the buzzer keeps beeping; distance<30, the car stop.

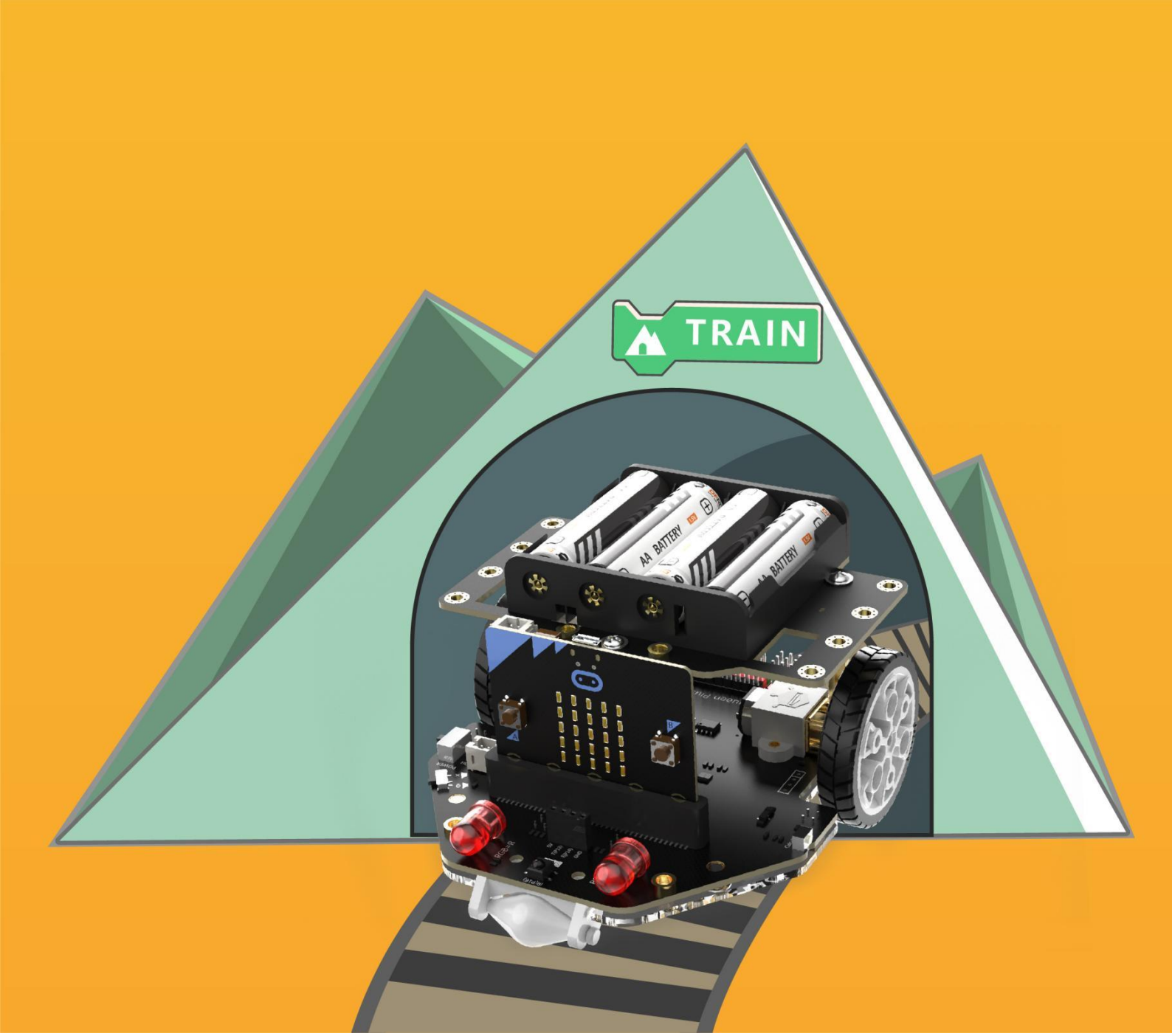


Think & Explore

We have turned Maqueen Plus V2 into an automated braking robot. The robot car stops when it is very near an obstacle, but can you let Maqueen Plus V2 bypass it? Make an obstacle avoidance car according to the flowchart below.



Tip: the ultrasonic sensor should point to the direction the car moves toward.



Chapter 9

Line-tracking Robot

Our line-tracking robot is fond of exploring things with a map. No matter how complicated the road is, give him a long enough track, he will trace it to the end of the world. Let's step on an adventure with the Maqueen Plus V2 robot!

Goal

1. Learn the principle of the line-tracking sensor
2. Learn the logic "and"

Electronic Component

Line Tracking Sensor Brief

Line-tracking sensor

Detect white lines in black background and black lines in white background

Command Learning

Block Brief

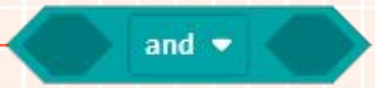
Line-tracking sensor

Read the output value from sensors L1, L2, M, R1 and R2.

Comparison operator "="

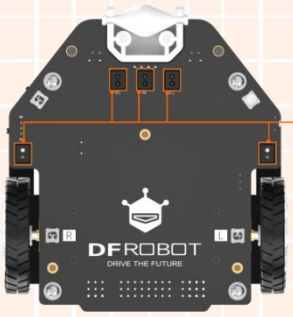
Return true if both inputs equal to each other.

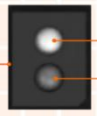
Logic "and"



Return true if both inputs are true.

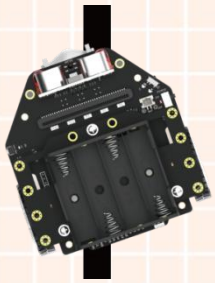
How does a line-tracking sensor work?





IR Transmitter

IR Receiver



There are 5 line-tracking sensors integrated on Maqueen Plus V2 board. Each sensor includes an IR transmitter and receiver.

When a line-tracking sensor detected the black line on the map, output "1", otherwise, output "0".

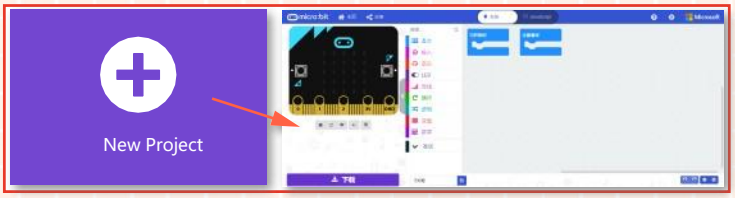
Note:
1. Since dark colors absorb light(including IR light), when the line-tracking sensor detected black, the IR light emitted by the transmitter cannot be reflected back to the receiver.
2. The output "0" or "1" does not refer to a high/low level, it's just a value obtained by processing the read grayscale.

Hands-on Practice

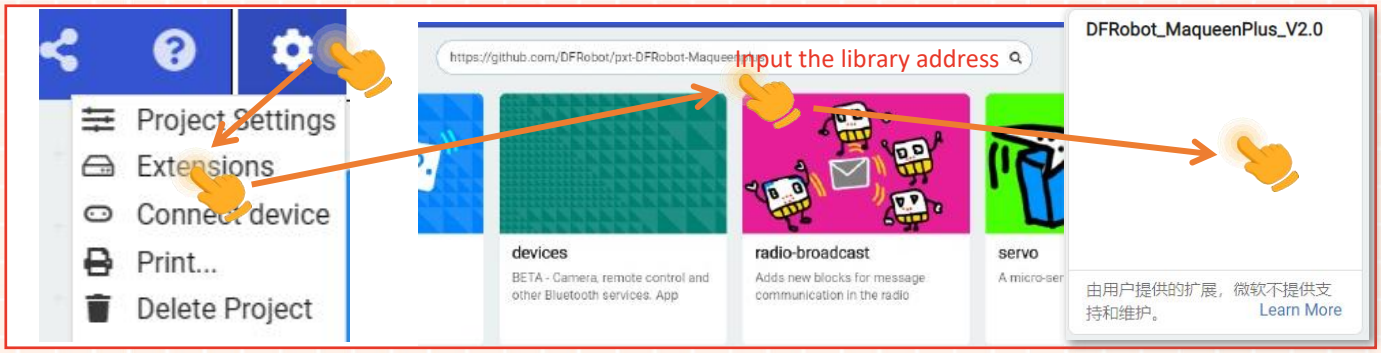
Step 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



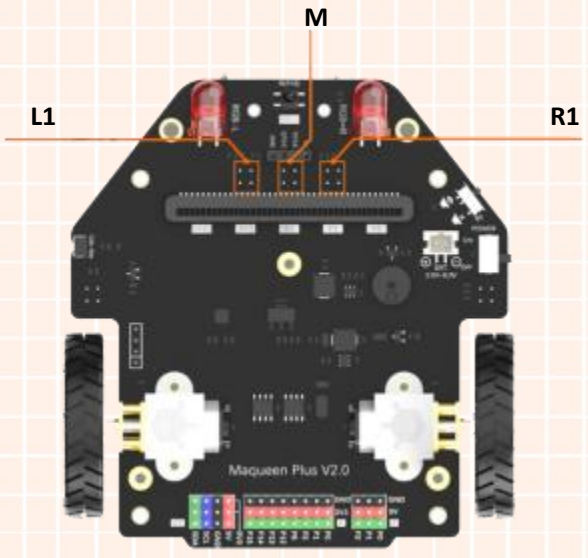
2. Enter programming interface



3. Add the extension library

Step 2 Programming

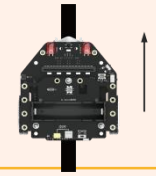
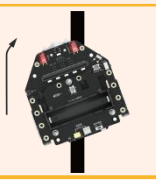
Maqueen Plus V2 moves along the black line on the map. If you don't have a map, you can make one using black adhesive tape. (Line-tracking Sensor R1, L1 and M will be used in this project.)



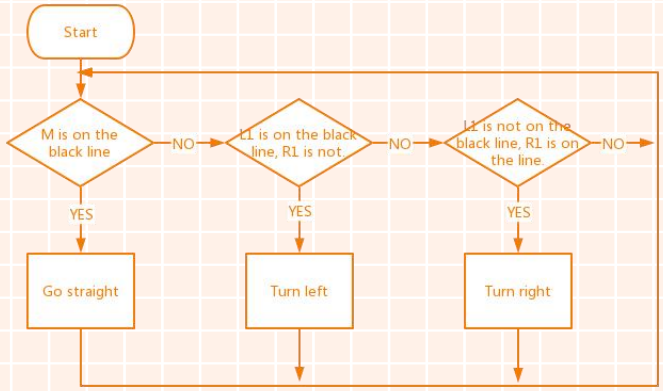
Knowledge Expansion

When you draw your own map, please make sure that **all** sensors L1, R1 **and M** can be placed on the black line.

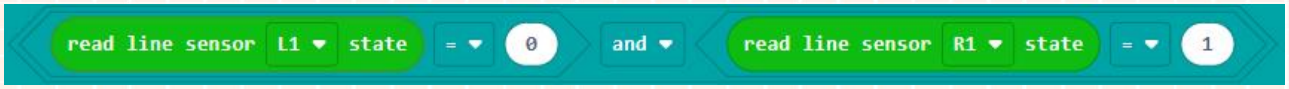
1. There are three possibilities when Maqueen Plus V2 drives on the map.

Status Image	Sensor Status	Detection	Output	Motor Movement
	M is on the black line	The middle sensor detected the black line.	M = 1	Go straight
	L1 is on the black line, R1 is not.	Only the left sensor detected the black line.	L1 = 1 R1 = 0	Turn left
	L1 is not on the black line, R1 is on the line.	Only the right sensor detected the black line	L1 = 0 R1 = 1	Turn right

2. Draw the corresponding program flowchart.

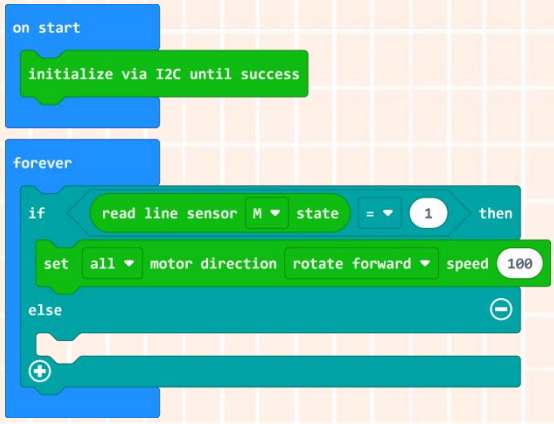


3. Since there are two conditions to judge, the outputs from sensor R1 and L1, we need to use a "and" block to combine them.

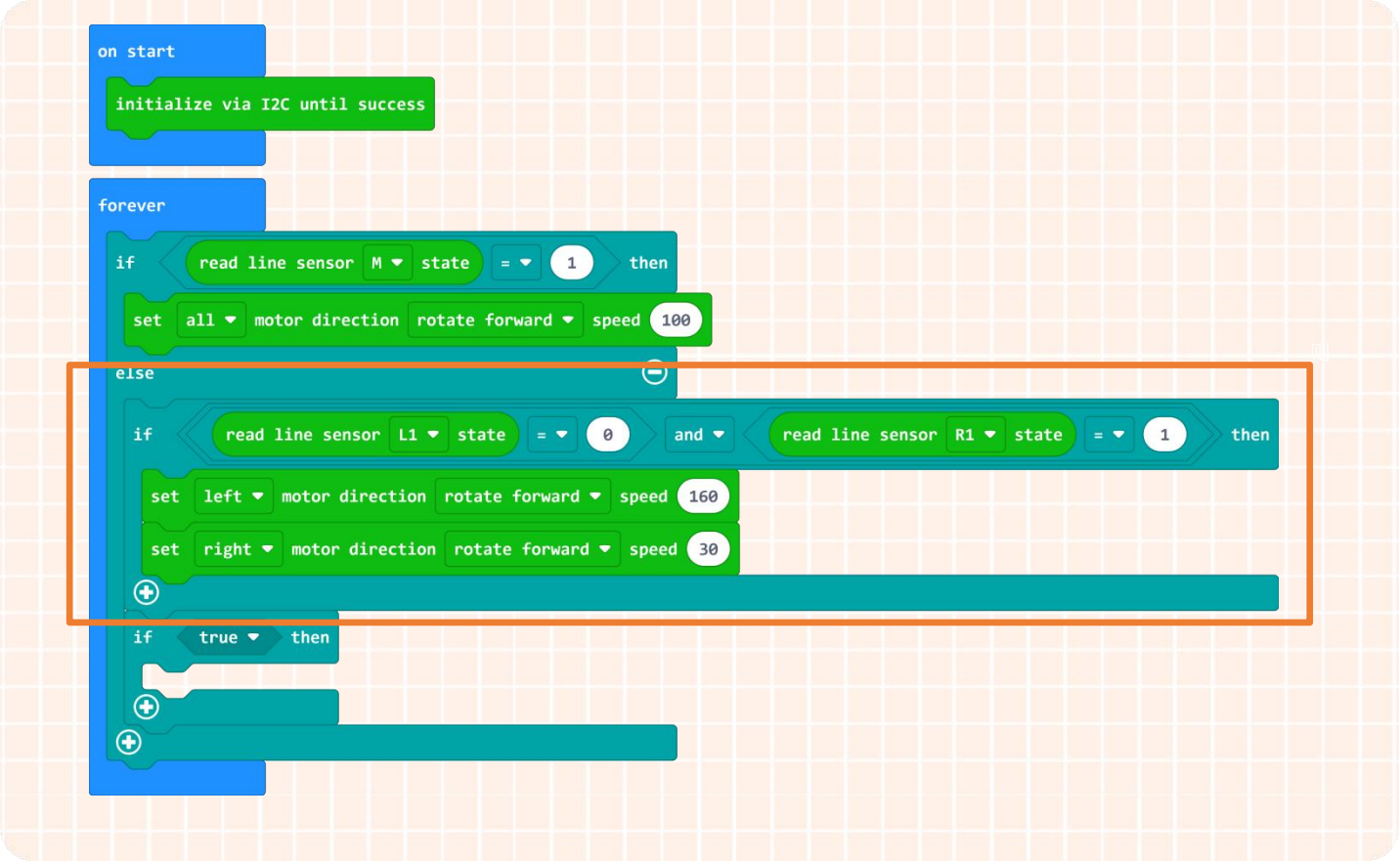


Note: the program means that only the sensor R1 detected the black line.

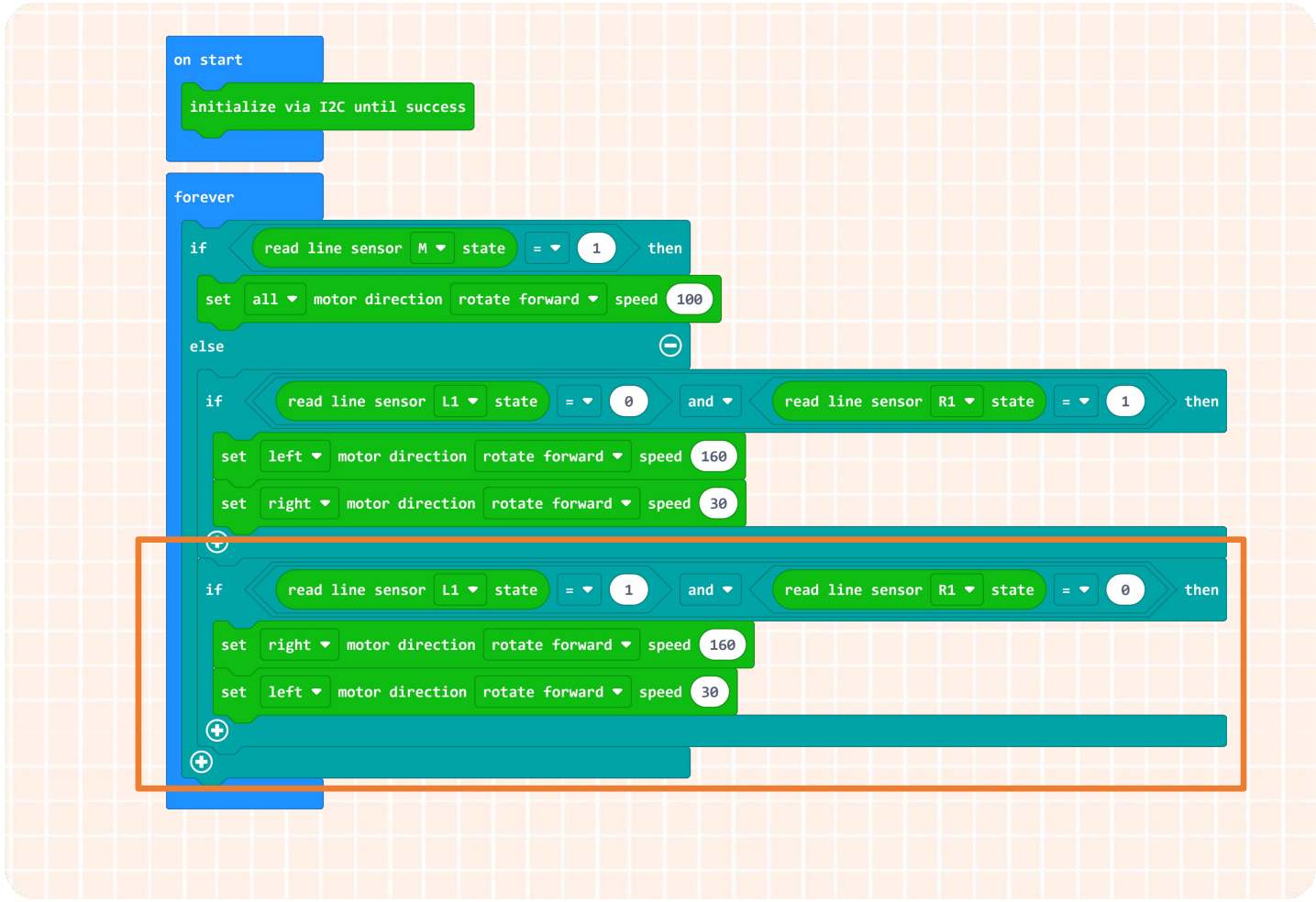
4. When sensor M detected the black line, Maqueen Plus V2 car moves forward.



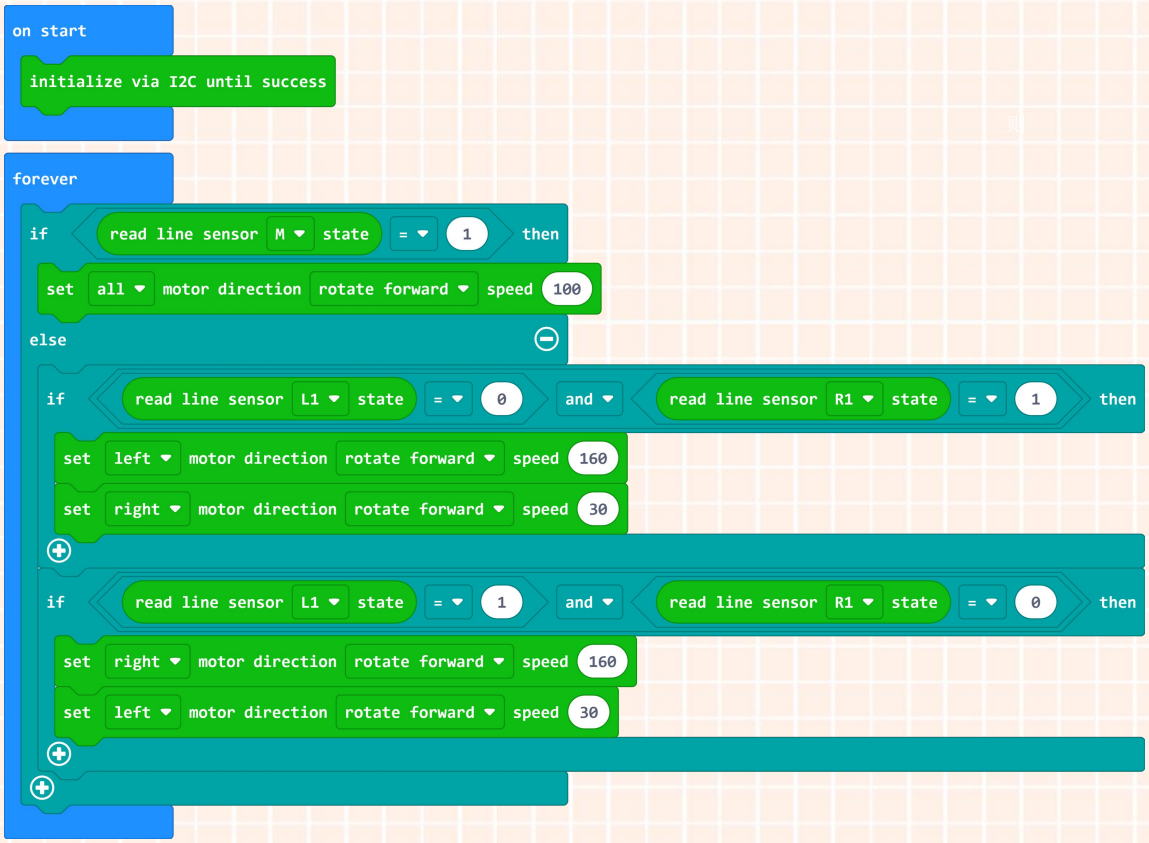
5. When only the right sensor (R1) detected the black line, Maqueen Plus V2 car turns right.



6. When only the left sensor (L1) detected the black line, Maqueen Plus V2 car turns left.



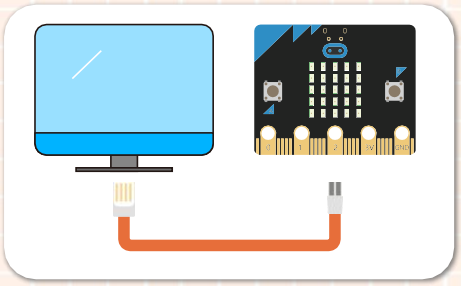
7. The complete program is shown below:



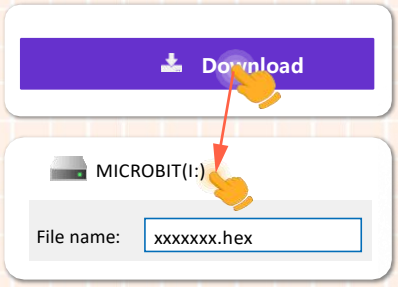
Note: if the Maqueen Plus V2 car turns left or right too much, you can change its motor speed to adjust.

8. Name your project as "Line-tracking Robot" and save it.

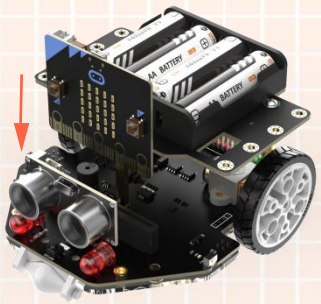
Step 3 Download Program



1. Connect to computer



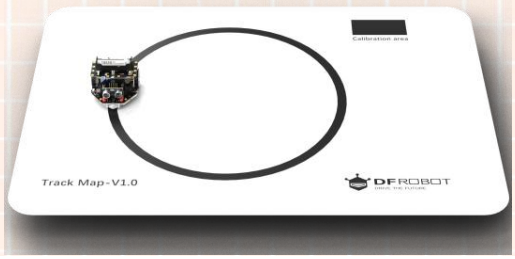
2. Download program



3. Plug in micro:bit

Step 4 Effect Display

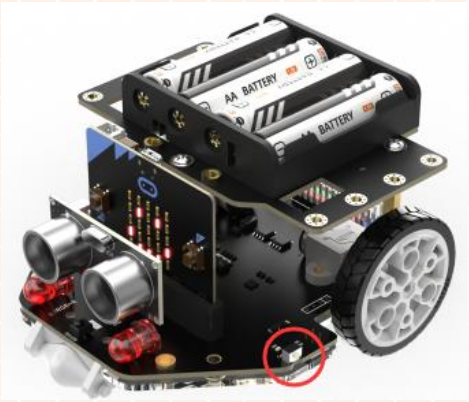
Turn on the power switch after all the above steps are done, put Maqueen Plus V2 car on the map, then it will automatically move along the black line, just like a track train!



Extension-Sensor Calibration

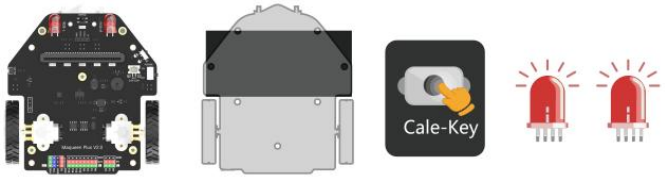
The line-tracking sensors on Maqueen Plus V2 can be directly used since they are factory calibrated. But if you find that your sensors cannot detect black line accurately, you can calibrate them in the way shown below:

The button circled in red is the calibration button.



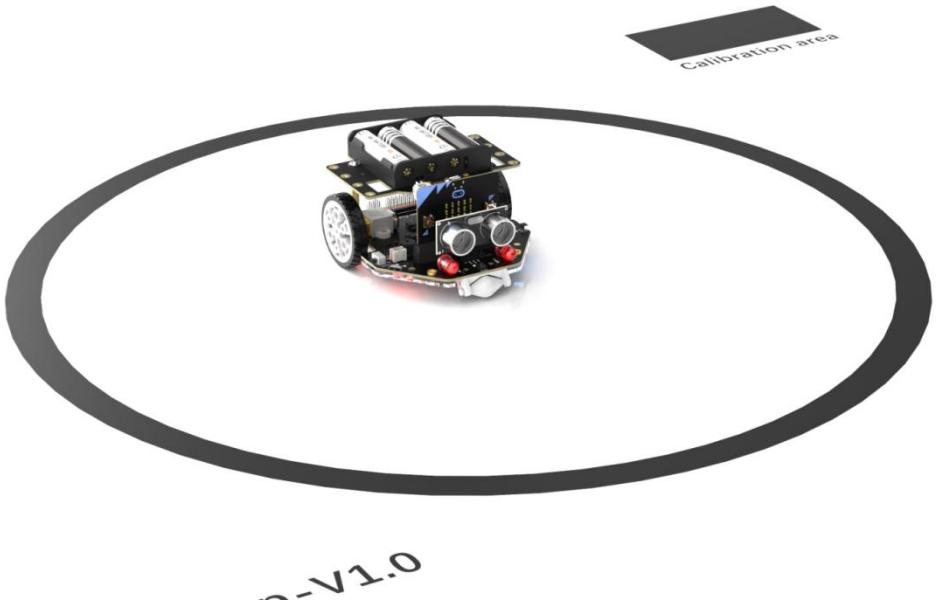
Place Maqueen Plus V2 on the black calibrating area of the map, and make sure all the line-tracking sensors are within that area. Press down the calibration button, when the two RGB LEDs flashes red light, release the button, then sensor calibration is done.

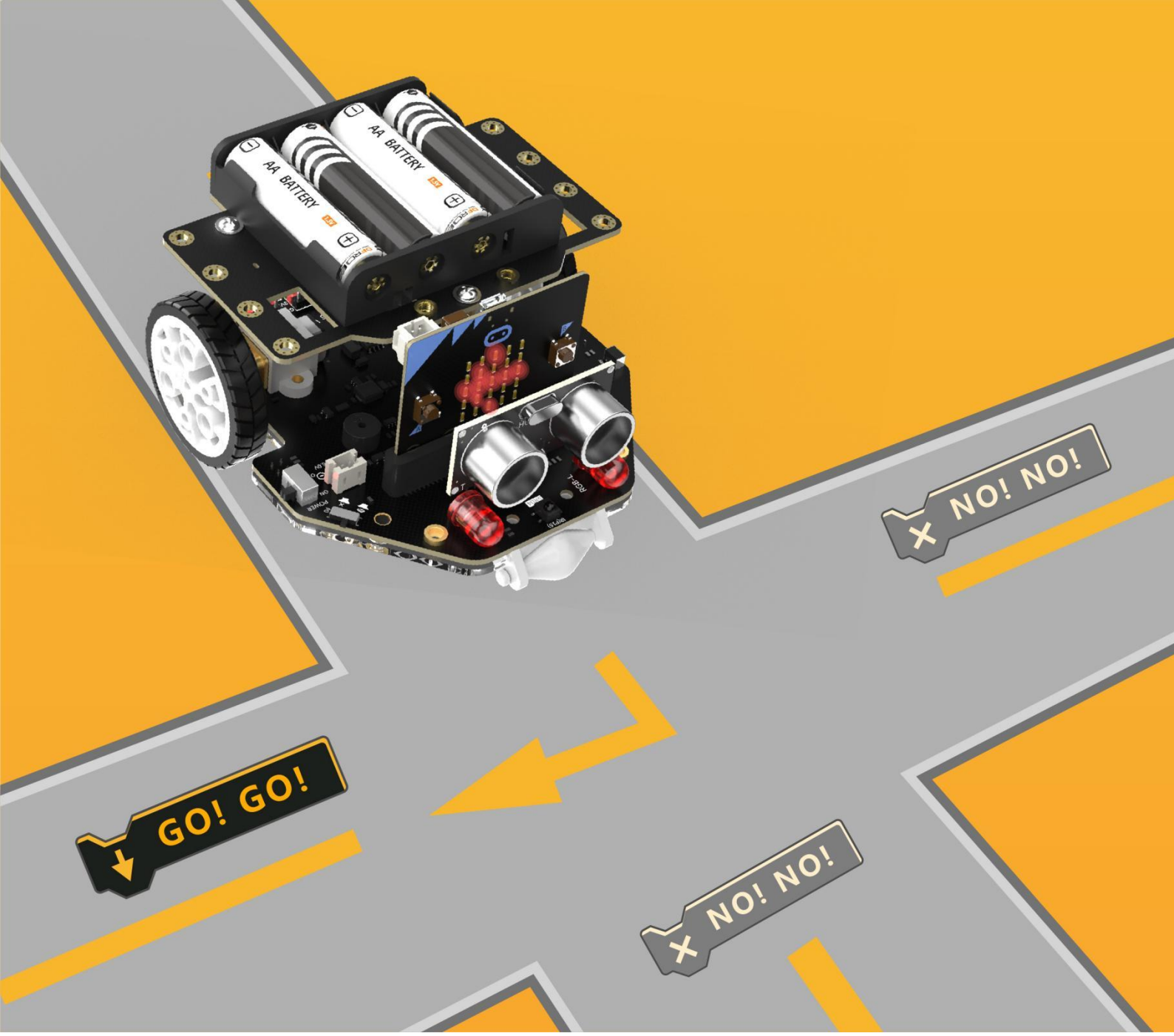
Note: the black line printed by printer cannot be accurately detected sometimes. You can use black tape or marker to make a map if necessary.



Think & Explore

With the development of technology, sweeping robot is gradually becoming a part of our family life. Simply place it on the floor and turn it on, here it goes! To prevent the robot from falling off the stairs, the bottom of the robot usually is surrounded by many sensors. Our Maqueen Plus V2 has 5 line-tracking sensors, so it can totally meet the requirements. Let's make a sweeping robot with Maqueen Plus V2. Take the black line on the map as the edge of the stairs, and the robot will be only allowed to move within that area.





Chapter 10

Tour of Crossroad

Standing at a crossroad, Maqueen Plus V2 is wondering which way he should go. Every road is so unique. Well, why not try all the roads? That sounds like a good idea, right? Let's help Maqueen Plus V2 to start his tour of the crossroads!

Goal

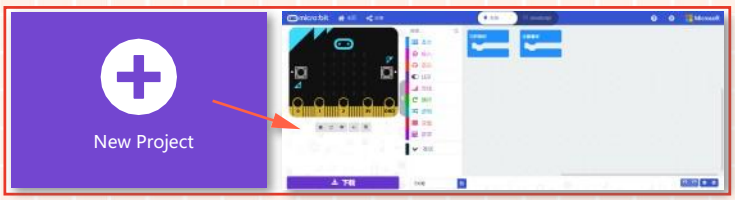
Learn the use of multiple line-tracking sensors

Hands-on Practice

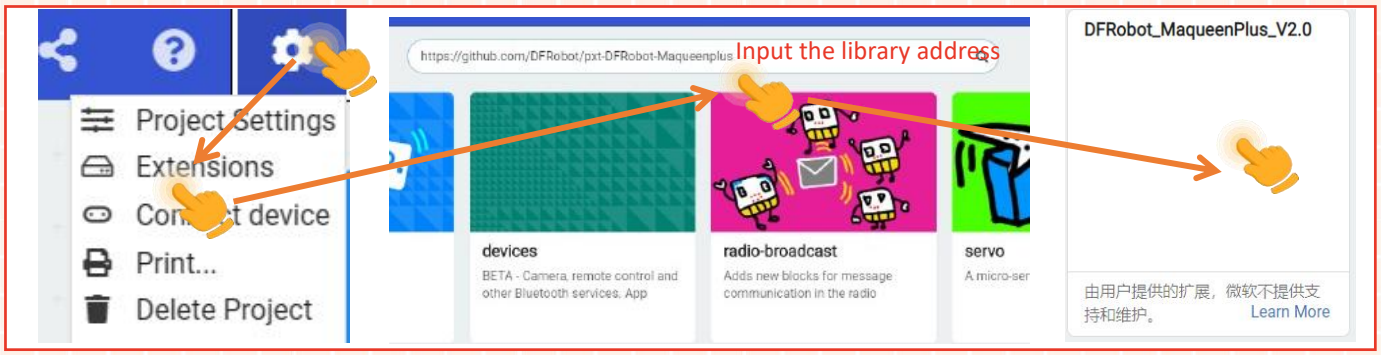
Step 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



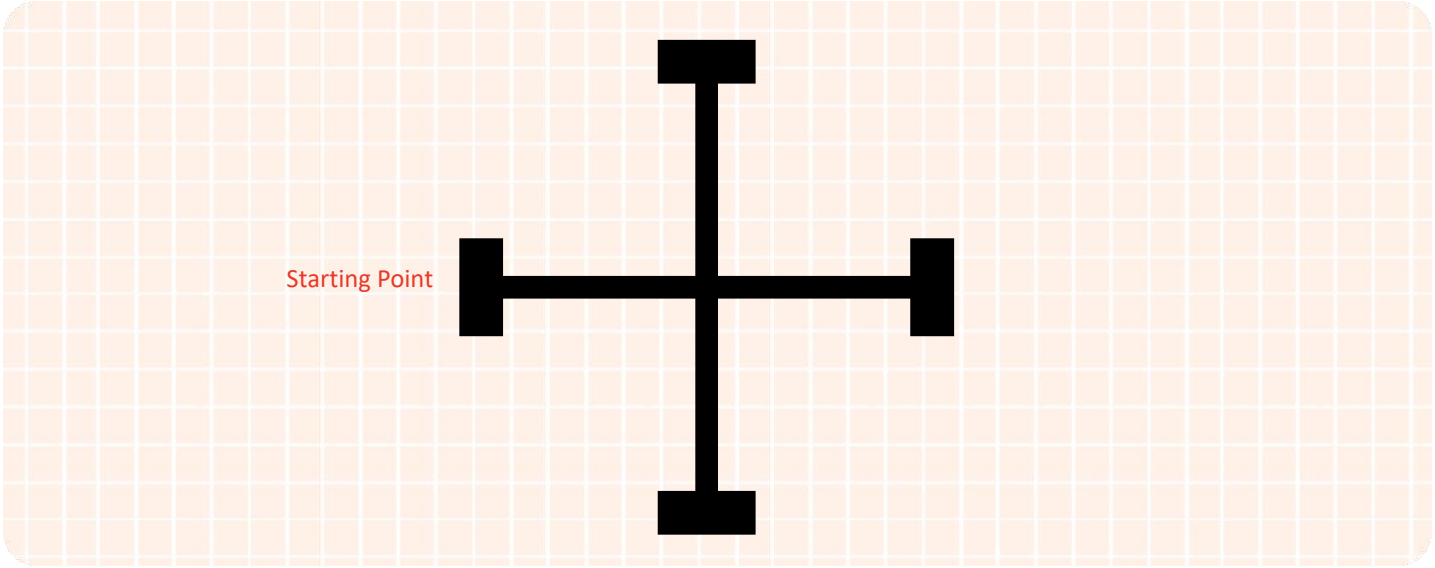
2. Enter programming interface



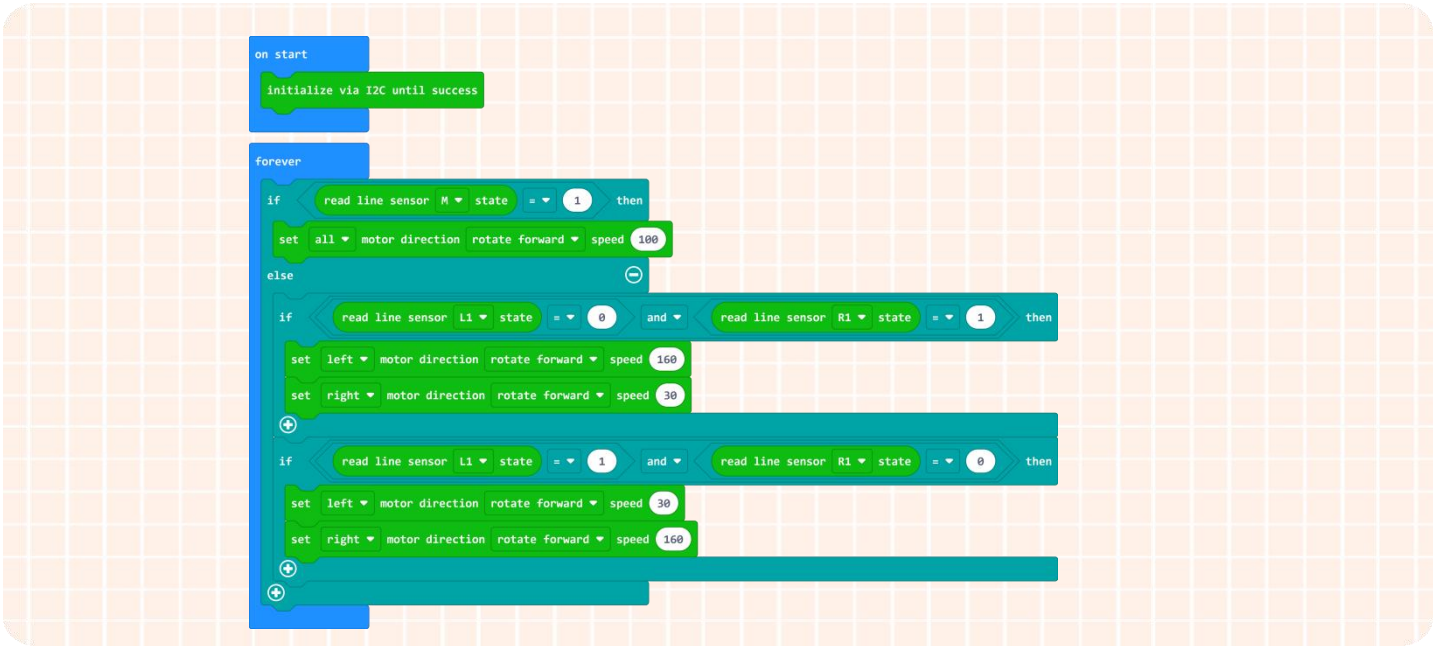
3. Add the extension library

Step 2 Programming

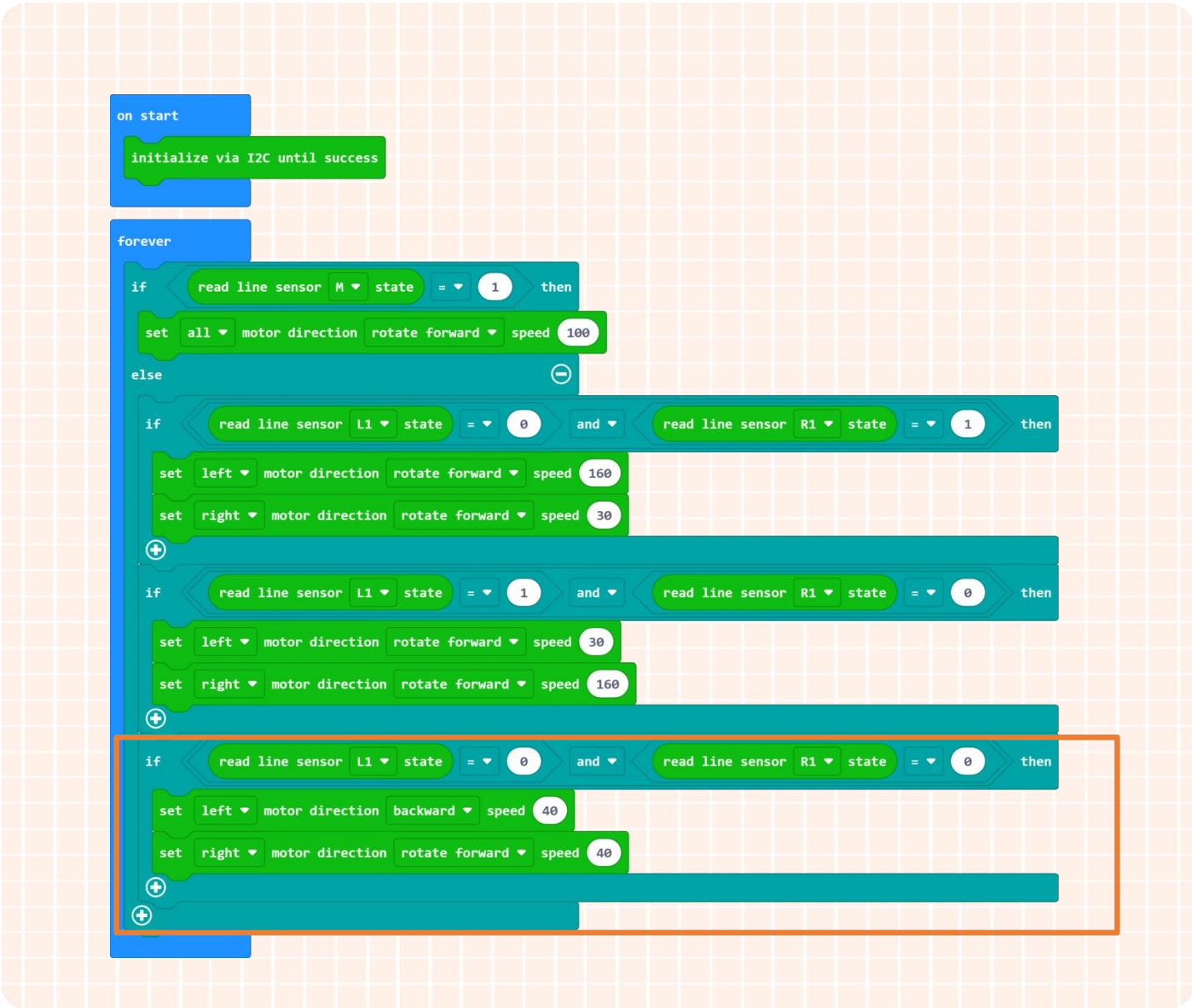
1.In this project, we need to prepare a crossroad map. Maqueen Plus V2 will try all roads of the crossroad, and then back to the starting point. How can we realize that by programming? In the process of line-tracking, Maqueen Plus V2 turns left/right at the intersection, and then turns around at the end of the road. Repeat this series of actions all the time.



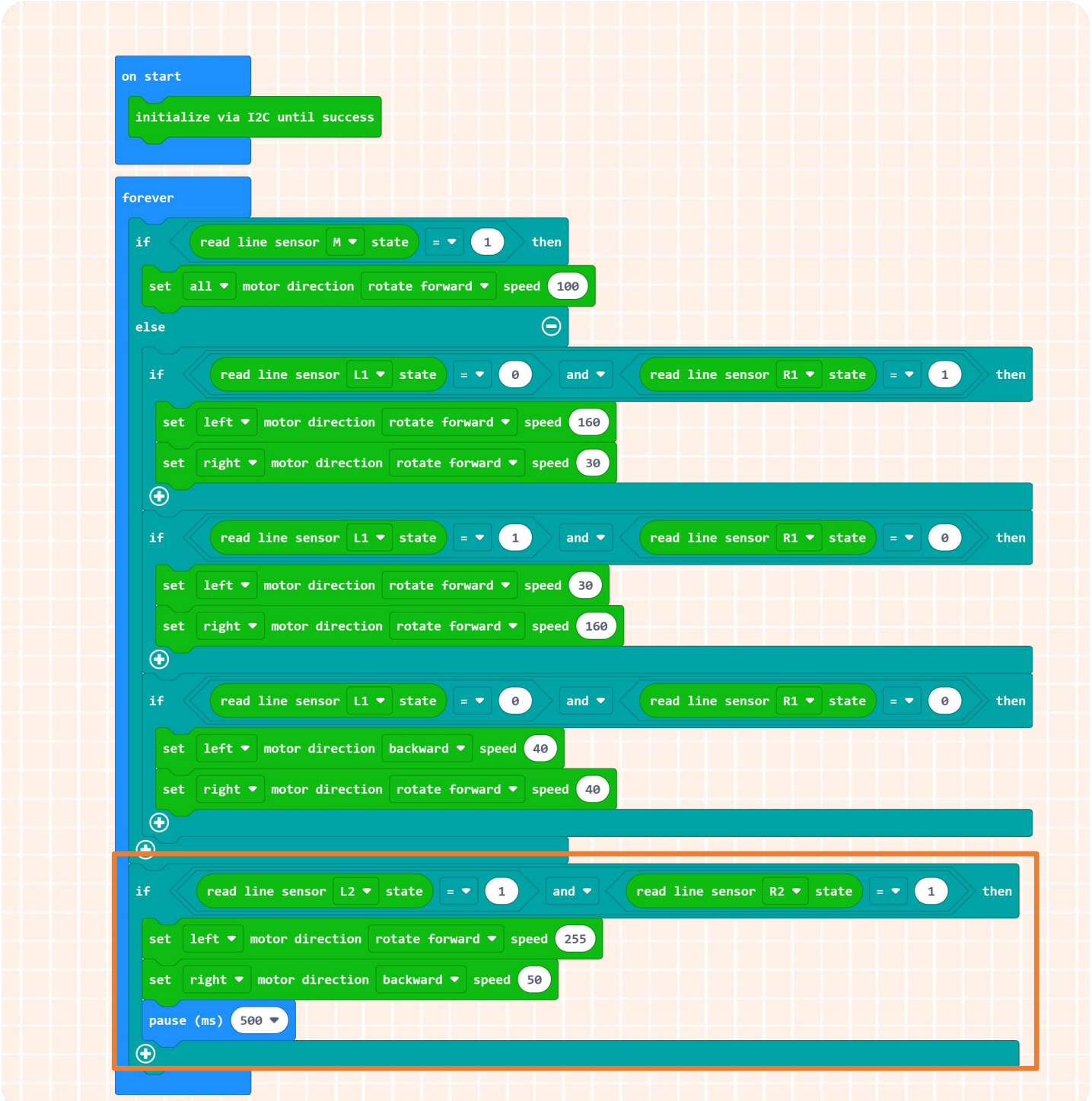
2.First, complete the program of line-tracking.



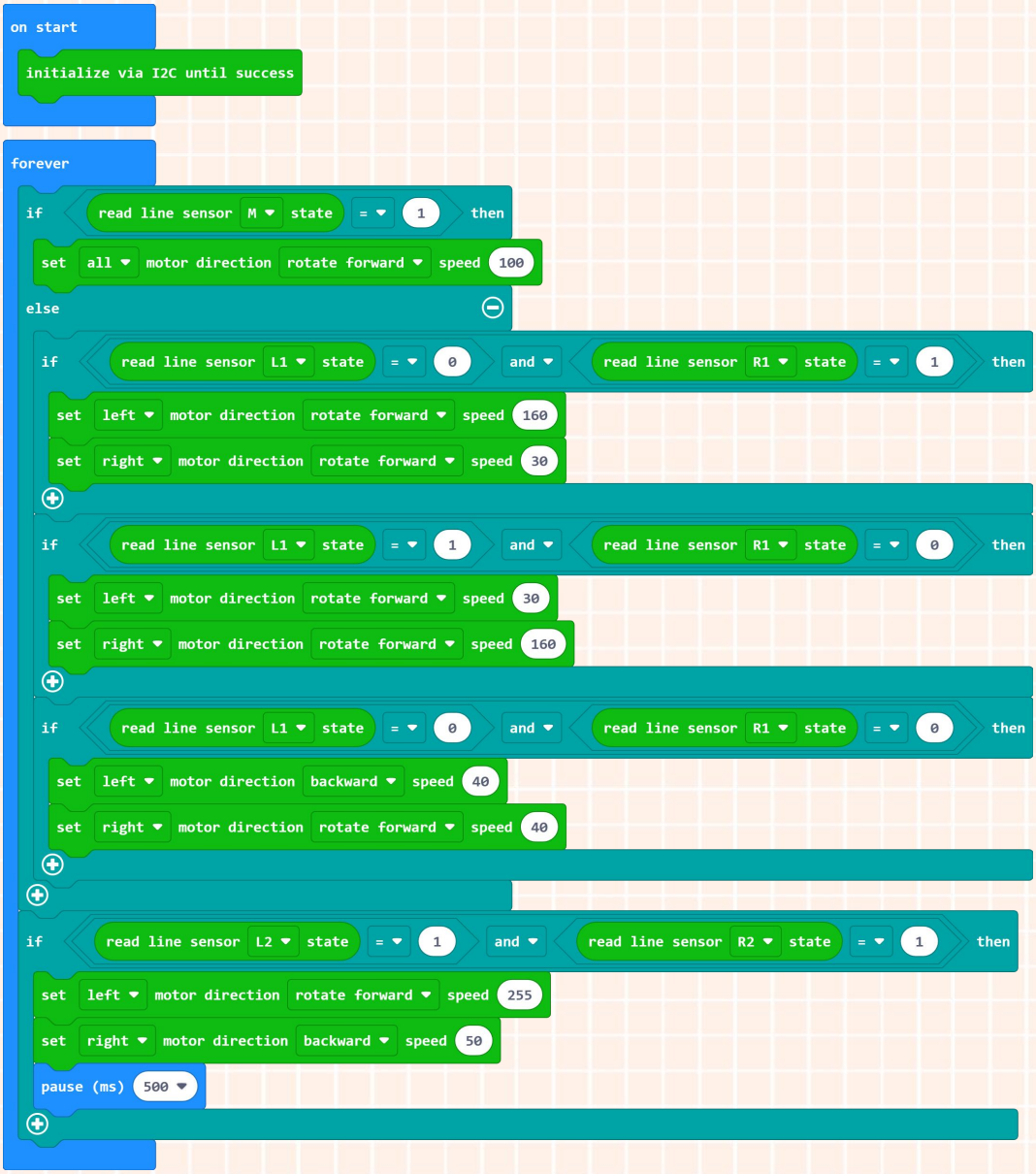
3.When no black line is detected by sensors L1 and R1, it means that the car drives out of the black track. Now let the Maqueen Plus V2 spin around until the black track is found again.



4.When the sensors L2 and R2 detected the black line, it means that the car has arrived at the intersection or the end of the road, and the car needs to spin around to find the black track again.



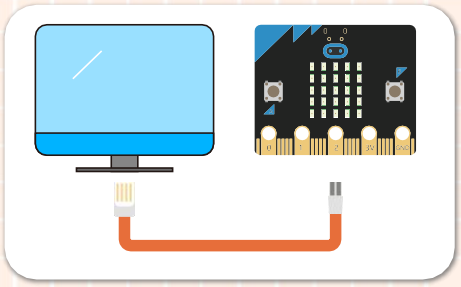
5.The complete program is shown below.



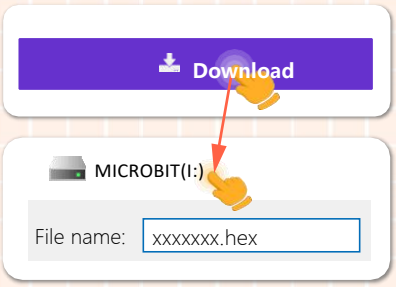
Note: if the Maqueen Plus V2 car turns left or right too much, you can change its motor speed

6.Name your project "Tour of crossroad" and save it.

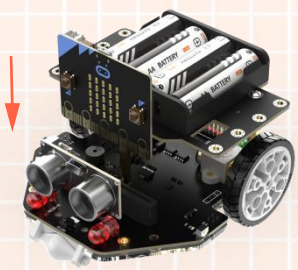
Step 3 Download Program



1. Connect to computer



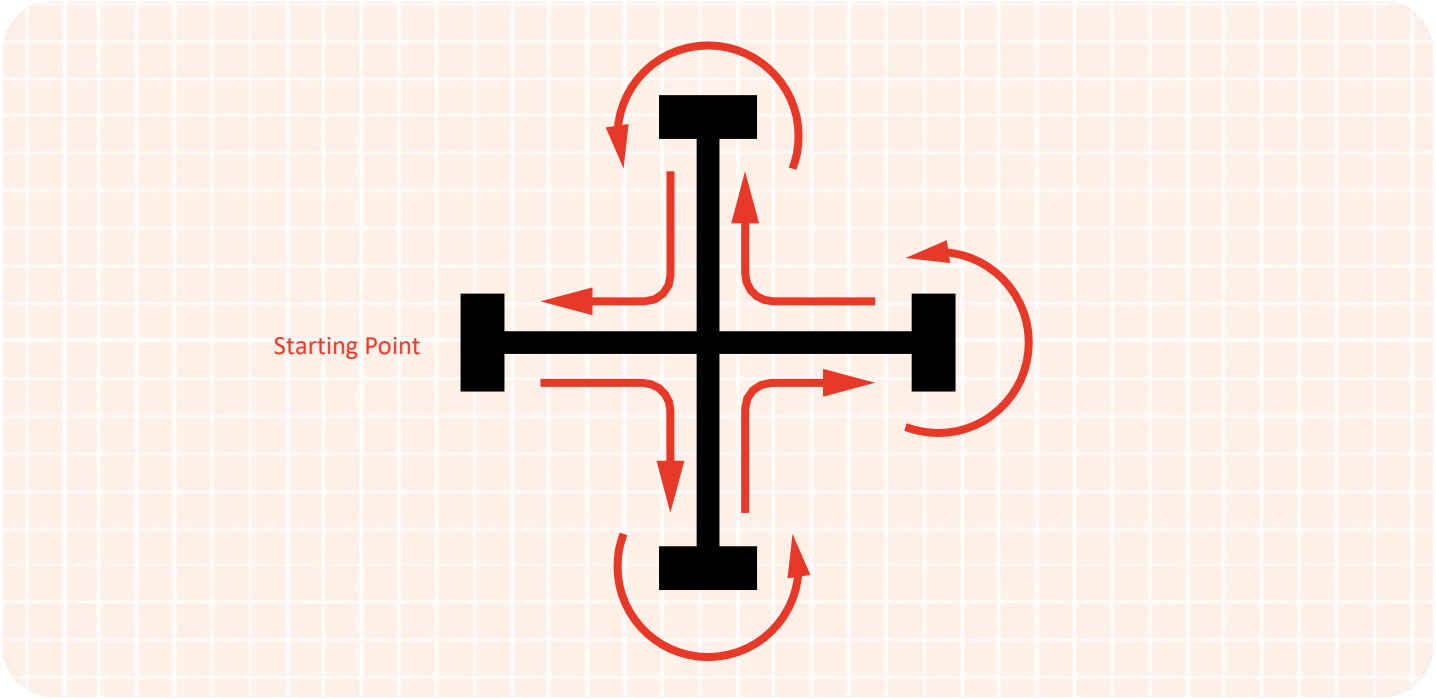
2. Download program



3. Plug in micro:bit

Step 4 Effect Display

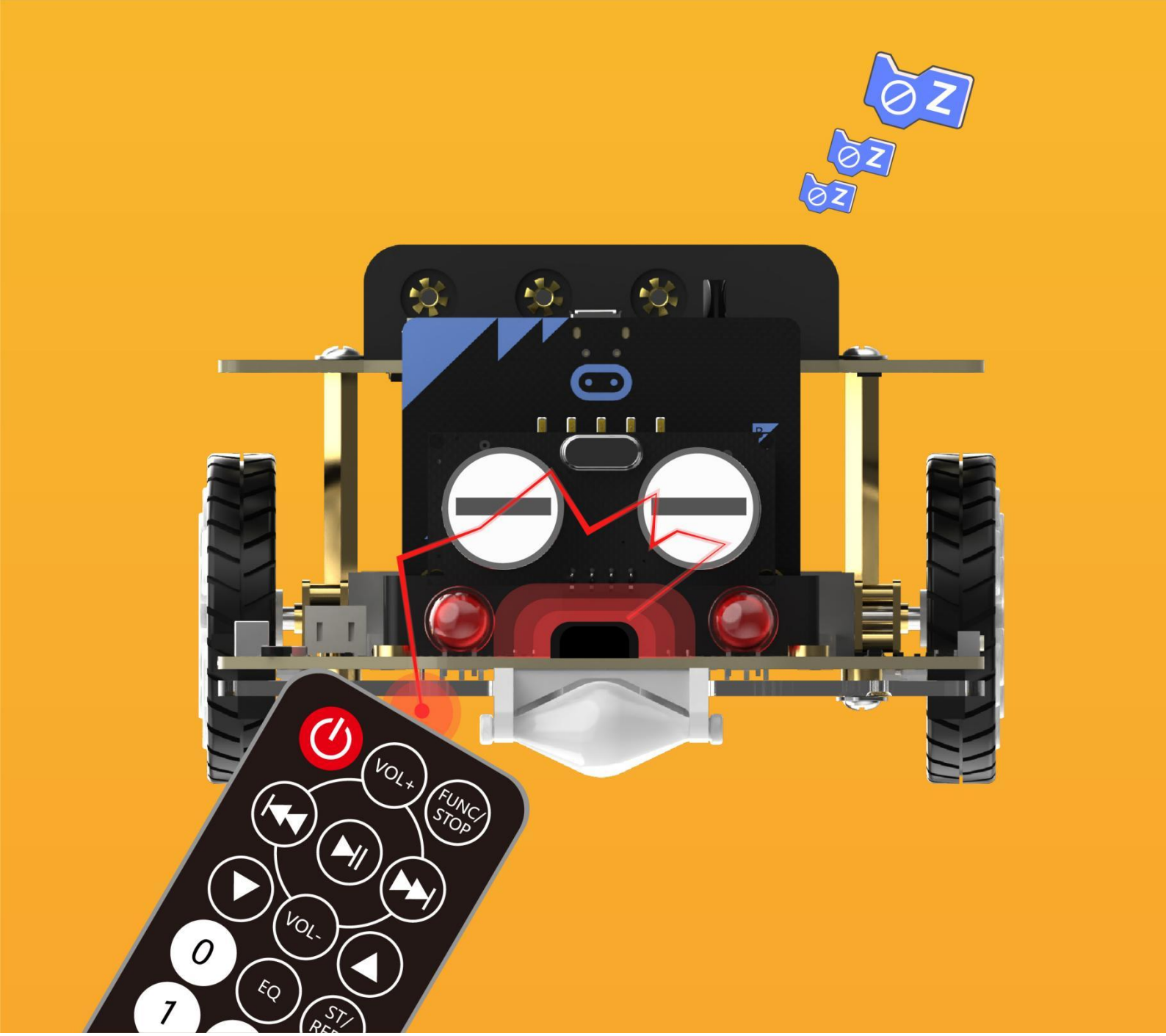
When you completed all the steps, put Maqueen Plus V2 on the crossroad map, and turn on its power switch.



Think & Explore

When Maqueen Plus V2 arrives at the intersection, he will turn right, well, now let's make its right RGB LED flash while turning right, and both RGB LEDs flash before turning around.

Tip: add RGB blocks in the program above.



Chapter 11

IR-controlled Robot

The invention of remote controllers allows people to operate devices from a certain distance, which brings a lot of convenience to our daily life. IR remote controller is the most commonly used one. In this chapter, we will use an IR remote controller with our Maqueen Plus V2 to make an "IR-controlled Robot".

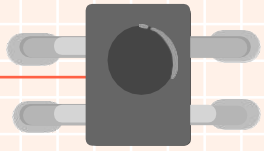
Goal

1. Learn how to use function block
2. Learn how to use IR remote controller to control motor

Electronic Component

IR Receiver Brief

IR receiver



Detect white lines in black background and black lines in white background

Command Learning

Block Brief

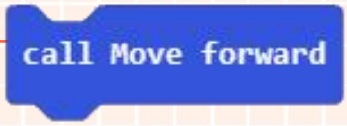
IR receiver

Receive and read the infrared value.

Create a sub-function

When a continuous action needs to occur multiple times in the program, to make the program more clear, we need to define a sub-function.

Call a sub-function



Drag the sub-function into the program to call it.

Hands-on Practice

We will use an IR remote controller to operate our Maqueen Plus V2, so we have to get the key value first. The decimal number of each key on the remote controller is shown below.

Key Value

Key	Value
Red key	0
VOL+	1
FUNC/STOP	2
Left arrow	4
Pause	5
Right arrow	6
Down arrow	8
VOL-	9

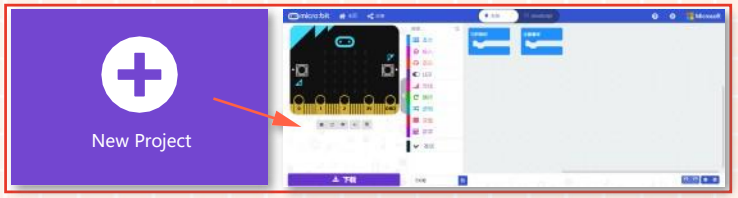
Key	Value
Up arrow	10
0	12
EQ	13
ST/REST	14
1	16
2	17
3	18
4	20

Key	Value
5	21
6	22
7	24
8	25
9	26

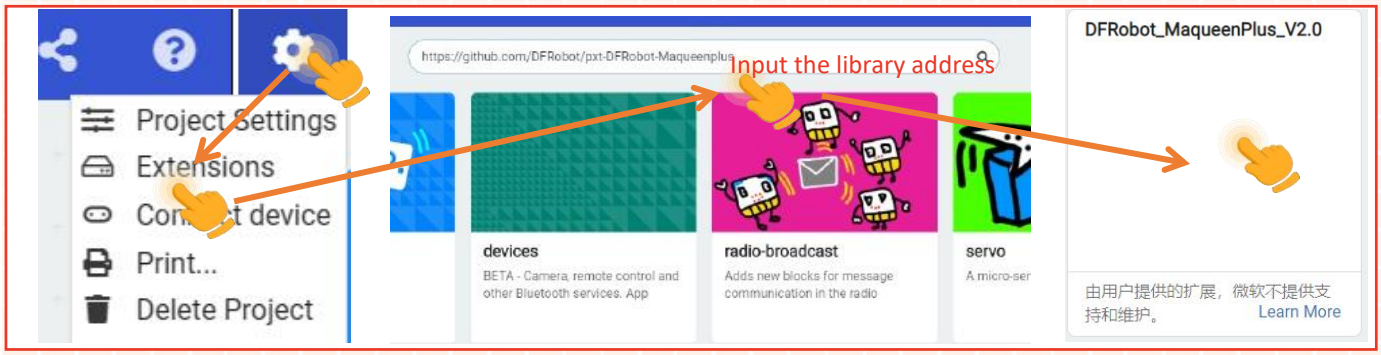
Step 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



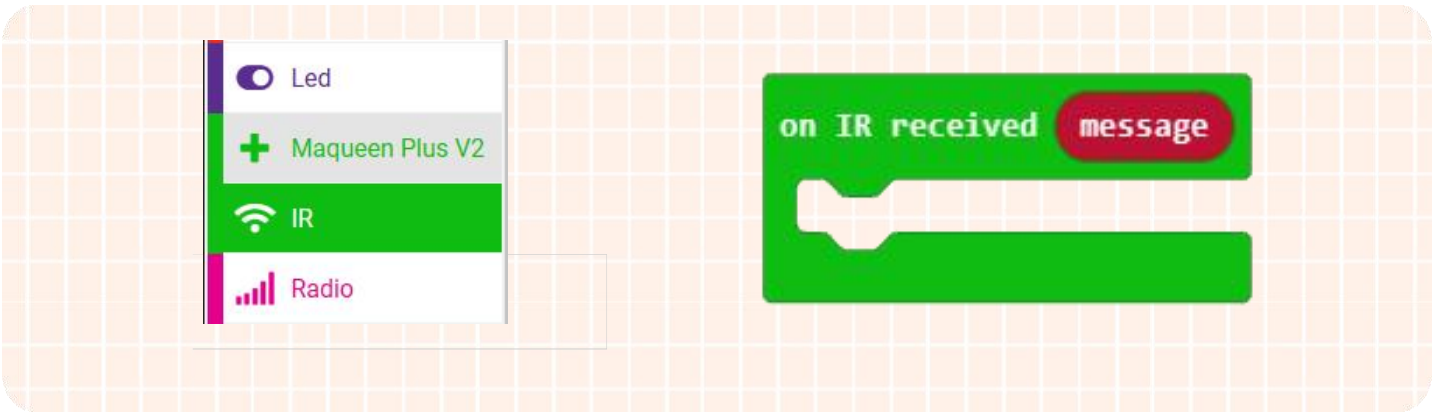
2. Enter programming interface



3. Add the extension library

Step 2 Programming

1. Drag the IR receive block to the editing section.



2. After we dragged the IR receive block to the editing section, there will be a variable named "message" appearing in the variable command section for storing the key value of the IR remote controller.

3. Create a function and name it "Move forward".

① Click "Advanced" -> "Function"

② Click "Make a function"

③ Name the function as "Move forward"

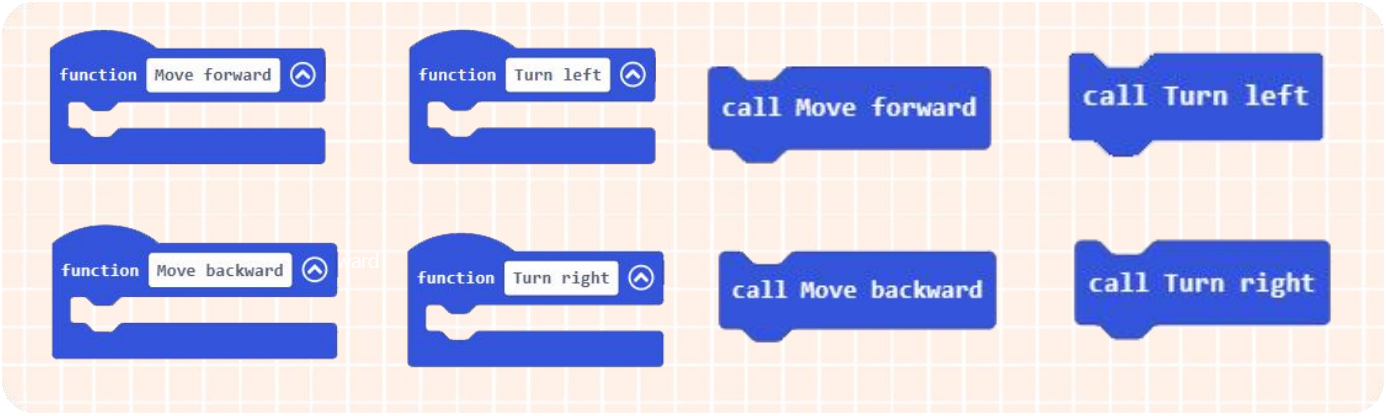
④ Click "OK"

Knowledge Expansion

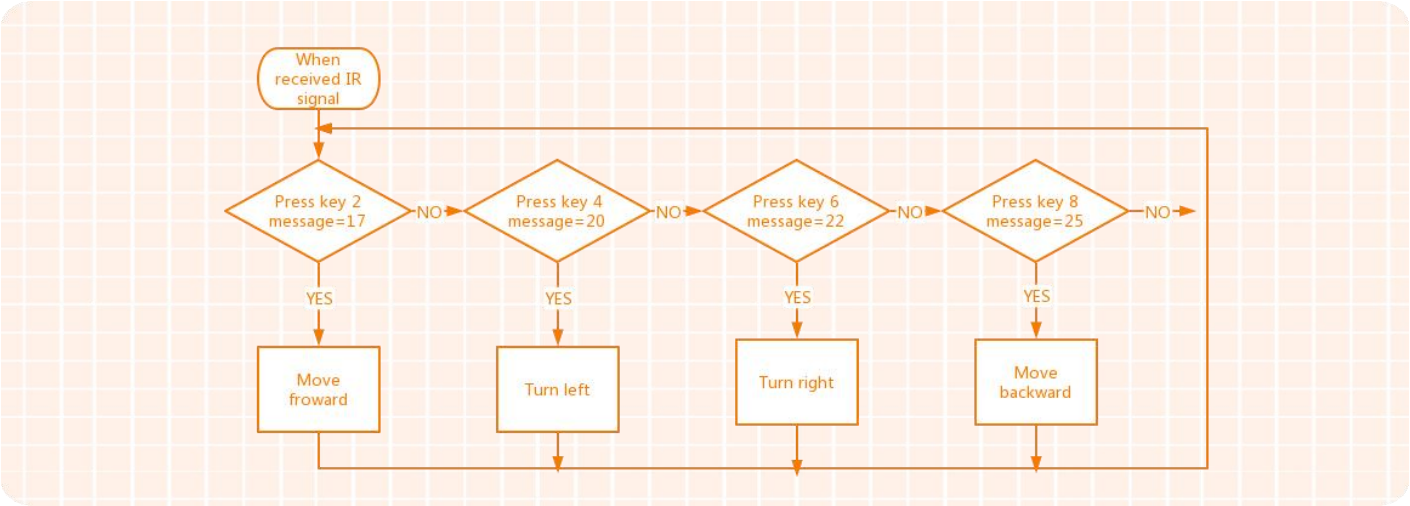
What is a function?

A function, also known as procedure or subroutine, can be defined as the organized block of reusable code which can be called whenever required. Generally, a large program can be divided into many basic building blocks and each block can realize a specified function. A function can be called multiple times by other main functions and sub-functions, which not only reduces the workload of rewriting the program segment, but also improves the utilization of the program.

4. Create functions "Move backward", "Turn left" and "Turn right" in the same way above. The customized function will be shown in the editing section and the command block section.



5. How can we use the keys 2, 4, 6, and 8 on the IR remote controller to operate our Maqueen Plus V2 car? Let's draw a flowchart to analyze this question:

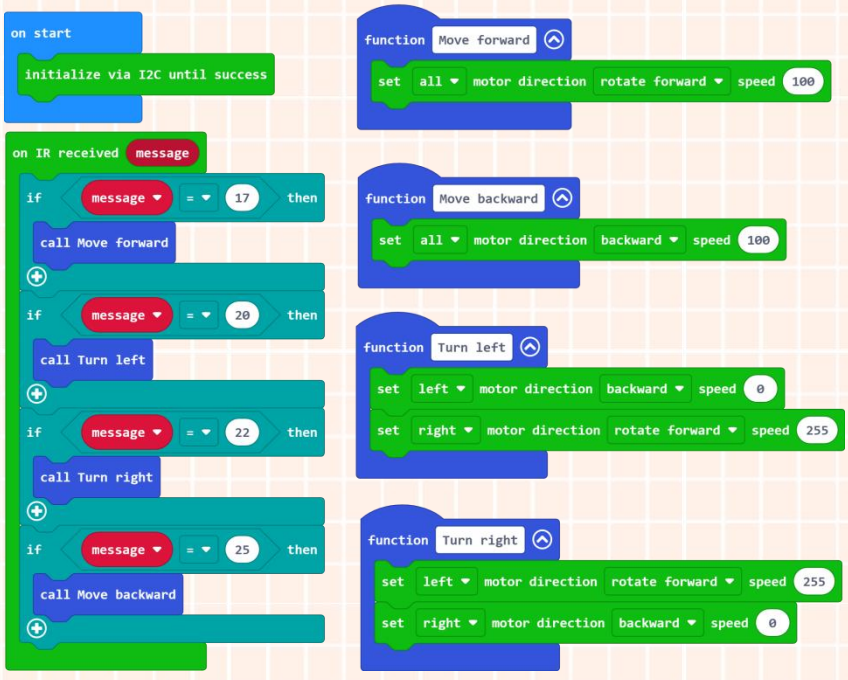


6. The flowchart above shows that we have to press the related key first, and then judge if the key value meets the condition. When the key value "message=17", call the function "Move forward"; when "message=20", call "turn left"; when "message=22", call "turn right"; when "message=25", call "move backward".

7. The above program is just a framework. Detailed operations need to be implemented in the functions. For example, press key 2, and the car moves forward. How to achieve that? Well, it's easy, just add a motor control block inside the move forward function.

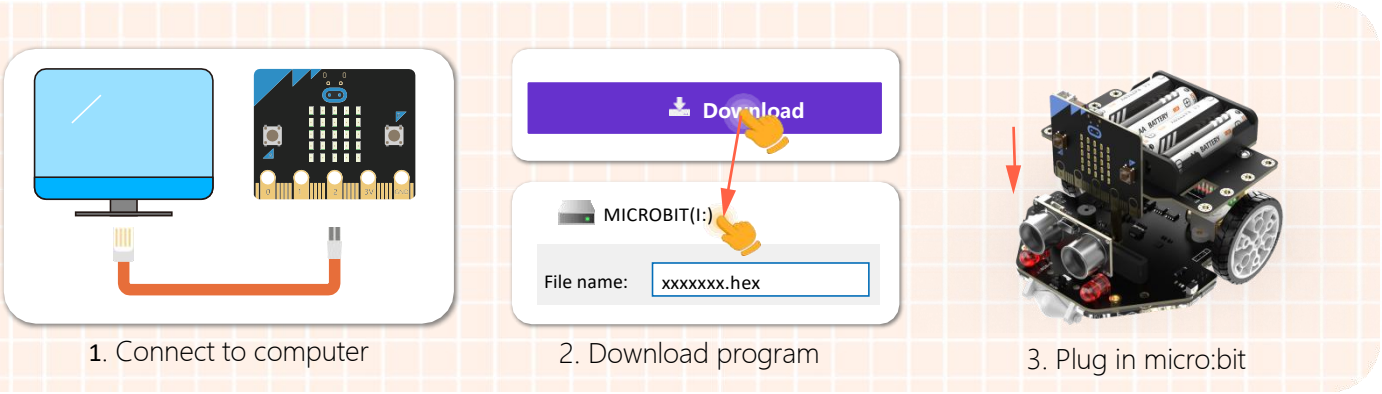
As long as we have a clear logic, the realization of the program will be not so hard. Try completing the program!

8. The complete program is shown below.



9. Name your project "IR-controlled robot" and save it.

Step 3 Download Program



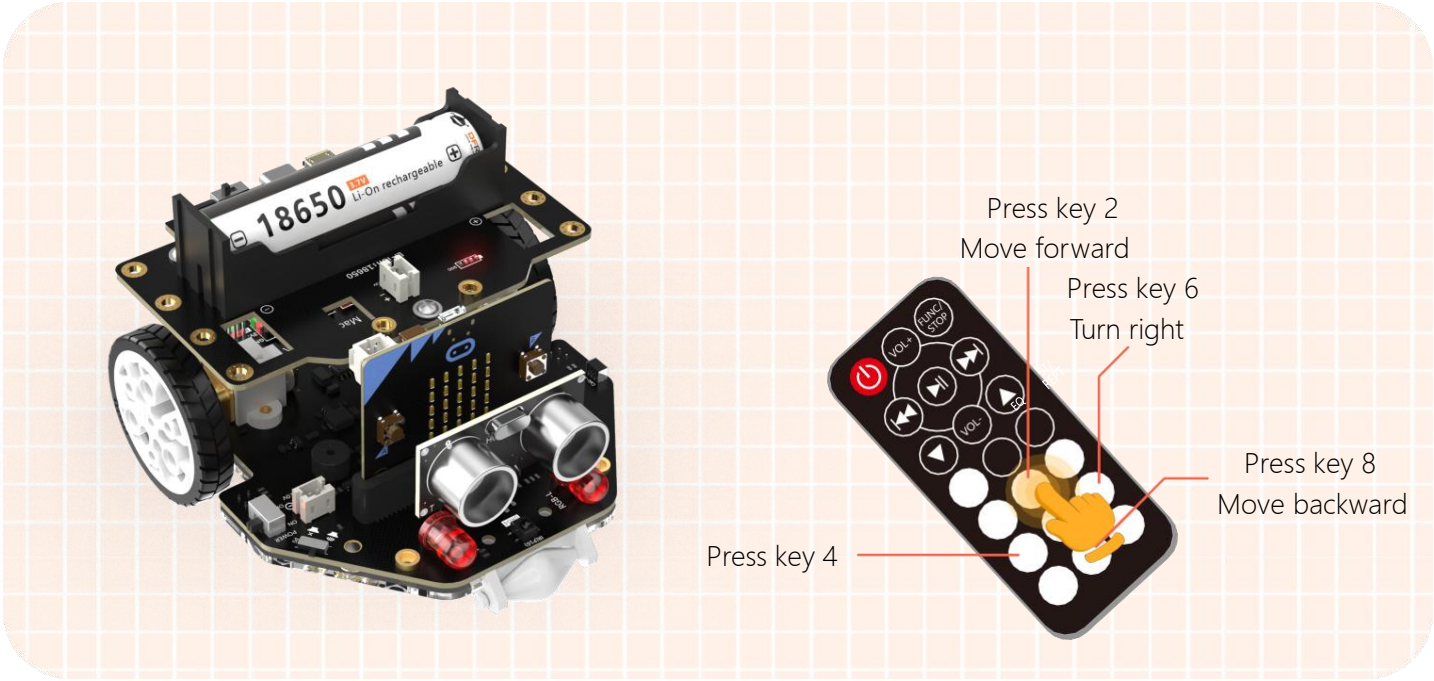
1. Connect to computer

2. Download program

3. Plug in micro:bit

Step 4 Effect Display

After completing all the above steps, use the IR remote controller to operate our Maqueen Plus V2!

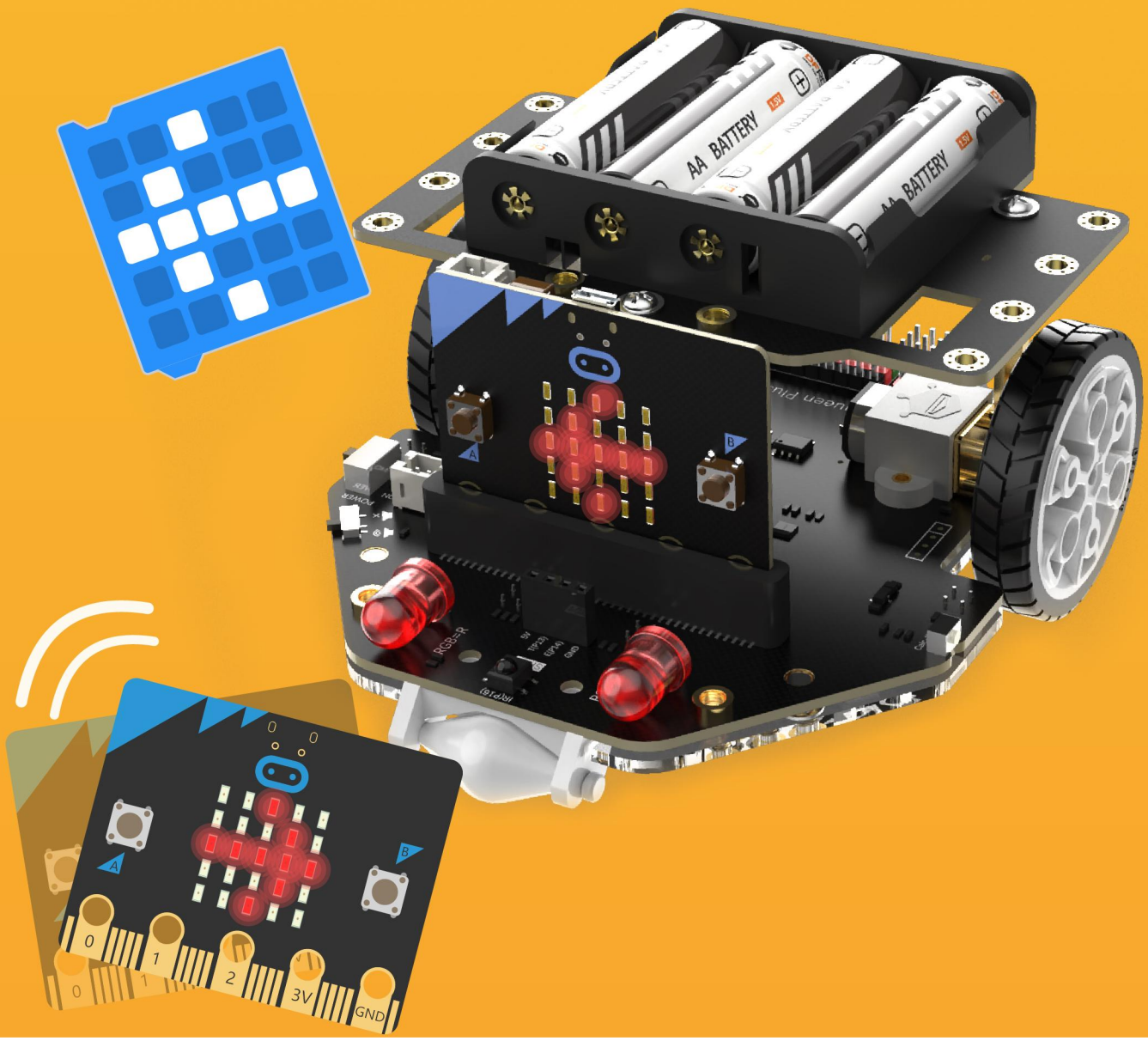


Think & Explore

We may find that once we enabled the Maqueen Plus V2 to move, it won't stop until we turn off its power switch, which could be inconvenient for us to operate.

Now here is a task for you: use the red button on the remote controller to make Maqueen Plus V2 stop.

Tip: create a function to stop the motor!



Chapter 12

Motion Sensing Robot

Have you ever tried motion-sensing games, like racing cars? This kind of game requires us to control the movement of the car on the screen by changing the direction of the remote controller. It is extremely exciting! Our Maqueen Plus V2 can also realize motion-sensing since the accelerometer on the micro:bit can detect the orientation of the board. With accelerometer and radio communication functions, we can make a similar motion-sensing game using Maqueen Plus V2.

Goal

1.The basics of radio communication

2.Learn how to use the accelerometer sensor

Electronic Component

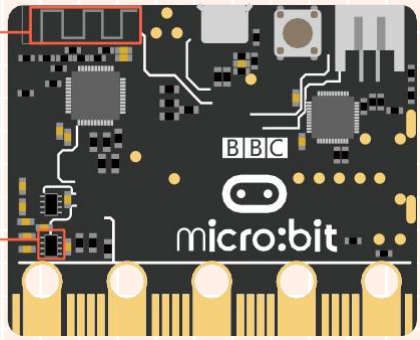
Figure of the Bluetooth and the acceleration sensor

Bluetooth

Allow two or more micro:bits communicate with each other, receive and send radio signals.

Accelerometer

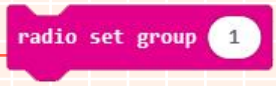
The accelerometer on the micro:bit detects the acceleration in 3 planes: x, y and z.



Command Learning

Block Brief

Radio set group



Sets the group ID for radio communications. A micro:bit can only listen to one group ID at any time.

Radio send number

Broadcasts a number via radio to any connected micro:bit in the group.

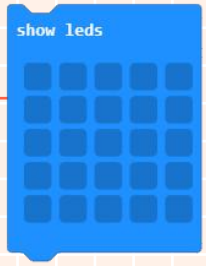
Radio received

Register code to run when the radio receives a number.

Gesture

Do something when a gesture is done (11 gestures).

Show LEDs



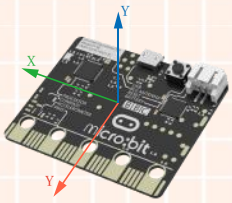
Draw an image on the LED screen.

What is an accelerometer sensor?

An accelerometer is a device that can detect the change in an object's acceleration. The onboard accelerometer of micro:bit can sense the motion of the micro:bit, including its movement, angle, gesture, and so on.

Introduction of the acceleration sensor

The micro: bit comes with a three-axis acceleration sensor that can detect gravity accelerations in three directions: x, y, and z. The measured value on each axis should be positive or negative. When the reading of one axis is 0, it indicates that the acceleration sensor is parallel to that axis. The different attitudes of the micro: bit is determined by calculating the vector sum of the three axes x, y, and z of the acceleration sensor. A vector is a quantity that has magnitude and direction, and vector sums refer to the sum of direction and magnitude.



Hands-on Practice

Motion Sensing Robot - Transmitting End

Before we start programming, let's analyze how to use the accelerometer sensor in this project.

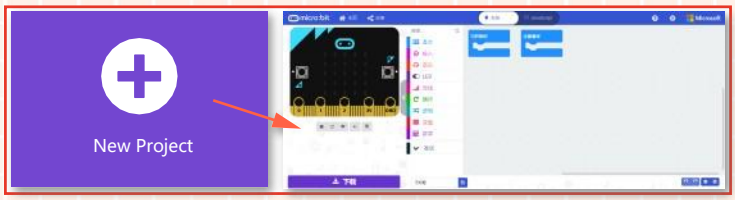
The movement of the robot car is controlled by the gesture of micro:bit. When micro:bit logo faces up, an "Up arrow" shows on the LED screen, and then the car moves forward; When logo down, show "Down arrow" and the car moves backward; tilt left, show "left arrow" and the car turns left; tilt right, show "right arrow" and the car turns right.

Tip: to realize radio communication, we need two micro:bits here. One for transmitting signal (Transmitting end), one for receiving signals (Receiving end).

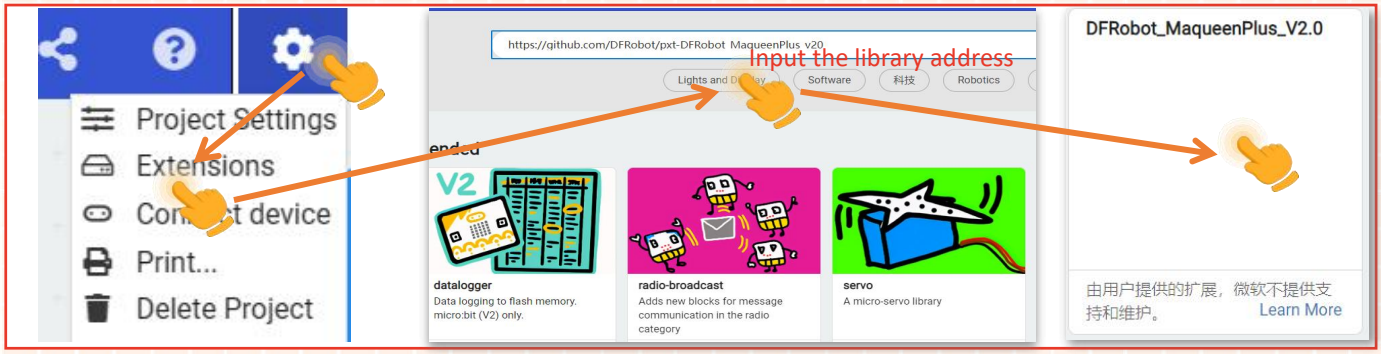
Step 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20

1. Enter MakeCode editor



2. Enter programming interface

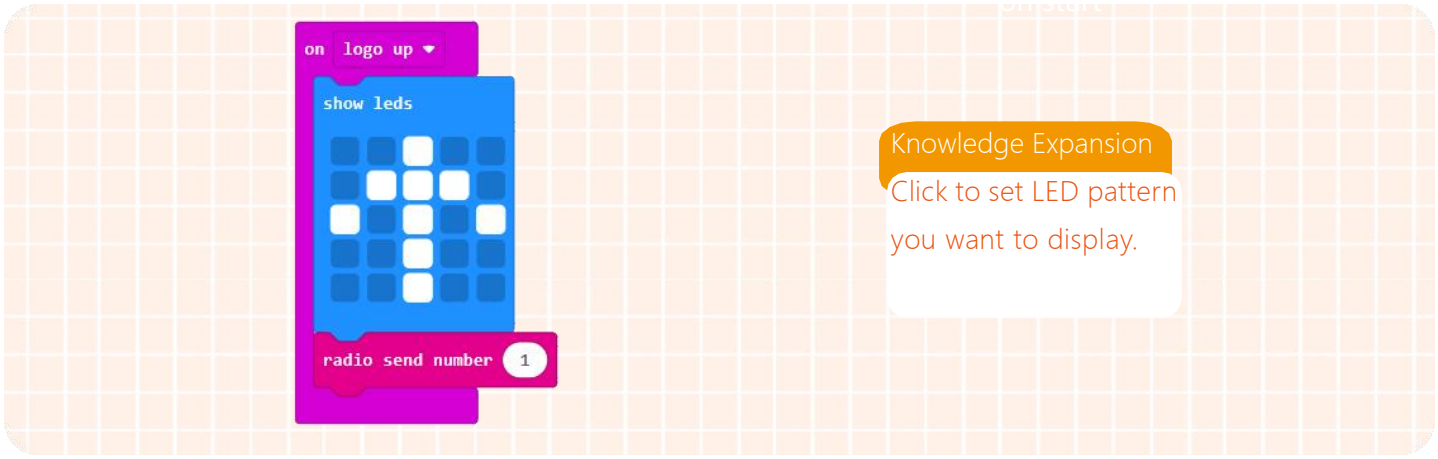


3. Add the extension library

Step 2 Programming

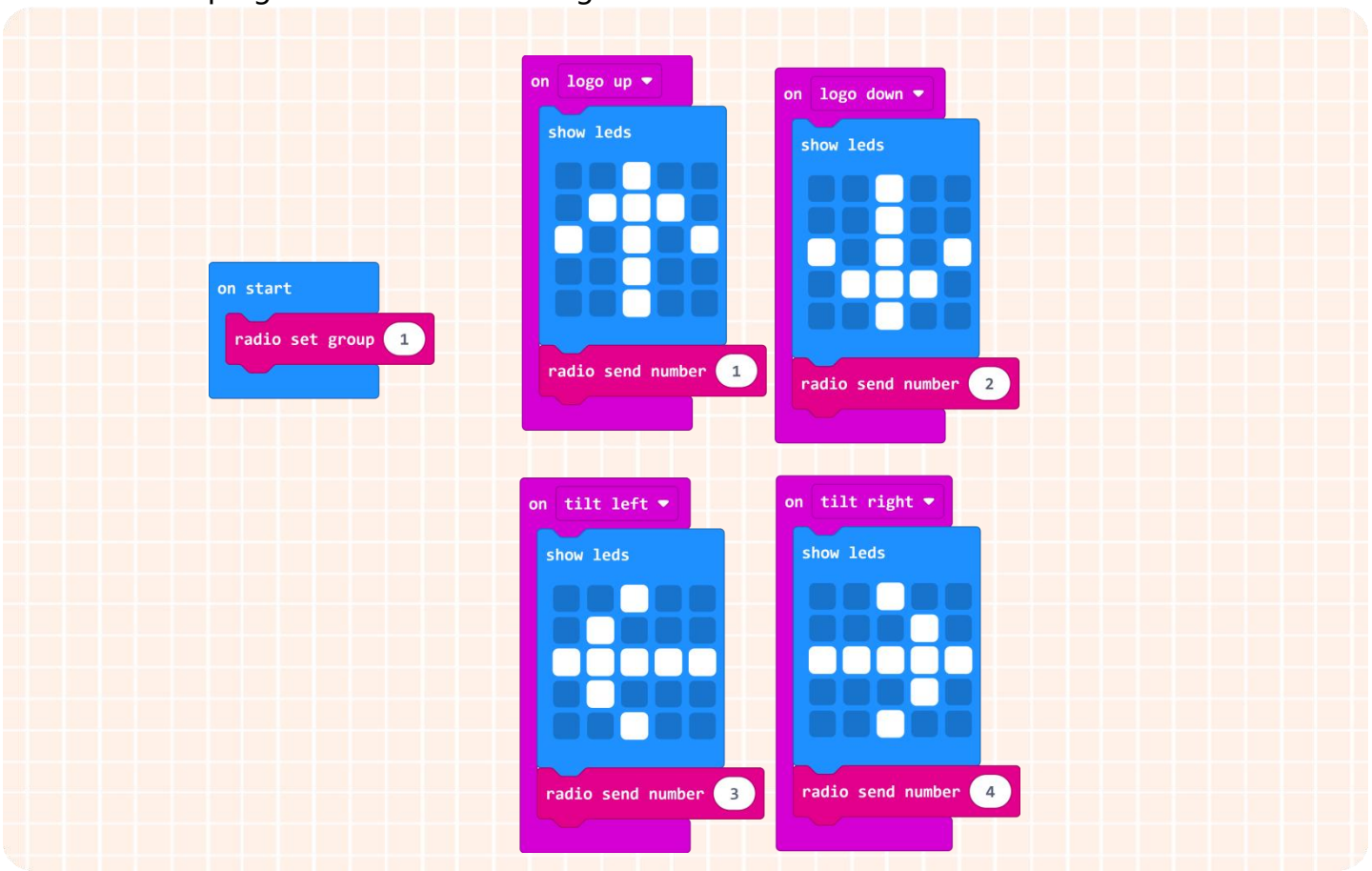
1. Enable the radio communication, and set the radio group to 1.

2. When the micro:bit logo is up, show an "Up arrow" on its LED screen, and send data 1 via radio to the Maqueen Plus V2.



The programs for the rest actions of micro:bit works in the same way. (Note: The numbers in the "radio send number" block can't be the same.)

3. The whole program for the transmitting end is shown below:

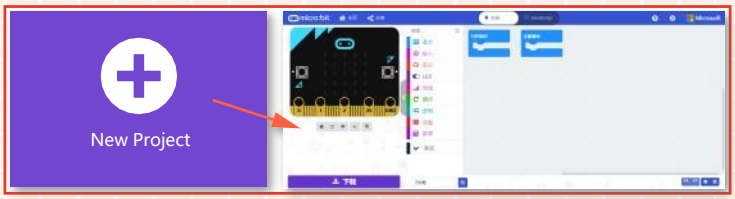


4. Name your project "Motion sensing robot-Transmitting end" and download it into the micro:bit of the transmitting end.

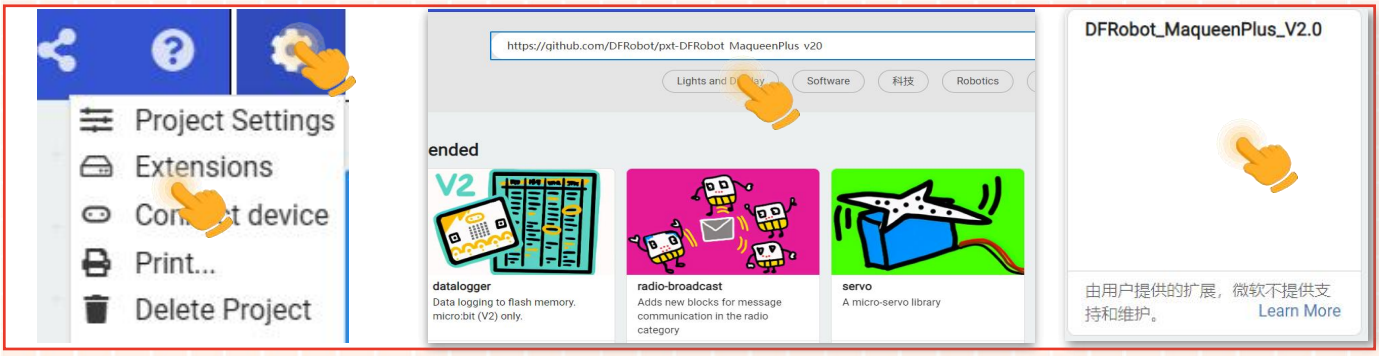
Motion Sensing- Receiving End:

Step 1 Create a new project

1. Enter MakeCode editor



2. Enter programming interface

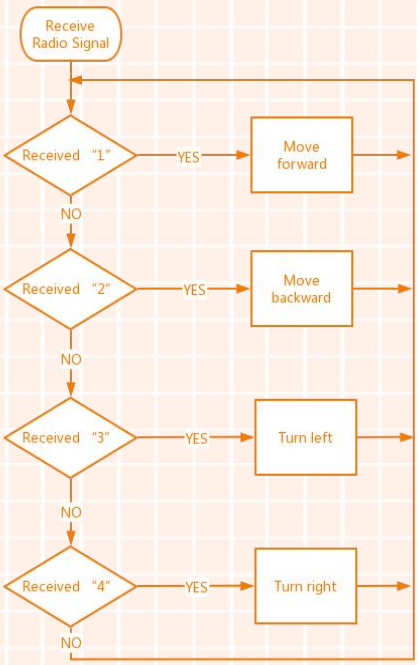


3. Add the extension library

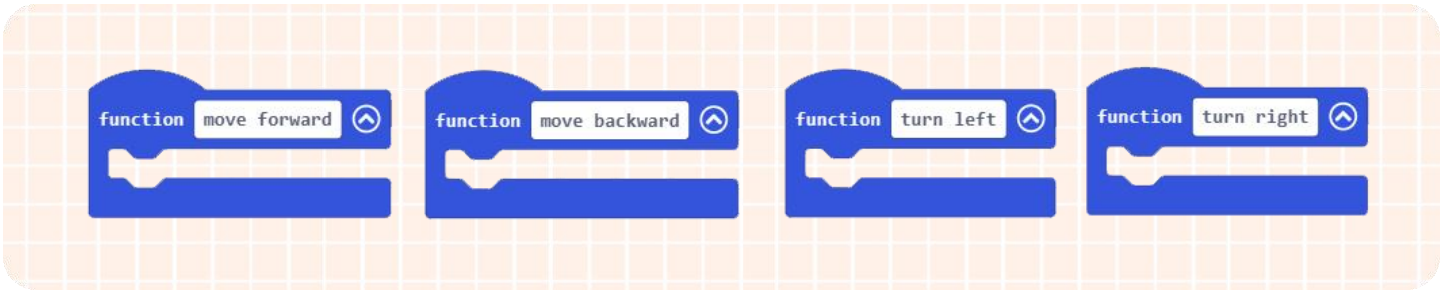
Step 2 Programming

1. Enable radio communication function, and set the radio group to 1. (The radio group of the transmitting end should be the same as that of the receiving end.)

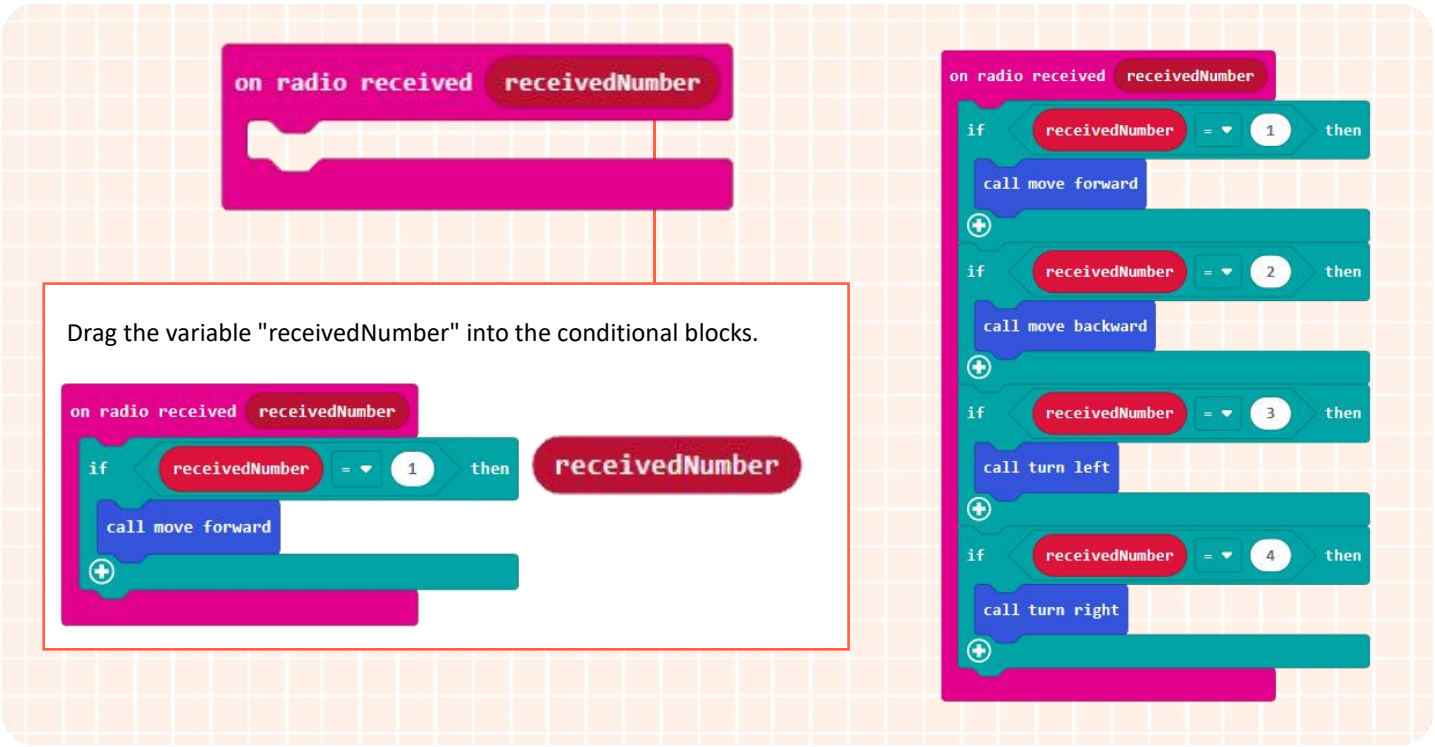
2. Program Maqueen Plus V2 to react according to the received signal.



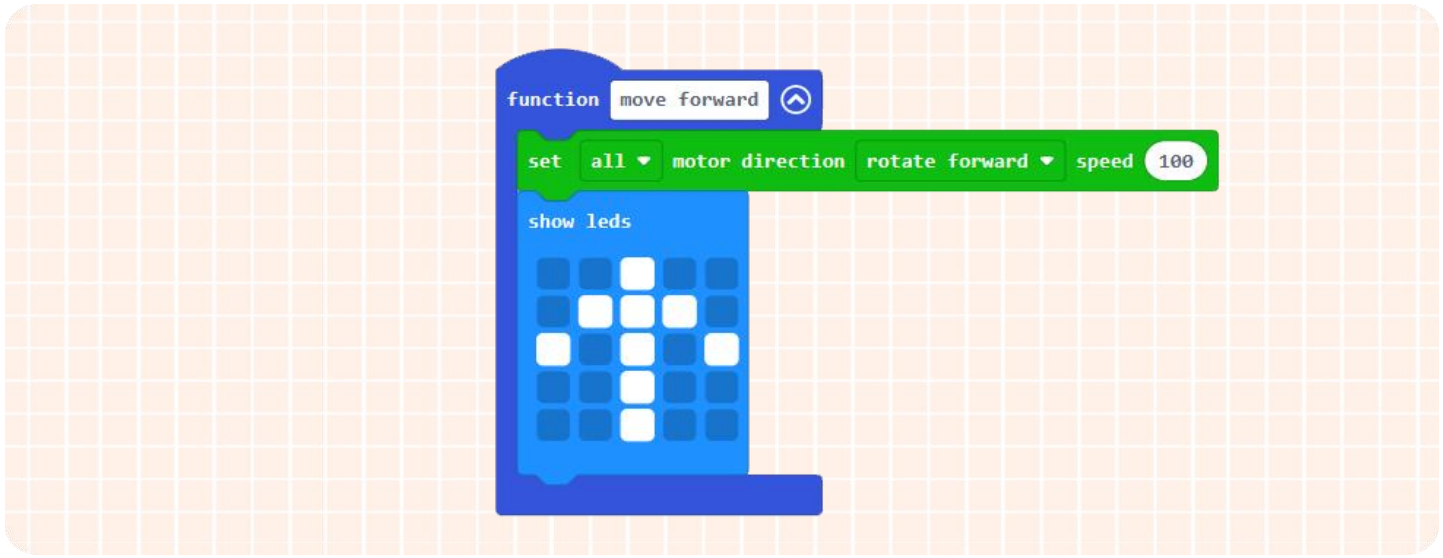
3. Create four functions "move forward", "move backward", "Turn left", and "Turn right".



4. We have to use radio receive block to control the Maqueen Plus V2 to do different reactions according to the received data.

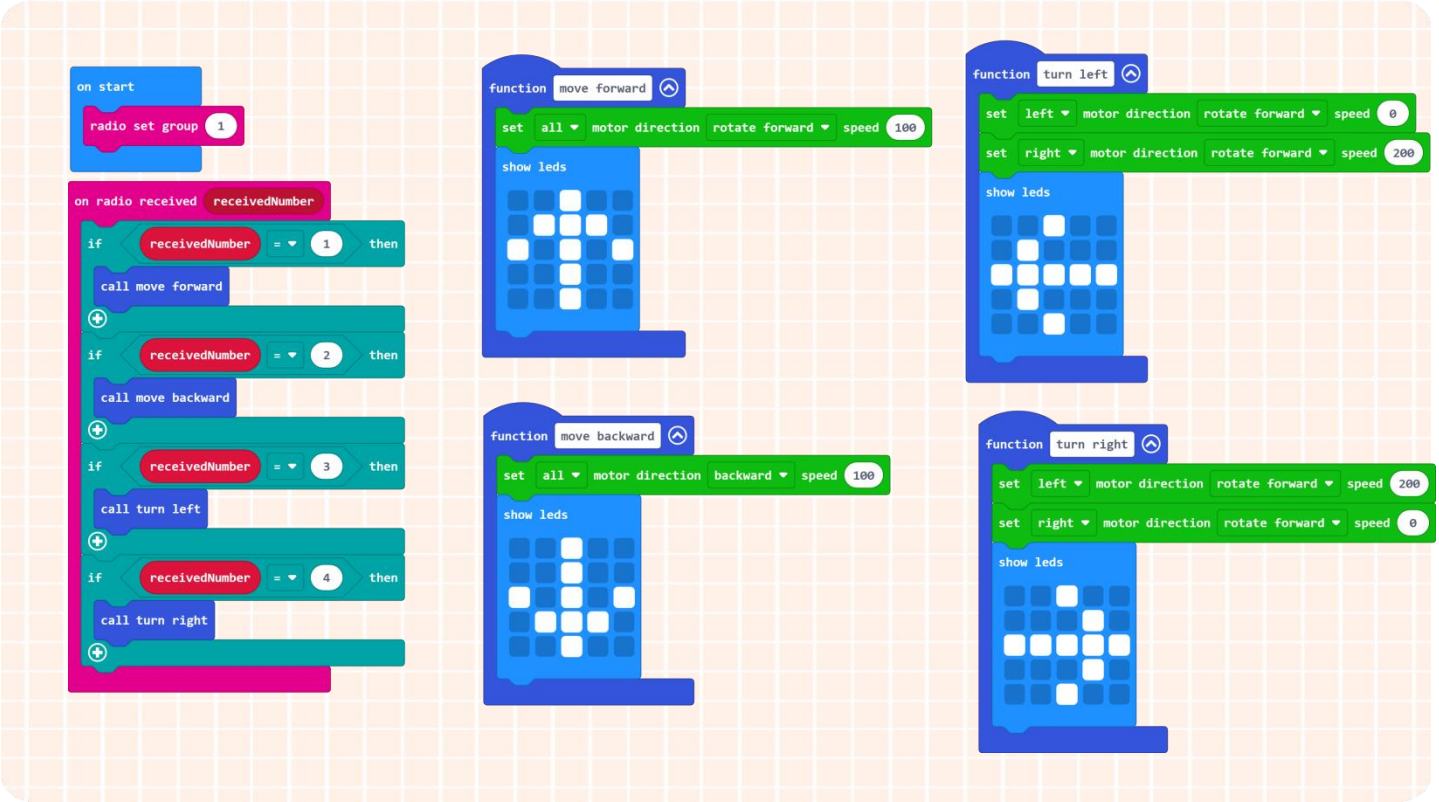


5. Take the function "move forward" as an example. When the Maqueen Plus V2 car moves forward, we let its LED screen display an "Up arrow".



Please complete the rest functions by yourself.

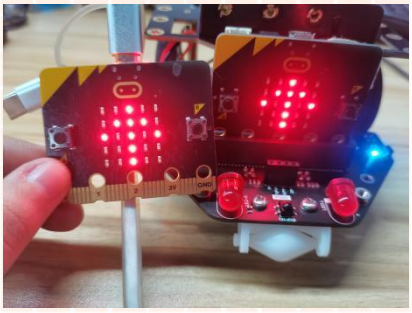
6. The program of the receiving end is shown below:



7. Name your project "Motion sensing robot-Receiving end" and then download the program to the micro:bit in Maqueen Plus V2.

Effect Display

When completed all the above steps, turn Maqueen Plus V2's power switch on, then we can use the micro:bit of the transmitting end to control our car. Give it a go!

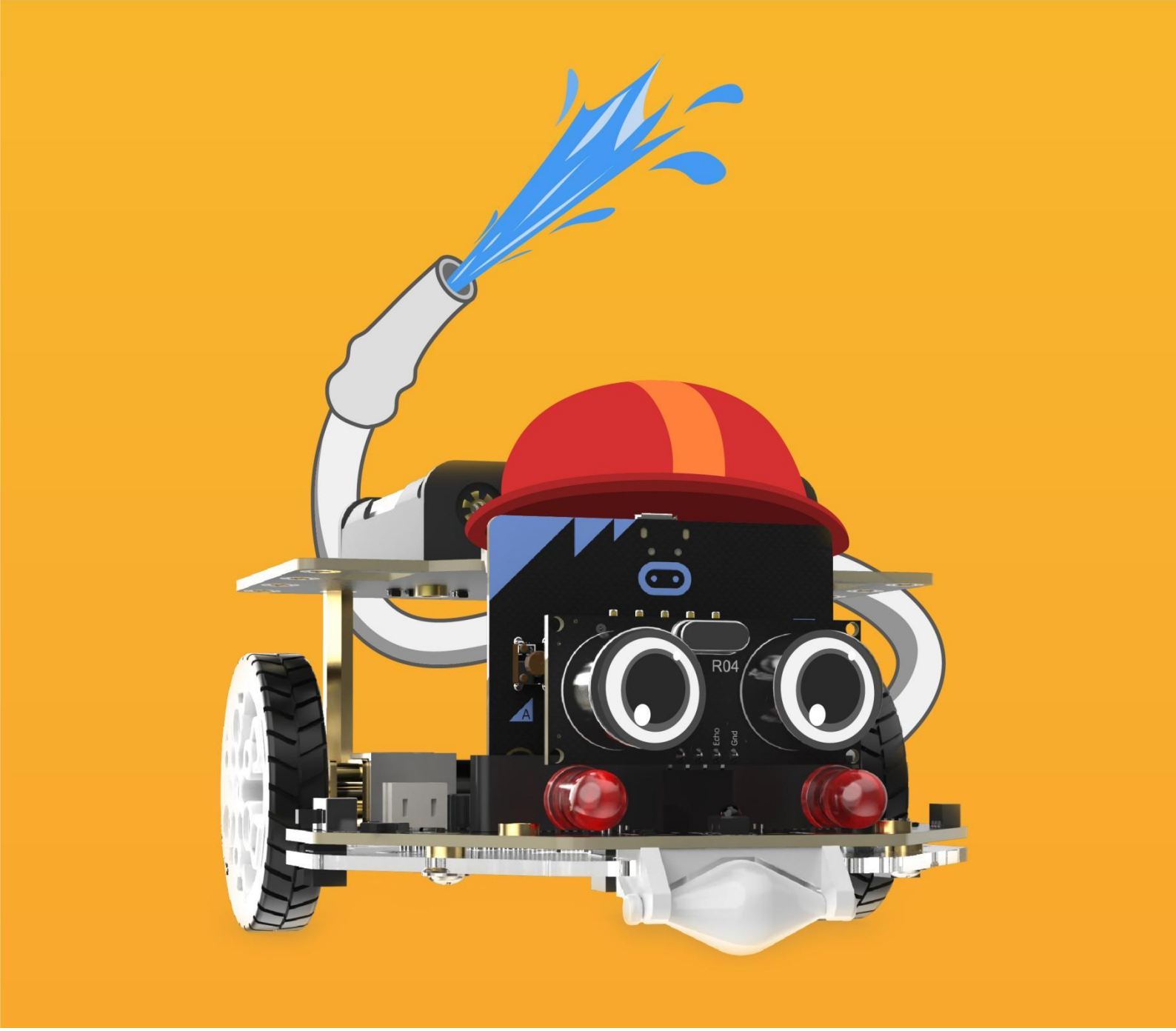


Move forward!

Note: only when powered on can the micro:bit board work, so the micro:bit of transmitting end should be always connected to a computer during operation.

Think & Explore

When playing a motion-sensing game, the larger angle we turn, the larger angle the object moves on the screen. How do we achieve this on our Maqueen Plus V2?



Chapter 13

Firefighting Robot

Firefighting is a highly dangerous occupation. When an emergency response call comes, firefighters must get themselves dressed in the appropriate gear and onto the emergency vehicles quickly to arrive at the fire scene, and then extinguish the fire. Firefighters are risking their lives to defend our cities and country. So everybody, can we use what we learned to help reduce the risk? How about a firefighting robot? Let's make a firefighting robot with Maqueen Plus V2, and let it complete three actions: Call out, Firefighting, and Mission done. A flame sensor and servo will be used in this project. Maqueen Plus V2 has 8 GPIO ports and 3 servo ports for connecting Gravity modules and servos. Everything is ready, let's start!

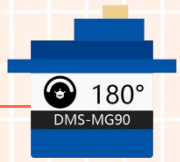
Goal

1. Learn how to drive a servo
2. Learn how to use a flame sensor
3. Learn how to use OLED display

Electronic Component

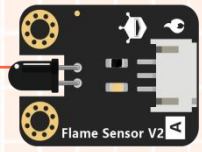
Figure of the servo and the flame sensor

Servo



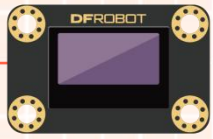
0°- 180° Micro Servo

Flame sensor



Detect fire source or other light sources of the wavelength in the range of 760nm - 1100nm

OLED-2864 Display



OLED-2864 display module can work without backlight, support IIC communication, high trans mission rate and 60Hz refresh frequency.

Command Learning

Block Brief

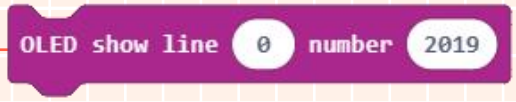
Analog read

Read analog value of the specified pin

Servo

Set the angle of servo

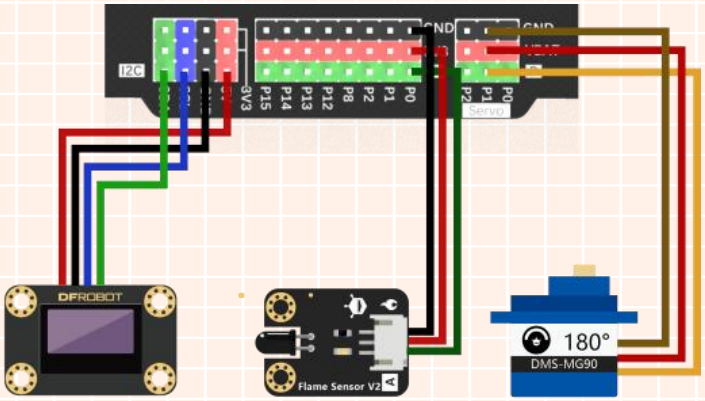
OLED-2864 Display



Display number at the n line of OLED screen

Hands-on Practic

Hardware Connection

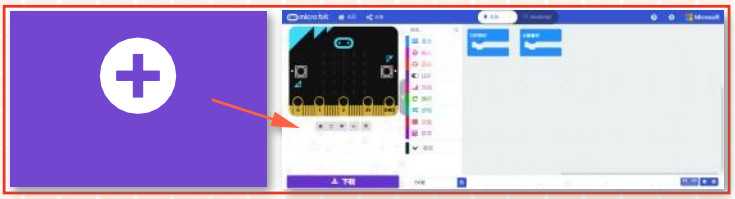


Flame sensor to P0;
Servo to the Servo P1;
OLED screen to I2C

Step 1 Create a New Project

1. Input https://makecode.microbit.org/ into your browser to enter the MakeCode editor.
2. Click "new project" to enter the MakeCode programming interface.
3. Add the Maqueen Plus V2 library: https://github.com/DFRobot/pxt-DFRobot_MaqueenPlus_v20
4. Click "Setting"->"Extensions", input the following address and click the result OLEDV1: https://github.com/DFRobot/pxt-OLED

1. Enter MakeCode editor



2. Enter programming interface

3. Add the extension library

Step 2 Programming

1. Generally, there are three parts to the firefighting process.

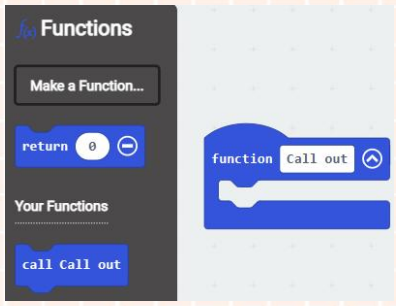
(1) Received the emergency call, and go to the fire scene;

(2) Arrived at the scene, and extinguish the fire.

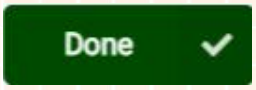
(3) Mission completed.

So, we will create three functions: "Call out", "Firefighting", and "Mission done". Take the first one as an example.

① Click "Advanced"->"Functions";



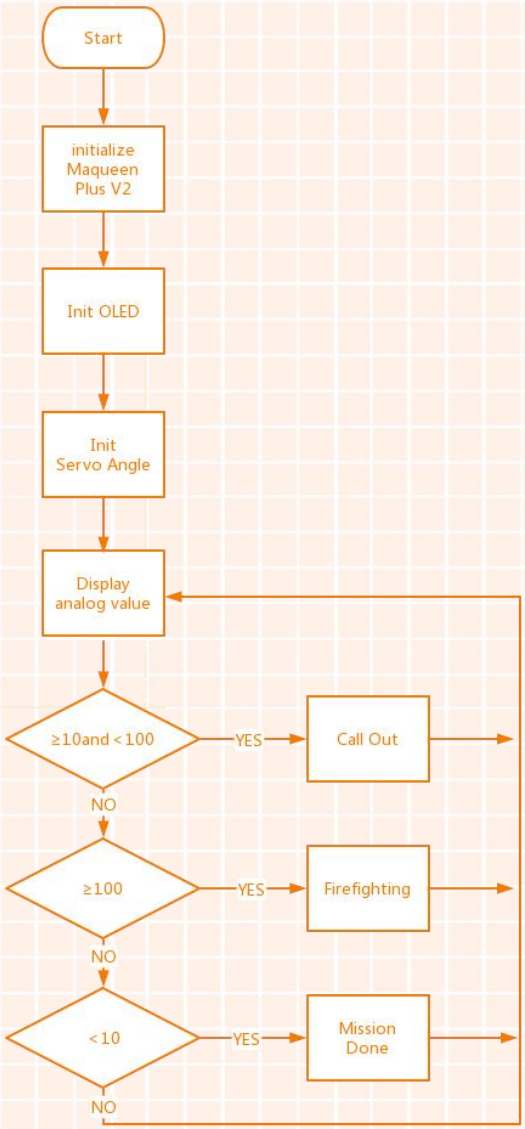
② Click "Make a Function";



③ Name the function "Call out";
④ Click "Done".

Create another two functions Firefighting and Mission done in the same way above.

2. Then do condition judgment as the flowchart shown below:



3. Initialize the Maqueen Plus V2 and OLED, and set the servo angle when the program starts.

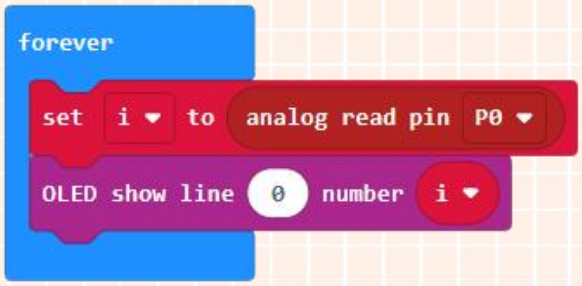
Knowledge Expansion

Initialize the servo angle to represent that the fire hose is in off state.

4. Create a variable "i" to store the analog value read from the flame sensor.

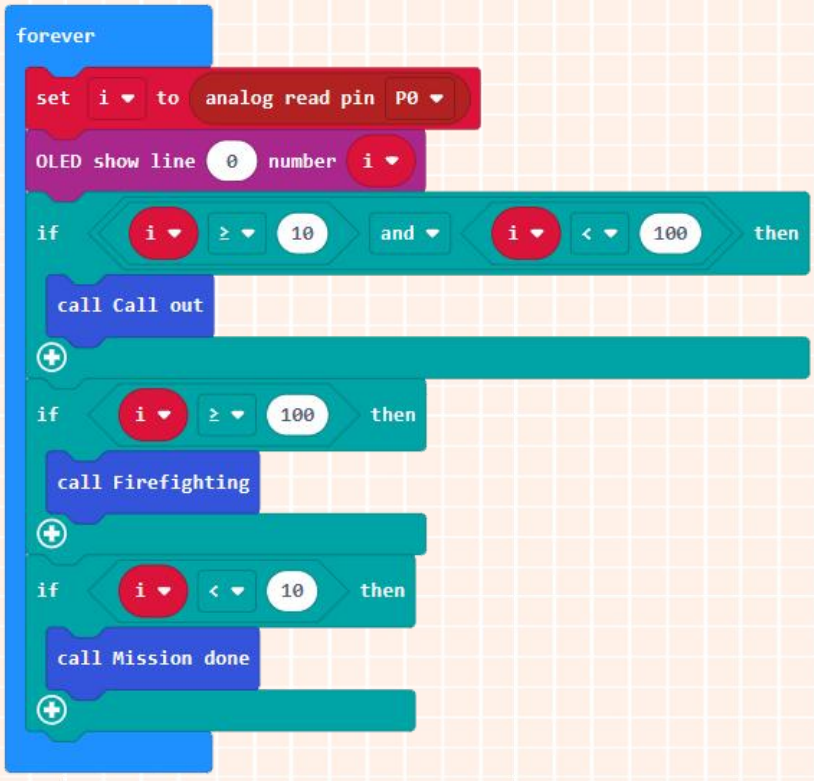
When calling the analog value of flame sensor, directly use the variable "i".

5. Display the analog value of the flame sensor on the first line of the OLED screen.

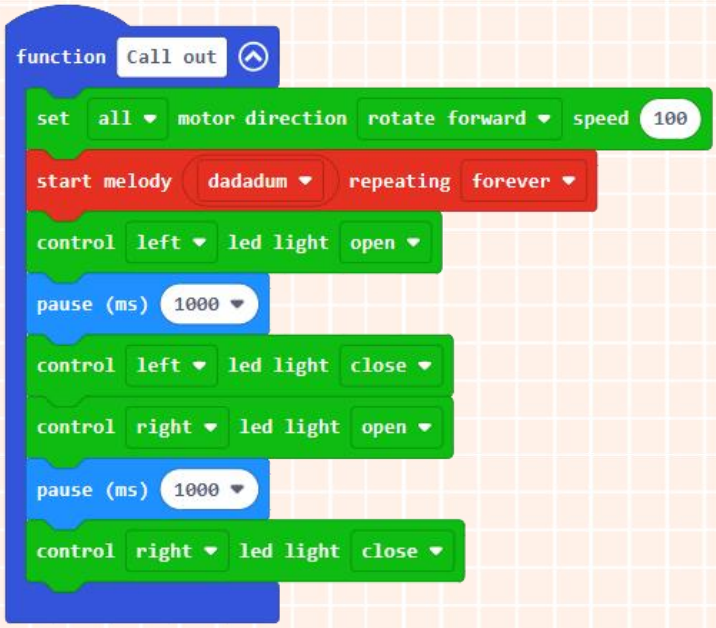


6. Call the related functions according to the program flowchart.

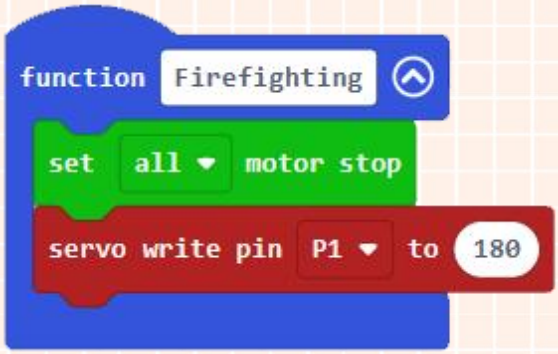
When the analog value detected by the flame sensor is between 10-100, it means that there is a certain distance between the firefighting robot and the fire scene, and the robot has to go forward to the scene; if the value is larger than 100, the robot has arrived at the fire scene, and starts to extinguish the fire; when it is less than 10, the firefighting mission is done.



7. The realization of the "Call out" function: when the firefighting robot drives to the fire scene, the two LEDs on the left and right will flash red alternatively, with the siren blaring (use the sound "dadadum" to simulate the siren).

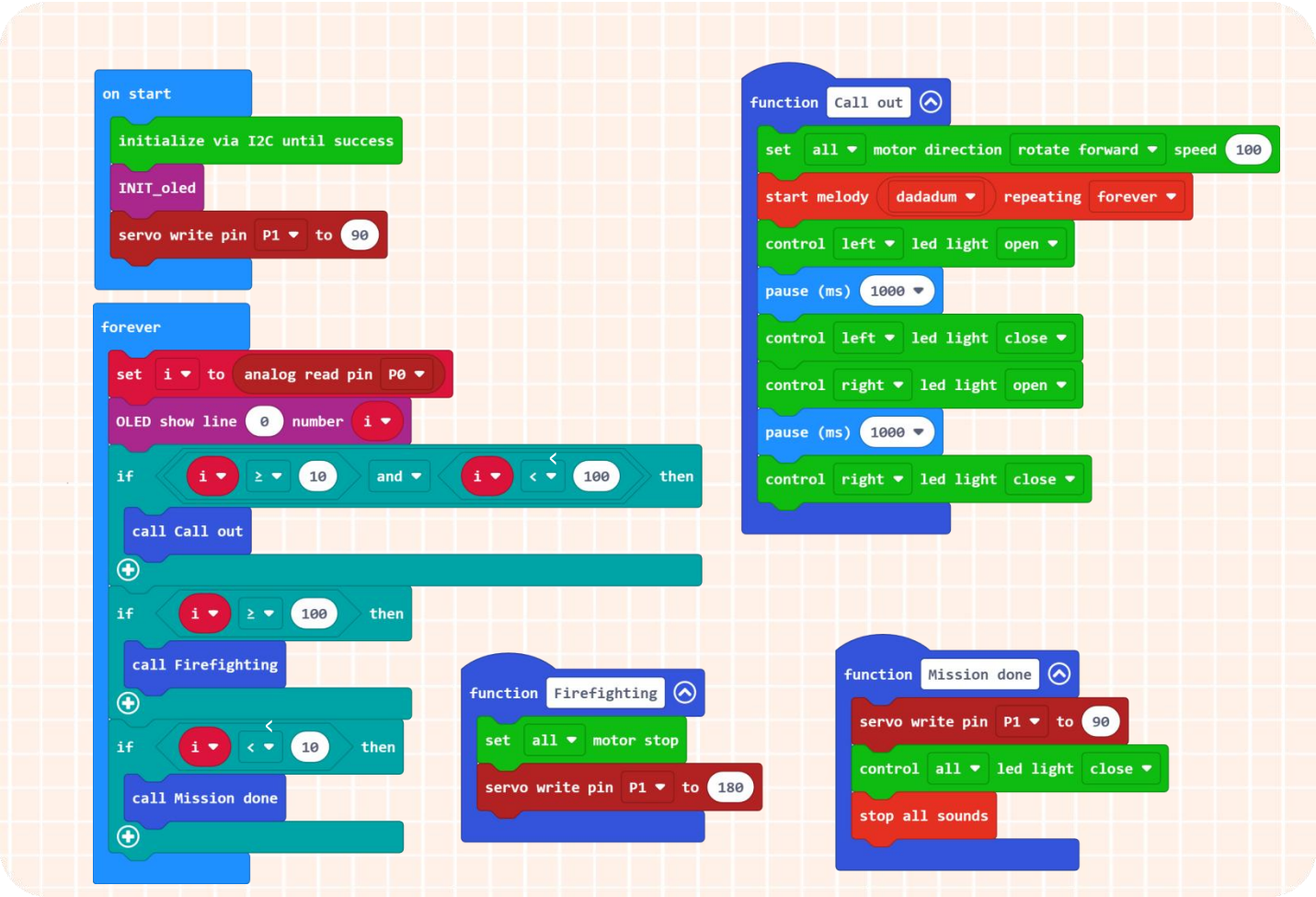


8. The realization of the function "Firefighting": when the firefighting robot arrived at the scene, turn on its fire hose to put out the fire (change the angle of the servo to simulate this process).



9. The function "Mission done": turn off the fire hose, LEDs, and siren.

10. The complete program is shown below.



11. Name your project "Firefighting robot", and download it into Maqueen Plus V2.

Step 3 Effect Display

Turn on the power switch when completed all the steps above. Then the analog value from the flame sensor will be constantly displayed on the OLED screen. When the value is in 10-100, the firefighting robot moves forward at the speed of 100 with its light flashing and siren blaring; when it is more than 100, the robot car stops, and its servo rotates 180 degrees; when less than 10, the servo back to 90 degrees, stop playing sound and turn off the RGB LEDs.

Note: we can use a lighter to imitate the fire scene. Although the flame sensor can detect fire, it is not fireproof. Please make sure the sensor always keeps a certain distance from the fire. This project involves dangerous action, please complete this chapter with the assistance of guardians or teachers.

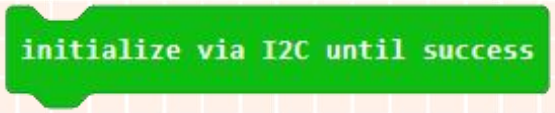
Think & Explore

We all know that smoking is not only harmful to our own health but effecting others. Therefore, smoking is forbidden in some public places. Can we make an "Anti-smoking robot" to monitor smoking in real-time?

Tip: add a gas sensor based on this project.

Appendix 1. Maqueen Plus V2 Block Description & Program Links

Technically, Maqueen Plus V2 is a device and the block is the tool to operate it, just like a TV set and remote controller. So we have to use the following blocks to make our Maqueen Plus V2 "come alive". Let's get to know how these blocks work.



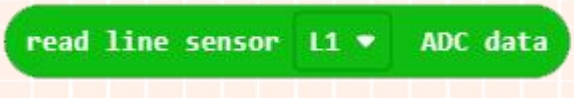
Initialize Maqueen Plus V2 to restore it to the default state

Set the direction and speed for "left/right/all" motor.
Direction: forward, backward
Speed: 0-255

Stop the "left/right/all" motor.

Turn "left/right/all" LED ON or OFF.
Color: red

Read the value returned by line-tracking sensors "L1, L2, M, R1, and R2".
Return value 1: detected black line.
Return value 0: no black line detected



Read the analog value returned by line-tracking sensors "L1, L2, M, R1, and R2"

Detect the distance between the ultrasonic sensor and the obstacle ahead.
TRIG: transmitting end; ECHO: receiving end.
Set the GPIO port corresponding to TRIG & ECHO according to hardware connection.

Set brightness of the RGB LED
Brightness Range: 0-255

Set the display color of the RGB LED, 10 colors available.

Set the display color of the RGB LED 1, 2, 3 and 4, 10 colors available.

Set RGB LED number range
LED Number: 1-4

Set RGB(Red, Green, Blue) value
Range: 0-255

Set RGB LED to display any color
Range: 1-360

Turn off all RGB LEDs

Read the current version number of Maqueen Plus V2

Read the received IR signal
Display the code value in decimal

The codes inside this block will run when IR receiver module received an external IR signal.

Program links for Maqueen Plus V2 Basic Tutorial

Chapter 2-Let's move, Maqueen!:https://makecode.microbit.org/_a0j7UdA8dHbo
Chapter 3-Walking Emoji:https://makecode.microbit.org/_AgdcJu3uKJqU
Chapter 4-City Defender-A Police Car:https://makecode.microbit.org/_V997HvFrcKm1
Chapter 5-Light Sensing Robot: https://makecode.microbit.org/_PA4XVHCPyVdL
Chapter 6-Moth Robot: https://makecode.microbit.org/_RskYed68Y0z5
Chapter 7-Little Ranging Expert: https://makecode.microbit.org/_YXFM1sXVwTeu
Chapter 8-Auto-braking System: https://makecode.microbit.org/_gcH2gf7YrDVC
Chapter 9-Line-tracking Robot: https://makecode.microbit.org/_5zW0gjTe5Dcp
Chapter 10-Tour of Crossroad: https://makecode.microbit.org/_9wg0fWLaciJo
Chapter 11-IR-controlled Robot: https://makecode.microbit.org/_4t4KibbFU83r

Chapter 12-Motion Sensing Robot-Transmitting End: https://makecode.microbit.org/_D2Eg5p2rv4K9
Chapter 12-Motion Sensing Robot-Receiving End: https://makecode.microbit.org/_6AcdRw2us2DY
Chapter 13-Firefighting Robot: https://makecode.microbit.org/_3MKY9xe5H93P