

1. Getting Started with Arduino .....	3
1.1 What is Arduino? .....	3
1.2 Installing the Arduino IDE for Windows .....	3
1.3 Install Driver for KEYESTUDIO V4.0 Control Board .....	6
1.4 Introduce of Arduino IDE 2.0 .....	9
1.5 Add Libraries to Arduino IDE .....	10
2. Adjust the Angle of the Servo Before Assembly .....	13
3. How to Assemble the Robot .....	14
3.1 Things to note before assembly: .....	15
3.2 Assembly Steps .....	16
4. Wiring .....	28
5. Parameters of the robot .....	34
6. Lessons .....	36
Lesson 1.1: LED Blinks .....	36
Lesson 1.2: Changing the blinking frequency .....	39
Lesson 2.1: Adjust the Brightness of the LED .....	40
Lesson 2.2: Slow down the change of the brightness of the LED .....	42
Lesson 3.1: Analog value of photoresistor .....	43
Lesson 3.2: Control the brightness of the LED by PWM .....	46
Lesson 4.1: Digital Value of Line Tracking Sensor .....	48
Lesson 4.2: Use Line Tracking Sensor to control LED .....	50
Lesson 5.1: Control the Servo Motor-1 .....	52
Lesson 5.2: Control the Servo Motor-2 .....	55
Lesson 6.1: Detect distance with ultrasonic module .....	56
Lesson 6.2: Use Ultrasonic Sensor to control LED .....	60

Lesson 7.1: Infrared Transmitting and Receiving .....	62
Lesson 7.2: Use infrared remote control to control LED .....	64
Lesson 8: Motor Driving and Speed Control .....	66
Lesson 9.1: 8*16 LED Dot Matrix Displays Smiley Face .....	69
Lesson 9.2: 8*16 LED Dot Matrix Displays Direction and Text .....	74
Lesson 10: Light Following Tank .....	76
Lesson 11: Ultrasonic Follow Robot .....	79
Lesson 12: Ultrasonic Obstacle Avoidance Robot .....	82
Lesson 13: Robot Restricted by a circle .....	85
Lesson 14: Line-tracking Tank .....	87
Lesson 15: IR Remote Control Tank .....	90
Lesson 16: Install and Test the Bluetooth APP .....	93
Lesson 17: Bluetooth controlled LDE .....	101
Lesson 18: Bluetooth controlled Tank .....	103
Lesson 19: Control the Move Speed of the Robot via Bluetooth .....	108
Lesson 20: Multifunctional Tank .....	110
Expand Experiments: Install flame sensors and a fan on the Robot .....	114
Lesson 21: Read the Value of Flame Sensor .....	116
Lesson 22: Flame Warning .....	118
Lesson 23: Get the Fan Rotating .....	120
Lesson 24: Turn on the fan when Robot detect a flame. ....	122
Lesson 25: Bluetooth Controlled Robot with Fan .....	123

# 1. Getting Started with Arduino

## 1.1 What is Arduino?

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by writing the program code in the IDE and sending the instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software (IDE), based on Processing.

## 1.2 Installing the Arduino IDE for Windows

1. Visit <https://www.arduino.cc/en/software> to download the latest Arduino IDE version for your computer's operating system. There are versions for Windows, Mac, and Linux systems.

### The Arduino IDE 2

The Arduino IDE 2 is a big step from its sturdy predecessor, Arduino IDE 1.x, and comes with revamped UI, improved board & library manager, debugger, autocomplete feature and much more.

Here we will show how to download and install the Arduino IDE 2.21 on your Windows

You can choose between the Installer (.exe) and the Zip packages. We suggest you use the first one that installs directly everything you need to use the Arduino Software (IDE), including the drivers. With the Zip package you need to install the drivers manually. The Zip file is also useful if you want to create a portable installation.

At the download page, click on the **"Windows Installer"** option for the easiest installation.



## Downloads



### Arduino IDE 2.2.1

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE

The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

#### DOWNLOAD OPTIONS

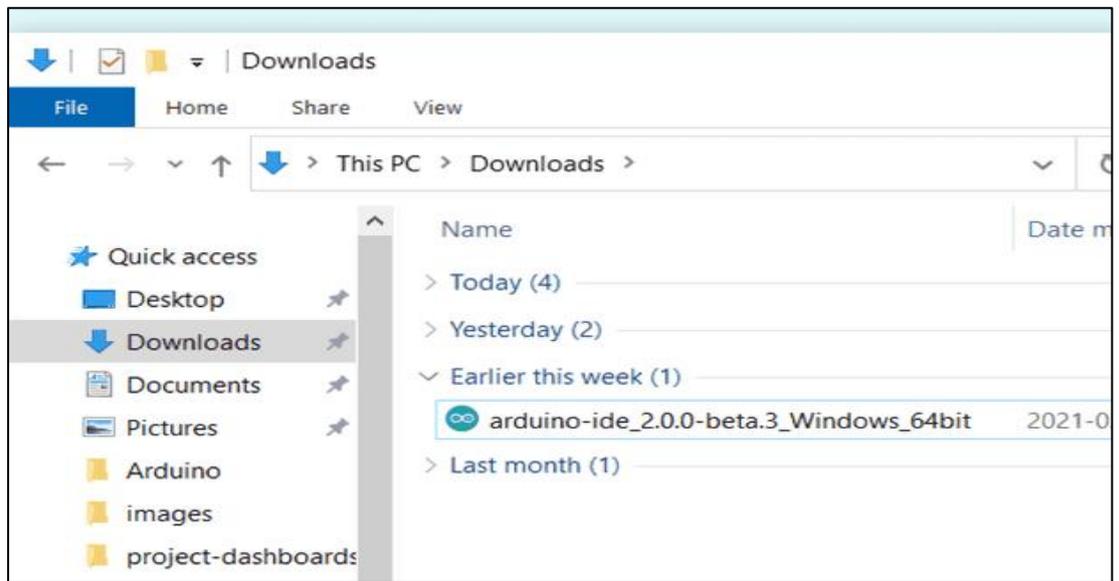
**Windows** Win 10 and newer, 64 bits  
**Windows** MSI installer  
**Windows** ZIP file

**Linux** AppImage 64 bits (X86-64)  
**Linux** ZIP file 64 bits (X86-64)

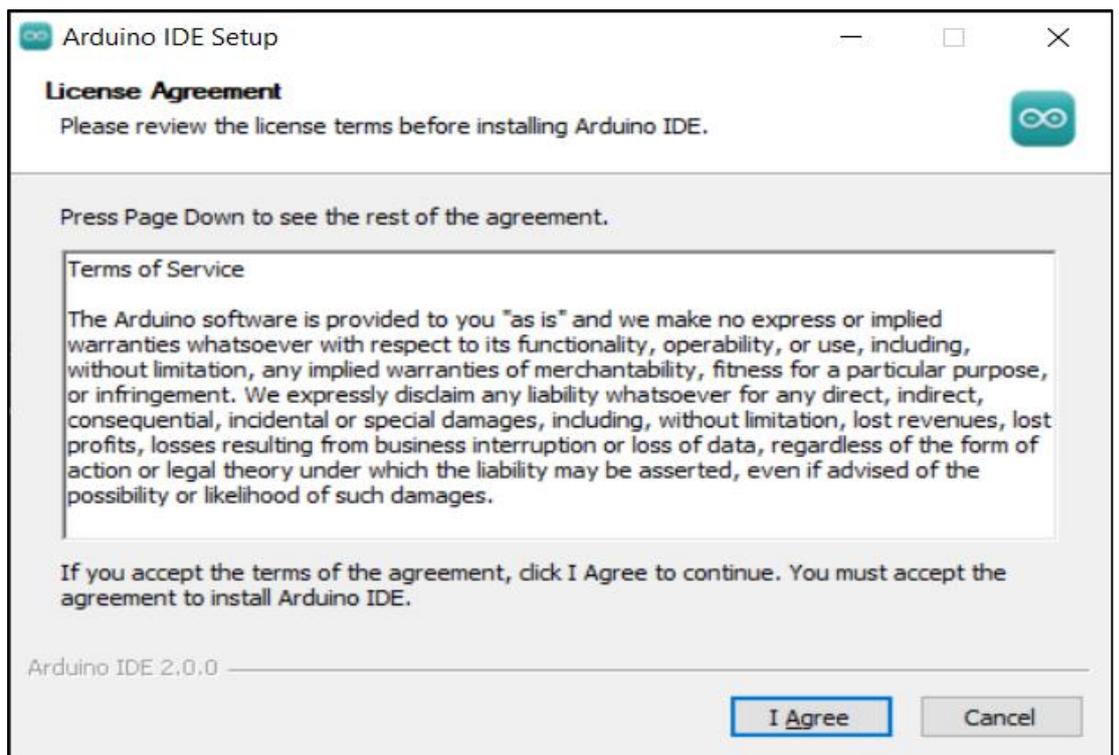
**macOS** Intel, 10.14: "Mojave" or newer, 64 bits  
**macOS** Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

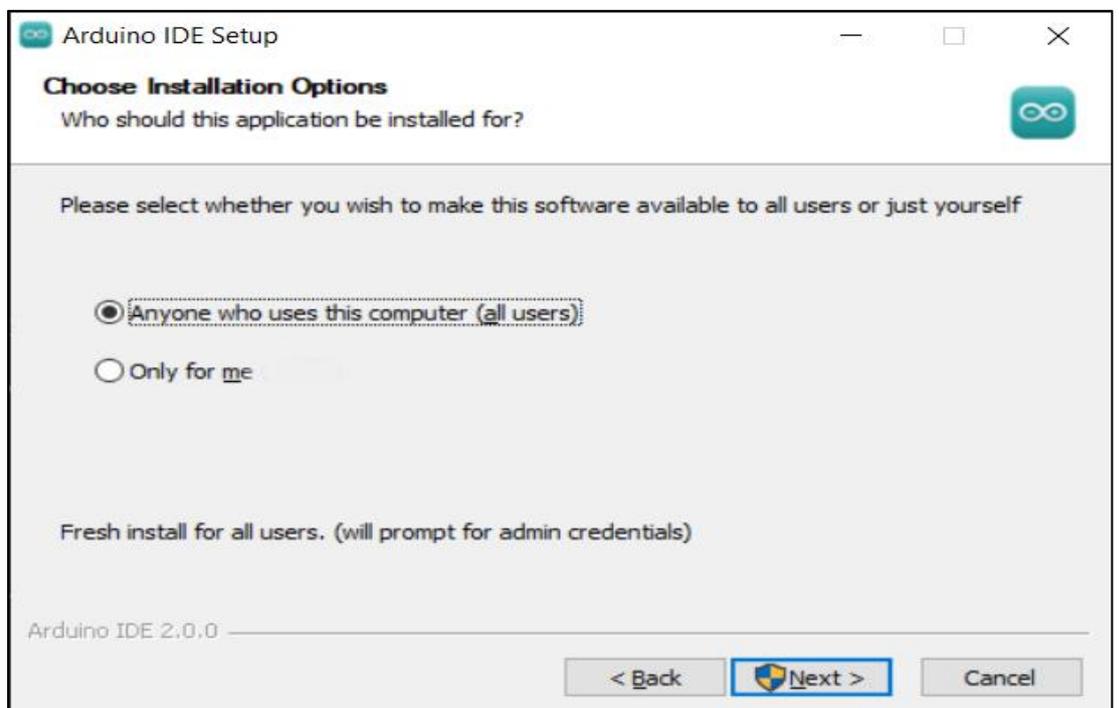
2. Save the .exe file downloaded from the software page to your hard drive and simply run the file .



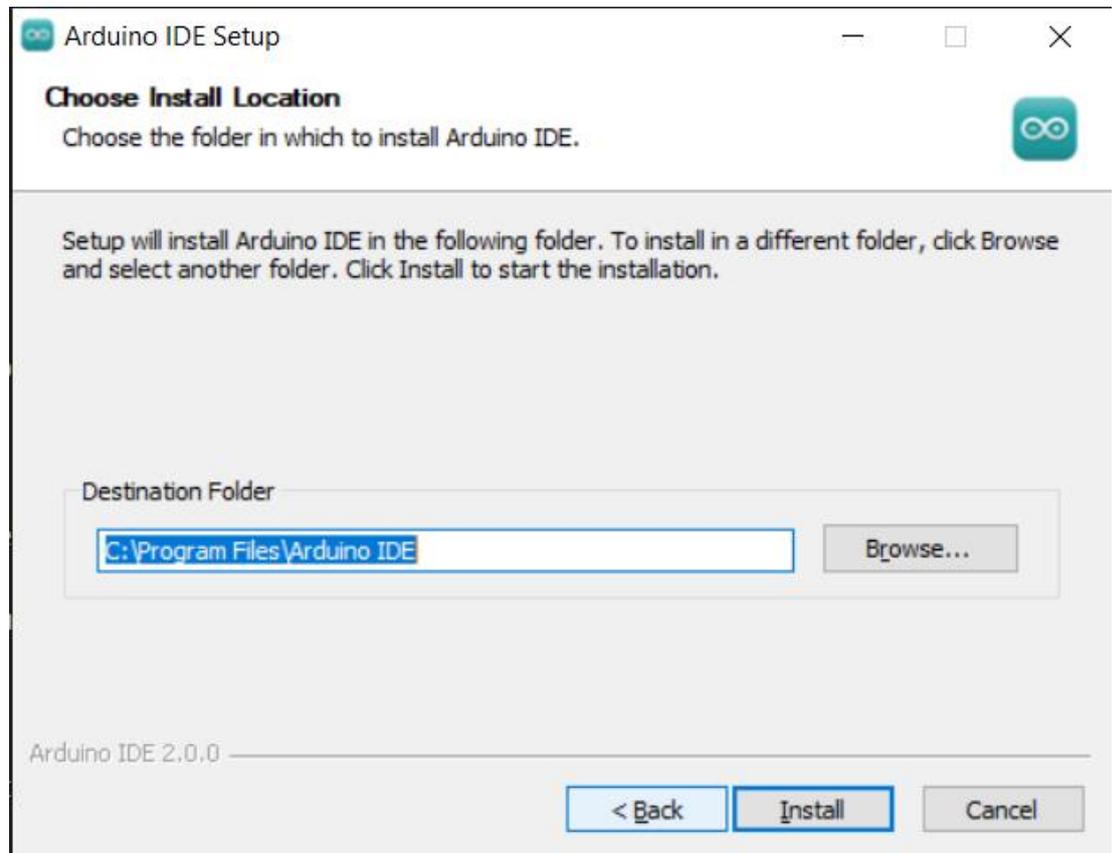
3. Read the License Agreement and agree it.



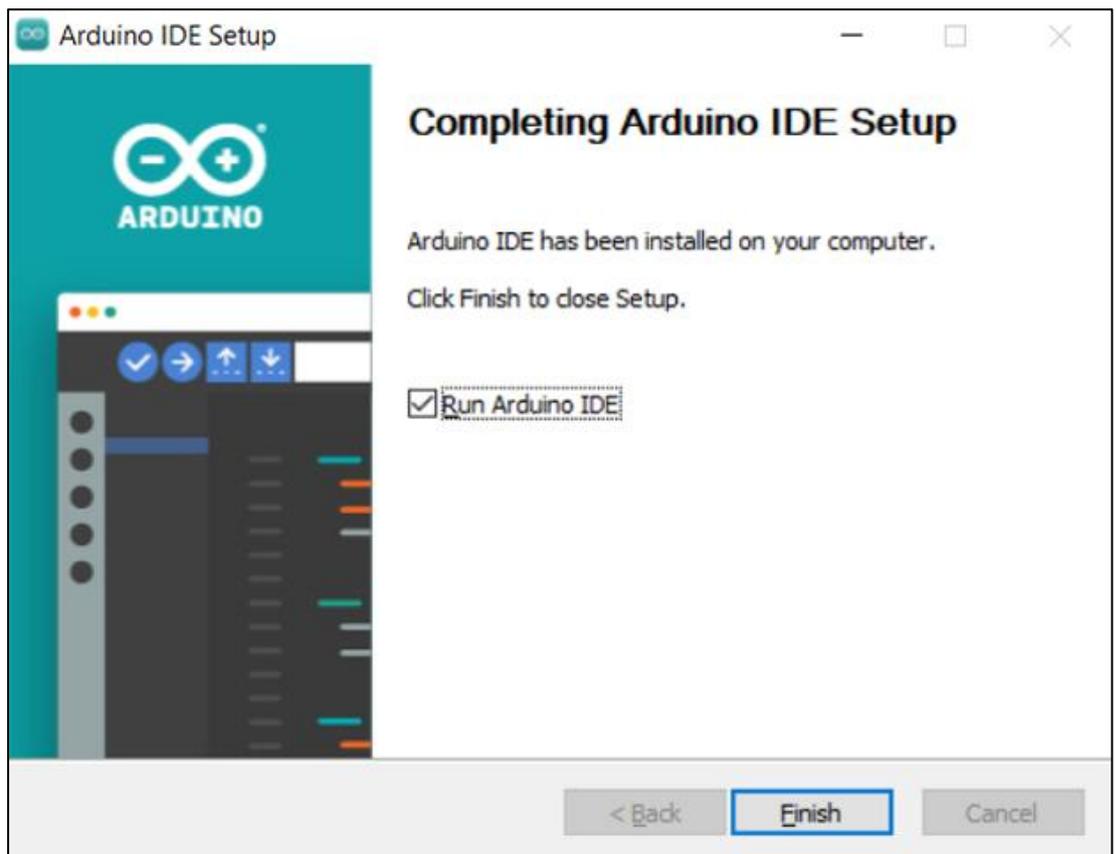
4. Choose the installation options.



5. Choose the install location.



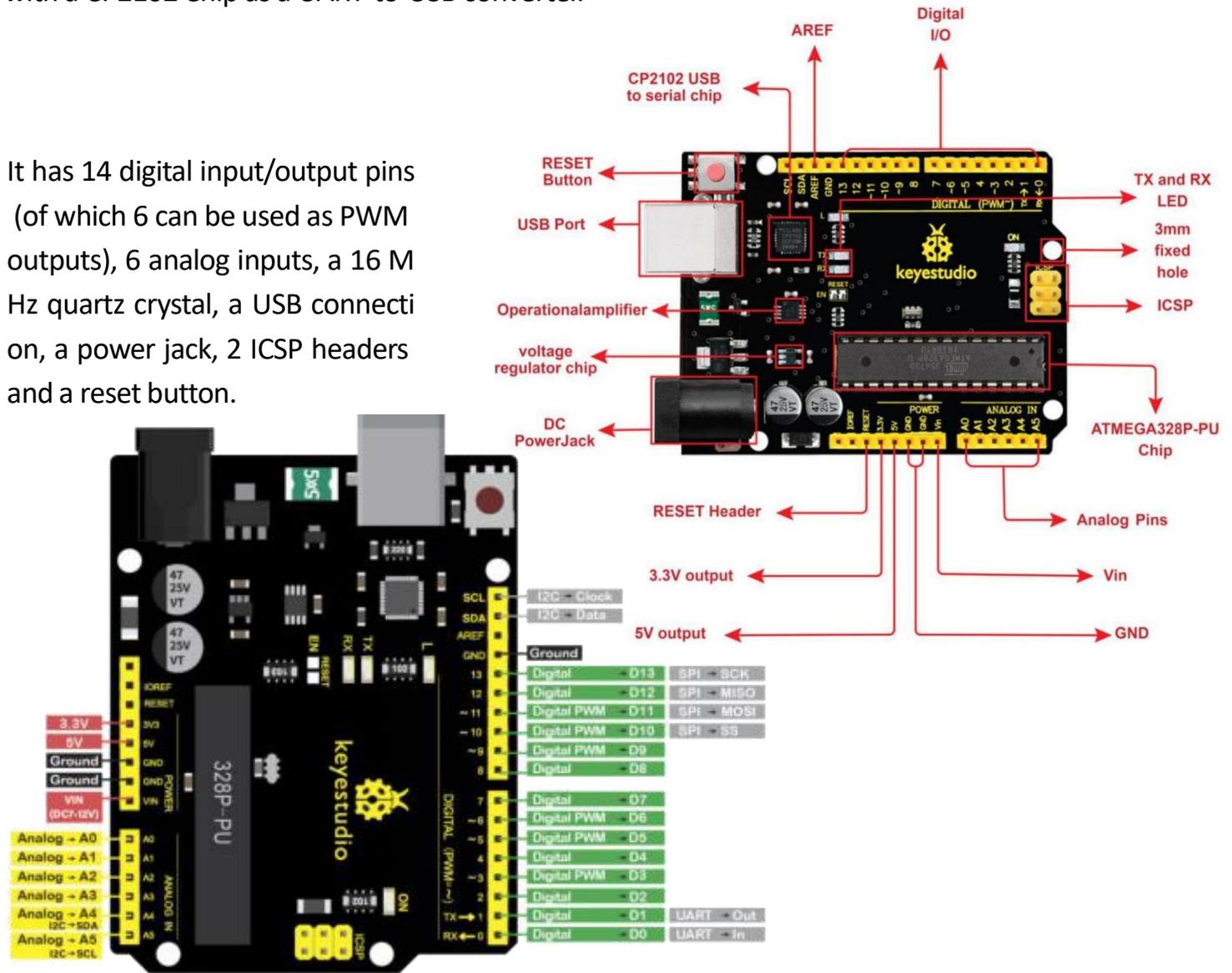
6. Click finish and run Arduino IDE



## 1.3 Install Driver for KEYESTUDIO V4.0 Control Board

KEYESTUDIO V4.0 control Board is the brain of this robot, based on ATmega328P MCU, and comes with a CP2102 Chip as a UART-to-USB converter.

It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 M Hz quartz crystal, a USB connection, a power jack, 2 ICSP headers and a reset button.



Micro controller	ATmega328P-PU
Operating Voltage	5V
Input Voltage (recommended)	DC7-12V
Digital I/O Pins	14 (D0-D13)
PWM Digital I/O Pins	6 (D3, D5, D6, D9, D10, D11)
Analog Input Pins	6 (A0-A5)
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	32 KB (ATmega328P-PU) of which 0.5 KB used by bootloader
SRAM	2 KB (ATmega328P-PU)
EEPROM	1 KB (ATmega328P-PU)
Clock Speed	16 MHz
LED_BUILTIN	D13

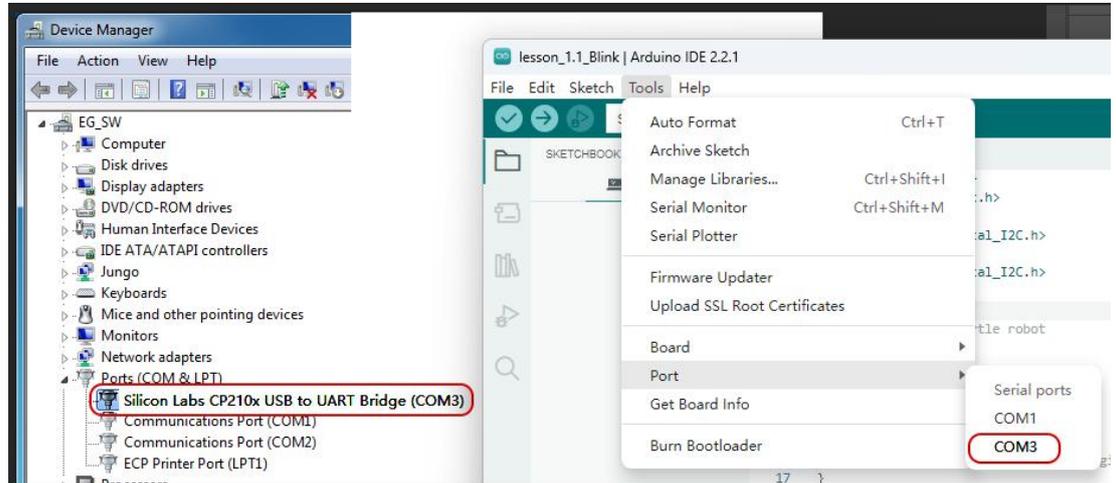
## Install driver

Plug in your board and wait for Windows to begin its driver installation process. Often CP2102 drivers will be automatically installed by your system when using Arduino. You can check the Device Manager or the port of the Arduino IDE to see if the driver is successfully installed.

Open the Device Manager by right clicking “My computer” and selecting control panel.

Look under Ports (COM & LPT). You should see an open port named Silicon Labs CP210x USB to UART Bridge (COM-X)

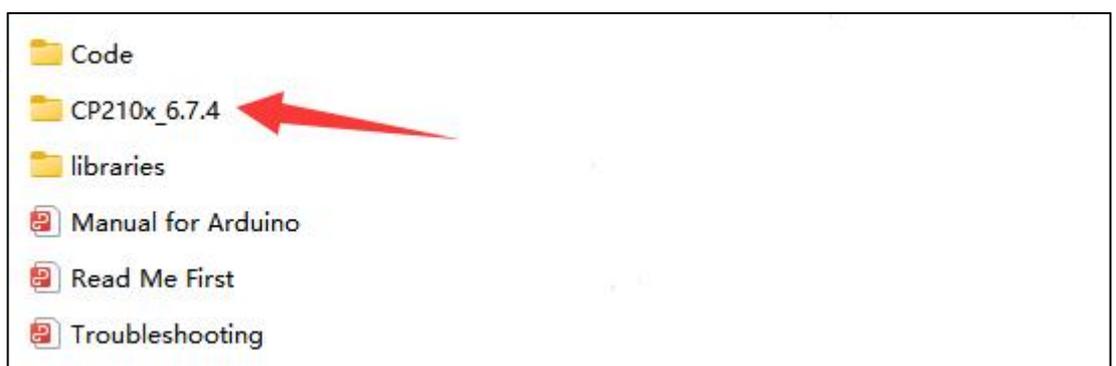
Click Tools>Port at Arduino IDE, you can find the com port displayed by device manager



If the installation process fail, you should see a device with a tiny yellow triangle and exclamation mark next to it.



Our tutorial package comes with a driver for this board, which you can install manually.



1. Right click on the "CP210x USB to UART Bridge Controller" and choose the "Update Driver Software" option.



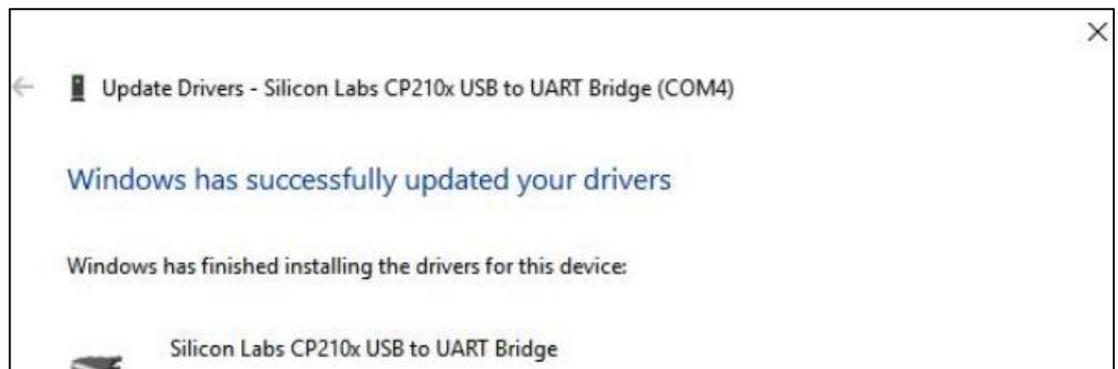
2. Choose the "Browse my computer for Driver software" option.



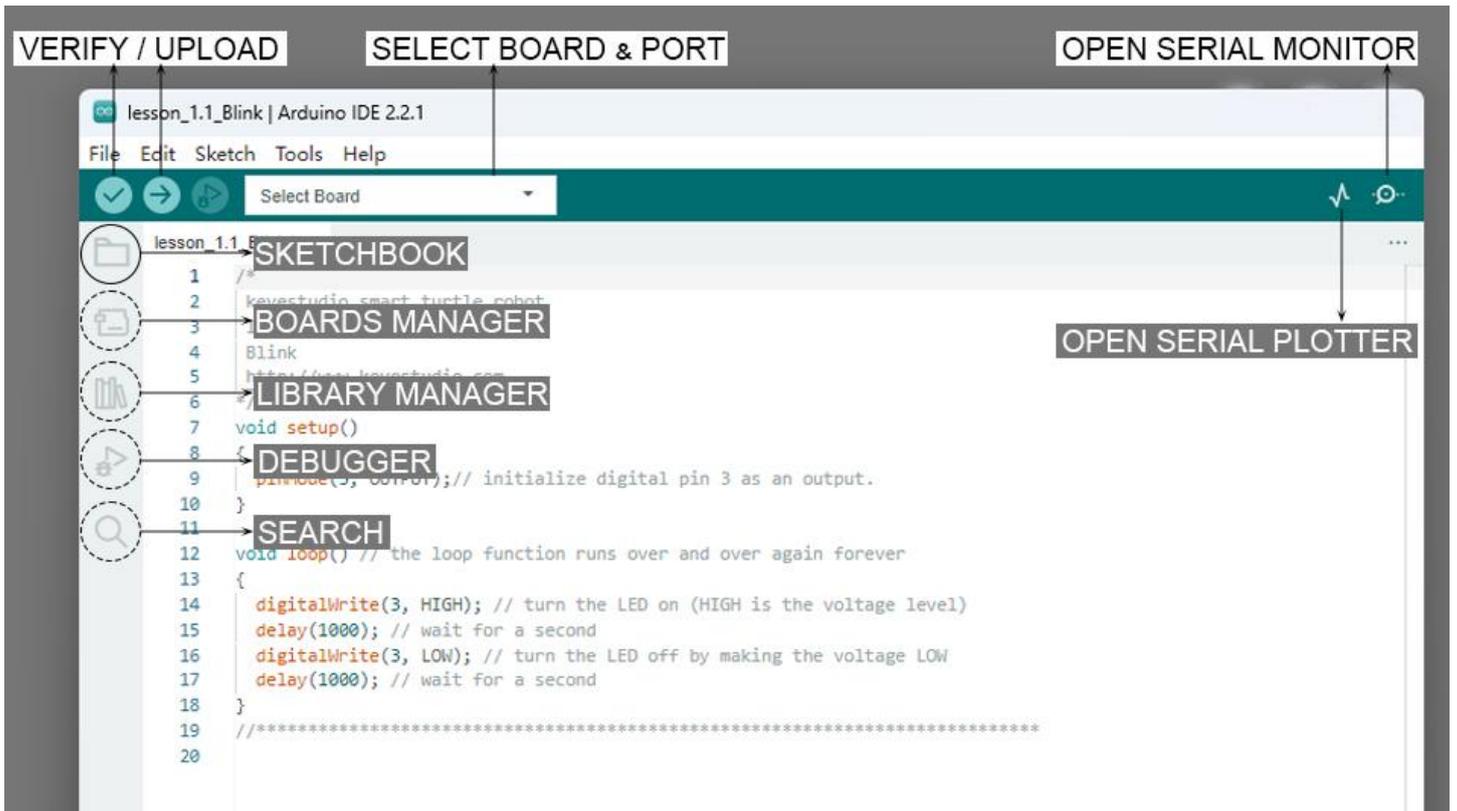
3. Select the driver file named "CP210x\_6.7.4", located in the tutorial package you have downloaded. Then click Next.



4. Wait for a few seconds to complete the installation process.



## 1.4 Introduce of Arduino IDE 2.0



**Verify / Upload** - compile and upload your code to your Arduino Board.

**Select Board & Port** - detected Arduino boards automatically show up here, along with the port number.

**Sketchbook** - here you will find all of your sketches locally stored on your computer. Additionally, you can sync with the Arduino Cloud, and also obtain your sketches from the online environment.

**Boards Manager** - browse through Arduino & third party packages that can be installed. For example, using a MKR WiFi 1010 board requires the Arduino SAMD Boards package installed.

**Library Manager** - browse through thousands of Arduino libraries, made by Arduino & its community.

**Debugger** - test and debug programs in real time.

**Search** - search for keywords in your code.

**Open Serial Monitor** - opens the Serial Monitor tool, as a new tab in the console.

If you want to learn more about Arduino IDE, please refer to this document: [Getting Started with Arduino IDE 2](#)

## 1.5 Add Libraries to Arduino IDE

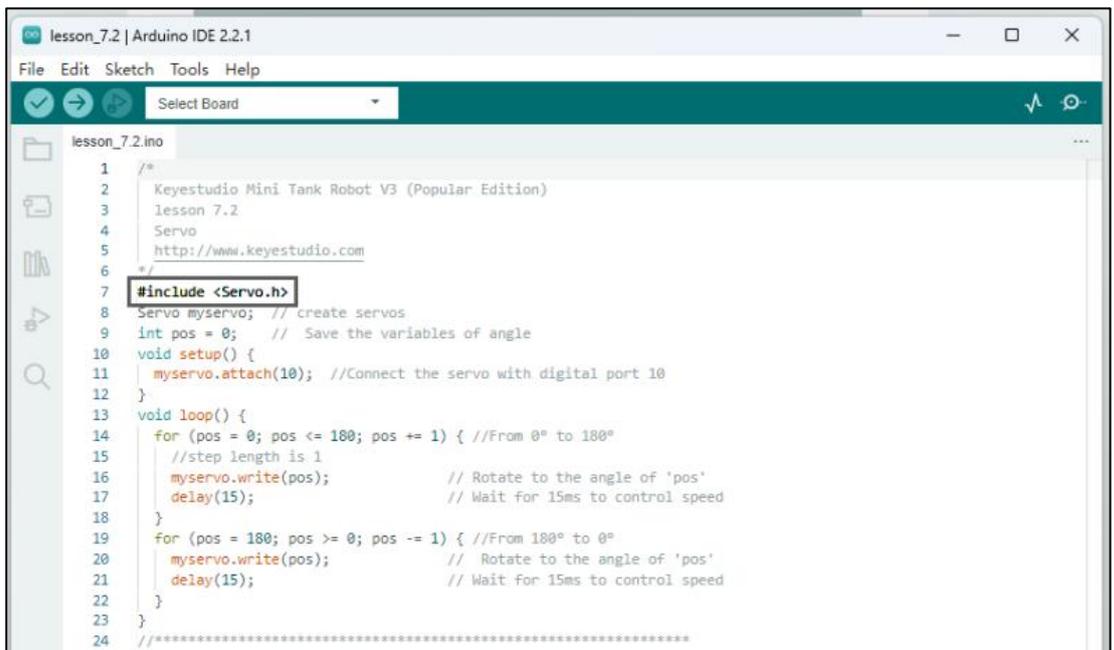
### Why Use Libraries?

Libraries are incredibly useful when creating a project of any type. They make our development experience much smoother, and there almost an infinite amount out there. They are used to interface with many different sensors, RTCs, Wi-Fi modules, RGB matrices and of course with other components on your board.

### Including a Library in the sketch

To use a library, you first need to **include the library at the top of the sketch.**

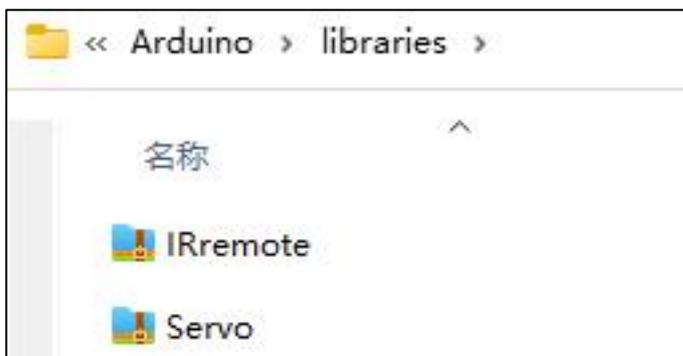
If you find a line of code in the format of `#include <library name>` at the beginning of the code when using our code, it means that you need to add this library file to arduino IDE first before you can successfully upload this code.



```
1  /*
2  Keystudio Mini Tank Robot V3 (Popular Edition)
3  Lesson 7.2
4  Servo
5  http://www.keystudio.com
6  */
7  #include <Servo.h>
8  Servo myservo; // create servos
9  int pos = 0; // Save the variables of angle
10 void setup() {
11   myservo.attach(10); //Connect the servo with digital port 10
12 }
13 void loop() {
14   for (pos = 0; pos <= 180; pos += 1) { //From 0° to 180°
15     //step length is 1
16     myservo.write(pos); // Rotate to the angle of 'pos'
17     delay(15); // Wait for 15ms to control speed
18   }
19   for (pos = 180; pos >= 0; pos -= 1) { //From 180° to 0°
20     myservo.write(pos); // Rotate to the angle of 'pos'
21     delay(15); // Wait for 15ms to control speed
22   }
23 }
24 //*****
```

To make the robot work, we need to add these two library files to the Arduino IDE.

The IRremote and the Servo library are came with the tutorial package.

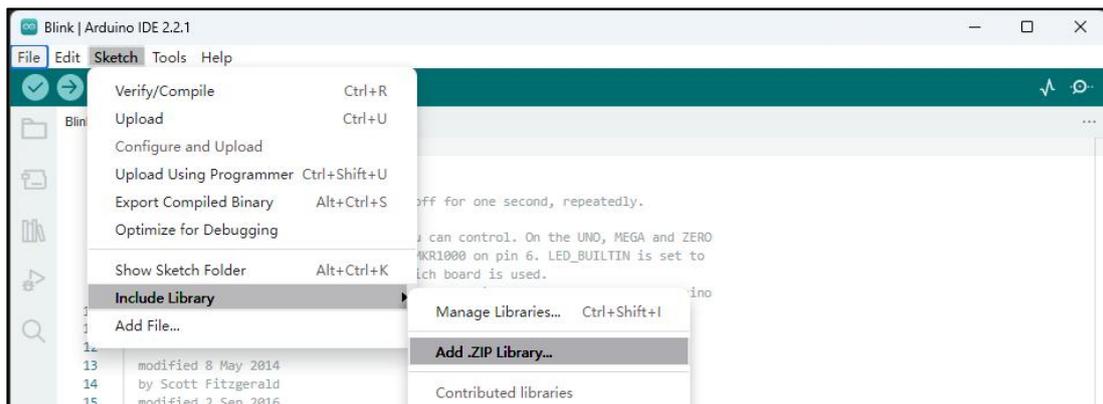


## Two ways to add libraries to Arduino IDE

### 1) Method One: Importing a .zip Library

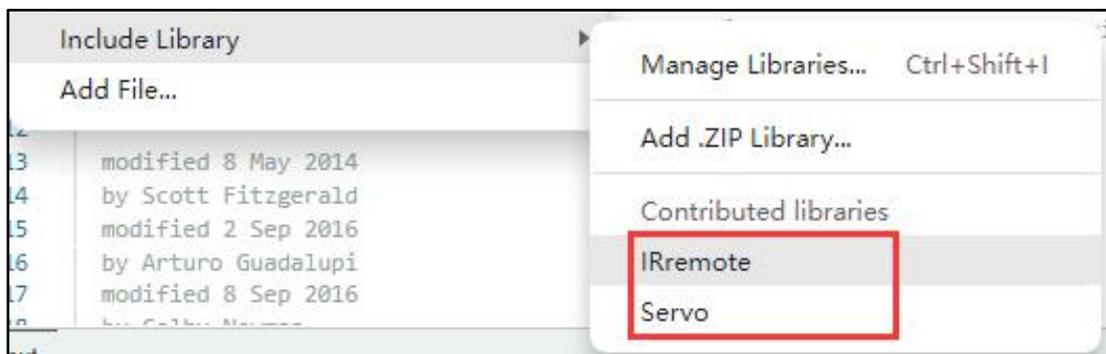
In the menu bar, go to Sketch > Include Library > Add .ZIP Library...

You will be prompted to select the library you want to add. Navigate to the .zip file's location and open it.



You may need to restart the Arduino IDE for the library to be available.

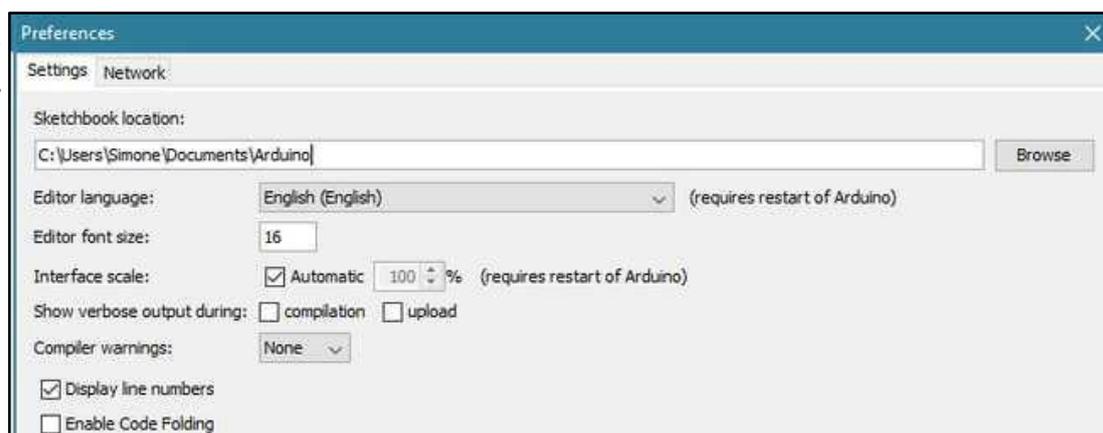
After successful installation, you will see them in the list.



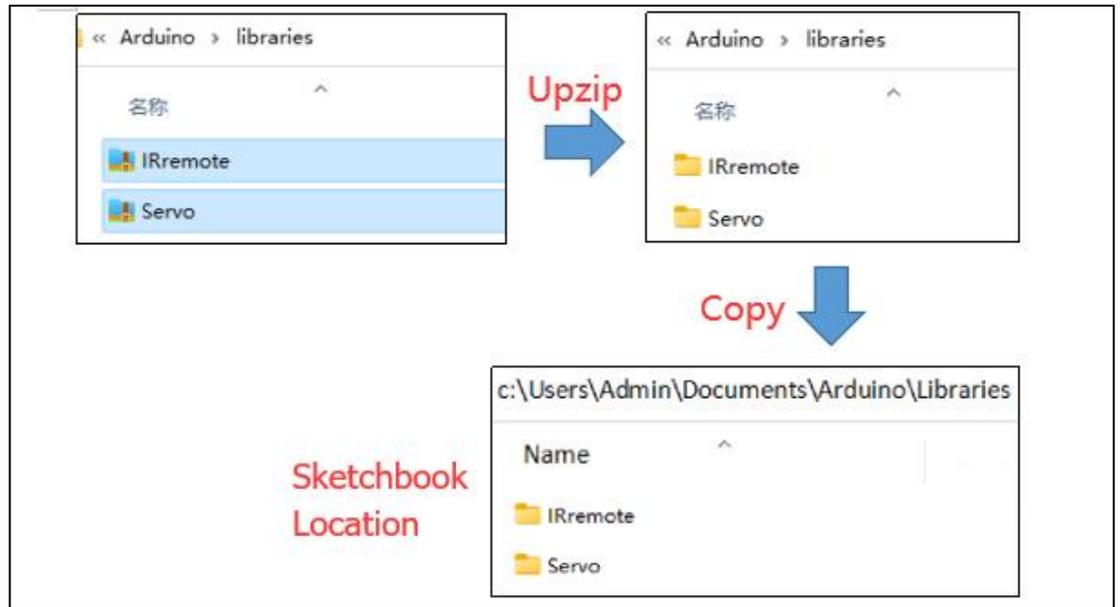
### 2) Method Two: Manual Installation

If you want to add a library manually, you need to download it as a ZIP file, expand it and put it in the **libraries** folder of your sketchbook by yourself.

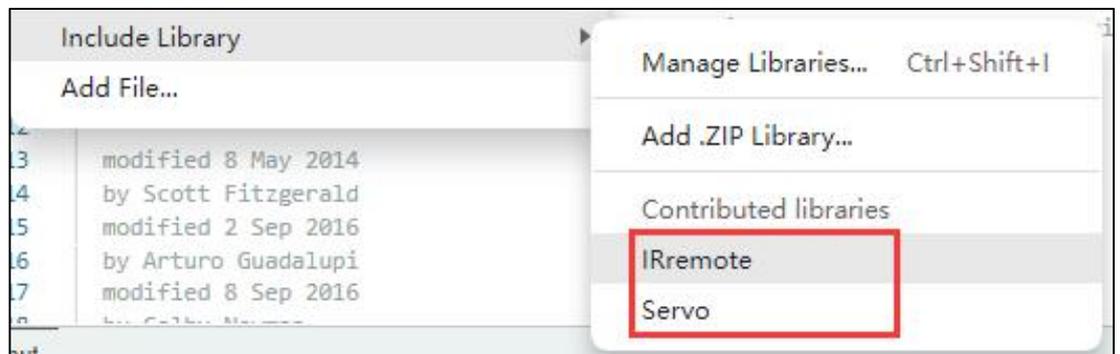
You can find or change the location of your sketchbook folder at **File > Preferences > Sketchbook** location



Go to the directory where you have downloaded the ZIP file of the library. Unzip the ZIP file and copy it into the "libraries" folder inside your sketchbook.



Start the Arduino Software (IDE), go to Sketch > Include Library. Verify that the library you just added is available in the list.



## Uninstalling an Arduino Library

Uninstalling an Arduino Library is simpler than installing it. Find the sketchbook folder on your computer (same as in the "Manually installing a library" chapter). Go to the location and open the "libraries" folder. Select the folder containing the library you want to delete, and then simply delete it. Next time you open your Arduino IDE, there won't be the deleted library under the Sketch > Include Library menu.

Libraries don't take much space and most of the time there is no reason to remove them. If you don't intend to use them again, though, and want to declutter the list, you can safely delete them. You can always install any Arduino Library again if you need to use it in the future.

## 2.Adjust the Angle of the Servo Before Assembly.

We need to adjust the servo to 90° so that it can work with the ultrasonic module to make the robot work as intended.

You need to prepare:



V4.0 board\*1



Motor Driver Shield\*1



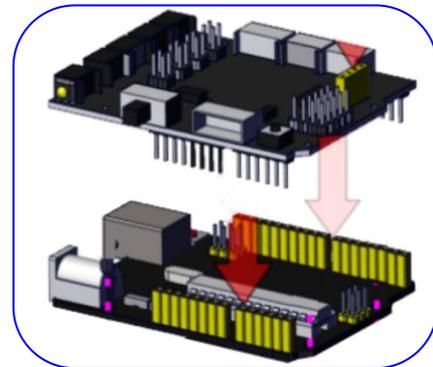
9G 180 ° Servo\*1



USB Cable\*1

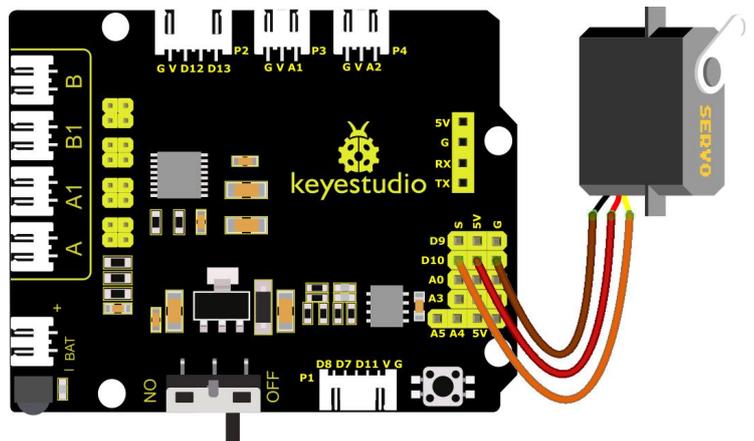
### Wiring

1. Plug the Motor Driver Shield into the arduino V4.0 control board

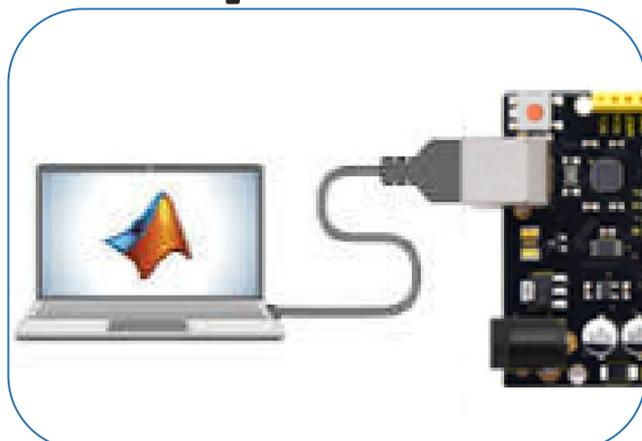


2. Connect the servo to the Motor Driver Shield

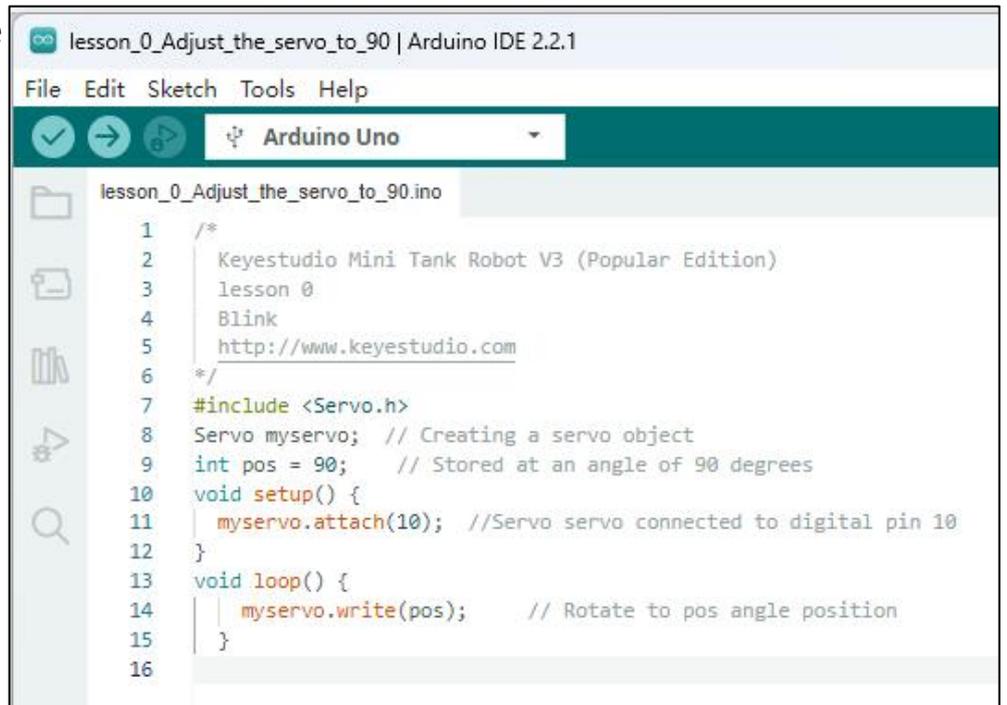
**Note:** The brown, red and orange wire of the servo are respectively attached to Gnd(G), 5v(V) and **Pin 10** of the shield.



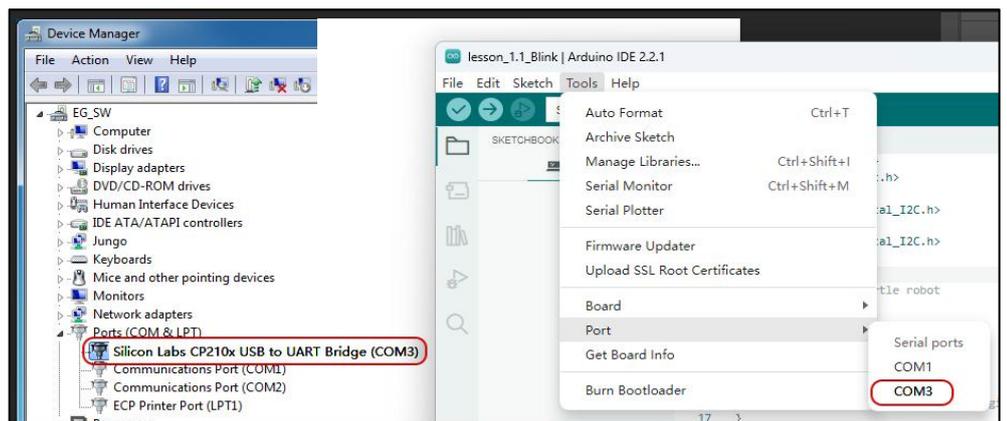
3. Connect the V4.0 board to the computer



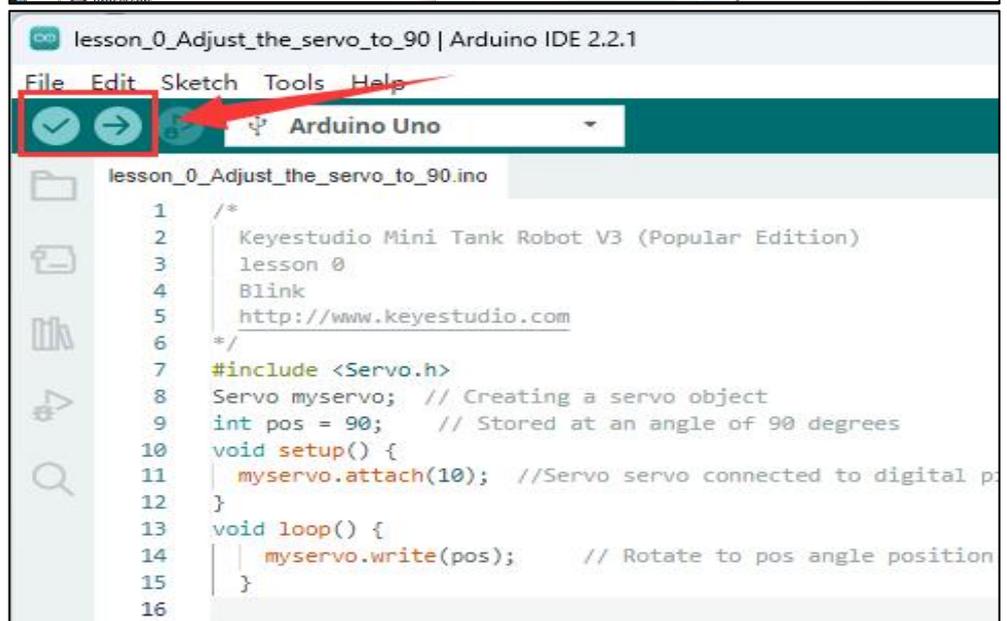
1. Open the INO file inside the `lesson_0_Adjust_the_servo_to_90` folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.



3. Click on upload >>>done uploading, the servo will be adjusted to 90°.



## 3.How to Assemble the Robot

### 3.1 Things to note before assembly:

1.Before assembly, please tear off the protective film on the acrylic boards.



2.To run this robot, you will need to prepare yourself two 18650 batteries as well as a battery charger.

The following parameters are available for your purchase:

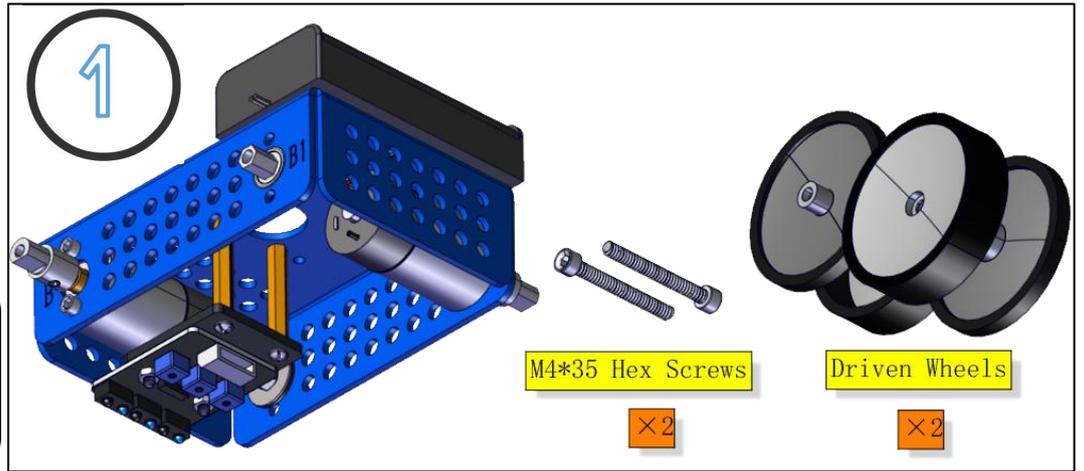
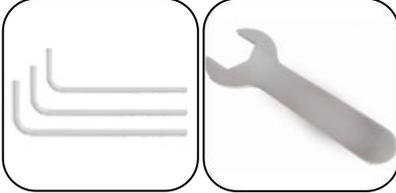
Specifications	
Size	18650
Positive Terminal:	Flat Top or with a top
Capacity	1500-3000mAh
Nominal Voltage	3.7V
Maximum Voltage	4.2V
Discharge cut-off Voltage	2.5V
Rechargeable	Yes
Approx. Dimensions	18.5mm x 65.2mm



### 3.2 Assembly Steps

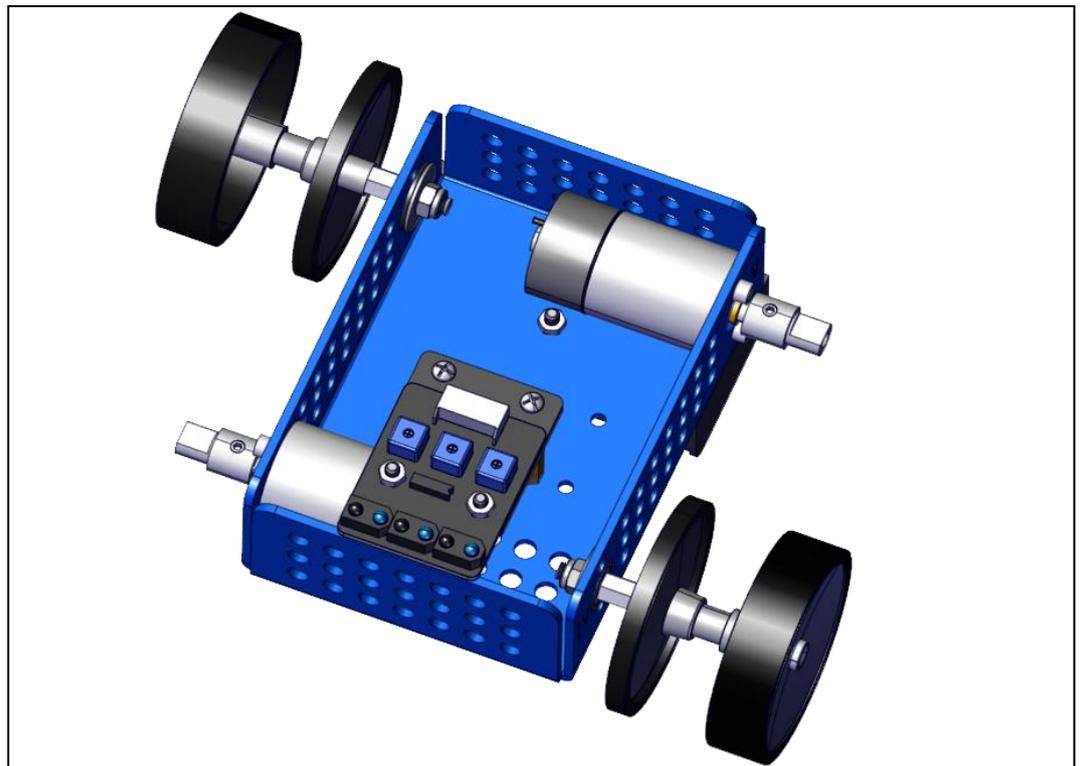
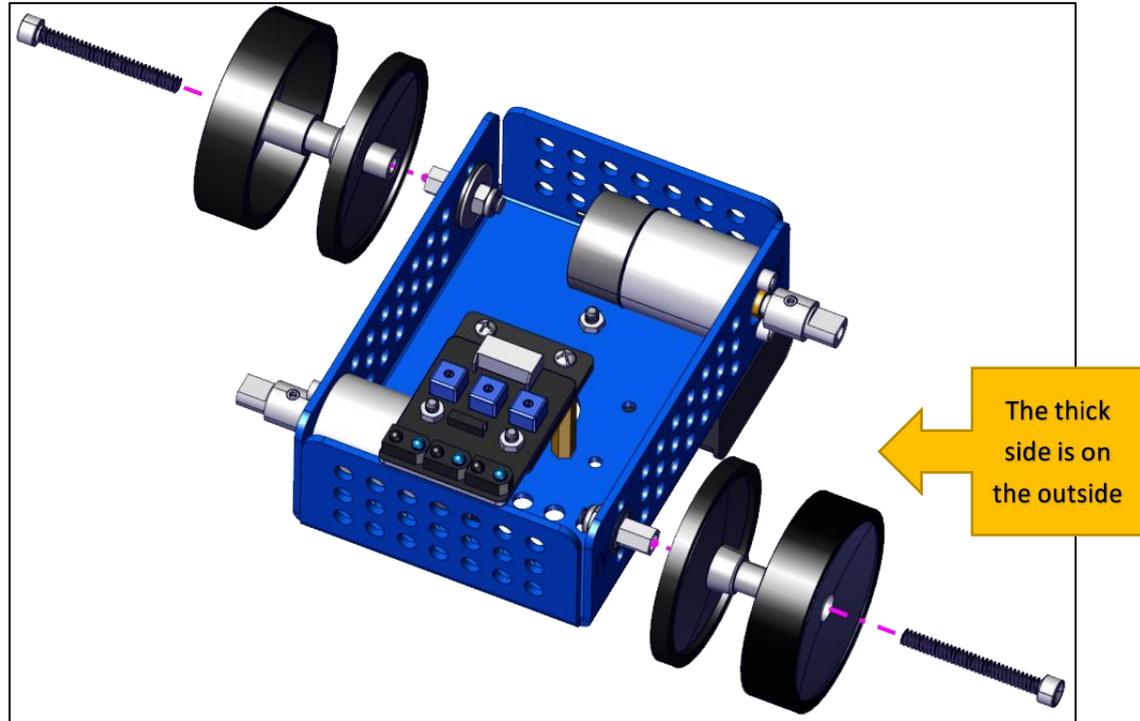
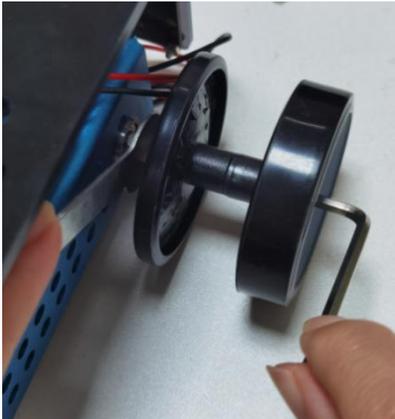
#### 1. Install the driven wheels

Tools needed:

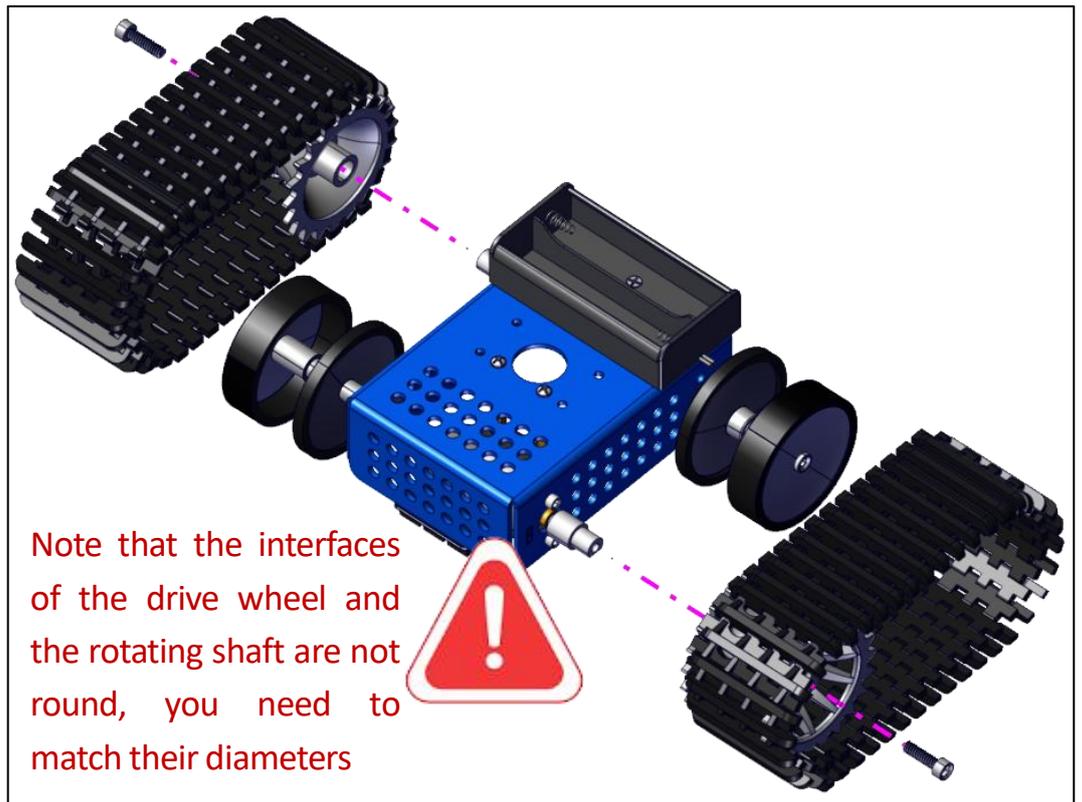
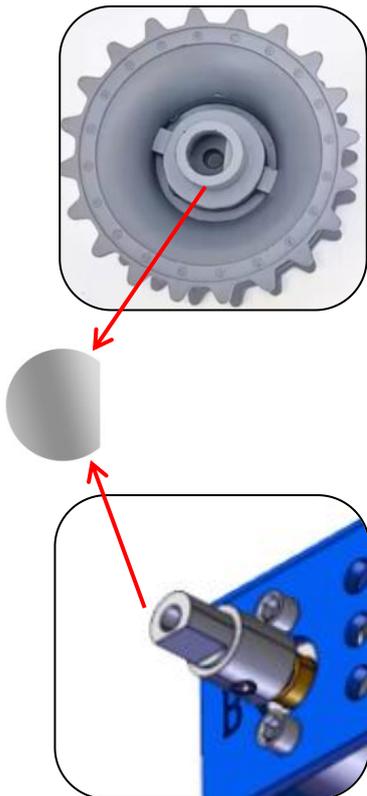
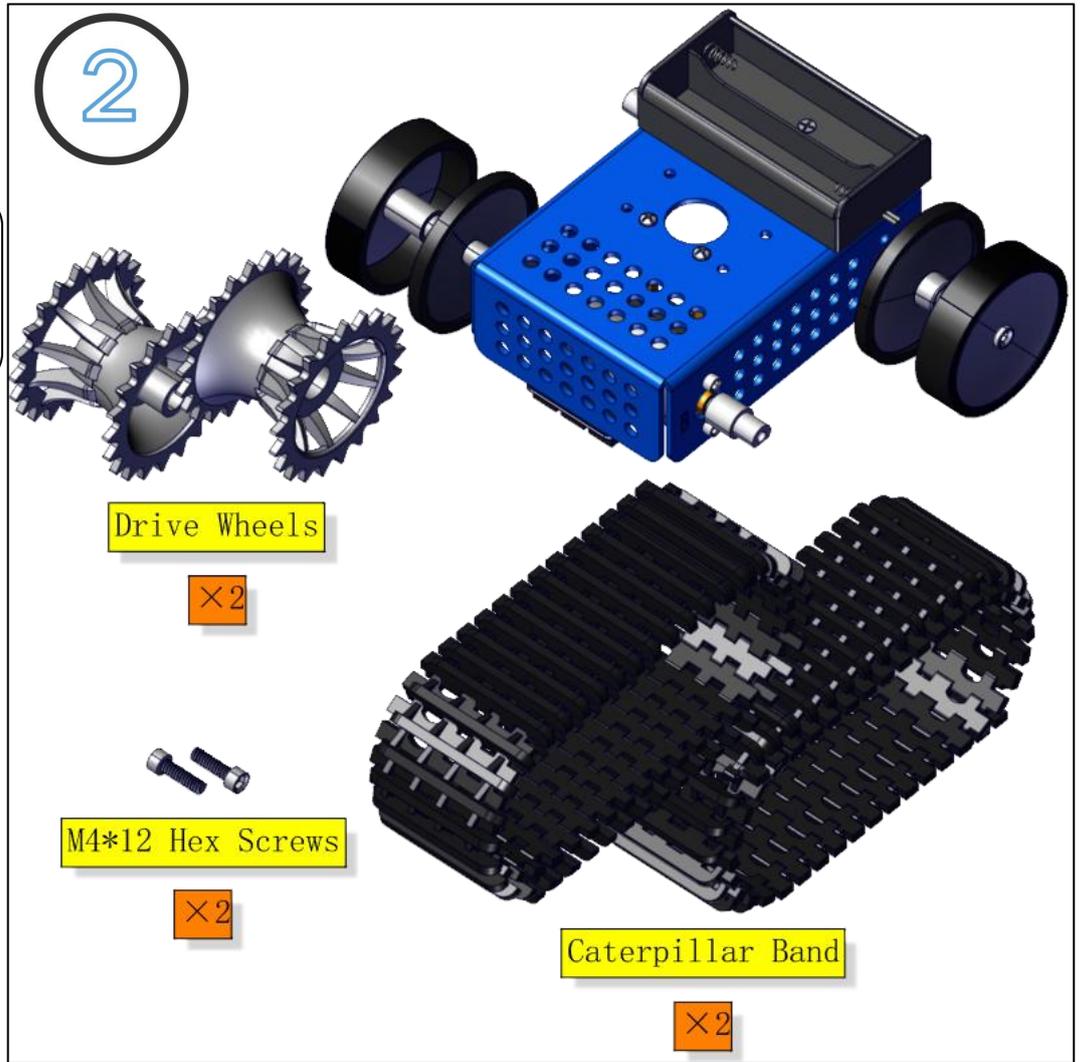
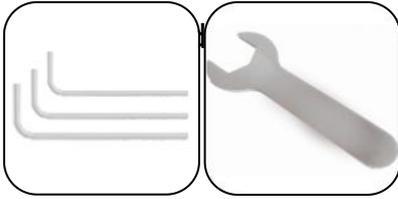


Pay attention to the installation direction of the wheels.

**The thick side is on the outside.**

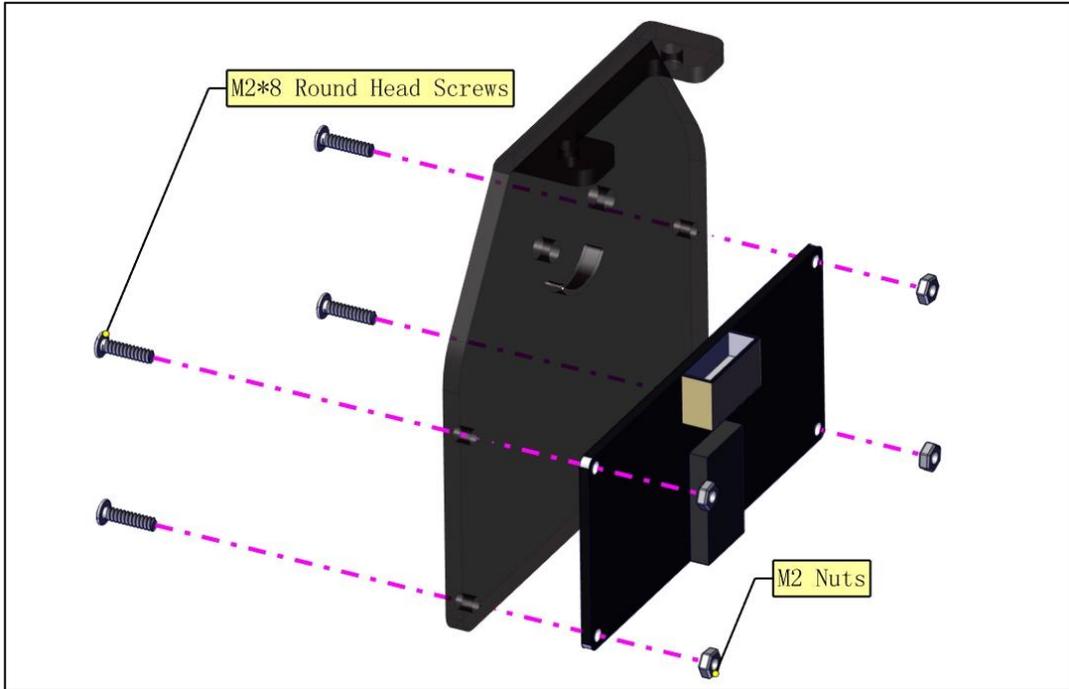
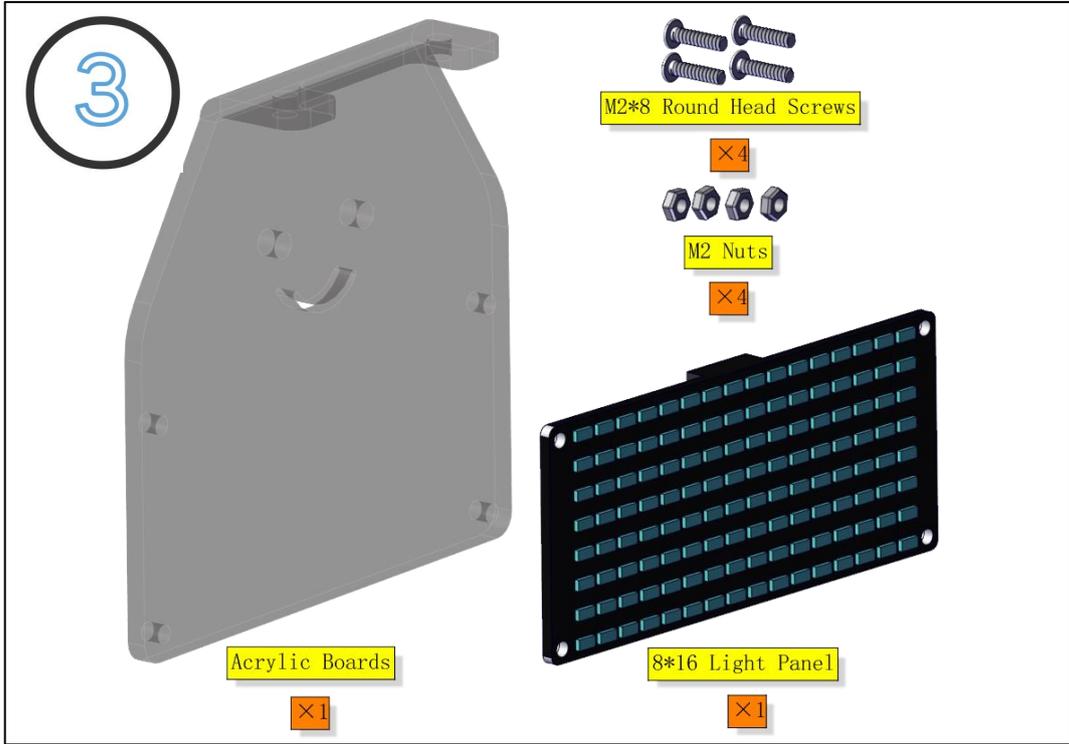


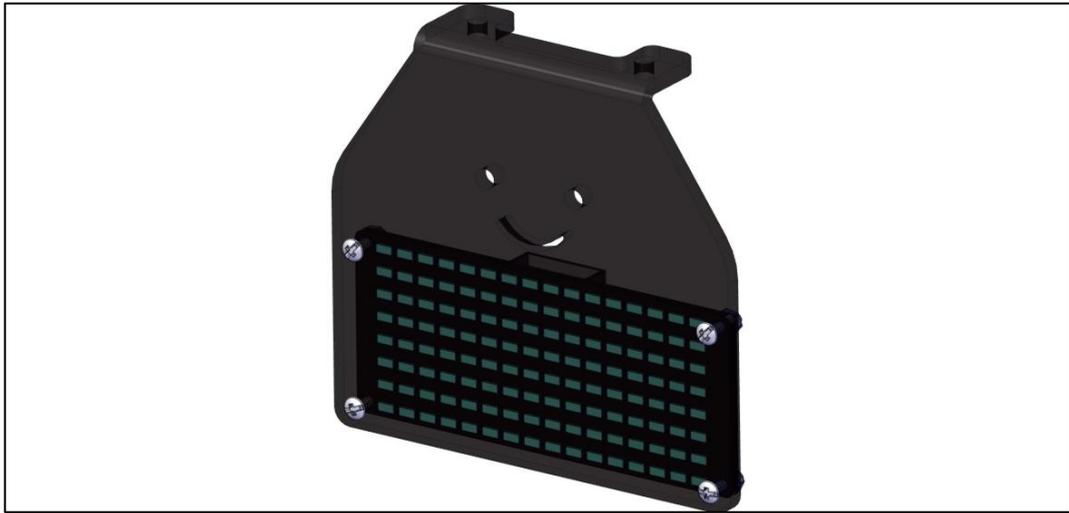
2. Install the drive wheels and the track



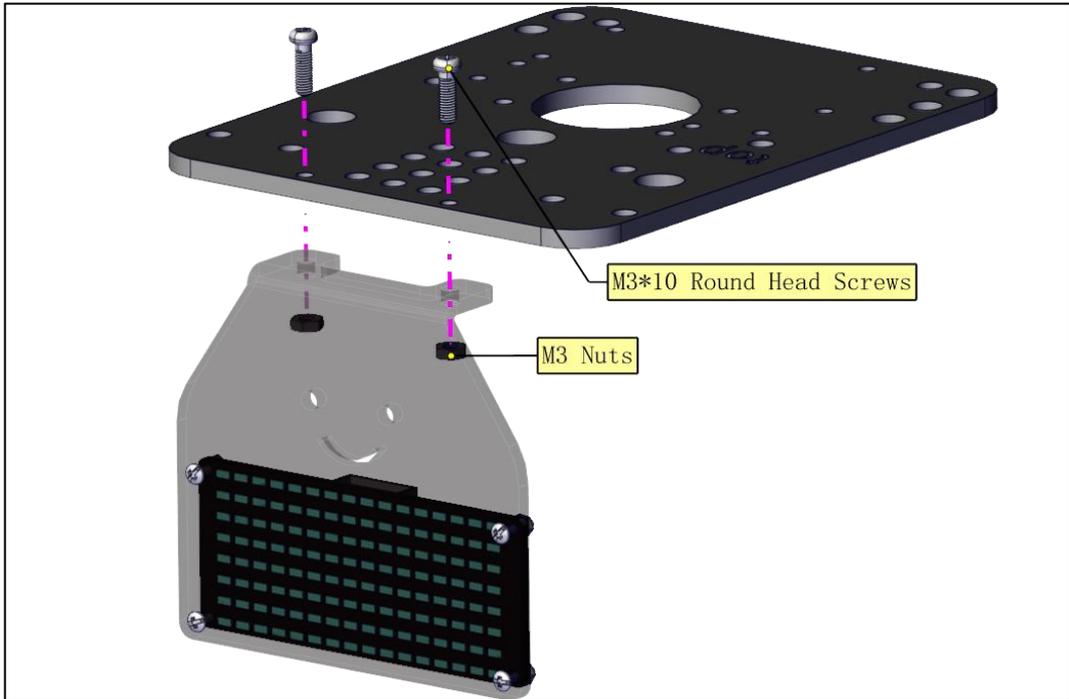
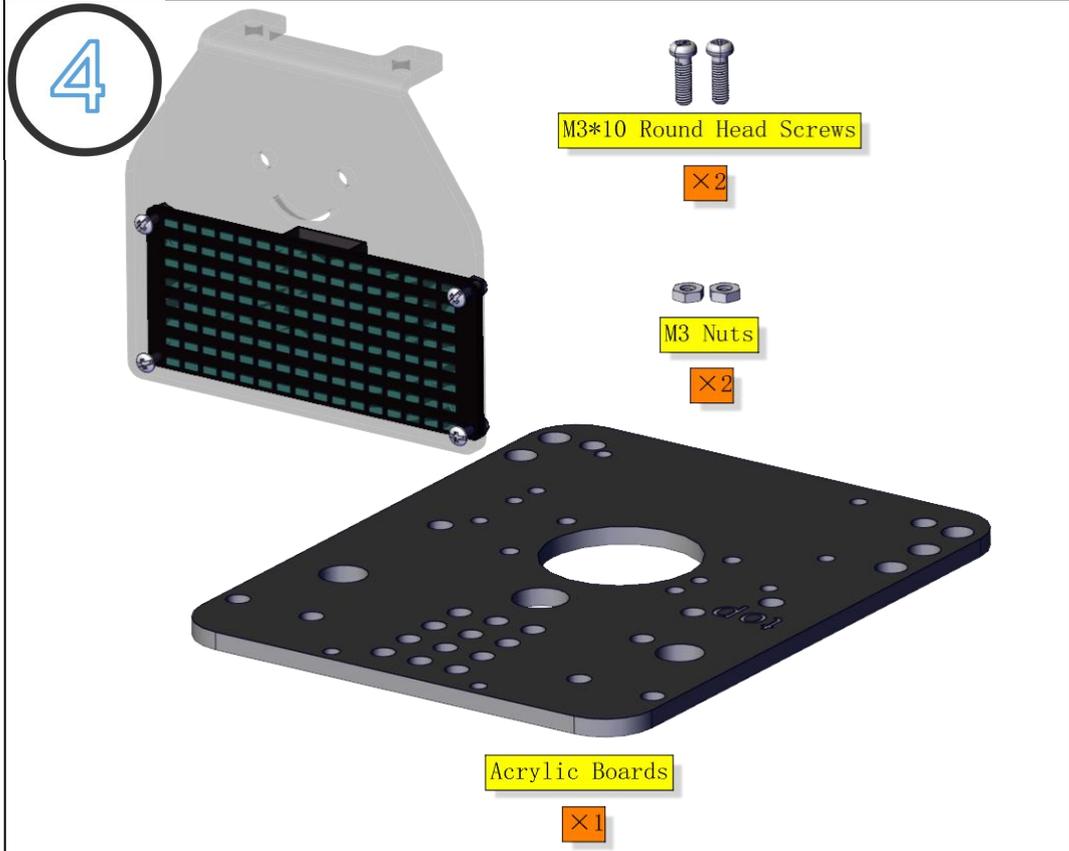


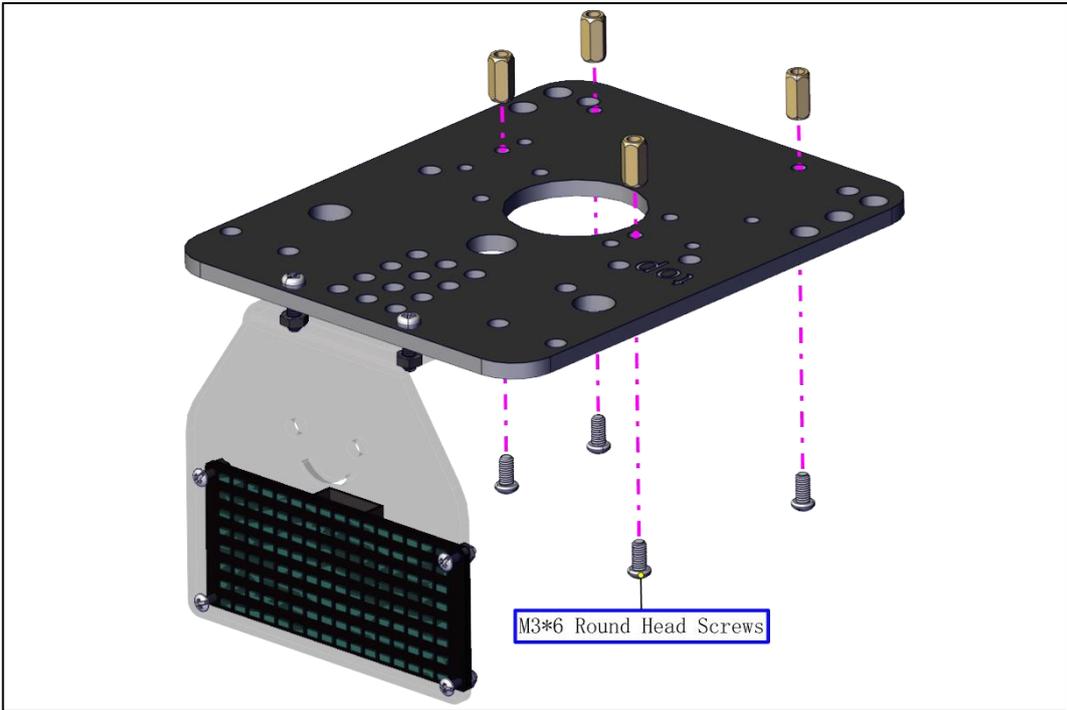
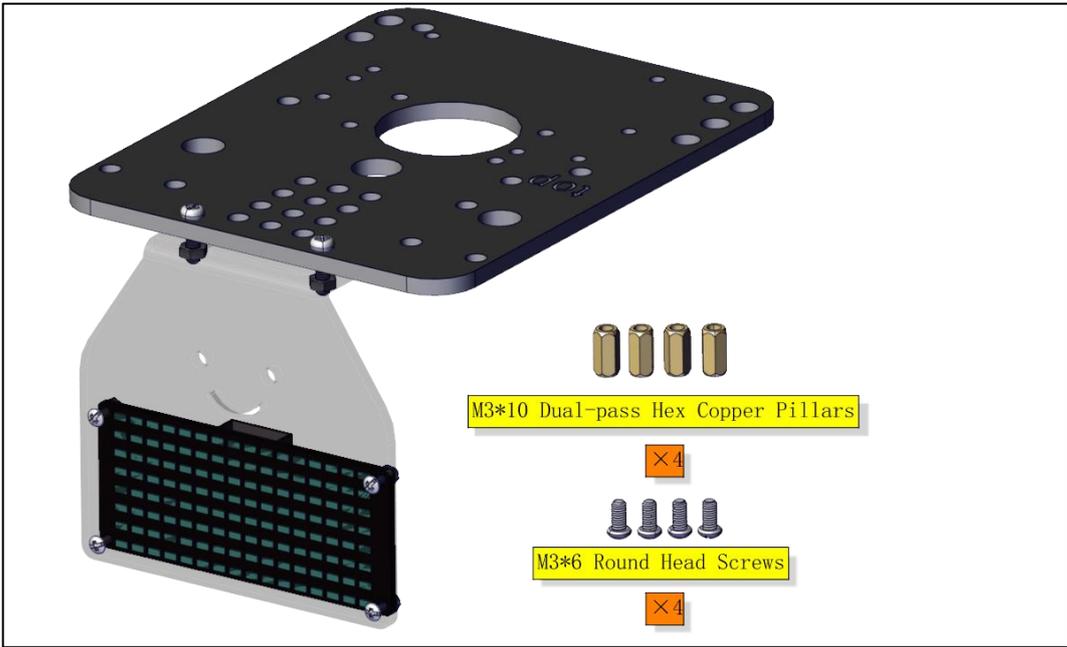
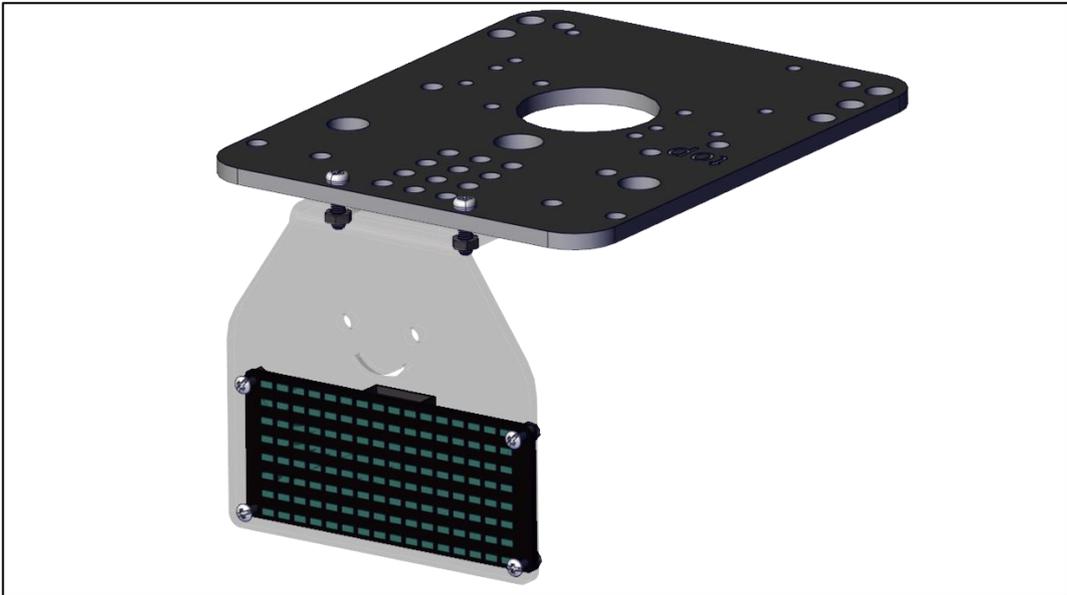
**3. Assemble 8\*16 LED Dot Matrix on the front acrylic board**

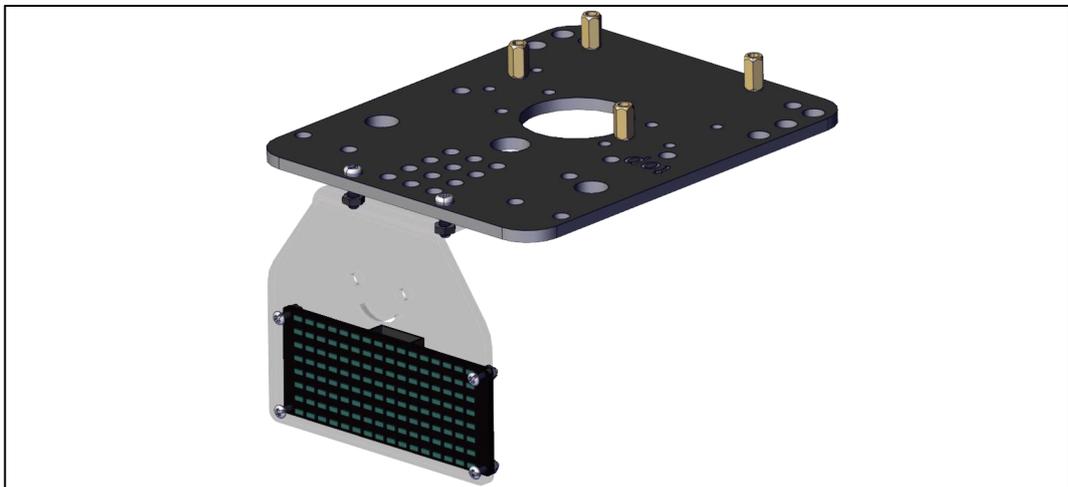




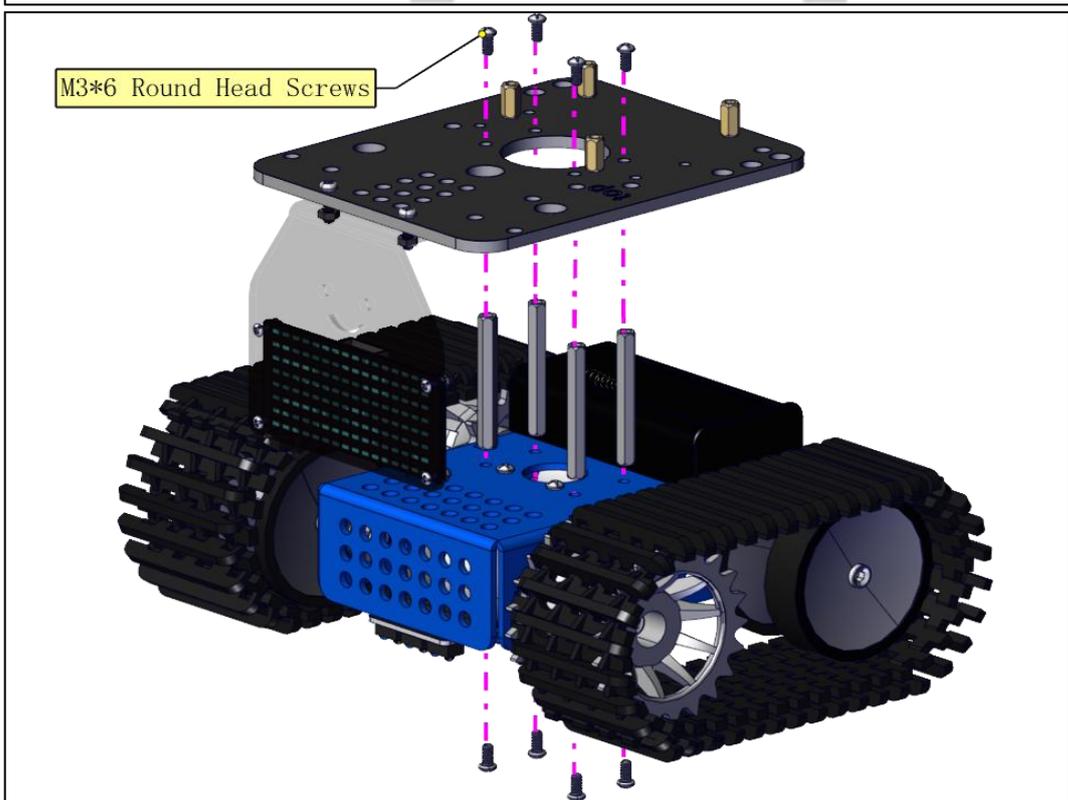
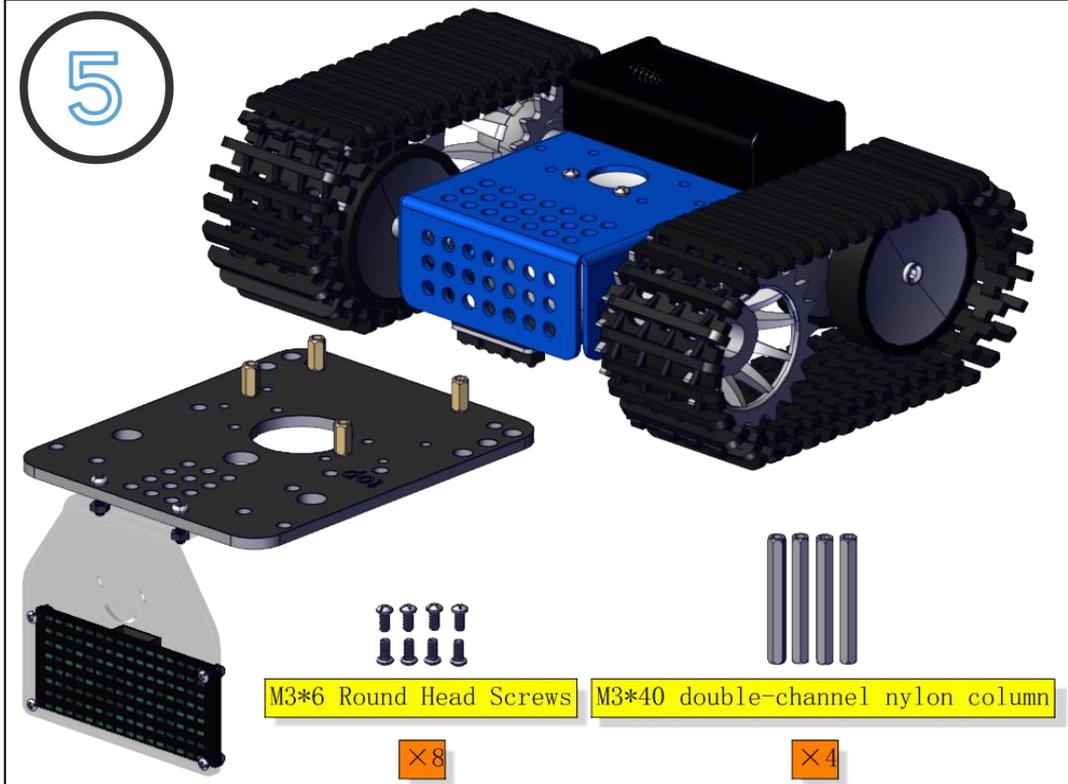
**4. Assemble 8\*16 LED Dot Matrix on the top acrylic board**

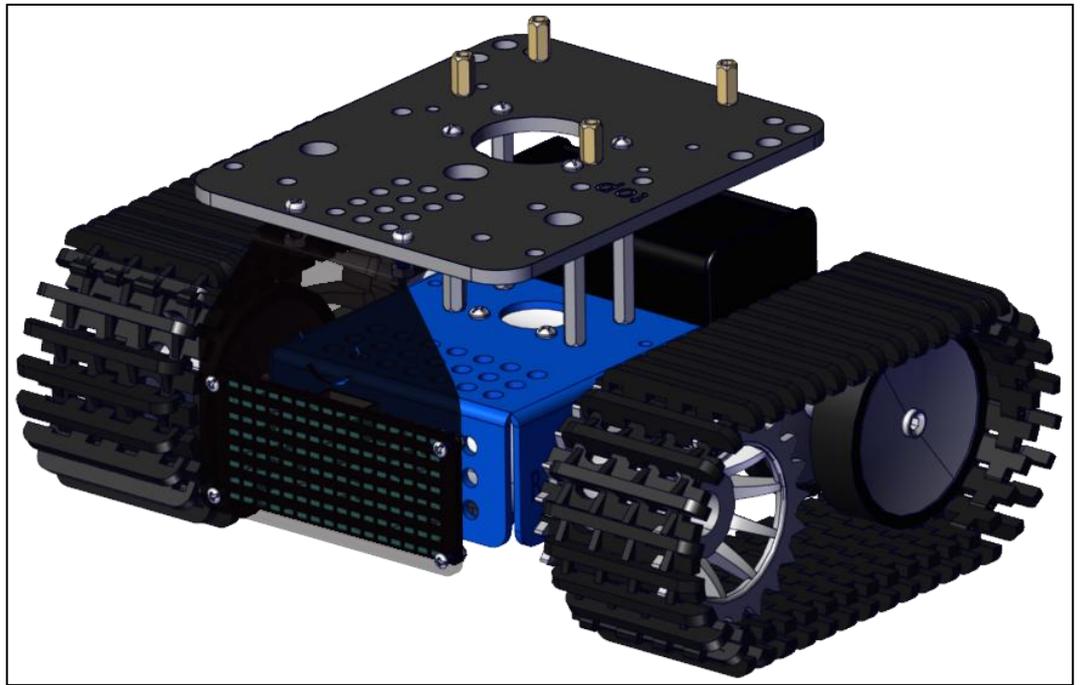




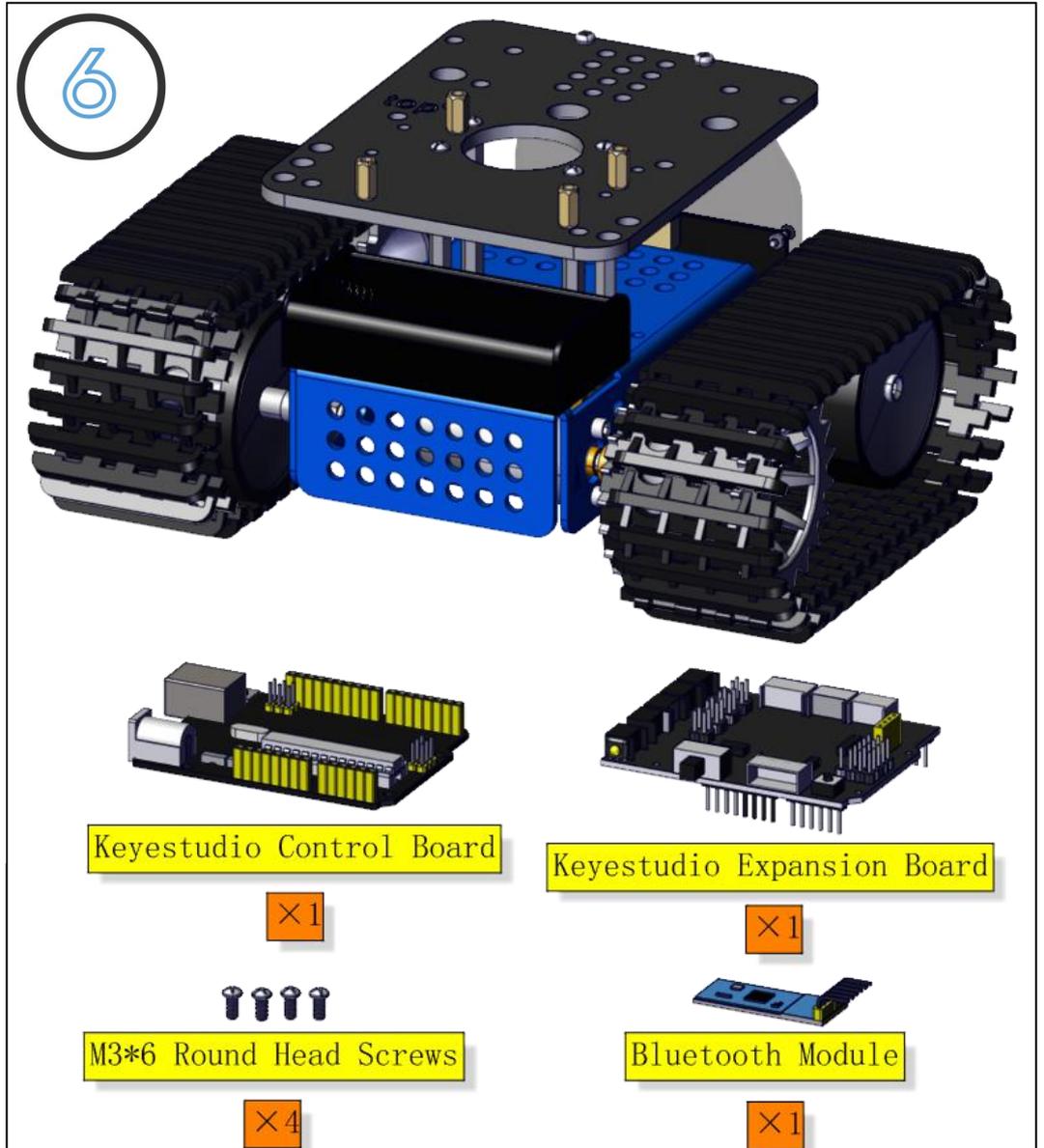


5. Assemble 8\*16 LED Dot Matrix and the top acrylic board on the car base

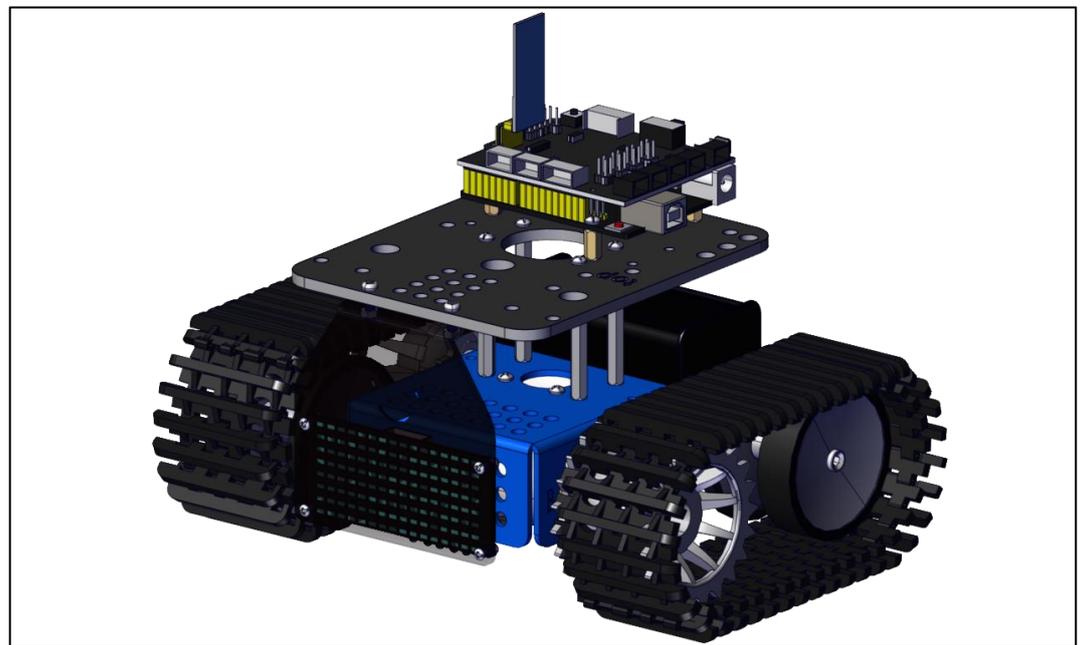
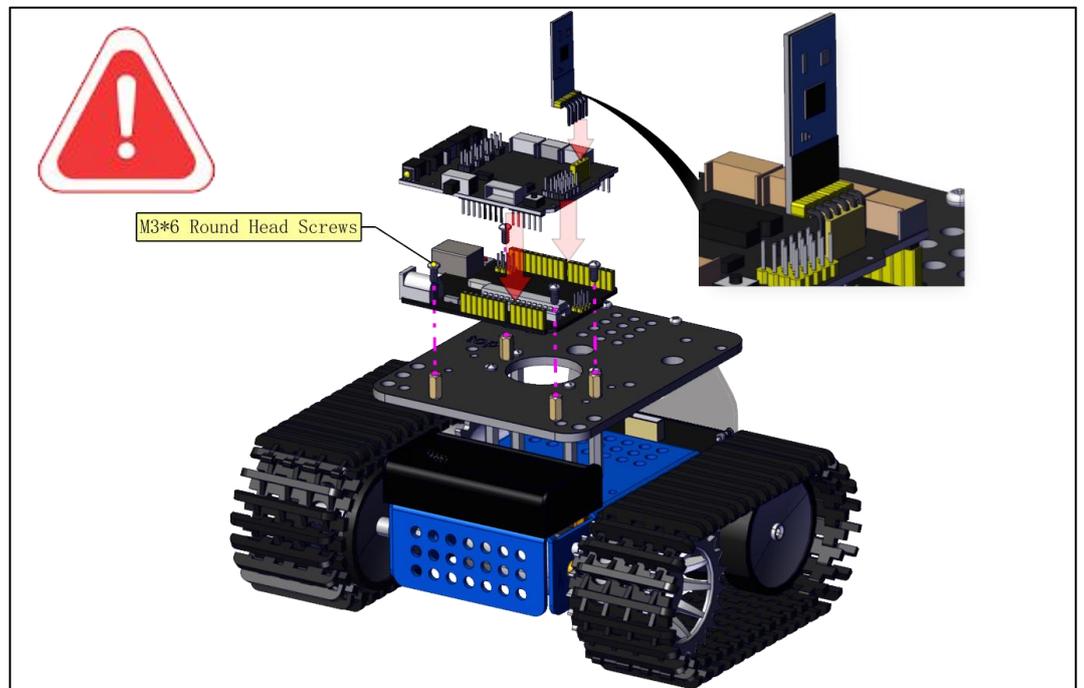




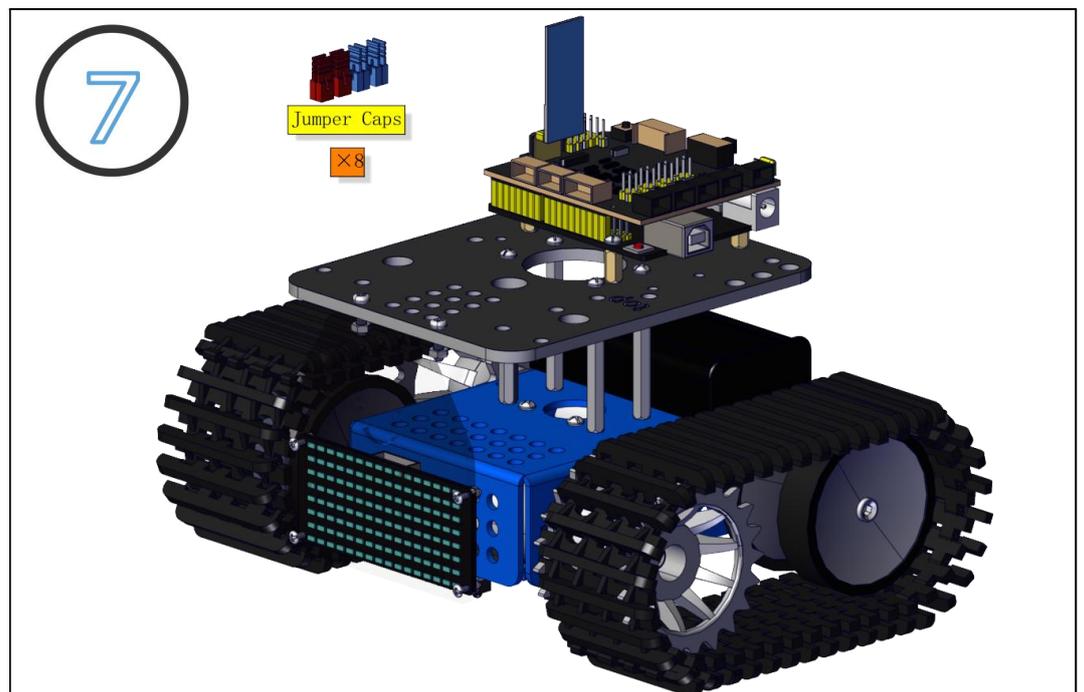
6. Assemble the control board, motor shield, and Bluetooth module on the car



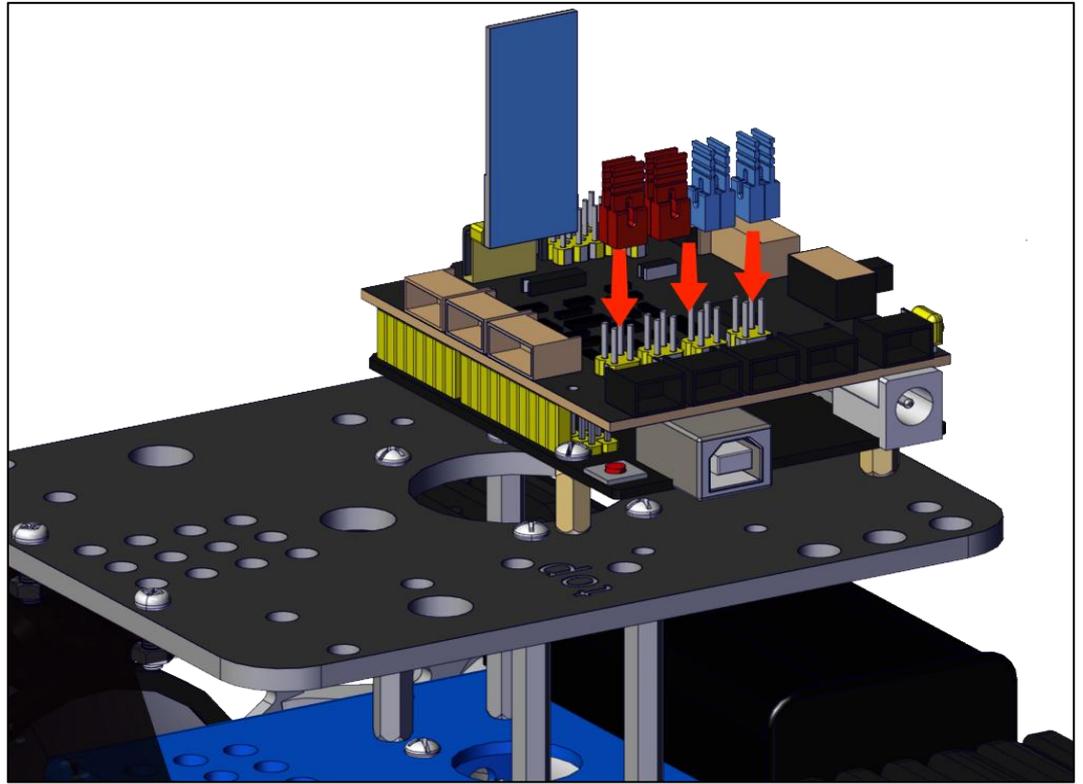
**Note: The Bluetooth module will occupy the serial port. Please do not connect to Bluetooth when uploading code.**



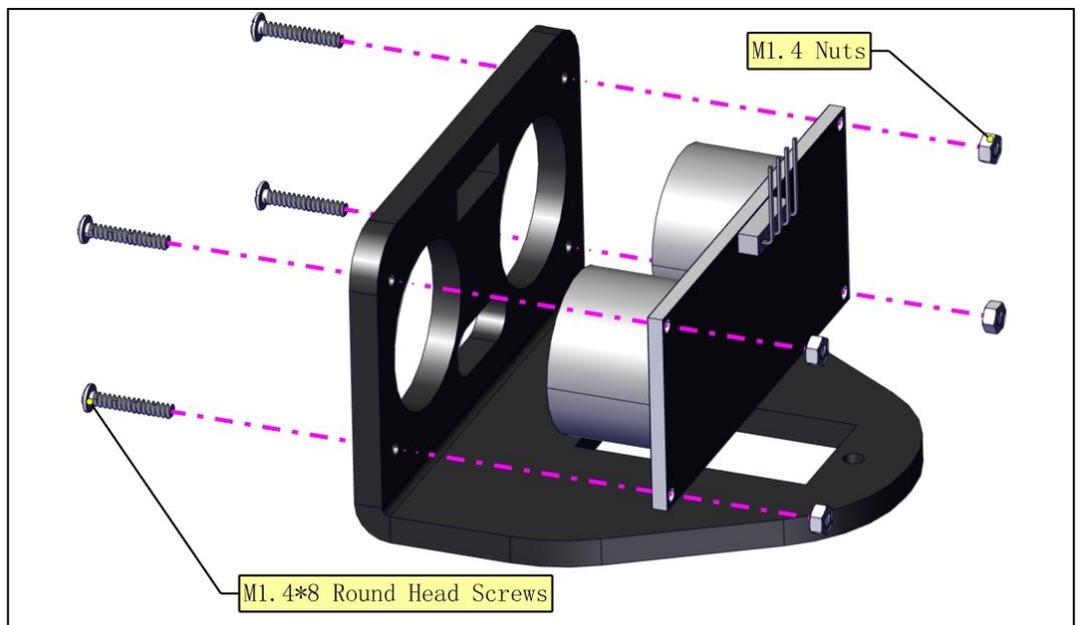
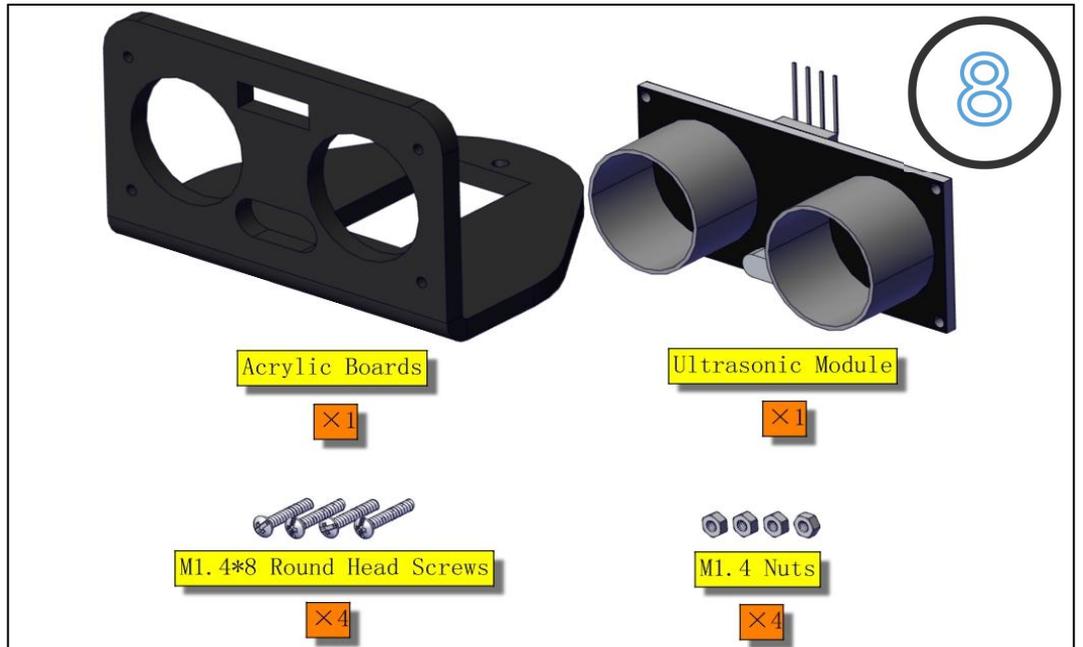
**7. Assemble the jumper caps on the shield**



Note the direction of jumper caps

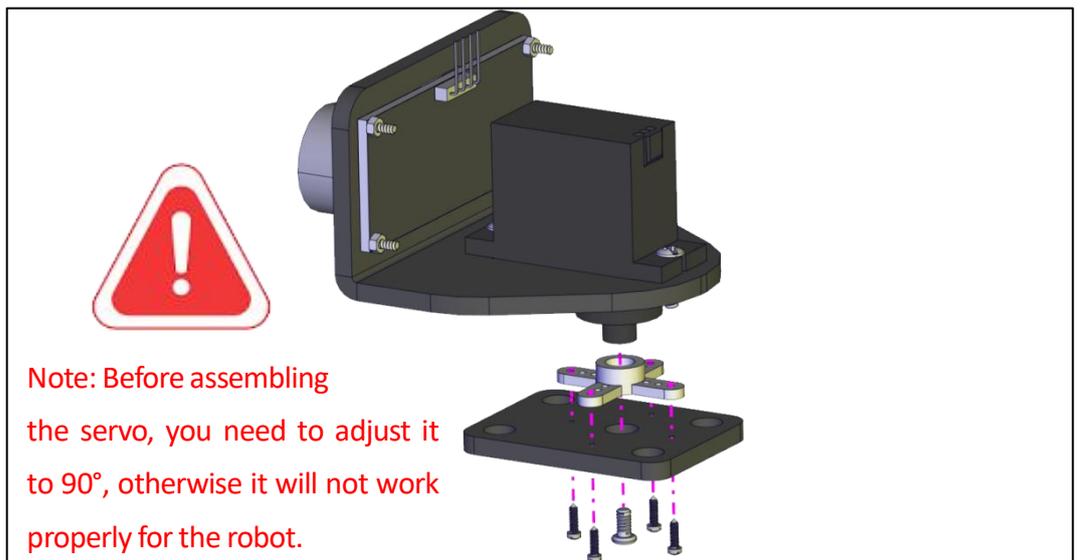
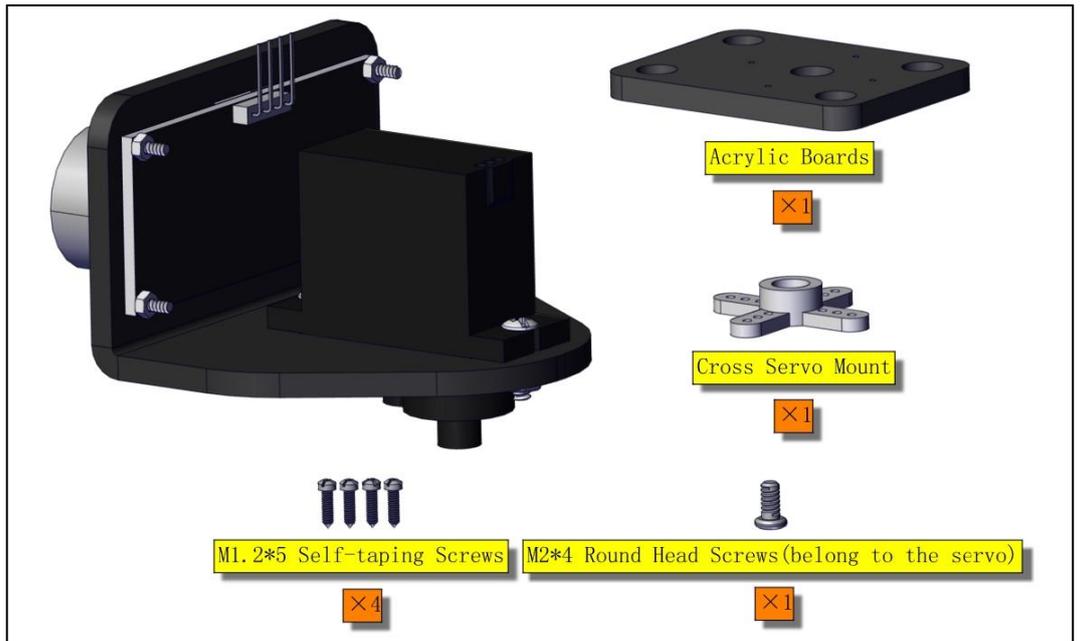
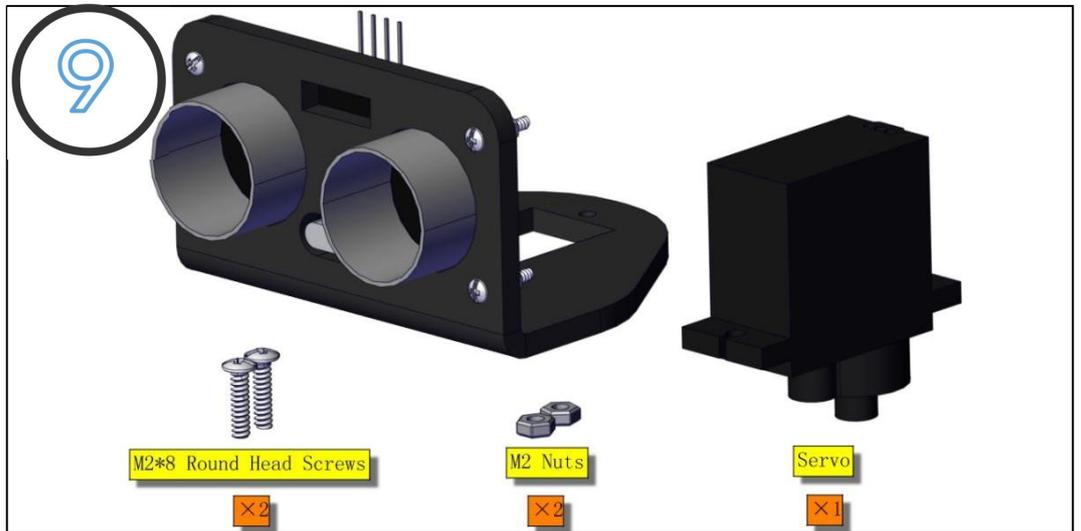
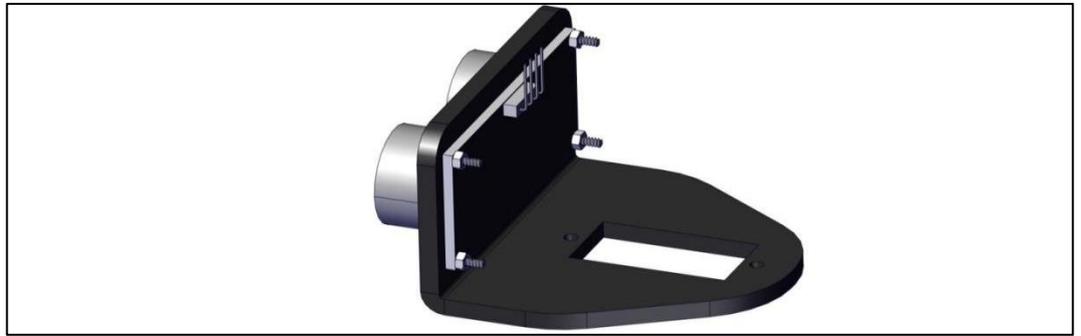


8. Assemble the ultrasonic module on the acrylic board



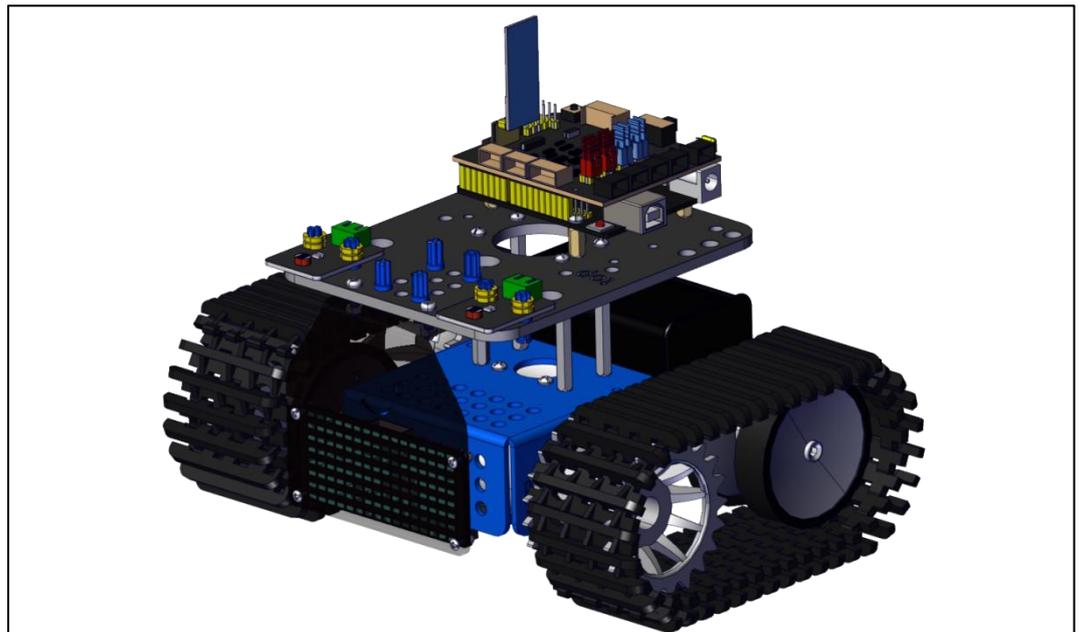
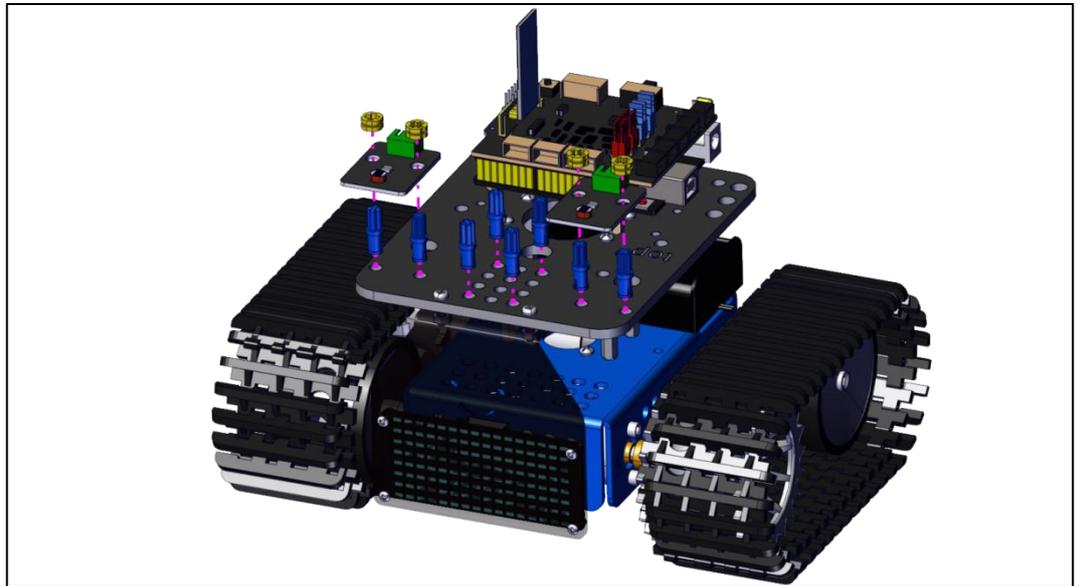
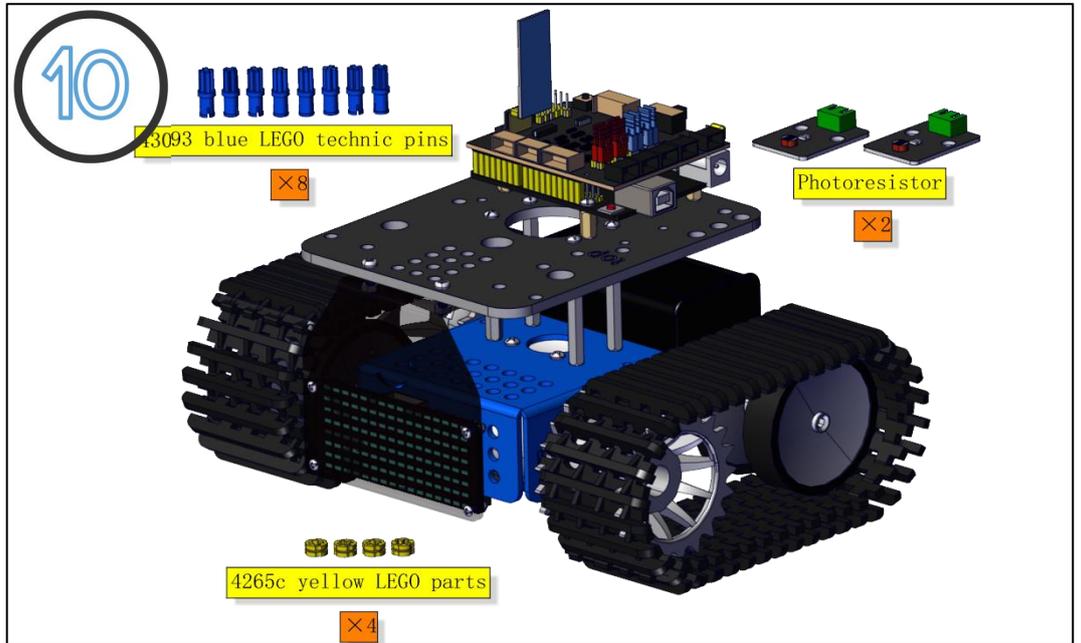
9. Assemble the servo on the acrylic board

**Note:** Before assembling the servo, you need to adjust it to 90°, otherwise it will not work properly for the robot.



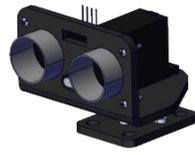
**Note:** Before assembling the servo, you need to adjust it to 90°, otherwise it will not work properly for the robot.

10. Assemble the photoresistor sensor on the car

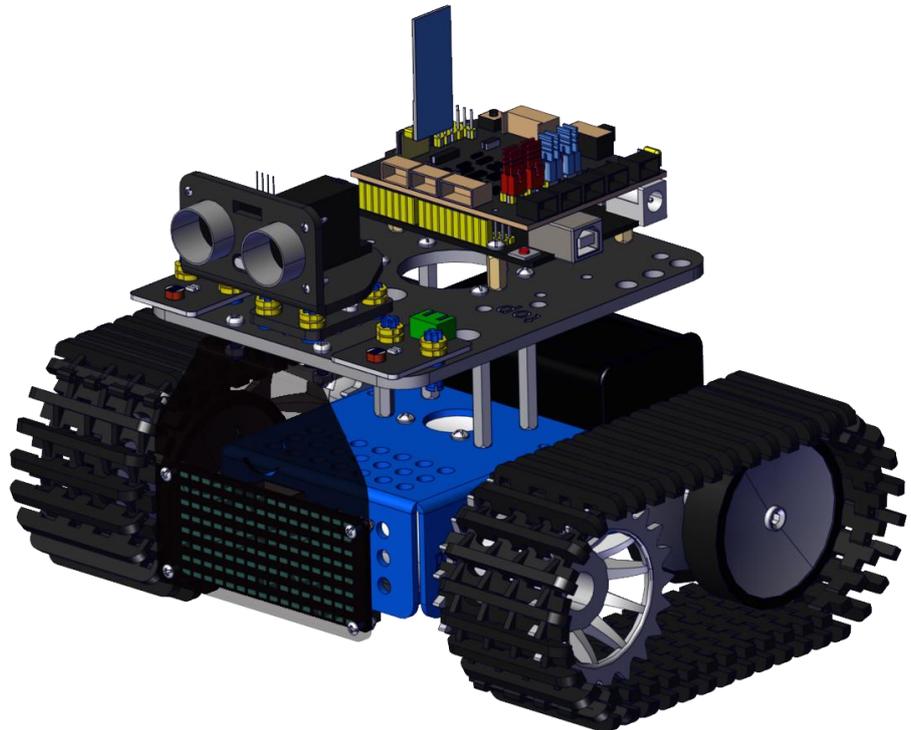
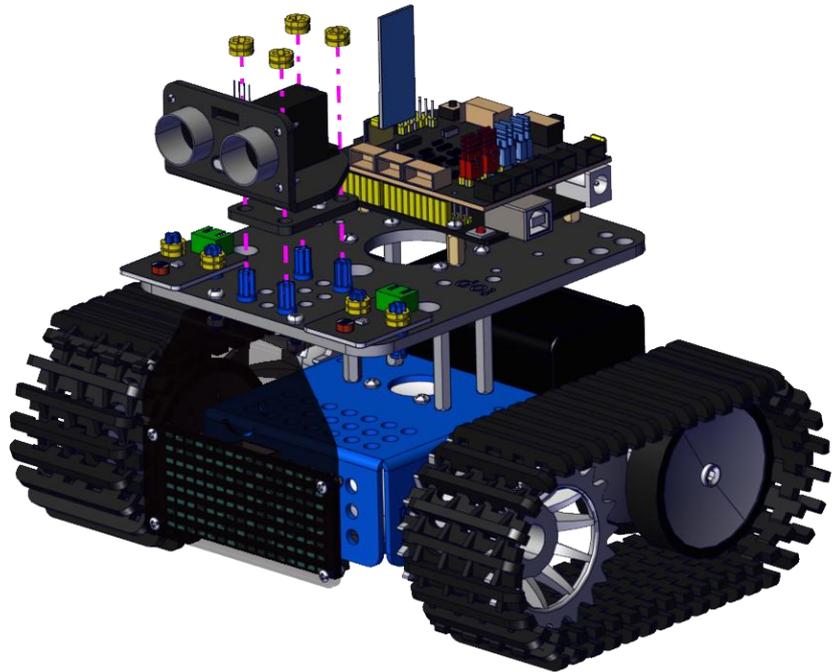
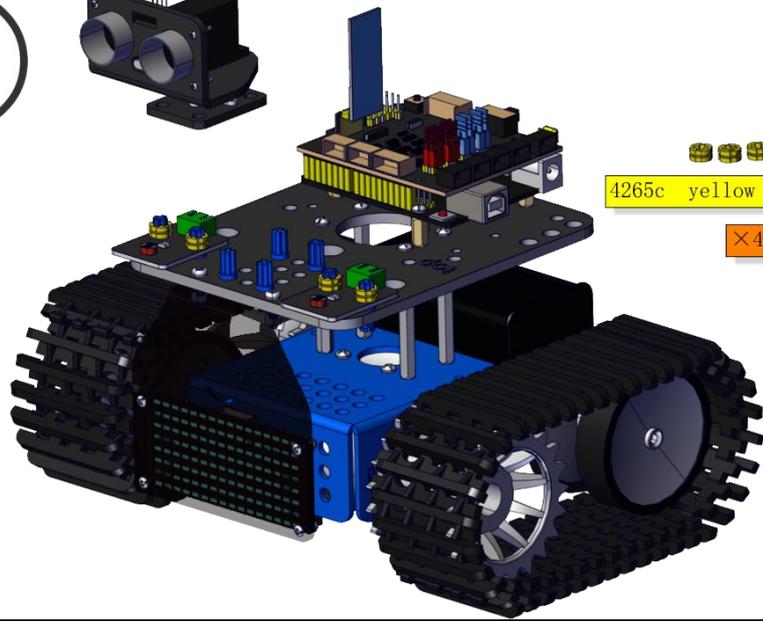


11. Assemble the ultrasonic module on the car

11



4265c yellow LEGO parts  
×4



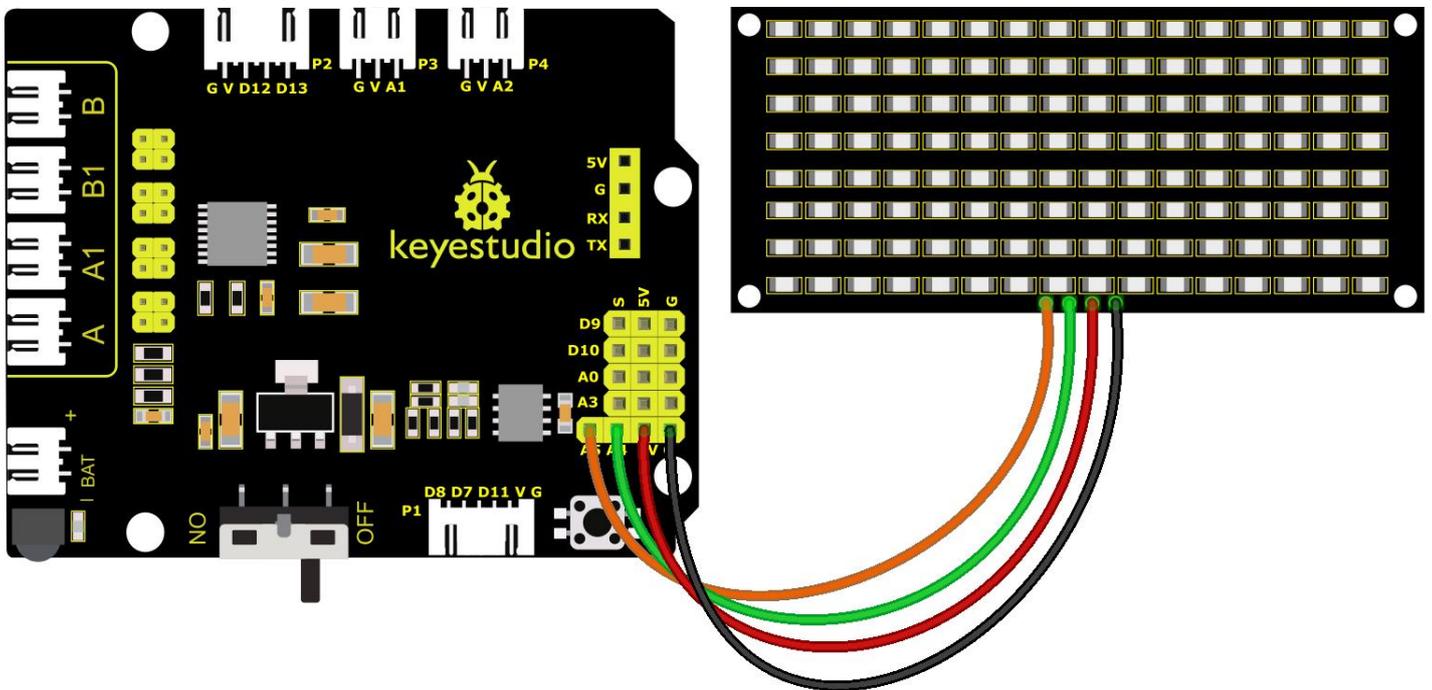
## 4.Wiring

1. Connect the 8\*16 LED Dot matrix to the motor drive shield with 4P HX-2.54 Dupont Wire

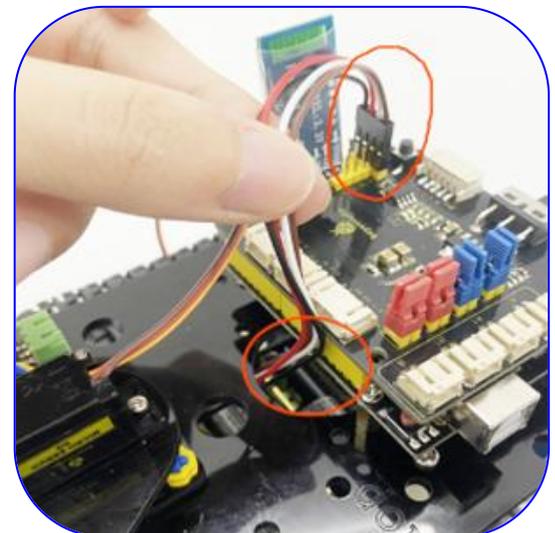
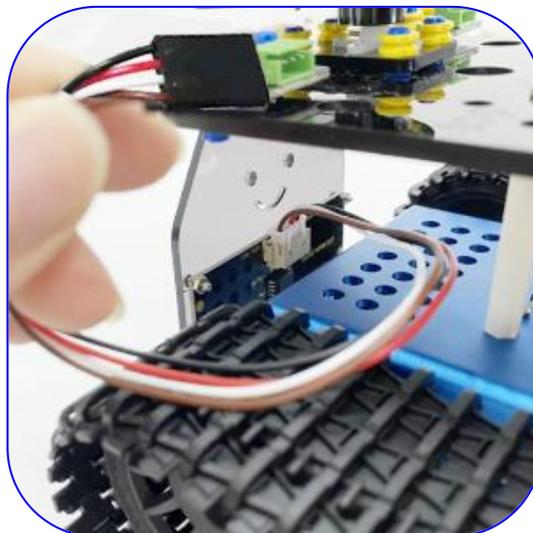
Pin Connection Table	
Pin of the 8*16 LED matrix	Pin of the Motor Driver Shield
GND	G
VCC	5V
SDA	A4
SCL	A5



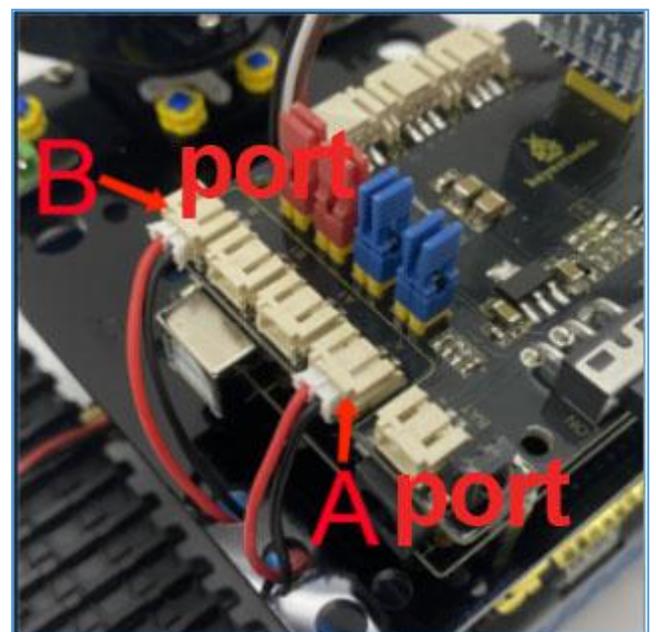
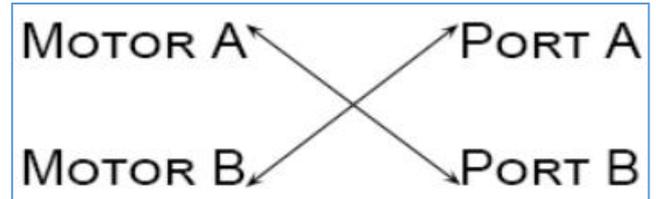
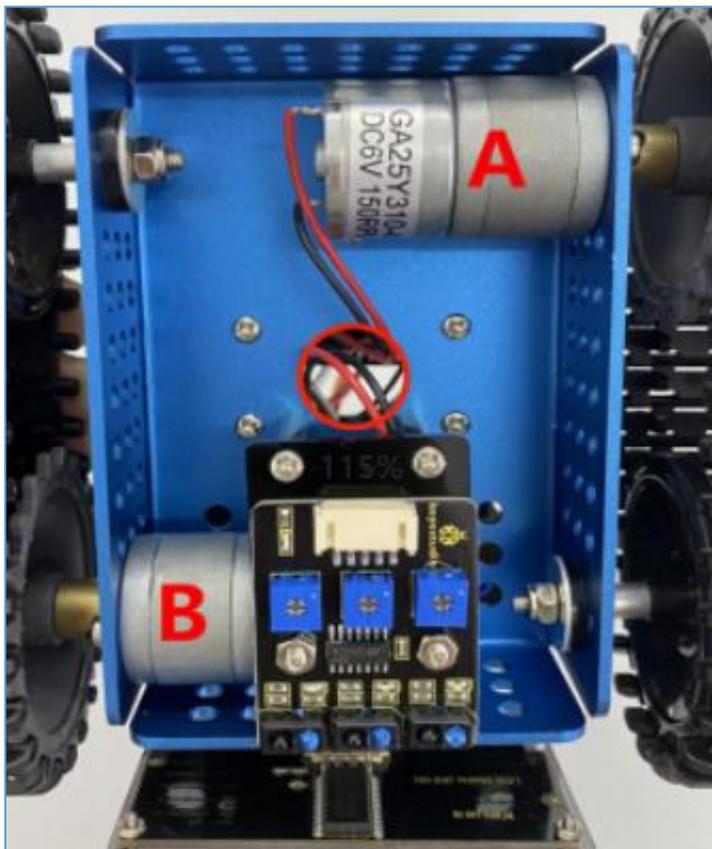
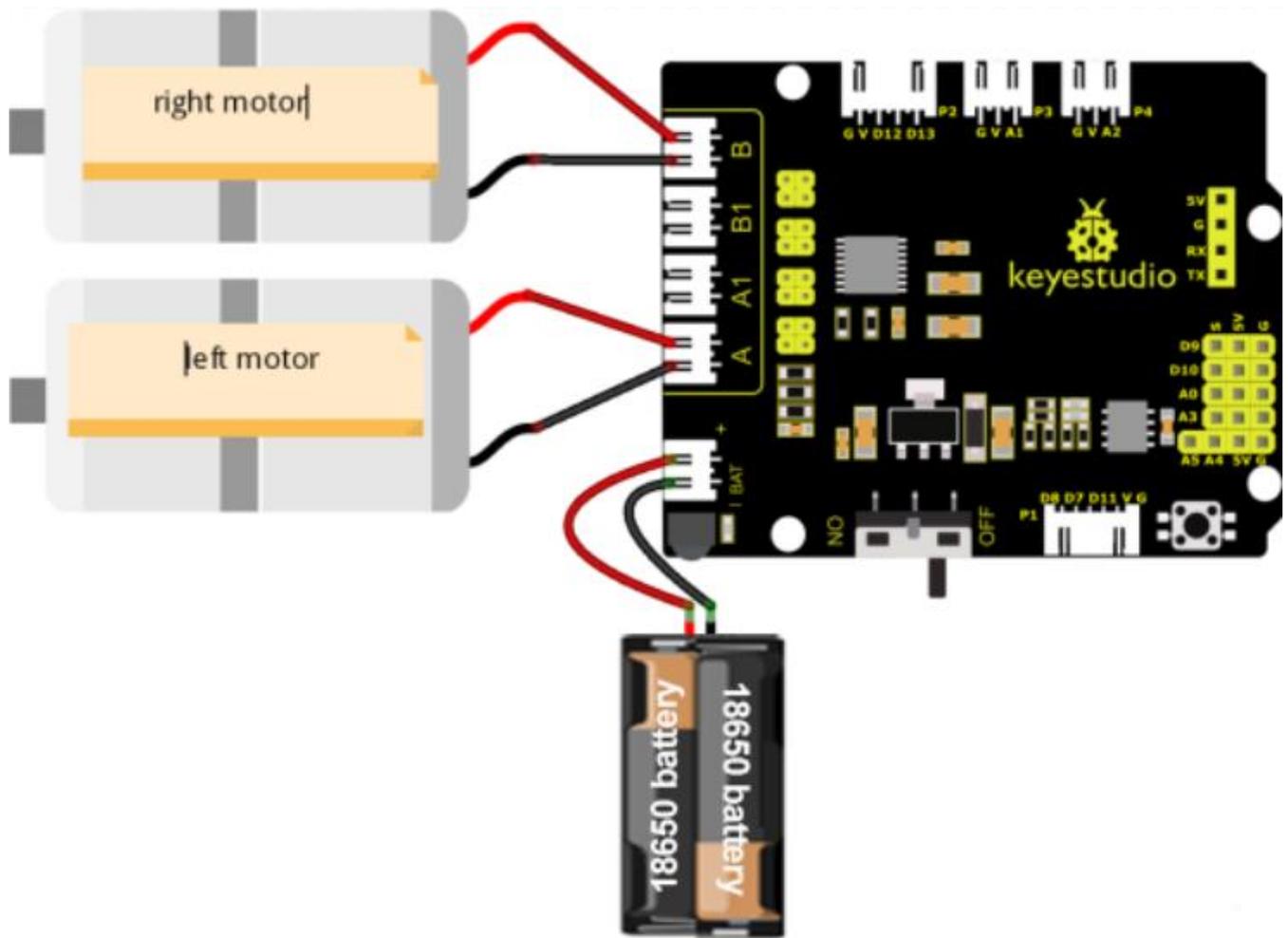
4P HX-2.54  
Dupont Wire\*1



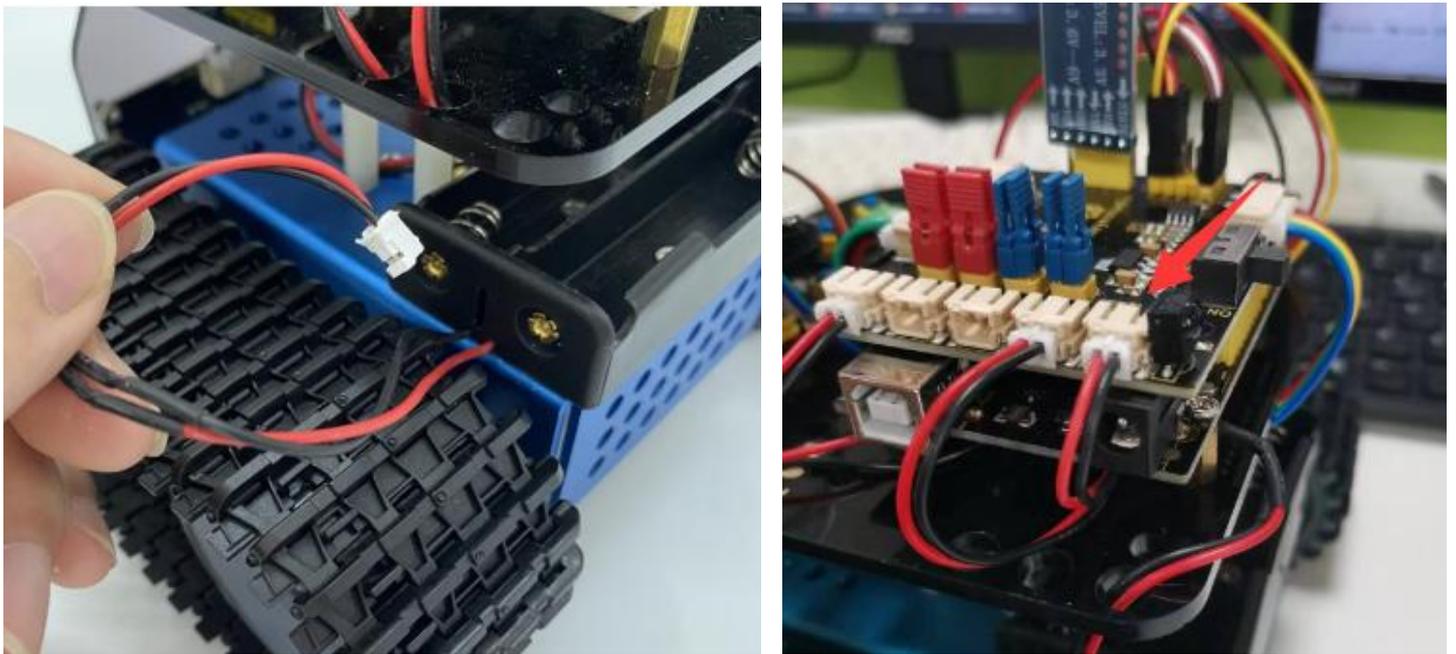
Connect one end of the wire to the 8\*16 LED matrix, the other end through the hole in the middle of the top acrylic board, and connect it to the Motor Driver Shield



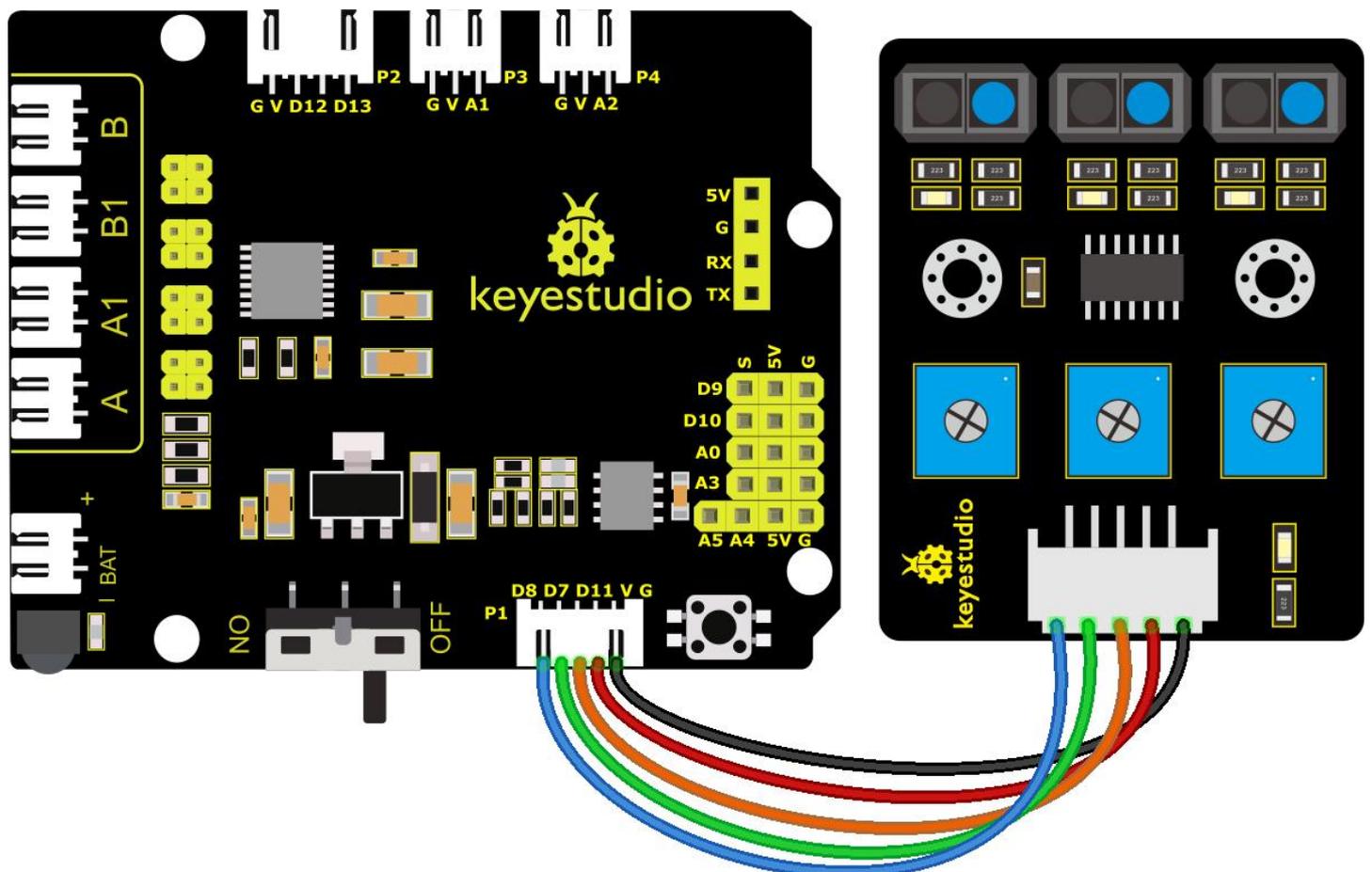
2. Connect the motor A to the **B port** of the Motor Driver Shield and the motor B to the **A port** of the Motor Driver Shield.



3. Connect the wire of the battery case to the **BAT port** of the Motor Driver Shield.



4. Connect the line tracking sensor to the P1 port of the Motor Driver Shield with 5P JST-PH2.0MM Dupont Wire.



One end of the 5P JST-PH2.0MM Dupont Wire is connected to the port of the line tracking module.

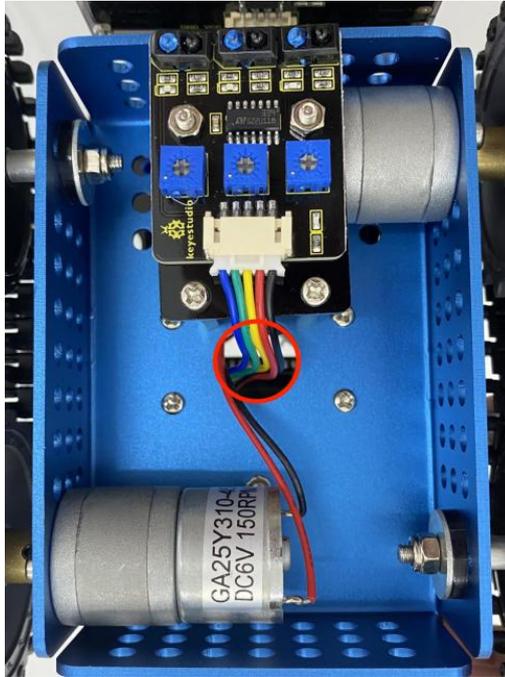
The other end first passes through the hole in the middle of the metal body of the robot and then connects to the P1 port of the Motor Driver Shield.



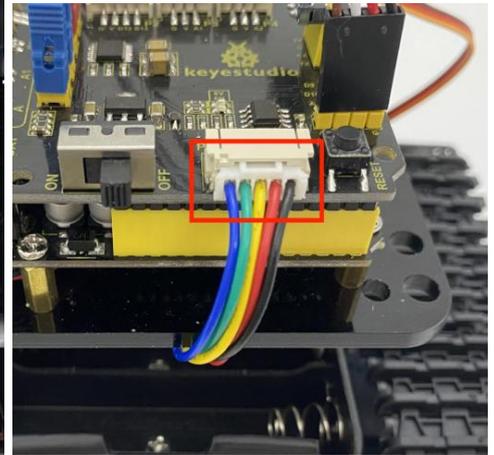
5P JST-PH2.0MM Dupont Wire\*1

1

2



3

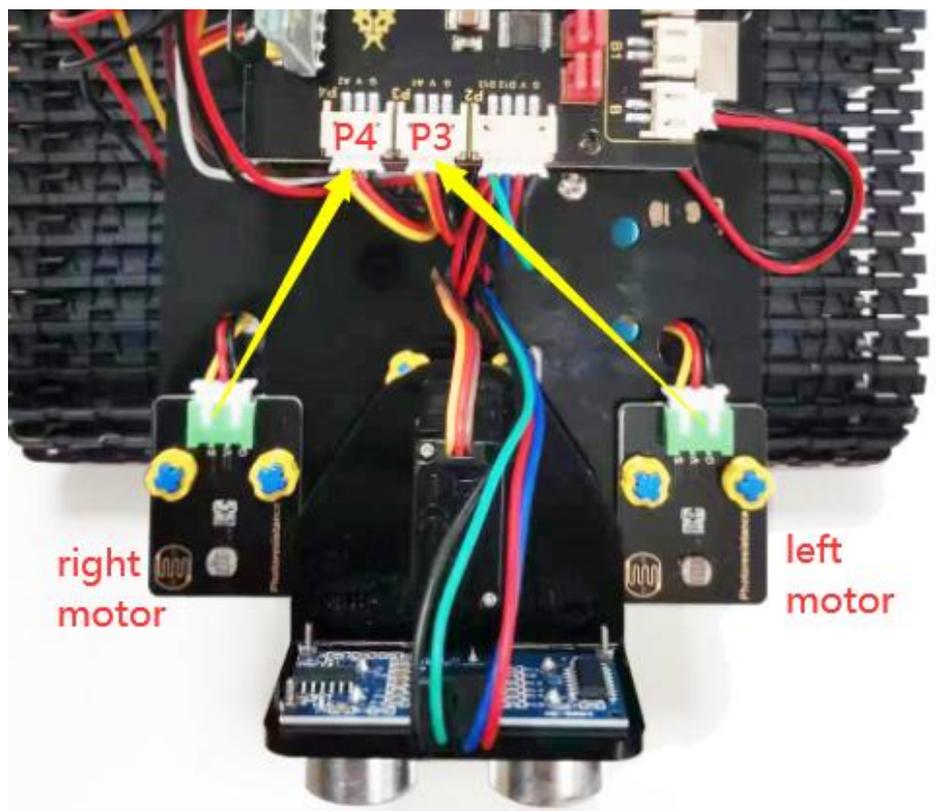


5. Connect the left photoresistor module to the P3 port of the Motor Driver Shield and the right photoresistor module to the P4 port of the Motor Driver Shield.



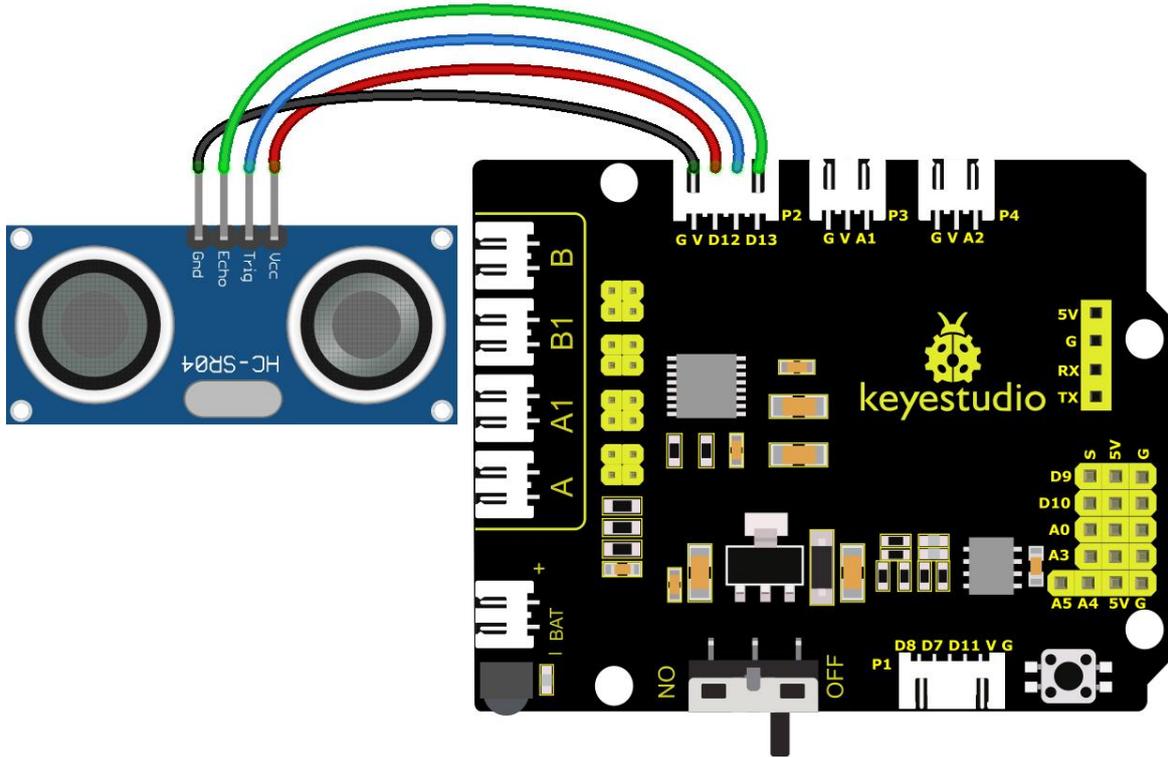
3P-3P XH2.54 to PH2.0 Dupont Wire\*2

Connect one end of the wire to the photoresistor module, and the other end to go through the two holes of the top acrylic board, and then connect to the Motor Driver Shield.

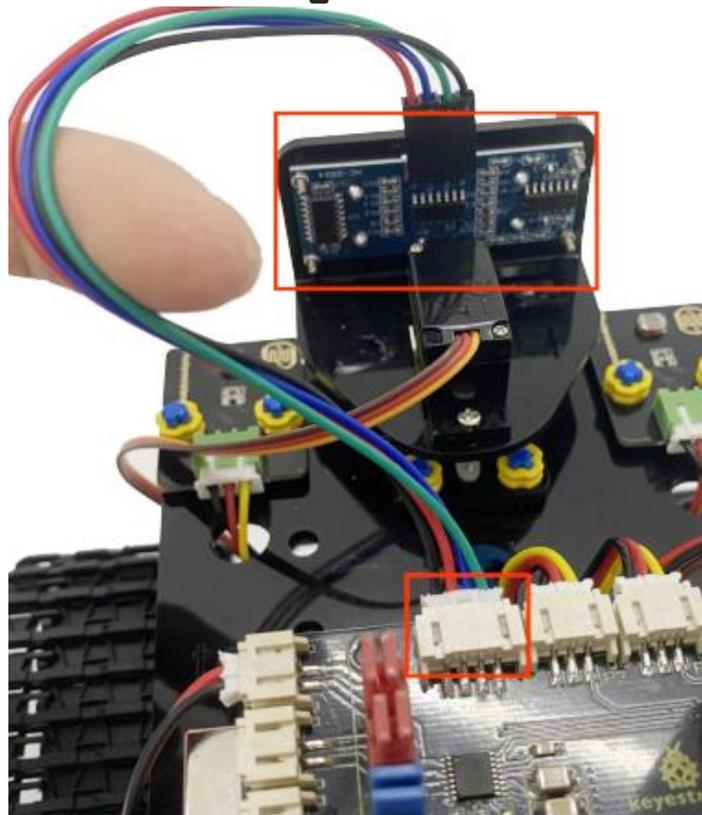


6. Connect the ultrasonic sensor to the **P2 port** of the Motor Driver Shield.

Pin Connection Table	
Pin of the Ultrasonic Sensor	Pin of the Motor Driver Shield
Vcc	V
Trig	D12
Echo	D13
Gnd	G

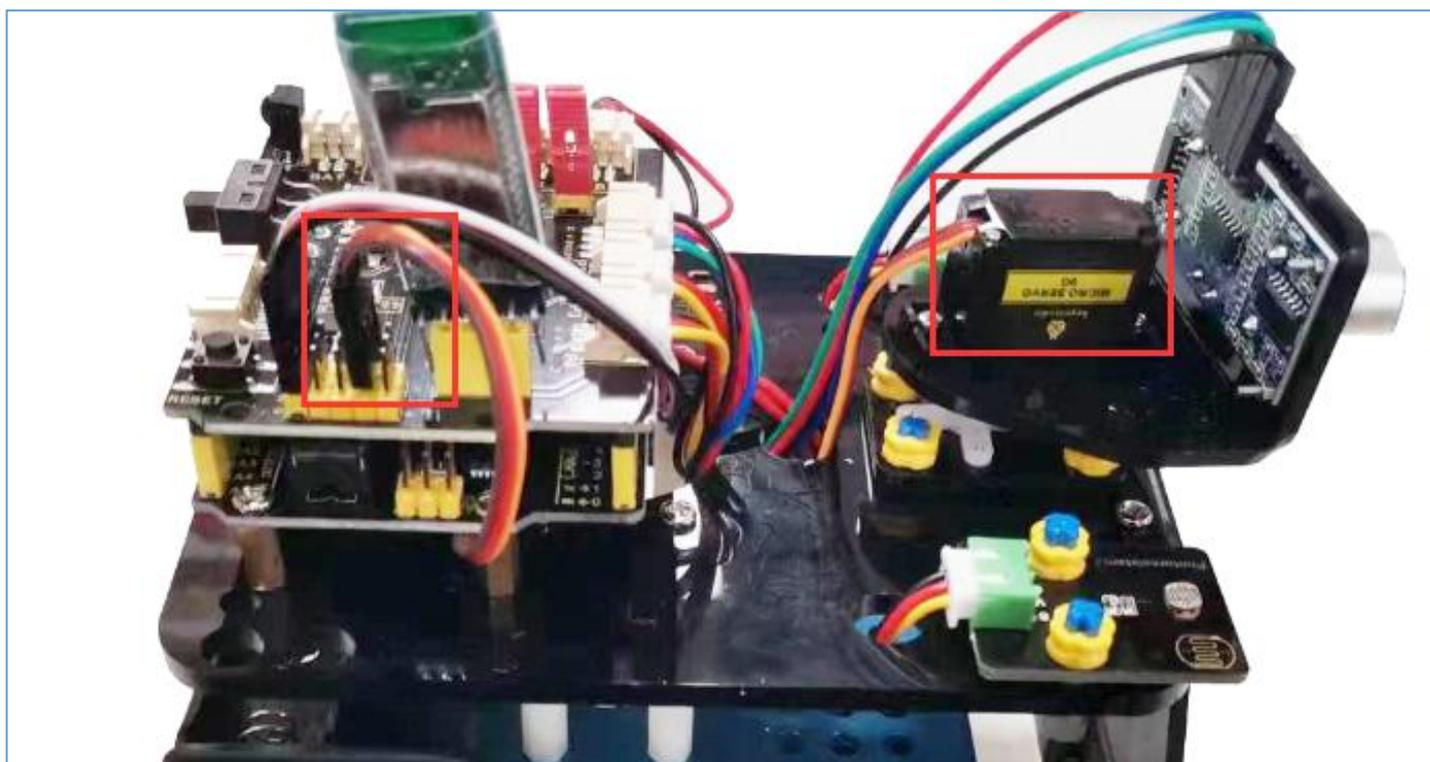


4P M-F PH2.0mm to 2.54  
Dupont Wire \*1



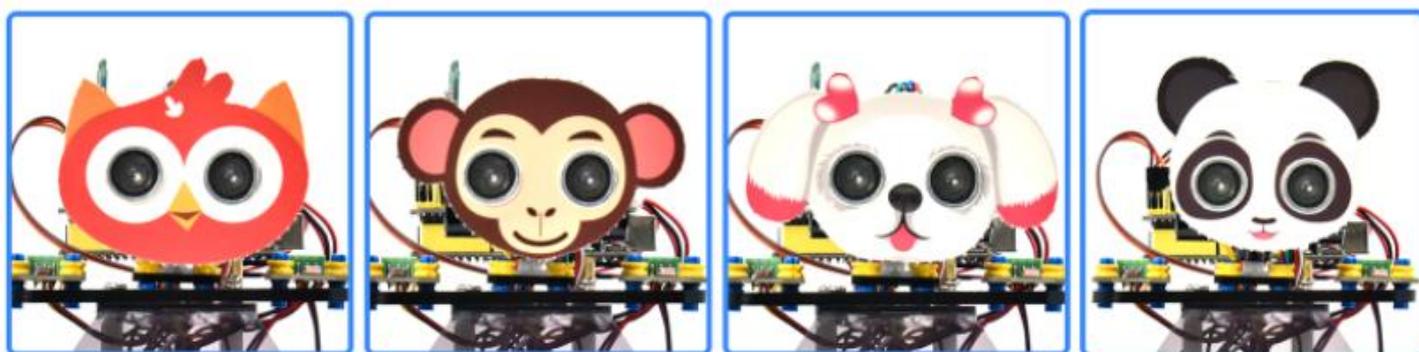
7. Connect the servo to the D10 pin of the Motor Driver Shield.

Pin Connection Table	
Pin of the Servo	Pin of the Motor Driver Shield
Brown	G
Red	V(5V)
Orange	D10



**Assembly of the robot is complete!**

We can also put animal cards on the robot's ultrasonic module to make it look more interesting.



## 5.Parameters of the robot

Working voltage: 5v

Input voltage: 6-9V

Maximum output current: 1.5A

Maximum power dissipation: 32W

Motor speed: 5V 200 rpm / min

Motor drive mode: dual H bridge drive(HR8833)

Ultrasonic induction angle:  $<15^{\circ}$

Ultrasonic detection distance: 2cm-300cm

Infrared remote control distance: 10 meters (measured)

BT remote control distance: 30 meters (measured)

### Parameters of Keystudio 8833 Motor Driver Shield

The Keystudio 8833 motor driver shield is compatible with the Arduino UNO. Just stack it onto the control board to use it.

Voltage USB 5V, DC 6-9V

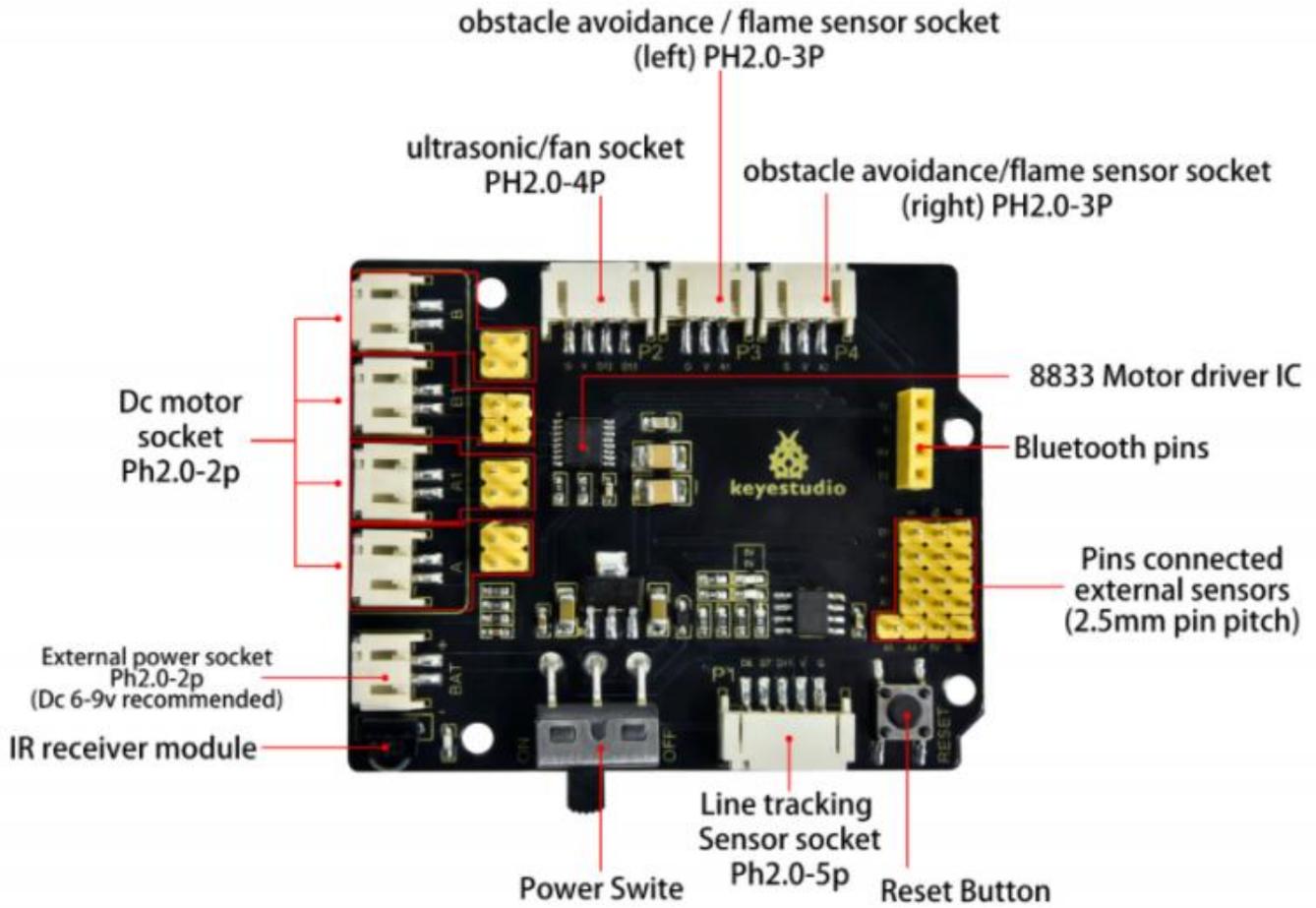
Current Max current is 3A, the real output is around 1A (2A, when the motor is loaded)

Max power is 27W; in fact, it is 9W

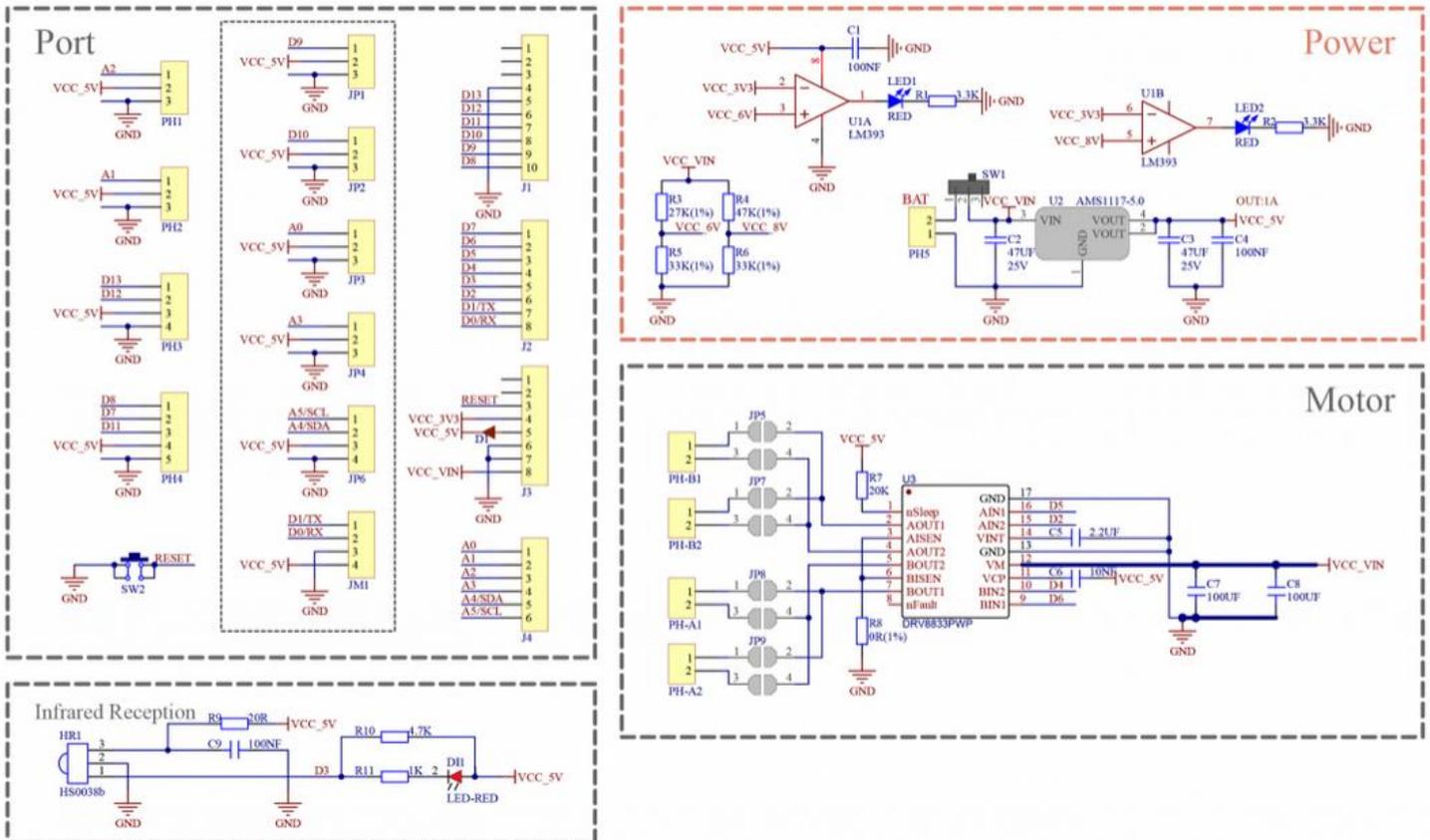
Working Temperature:  $-10\sim 50^{\circ}\text{C}$

Size: 69\*56\*18mm

Weight: 25.5g



### Schematic Diagram:



# 6.Lessons

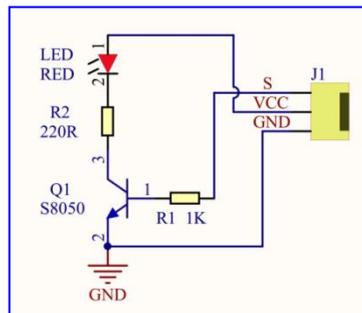
In this section, we'll start from testing individual electronic modules to making the robot perform complex tasks. Step by step, we'll show you how the tank robot works.

It is highly recommended that you complete all of the courses as this will help to verify that each electronic module is working correctly or verify that the robot is wired correctly, which is critical as they can affect whether the entire robot can work as expected in the end.

**NOTE:** Don't connect the Bluetooth module to the robot when uploading code, it will occupy the communication port and causes code to fail to upload. You can connect it later in a project that involves Bluetooth functionality.

## Lesson 1.1: LED Blinks

### (1) Description:



LED, the abbreviation of light emitting diodes, consists of Ga, As, P, N chemical compounds and so on. The LED can flash in diverse colors by altering the delay time in the test code. When in control, power on GND and VCC, the LED will be on if S end is in high level; nevertheless, it will go off.

### (2) Parameters:

Control interface: digital port

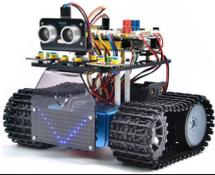
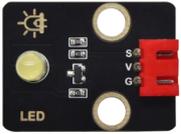
Working voltage: DC 3.3-5V

Pin spacing: 2.54mm

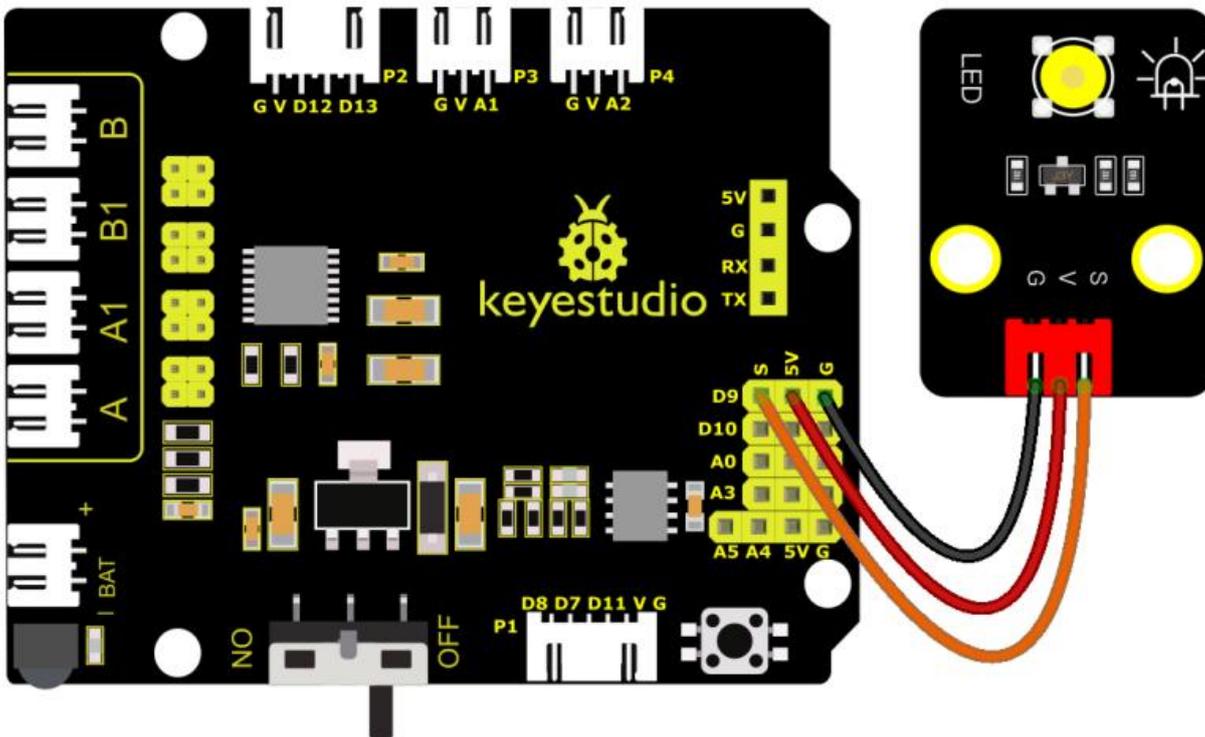
LED display color: yellow



**(3) You need to prepare:**

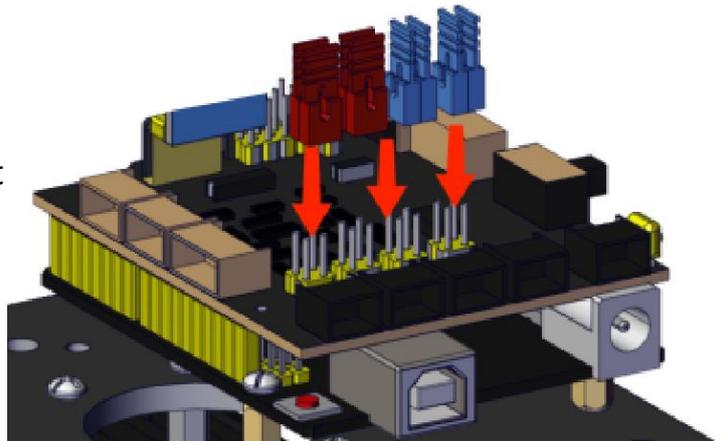
Robot without BT Module*1	USB Cable*1	Yellow LED Module*1
		
3P-3P XH2.54 to 2.54 Dupont Wire*1	Computer*1	
		

**(4) Connection diagram of the led module:**

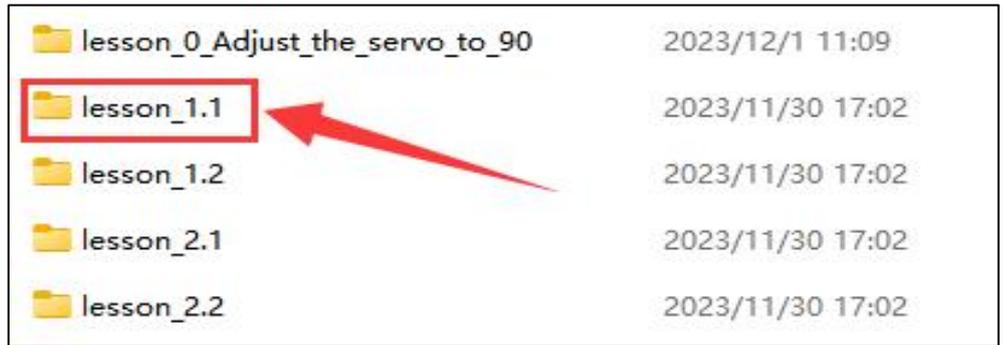


**Note:**

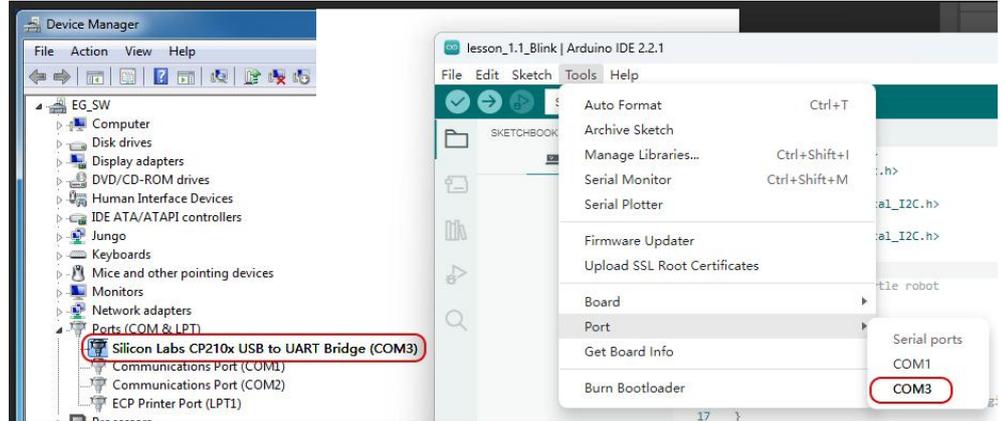
1. LED is connected to D9 port
2. Remember to install jumper caps onto the shield
3. Don't connect the Bluetooth module to the robot when uploading code.



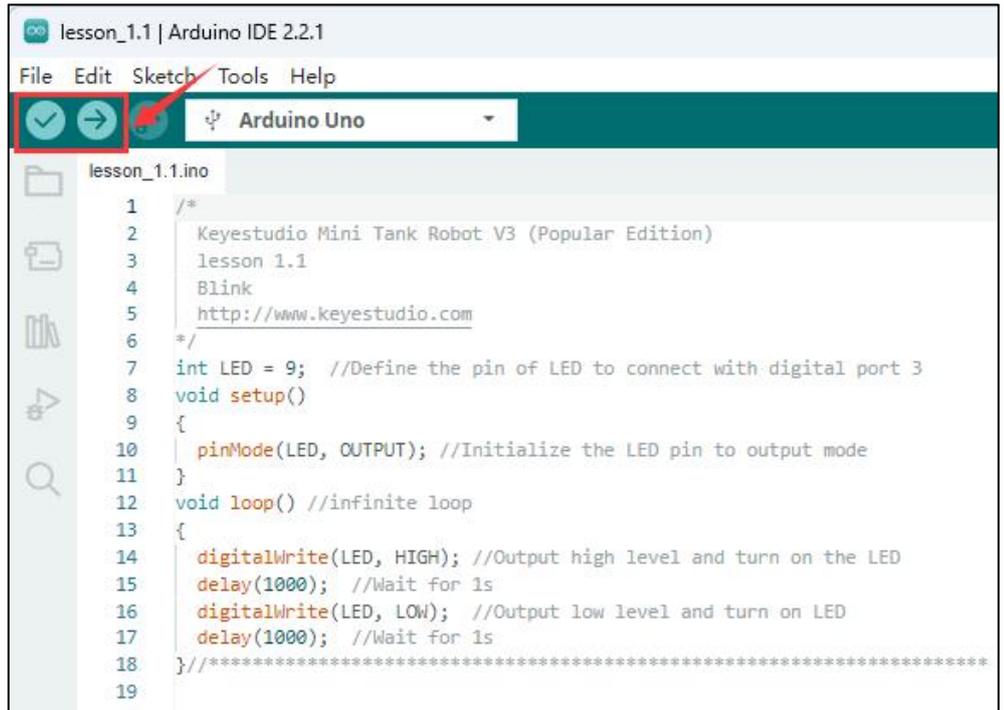
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_1.1** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.



3. Click upload >>> done uploading.  
**Test Results:** LED blinks at the interval of 1s



### (5) Code Explanation

pinMode(LED, OUTPUT) - This function can denote that the pin is INPUT or OUTPUT

digitalWrite(LED, HIGH) - When pin is OUTPUT, we can set it to HIGH(output 5V) or LOW(output 0V)

## Lesson 1.2: Changing the blinking frequency

>>>>>>>> *This lesson is an expansion of the Lesson 1.1*

Hardware required for this lesson, the configuration of the arduino IDE, and the wiring between the LDE module and the 8833 Motor Driver Shield is in the same way as [Lesson 1.1](#).

1. Open the INO file inside the [lesson\\_1.2](#) folder with Arduino IDE.

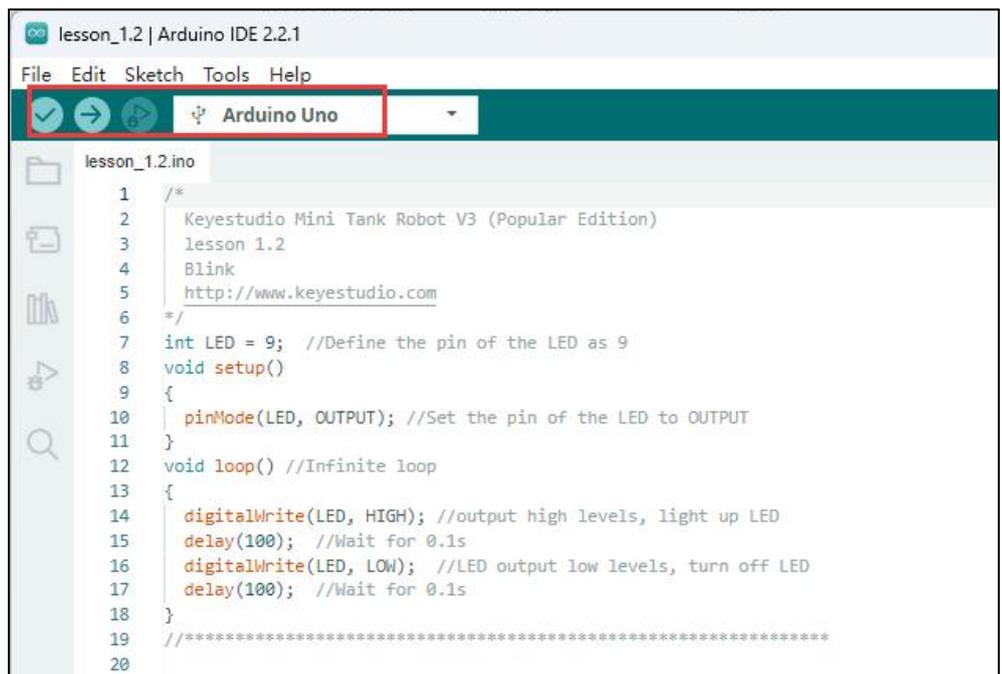


2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click upload >>>done uploading.

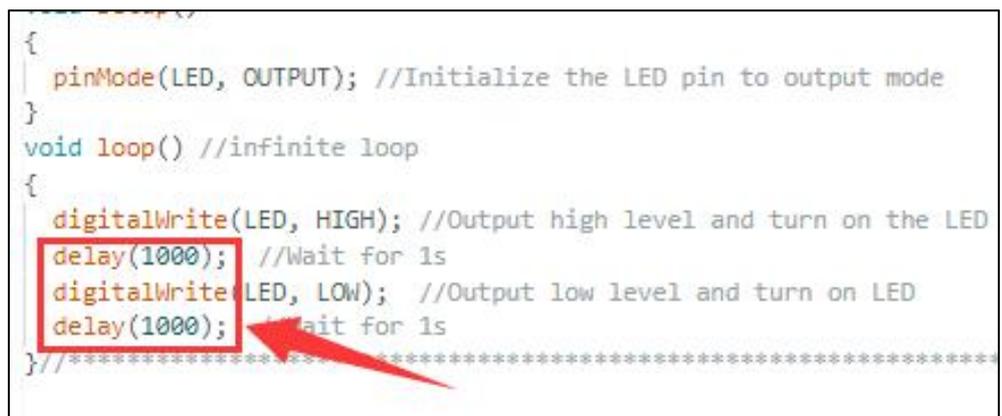
**Test Results:** You will see the LED flashes faster.



4. You can control the blinking frequency of the LDE by modifying the delay value in the code.

The larger the delay value, the slower the LDE blinks.

The smaller the delay value, the faster the LDE blinks.

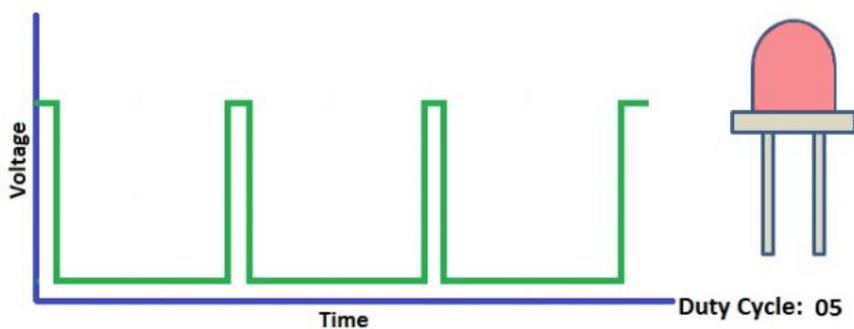


## Lesson 2.1: Adjust the Brightness of the LED

Hardware required for this lesson, the configuration of the arduino IDE, and the wiring between the LED module and the 8833 Motor Driver Shield is in the same way as [Lesson 1.1](#).

In this project, we will control brightness of LED through PWM.

Pulse Width Modulation, or PWM, is a technique for getting analog results with digital means. Digital control is used to create a square wave, a signal switched between on and off. This on-off pattern can simulate voltages in between the full Vcc of the board (e.g., 5 V on UNO, 3.3 V on a MKR board) and off (0 Volts) by changing the portion of the time the signal spends on versus the time that the signal spends off. The duration of "on time" is called the pulse width. To get varying analog values, you change, or modulate, that pulse width. If you repeat this on-off pattern fast enough with an LED for example, the result is as if the signal is a steady voltage between 0 and Vcc controlling the brightness of the LED.



For Arduino digital port voltage outputs, there are only LOW and HIGH levels, which correspond to the voltage outputs of 0V and 5V respectively. You can define LOW as “0” and HIGH as “1” , and let the Arduino output five hundred ‘0’ or ‘1’ within 1 second. If output five hundred ‘1’ , that is 5V; if all of which is ‘0’ ,that is 0V; if output 250 01 pattern, that is 2.5V.

This process can be likened to showing a movie. The movie we watch are not completely continuous. Actually, it generates 25 pictures per second, which cannot be told by human eyes. Therefore, we mistake it as a continuous process. PWM works in the same way. To output different voltages, we need to control the ratio of 0 and 1. The more ‘0’ or ‘1’ output per unit time, the more accurate the control.

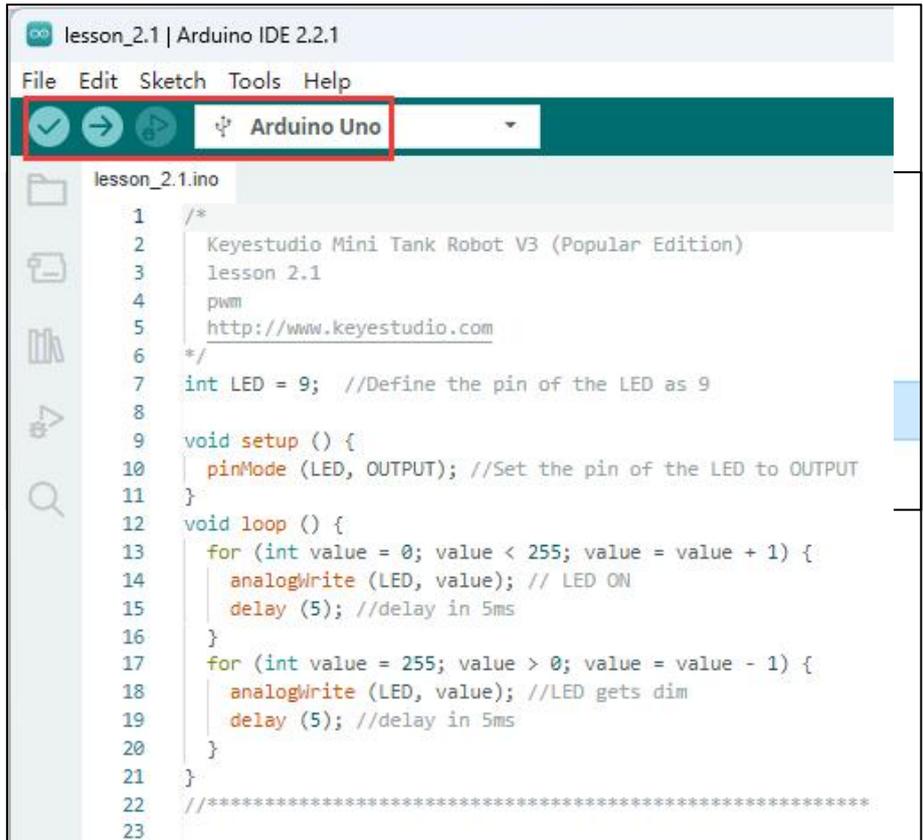
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_2.1** folder with Arduino IDE.



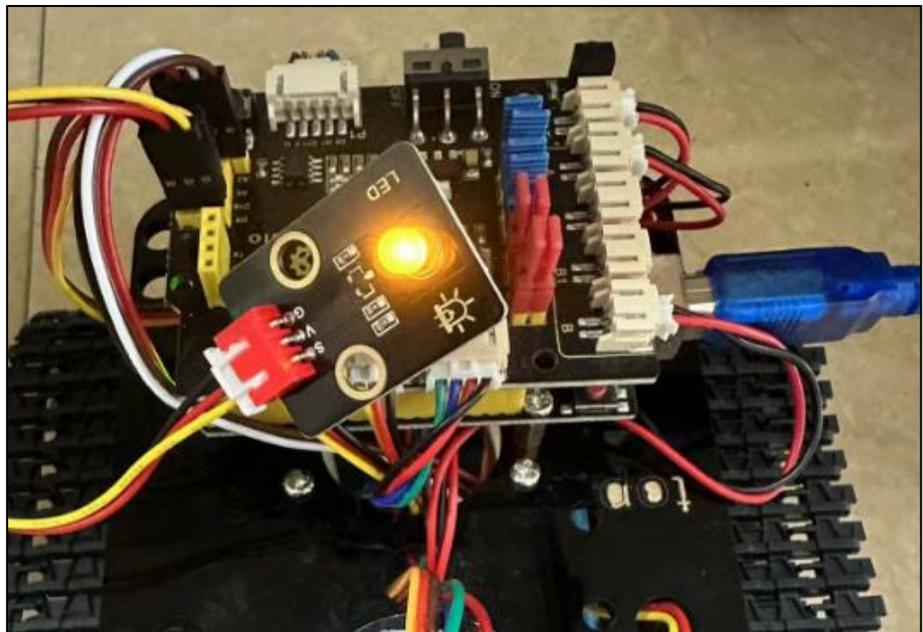
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click compile >>>done compiling.  
Click upload >>>done uploading.



**Test Results:** The LED gradually changes from bright to dark, rather than turning on and off.

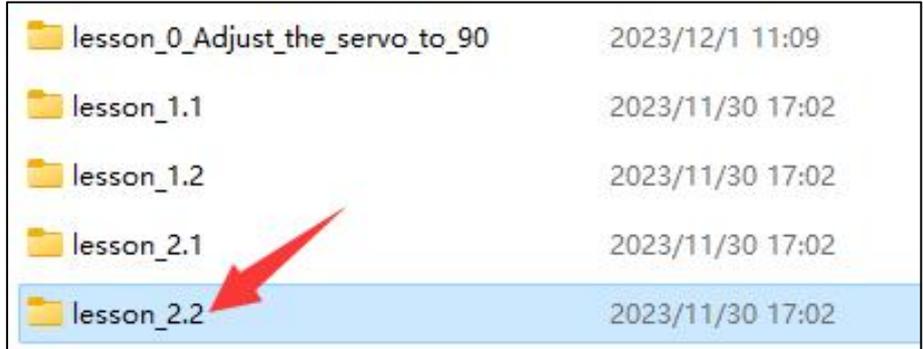


## Lesson 2.2: Slow down the change of the brightness of the LED

>>>>>>>>*This lesson is an expansion of the Lesson 2.1*

Hardware required for this lesson, the configuration of the arduino IDE, and the wiring between the LDE module and the 8833 Motor Driver Shield is in the same way as [Lesson 1.1](#).

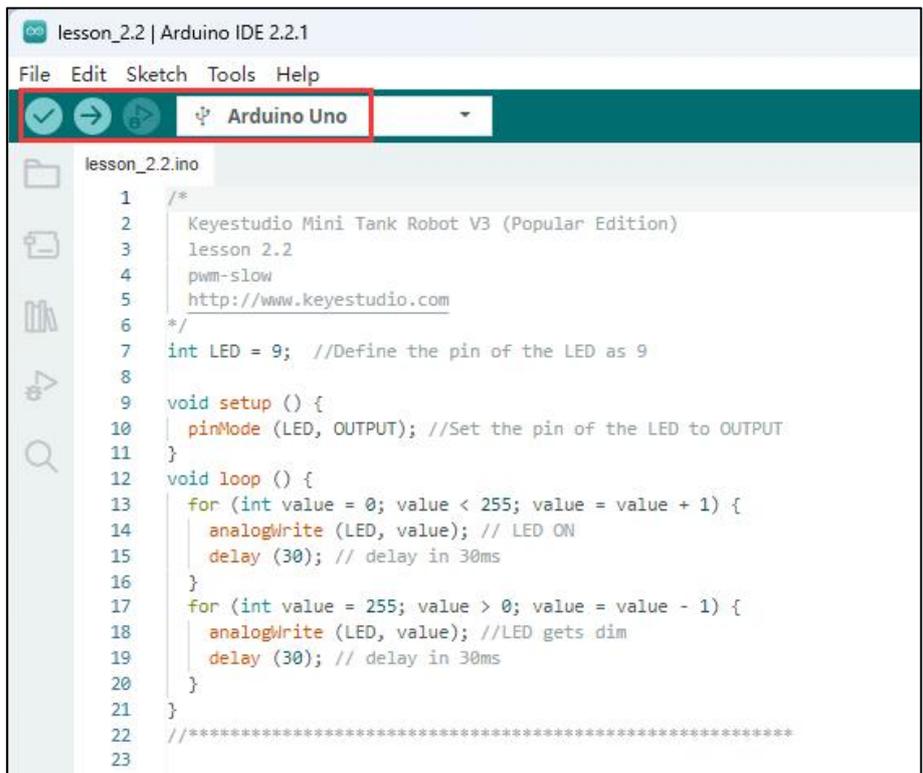
1. Open the INO file inside the [lesson\\_2.2](#) folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click compile >>>done compiling. Click upload >>>done uploading.

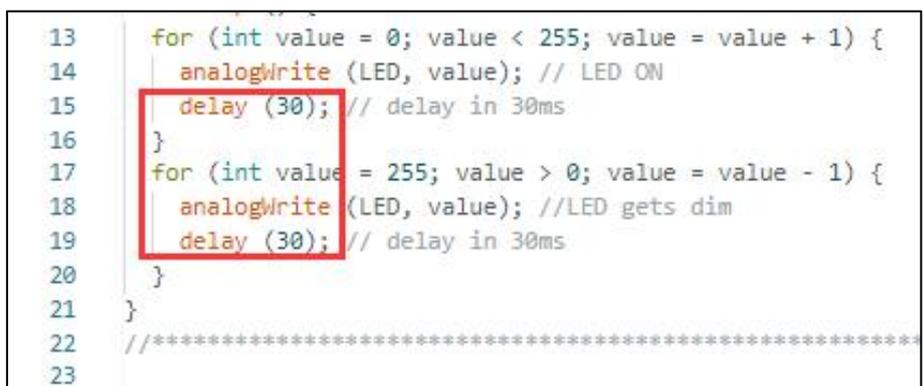


**Test Results:** Compared to the last lesson, the brightness of the LED changes slower.

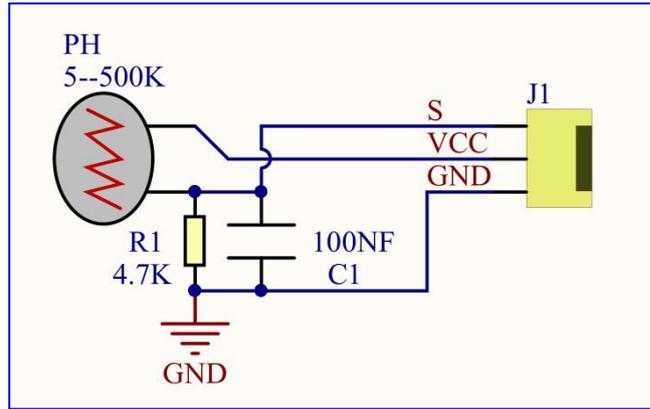
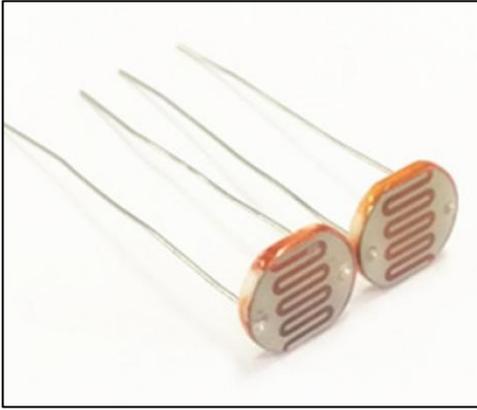
4. You can control the change frequency of the brightness of the LDE by modifying the delay value in the code.

The larger the delay value, the slower the LDE changes.

The smaller the delay value, the faster the LDE changes.



## Lesson 3.1: Analog value of photoresistor



The photosensitive resistor is a special resistor made of a semiconductor material such as a sulfide or selenium, and a moisture-proof resin is also coated with a photoconductive effect. The photosensitive resistance is most sensitive to the ambient light, different illumination strength, and the resistance of the photosensitive resistance is different. We use the photosensitive resistance to design the photosensitive resistor module. The module signal is connected to the microcontroller analog port. When the light intensity is stronger, the larger the analog port voltage, that is, the simulation value of the microcontroller is also large; in turn, when the light intensity is weaker, the smaller the analog port voltage, that is, the simulation value of the microcontroller is also small. . In this way, we can read the corresponding analog value using the photosensitive resistor module, and the intensity of the light in the inductive environment.

### Parameters:

Photosensitive resistance resistance value: 5K Ou-0.5m

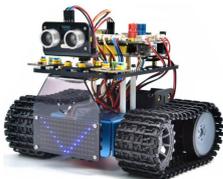
Interface type: simulation port A0, A1

Working voltage: 3.3V-5V

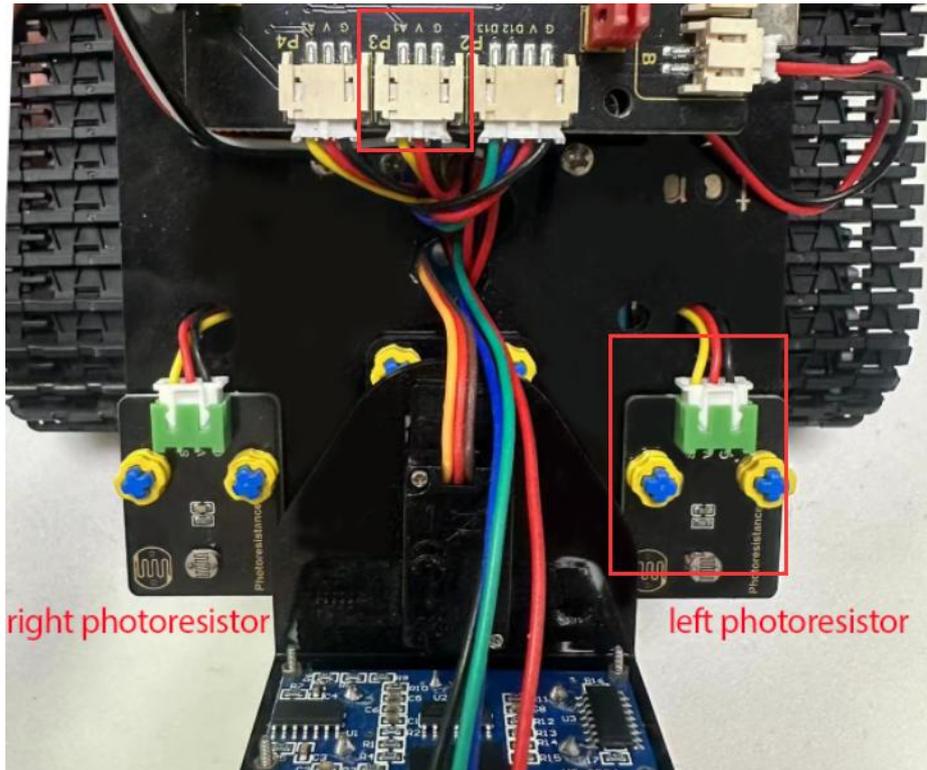
Pin spacing: 2.54mm



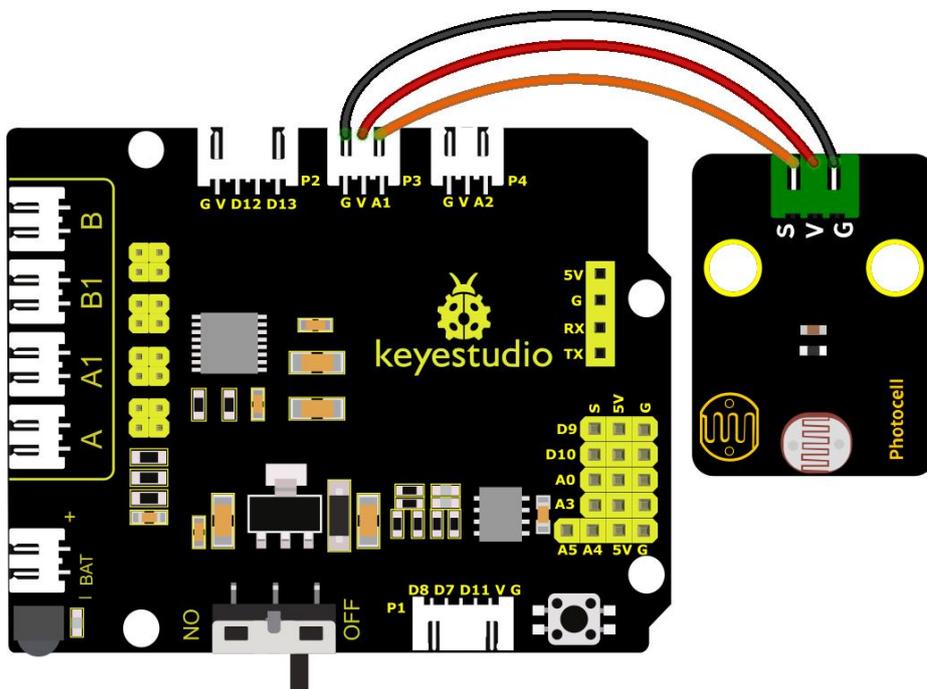
**You need to prepare:**

Robot without BT module*1	USB Cable*1	Computer*1
		

What we are going to test next is the photoresistor module on the left side of the robot.



The left photoresistor is connected to A1/P3 of the motor drive shield.



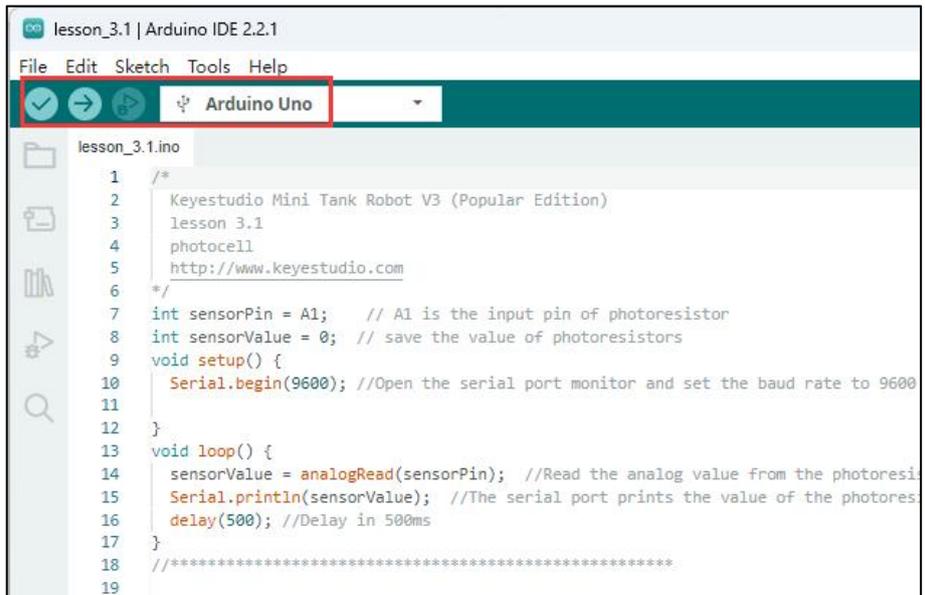
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_3.1** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click compile >>>done compiling.  
Click upload >>>done uploading.



**Test Results:** Open the Arduino IDE serial monitor and set its baud rate to 9600, serial monitor will display analog the value of the ambient light. Use your palm to cover the photoresistor module on the left side of the robot, you will be able to change the intensity of the light around it and the analog value of the serial monitor will change accordingly.



### Code Explanation:

`analogRead(sensorPin)`: read the analog value of photoresistors

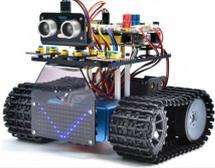
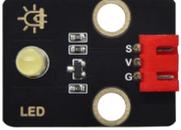
`Serial.begin(9600)`: initialize serial port and set baud rate to 9600

`Serial.println`: serial prints

## Lesson 3.2: Control the brightness of the LED by PWM

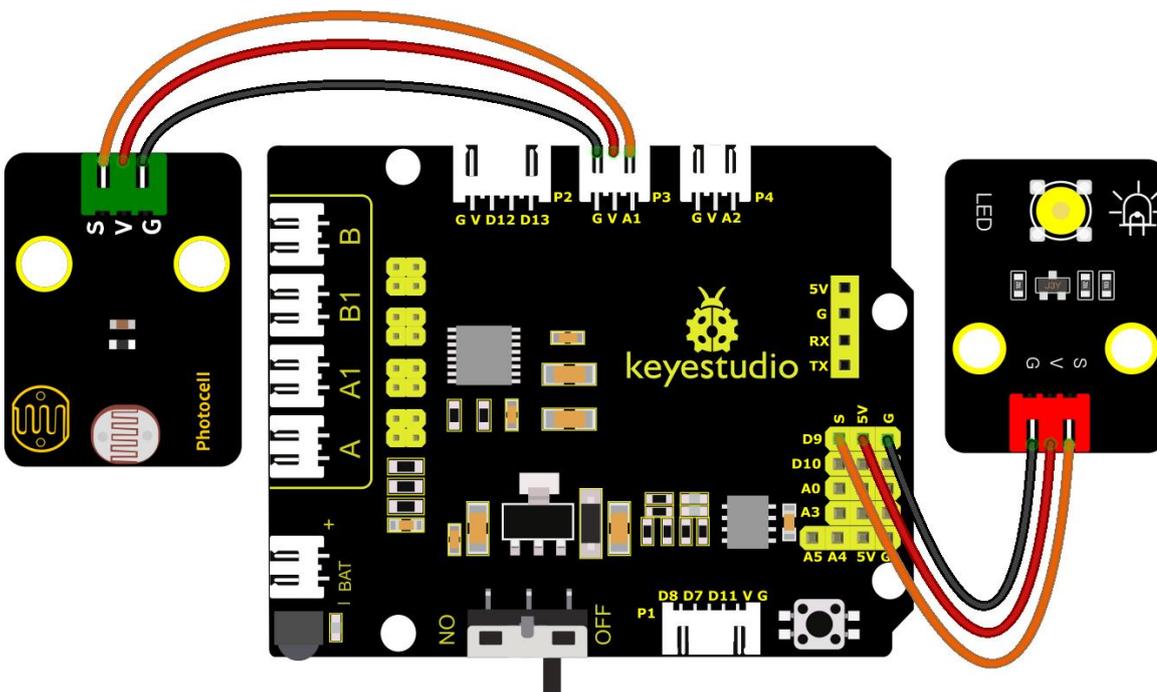
>>>>>>>> *This lesson is an expansion of the Lesson 3.1*

You need to prepare:

Robot without BT Module*1	USB Cable*1	Yellow LED Module*1
		
3P-3P XH2.54 to 2.54 Dupont Wire*1	Computer*1	
		

In this lesson, we will use the **left photoresistor module** of the robot to control the brightness of the LED module.

The brightness of the LED is controlled by PWM. We connect the LED to the PWM pin of the shield (**pin 9**).



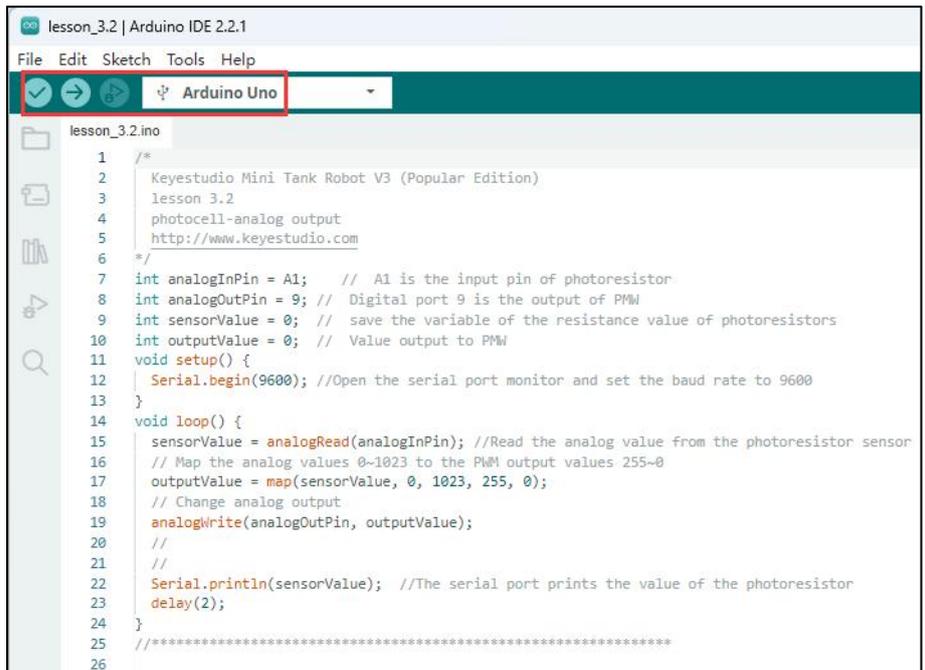
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_3.2** folder with Arduino IDE.



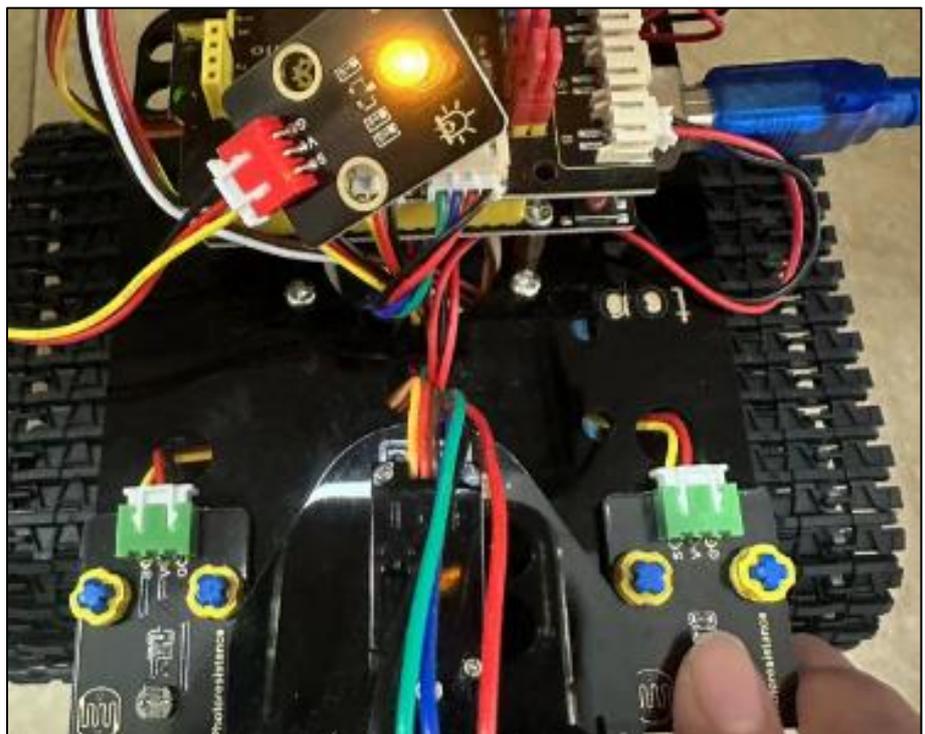
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click compile >>>done compiling.  
Click upload >>>done uploading.



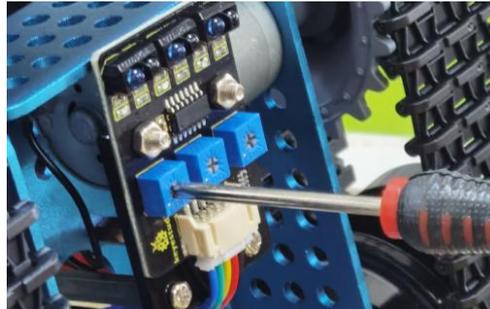
**Test Results:** Use your finger to cover the left photoresistor module on the robot, the brightness of the LDE module will change with the brightness of ambient light.



## Lesson 4.1: Digital Value of Line Tracking Sensor

The Line Tracking Sensor based on TCRT5000 is an type of infrared reflectance sensor, it detect the presence of a black line by emitting infrared (IR) light and detecting the light levels that return to the sensor. They do this using two components: an emitter and a light sensor (receiver).

On the right-hand side of the image you can see six circular components that look a bit like LEDs. The blue ones are the IR emitters and the black ones are the receivers. It also have three components called potentiometers, which adjusts the device's threshold. This is done by using a screwdriver to turn the blue dial that looks like a cross on the left-hand side of the image.



### Parameters:

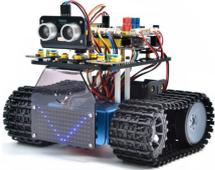
Operating Voltage: 3.3-5V (DC)

Interface: 5PIN

Output Signal: Digital signal

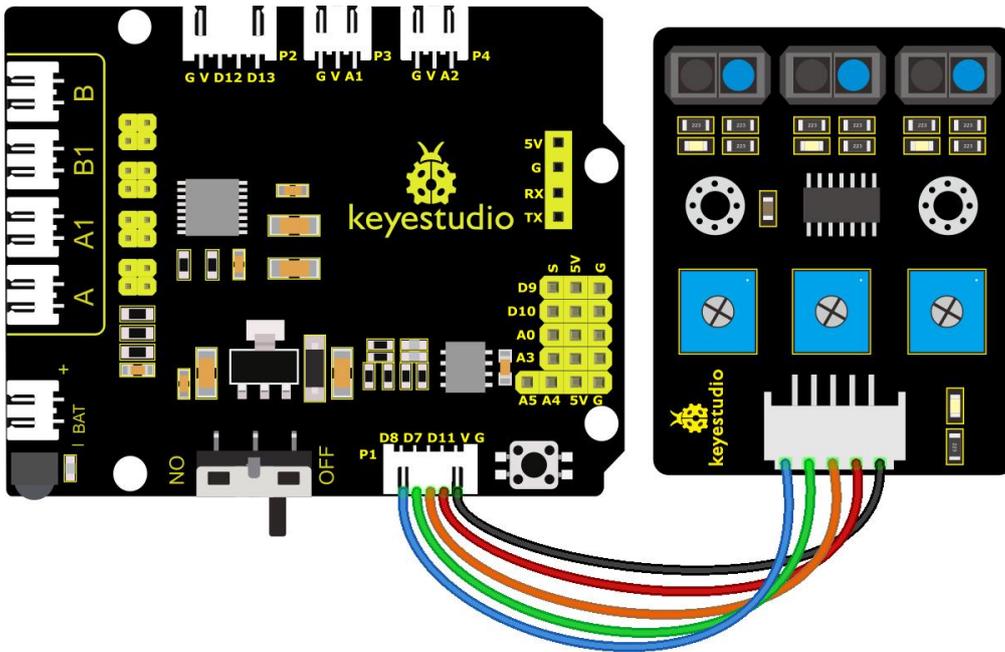
Detection Height: 0-3 cmm

### You need to prepare:

Robot without BT Module*1	USB Cable*1	Yellow LED Module*1
		
3P-3P XH2.54 to 2.54 Dupont Wire*1	Computer*1	
		

**Note:** the line tracking sensor is installed at the bottom of the robot.

The The Line Tracking Sensor is connected to the D8, D7 and D11 of the motor drive shield.



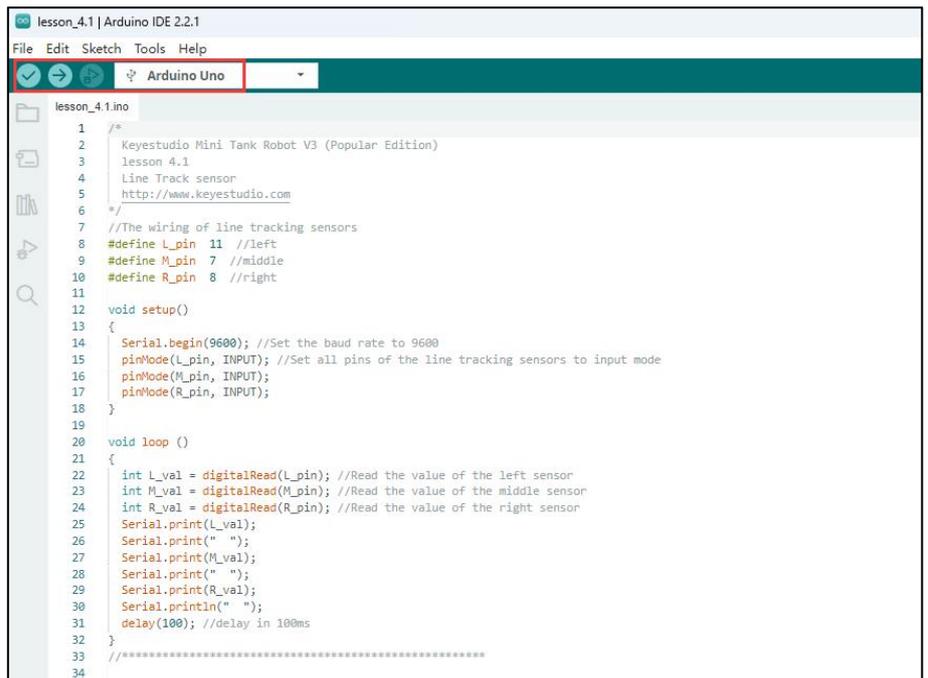
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the [lesson\\_4.1](#) folder with Arduino IDE.



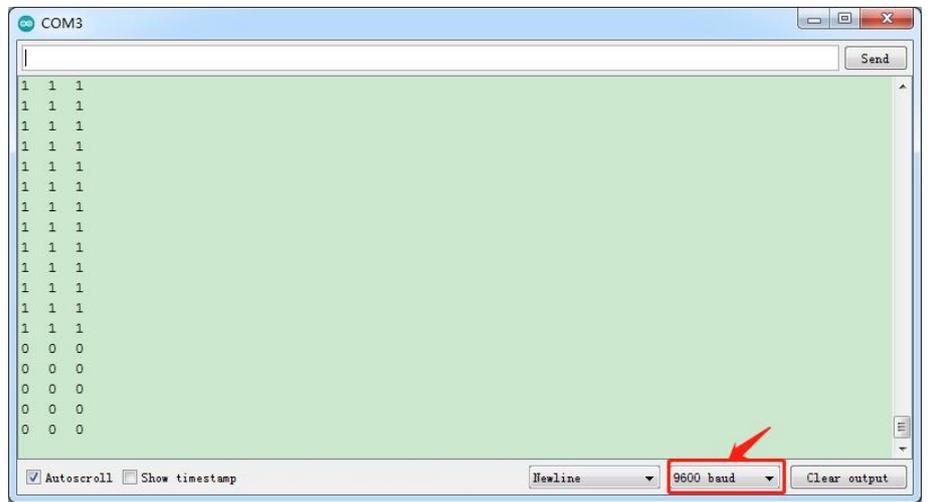
4. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

5. Click upload >>>done uploading.



**Test Results:** open serial monitor and set the baud rate to 9600. When the Line Tracking Sensor does not receive a return signal, the serial port monitor displays a value of 1 (high level). When the Line Tracking Sensor is covered by paper or other objects, the value becomes 0.

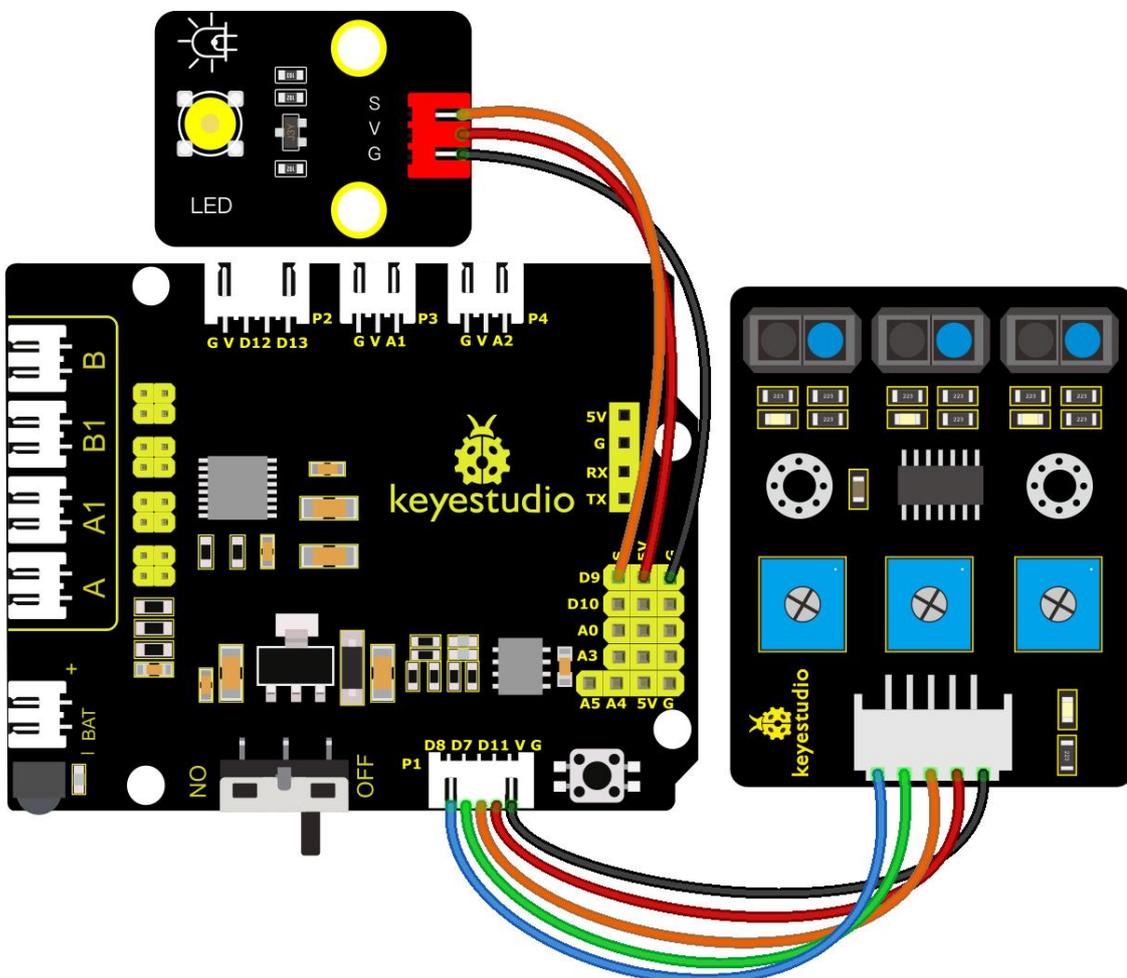


**Code Explanation:**

- Serial.begin(9600)- Initialize serial port, set baud rate to 9600
- pinMode- Define the pin as input or output mode
- digitalRead-Read the state of pin, which are generally HIGH and LOW level

**Lesson 4.2: Use Line Tracking Sensor to control LED**

The The Line Tracking Sensor is connected to the D8, D7 and D11 of the motor drive shield. The The LED module is connected to the D9 of the motor drive shield.



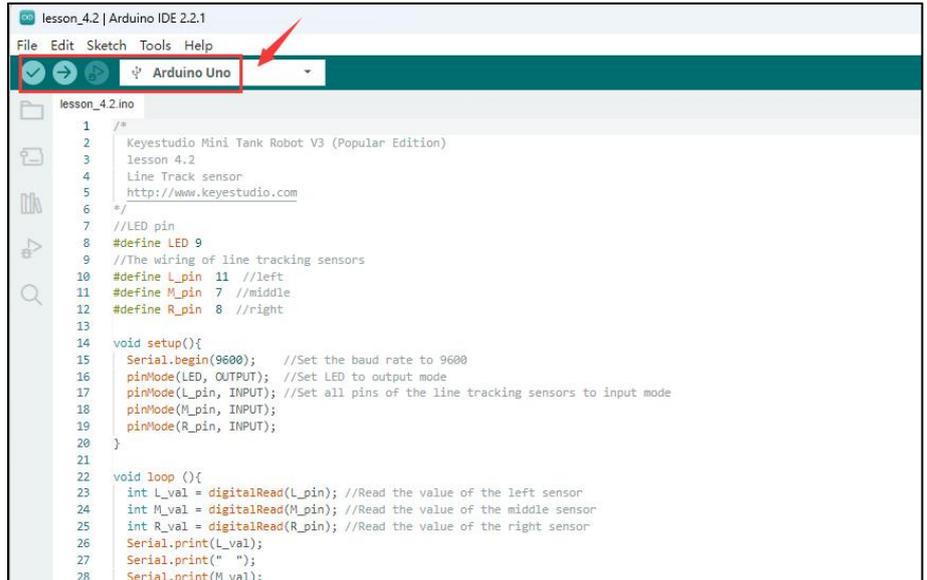
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_4.2** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click upload >>>done uploading.

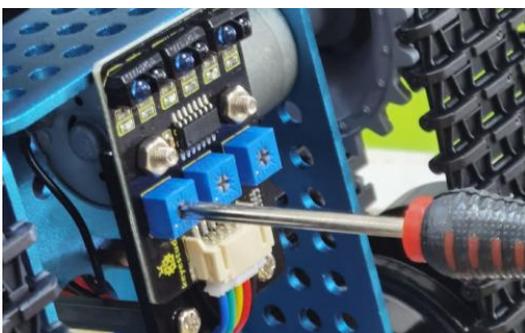


**Test Results:** When an object (such as paper or finger) approaches the line-following sensor, the sensor detects the return signal emitted by itself, and the LED module lights up. When the sensor does not detect any return signal, and the LED module turns off.



If the Line Tracking Sensor does not work as expected, you will need to use a screwdriver to adjust its potentiometer to make it more sensitive.

When your finger is close to the sensor, its on-board LED light turns on, and when your finger moves away, its on-board LED light turns off. At this time, its sensitivity is relatively good.

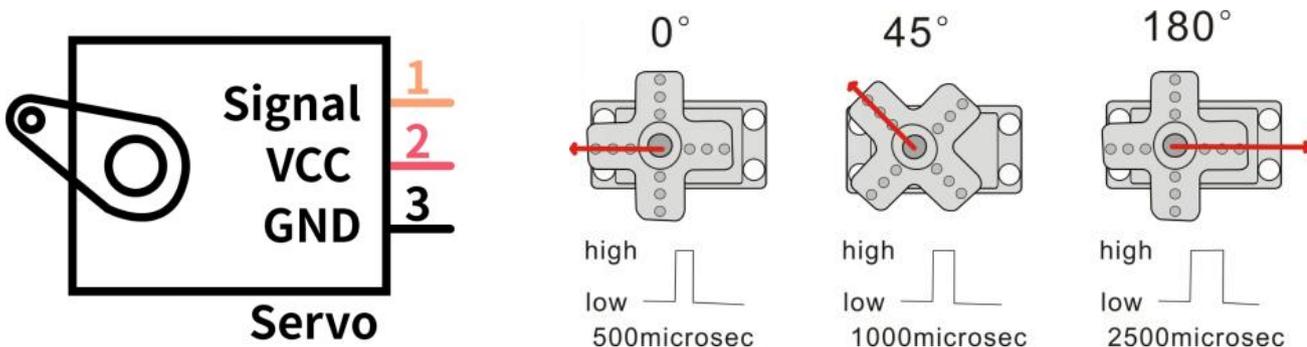


## Lesson 5.1: Control the Servo Motor-1

Servo motor is a position control rotary actuator. It mainly consists of a housing, a circuit board, a core-less motor, a gear and a position sensor. Its working principle is that the servo receives the signal sent by MCU or receiver and produces a reference signal with a period of 20ms and width of 1.5ms, then compares the acquired DC bias voltage to the voltage of the potentiometer and obtain the voltage difference output.

When the motor speed is constant, the potentiometer is driven to rotate through the cascade reduction gear, which leads that the voltage difference is 0, and the motor stops rotating. Generally, the angle range of servo rotation is  $0^\circ$  --  $180^\circ$

The rotation angle of servo motor is controlled by regulating the duty cycle of PWM (Pulse-Width Modulation) signal. The standard cycle of PWM signal is 20ms (50Hz). Theoretically, the width is distributed between 1ms-2ms, but in fact, it's between 0.5ms-2.5ms. The width corresponds the rotation angle from  $0^\circ$  to  $180^\circ$ . But note that for different brand motors, the same signal may have different rotation angles.



### The different wires

Almost all servos come with a set of 3 wires. These are PWR, GND and Signal. For a very simple circuit, all that is needed is to connect each of these two pins on the Arduino:

PWR (red) - connects to 5V on the Arduino.

GND (brown) - connects to GND on the Arduino.

Signal (yellow) - connects to a digital pin on the Arduino.

High level time	Servo angle
0.5ms	0 degree
1ms	45 degree
1.5ms	90 degree
2ms	135 degree
2.5ms	180 degree

## Parameters

Working voltage: DC 4.8V ~ 6V

Operating angle range: about 180 ° (at 500 → 2500 μsec)

Pulse width range: 500 → 2500 μsec

No-load speed: 0.12 ± 0.01 sec / 60 (DC 4.8V) 0.1 ± 0.01 sec / 60 (DC 6V)

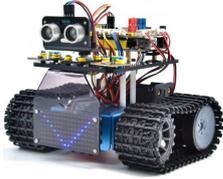
No-load current: 200 ± 20mA (DC 4.8V) 220 ± 20mA (DC 6V)

Stopping torque: 1.3 ± 0.01kg · cm (DC 4.8V) 1.5 ± 0.1kg · cm (DC 6V)

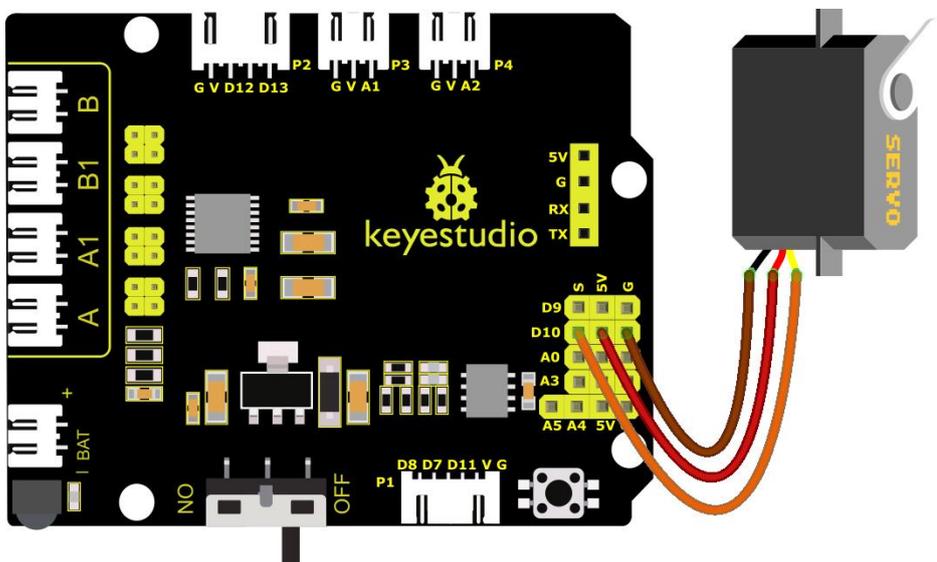
Stop current: ≅ 850mA (DC 4.8V) ≅ 1000mA (DC 6V)

Standby current: 3 ± 1mA (DC 4.8V) 4 ± 1mA (DC 6V)

## You need to prepare:

Robot without BT module*1	USB Cable*1	Computer*1
		
18650 Battery*2		
		

Note: The brown, red and orange wire of the servo are respectively connected to Gnd(G), 5v(V) and D10 of the shield. The working current of the servo is relatively large, so be sure to use an external power supply (two 18650 batteries) for power supply, otherwise the V4.0 motherboard may be burned out.



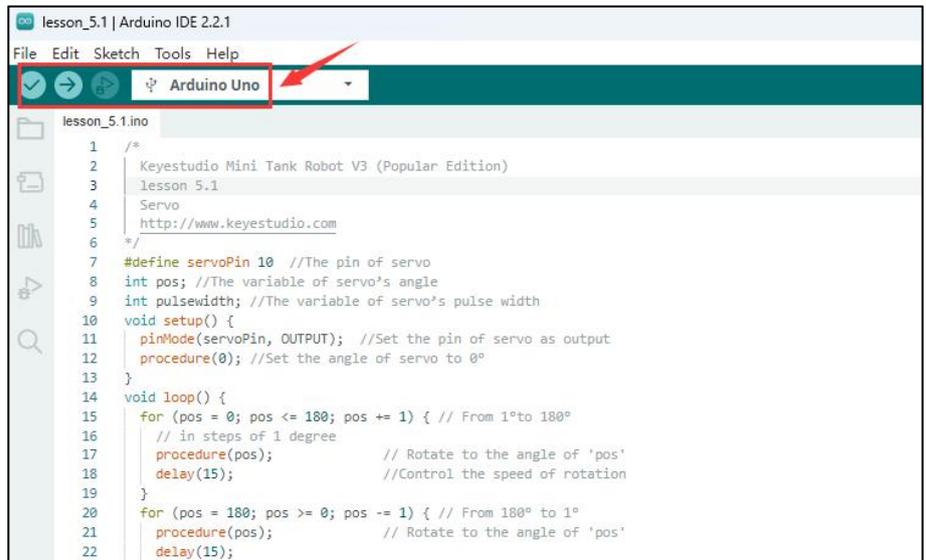
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_5.1** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

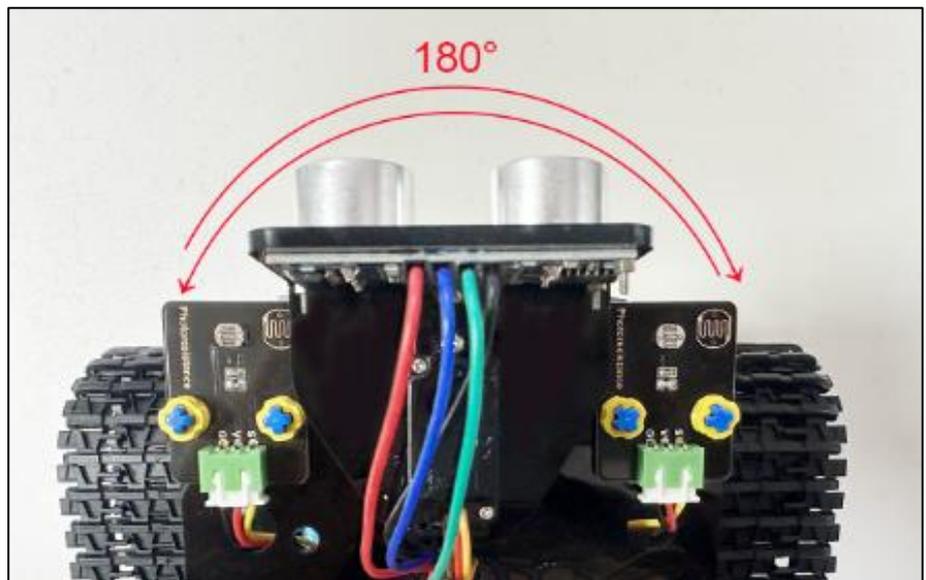
**Don't connect to the Bluetooth module when uploading code**

4. Click compile >>>done compiling.  
Click upload >>>done uploading.



**Test Results:** the shaft of the servo motor sweeps back and forth across 180 degrees.

Since the ultrasonic module is installed on the servo axis, you will see the head of the robot (ultrasonic module) swinging from side to side.



## Lesson 5.2: Control the Servo Motor-2

>>>>>>>> **This lesson is an expansion of the Lesson 5.1**

In this lesson, we will introduce how to use the Arduino servo library to control the servo.

You can refer to the use of the servo library: <https://www.arduino.cc/en/Reference/Servo>

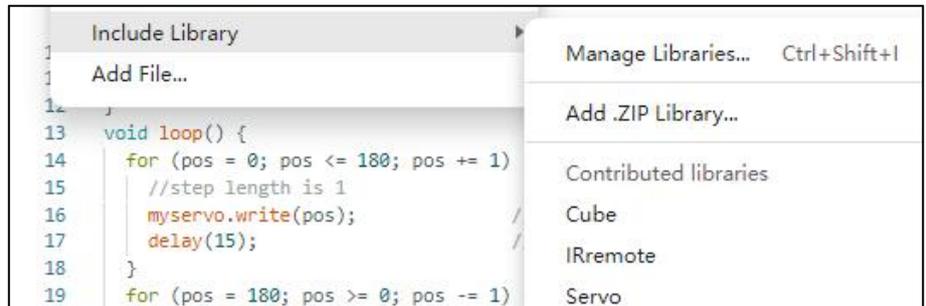
Hardware required for this lesson, the configuration of the arduino IDE, and the wiring between the servo motor and the 8833 Motor Driver Shield is in the same way as **Lesson 5.1**.

1. Connect the V4.0 board to the computer with the usb cable.

Open the INO file inside the **lesson\_5.2** folder with Arduino IDE.



2. Before uploading the code, please make sure you have installed the **Servo library** file in the arduino IDE.

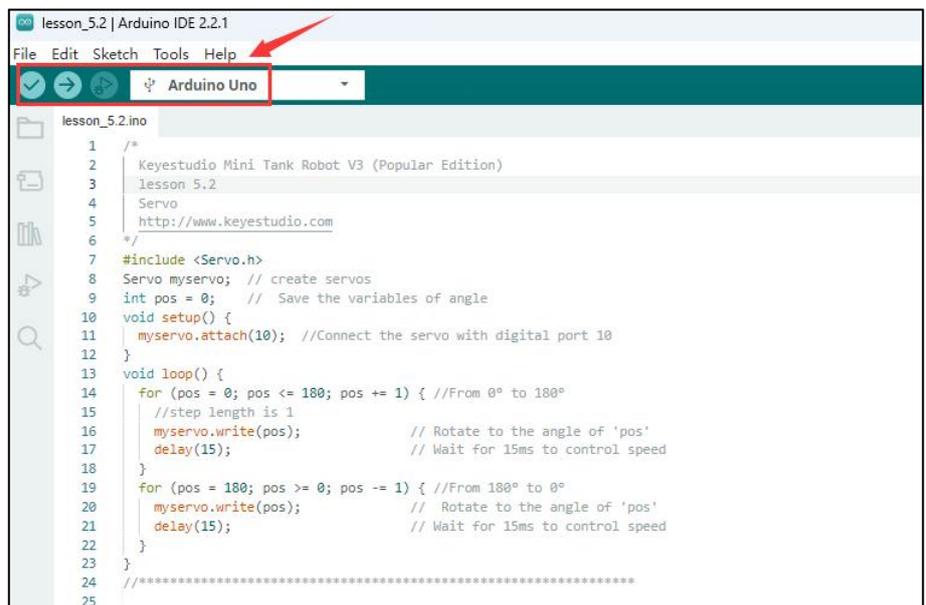


3. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

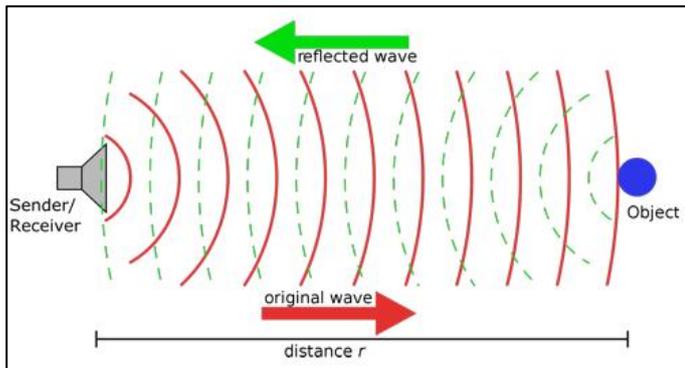
4. Click upload >>>done uploading.

**Test Results:** You will see the head of the robot swinging from side to side(180° ) .



## Lesson 6.1: Detect distance with ultrasonic module

In this lesson, we will learn how the ultrasonic sensor works and how to use it with Arduino. This is one of the most popular sensor for measuring distance and making obstacle-avoiding robots with Arduino.



### Specifications:

Power Supply :+5V DC

Quiescent Current : <2mA

Working Current: 15mA

Effectual Angle: <15°

Ranging Distance : 2cm – 400 cm

Resolution : 0.3 cm

Measuring Angle: 30 degree

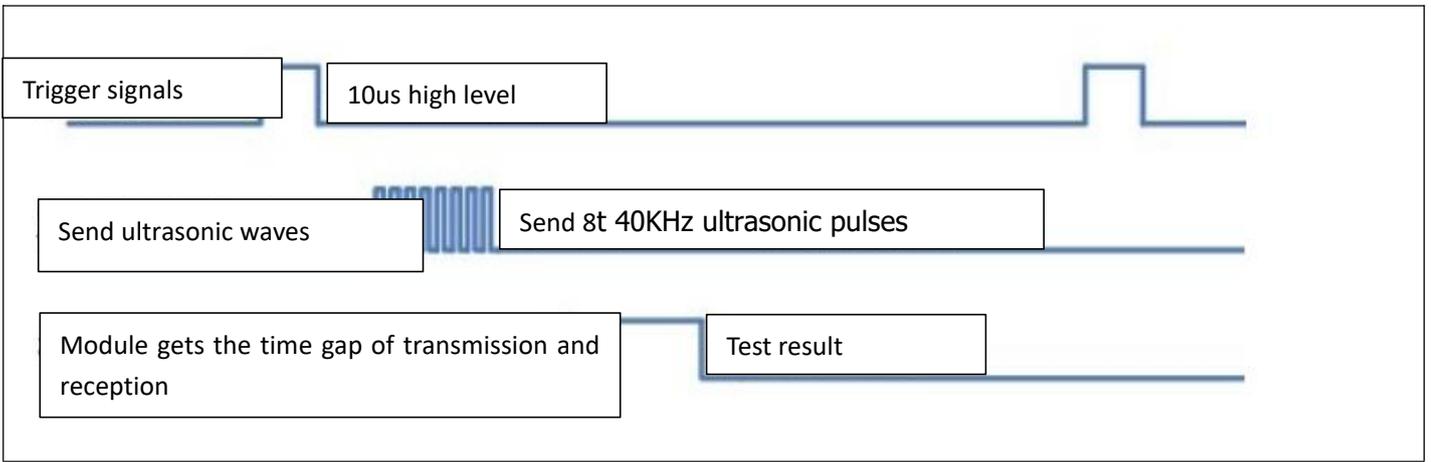
Trigger Input Pulse width: 10uS

### How the Ultrasonic Distance Sensor Works?

The sensor is composed of two ultrasonic transducers. One is transmitter which outputs ultrasonic sound pulses and the other is receiver which listens for reflected waves.

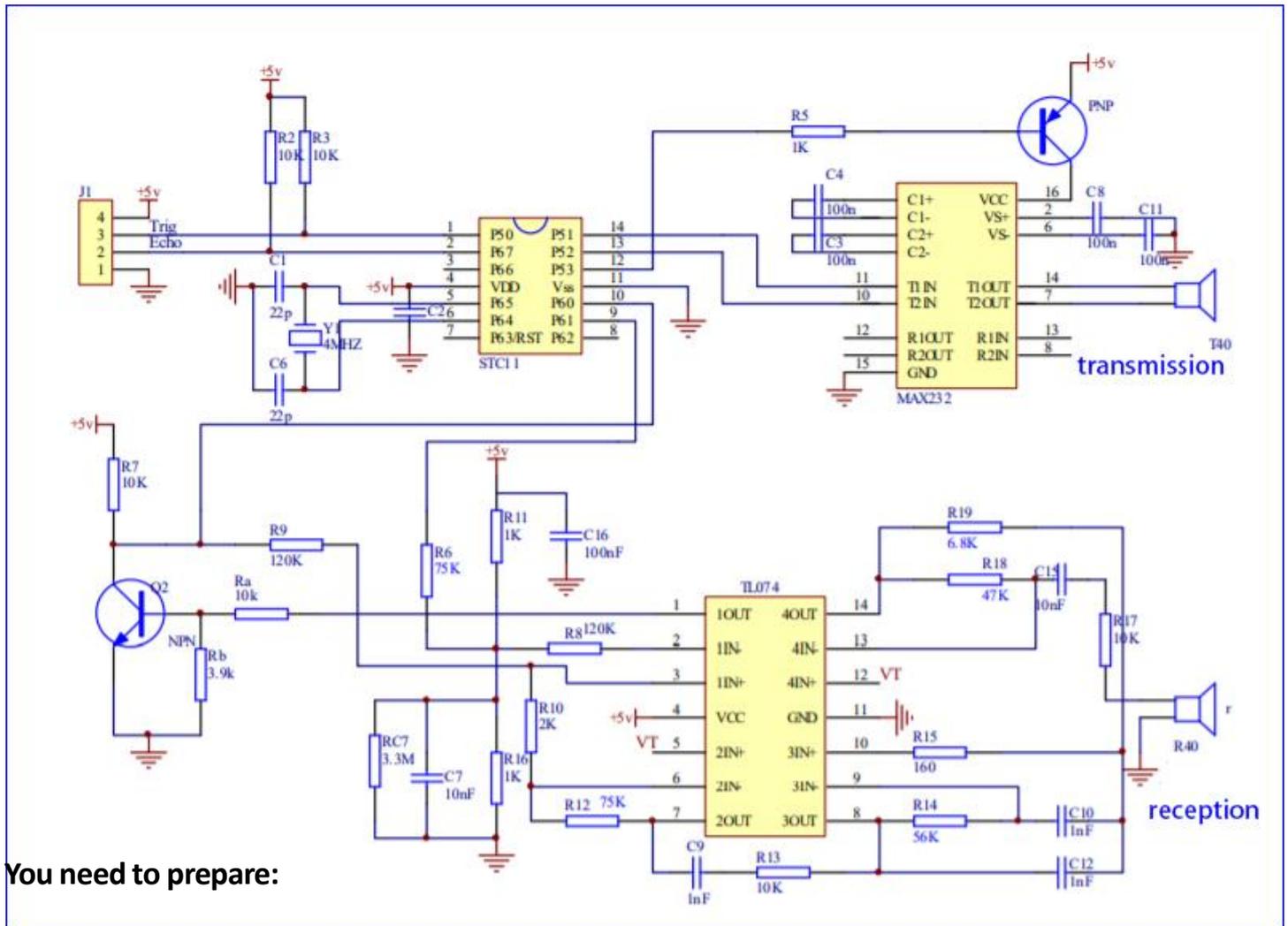
The ultrasonic module will emit the ultrasonic waves after triggering a signal. When the ultrasonic waves encounter the object and are reflected, the module outputs an echo signal, so it can determine the distance of the object from the time difference between the trigger signal and the echo signal.

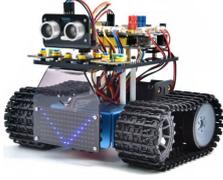
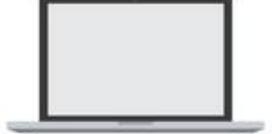
The  $t$  is the time that the emitting signal meets the obstacle and returns. And the propagation speed of sound in the air is about 343m/s, and distance = speed \* time. However, the ultrasonic wave emits and comes back, which is 2 times of distance. Therefore, it needs to be divided by 2, the distance measured by ultrasonic wave = (speed \* time)/2



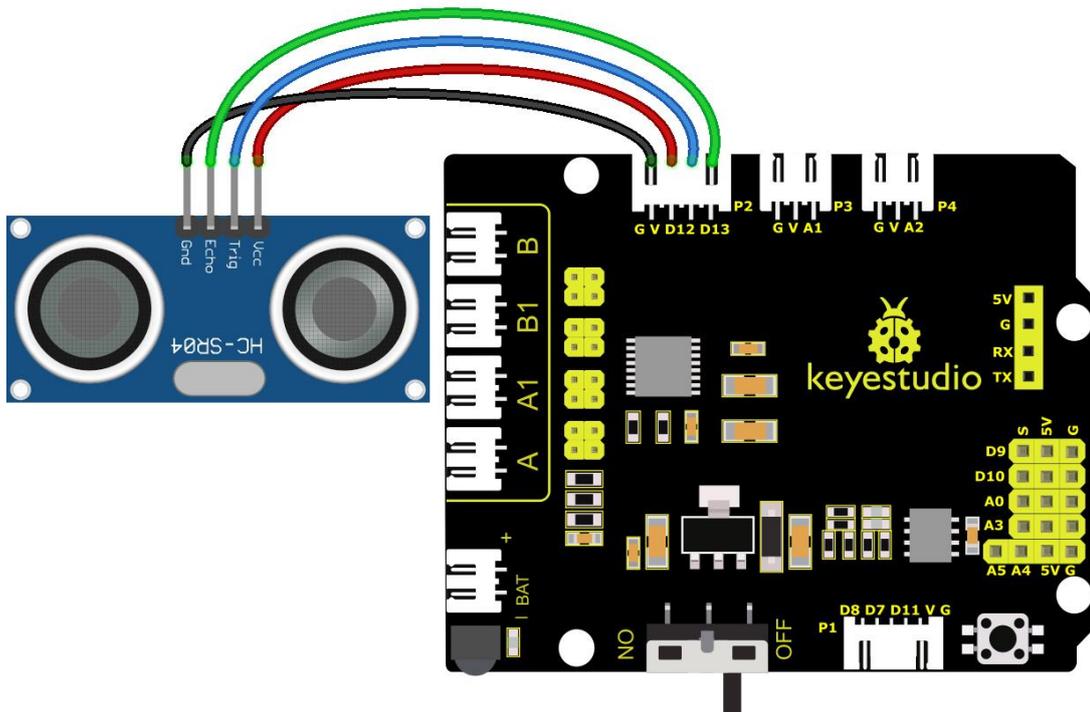
1. Use the method and timing chart of the ultrasonic module:
2. Setting the delay time of Trig pin of SR04 to  $10\mu\text{s}$  at least, which can trigger it to detect distance.
3. After triggering, the module will automatically send eight 40KHz ultrasonic pulses and detect whether there is a signal return. This step will be completed automatically by the module.
4. If the signal returns, the Echo pin will output a high level, and the duration of the high level is the time from the transmission of the ultrasonic wave to the return.

**Circuit diagram of ultrasonic sensor:**



Robot without BT module*1	USB Cable*1	Computer*1
		

Note: The pin VCC, Trig, Echo and Gnd of the ultrasonic sensor are respectively connected to 5v(V), 12(S), 13(S) and Gnd(G) of the shield.



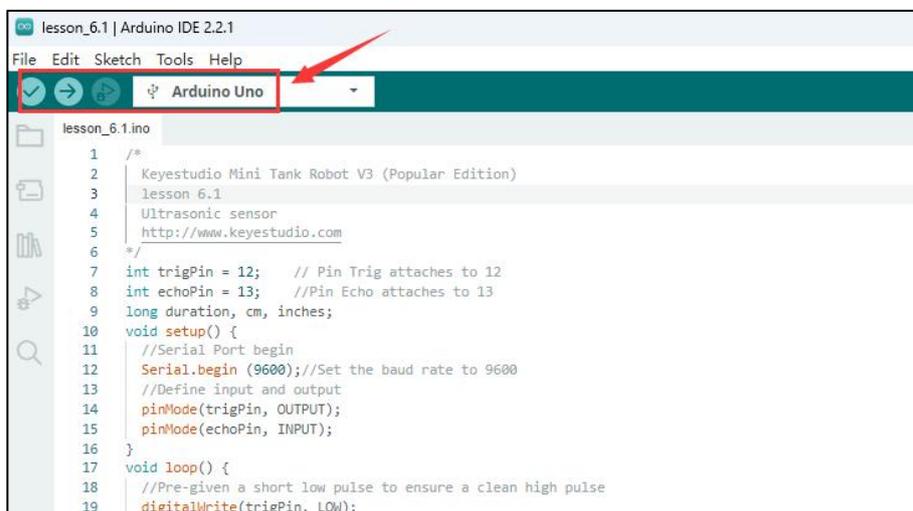
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the [lesson\\_6.1](#) folder with Arduino IDE.

lesson_4.2	2023/12/5 17:54
lesson_5.1	2023/12/5 17:57
lesson_5.2	2023/12/6 12:07
lesson_6.1	2023/12/6 15:53
lesson_6.2	2023/12/6 15:54

2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click compile >>>done compiling. Click upload >>>done uploading.



**Test Results:** Open the serial monitor of the Arduino IDE and set the baud rate to 9600. Place your palm or other object in front of the ultrasound, the serial monitor will display the distance value.



### Code Explanation:

`int trigPin = 12;` this pin is defined to transmit ultrasonic waves, generally output.

`int echoPin = 13;` this is defined as the pin of reception, generally input

`cm = (duration/2) / 29.1;` `inches = (duration/2) / 74;` by 0.0135

We can calculate the distance by using the following formula:

$distance = (traveltime/2) \times \text{speed of sound}$

The speed of sound is:  $343\text{m/s} = 0.0343 \text{ cm/uS} = 1/29.1 \text{ cm/uS}$

Or in inches:  $13503.9\text{in/s} = 0.0135\text{in/uS} = 1/74\text{in/uS}$

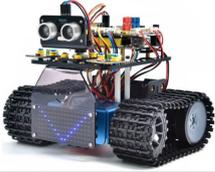
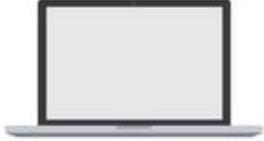
We need to divide the traveltime by 2 because we have to take into account that the wave was sent, hit the object, and then returned back to the sensor.

## Lesson 6.2: Use Ultrasonic Sensor to control LED

>>>>>>>> *This lesson is an expansion of the Lesson 6.1*

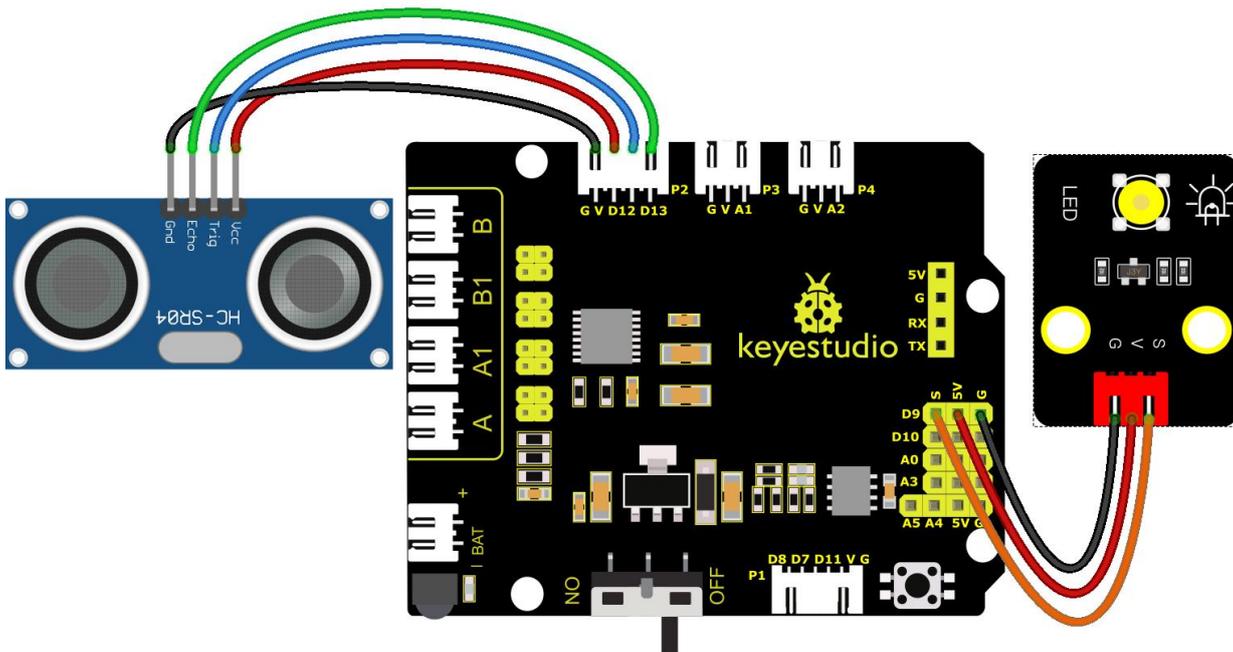
In this lesson, we will introduce how to use the ultrasonic sensor to control the LED module on and off.

**You need to prepare:**

Robot without BT Module*1	USB Cable*1	Yellow LED Module*1
		
3P-3P XH2.54 to 2.54 Dupont Wire*1	Computer*1	
		

Let's connect the LED light module to the D9 pin.

The pin VCC, Trig, Echo and Gnd of the ultrasonic sensor are respectively connected to 5v(V), 12(S), 13(S) and Gnd(G) of the shield.



1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the [lesson\\_6.2](#) folder with Arduino IDE.



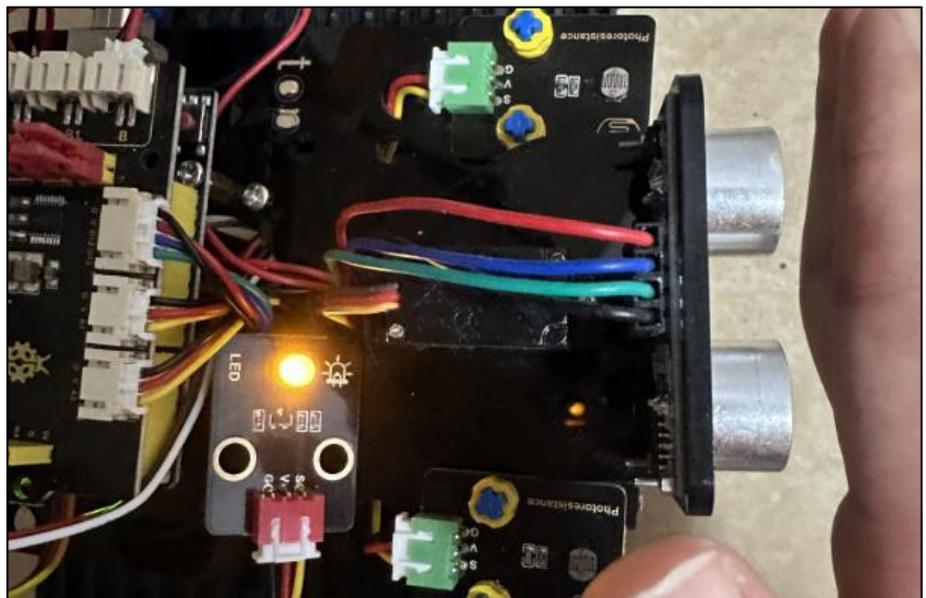
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click compile >>>done compiling. Click upload >>>done uploading.

```
lesson_6.2 | Arduino IDE 2.2.1
File Edit Sketch Tools Help
Select Board
lesson_6.2.ino
1  /*
2  Keyestudio Mini Tank Robot V3 (Popular Edition)
3  lesson 6.2
4  Ultrasonic LED
5  http://www.keyestudio.com
6  */
7  int trigPin = 12; // Trig is connected to 12
8  int echoPin = 13; // Echo is connected to 13
9  int LED = 9;
10 long duration, cm, inches;
11 void setup() {
12 //start serial port
13 Serial.begin (9600);//set baud rate to 9600
14 //define input and output
15 pinMode(trigPin, OUTPUT);
16 pinMode(echoPin, INPUT);
17 pinMode(LED, OUTPUT);
18 }
19 void loop() {
20 //Pre-given a short low pulse to ensure a clean high pulse
21 digitalWrite(trigPin, LOW);
22 delayMicroseconds(2);
23 digitalWrite(trigPin, HIGH);//Give at least 10us high level trigger
24 delayMicroseconds(10);
25 digitalWrite(trigPin, LOW);
26 // The duration of the high level is the time from the launch to the return of the ultrasonic wave
```

**Test Results:** When the ultrasonic sensor detects an obstacle in front of it, the LED module lights up, and when there is no obstacle in front of it, the LED module turns off.



## Lesson 7.1: Infrared Transmitting and Receiving

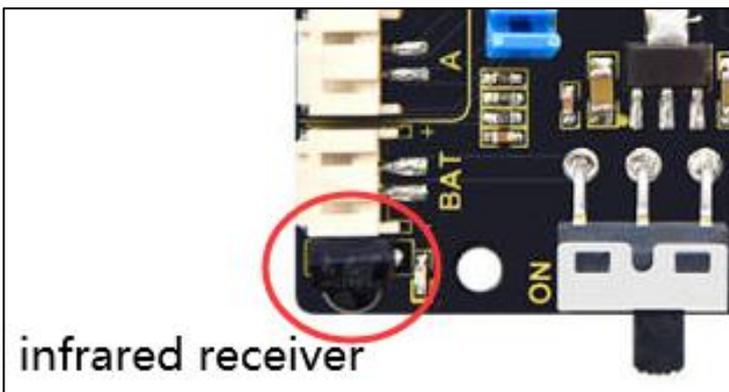
The infrared remote control is composed of infrared transmitting and infrared receiving systems, that is, an infrared remote control and infrared receiving module and a single-chip microcomputer capable of decoding.

The 38K infrared carrier signal emitted by remote controller is encoded by the encoding chip in the remote controller. It is composed of a section of pilot code, user code, user inverse code, data code, and data inverse code. The time interval of the pulse is used to distinguish whether it is a 0 or 1 signal and the encoding is made up of these 0, 1 signals.

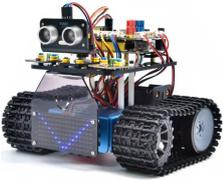
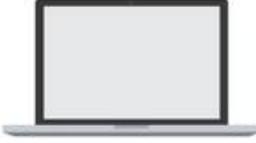
The user code of the same remote control is unchanged while the data code can distinguish the key. When the remote control button is pressed, the remote control sends out an infrared carrier signal. When the IR receiver receives the signal, the program will decode the carrier signal and determines which key is pressed. The MCU decodes the received 01 signal, thereby judging what key is pressed by the remote control.



The infrared receiver we use is integrated into the motor drive shield, which is a device that integrates reception, amplification, and demodulation. Its internal IC has completed demodulation, and can achieve from infrared reception to output and be compatible with TTL signals.



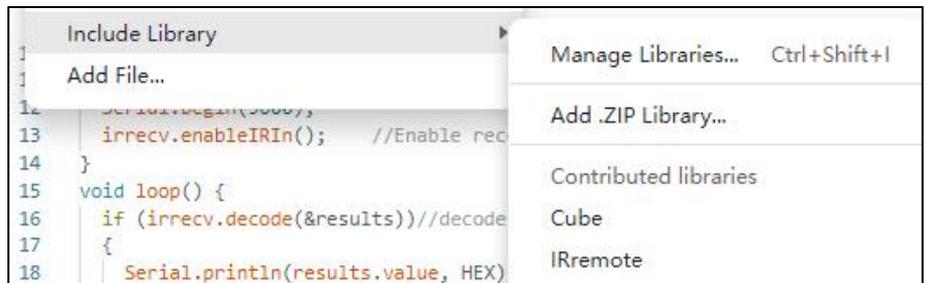
**You need to prepare:**

Robot without BT module*1	USB Cable*1	Computer*1	Remote Control*1
			

1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the [lesson\\_7.1](#) folder with Arduino IDE.



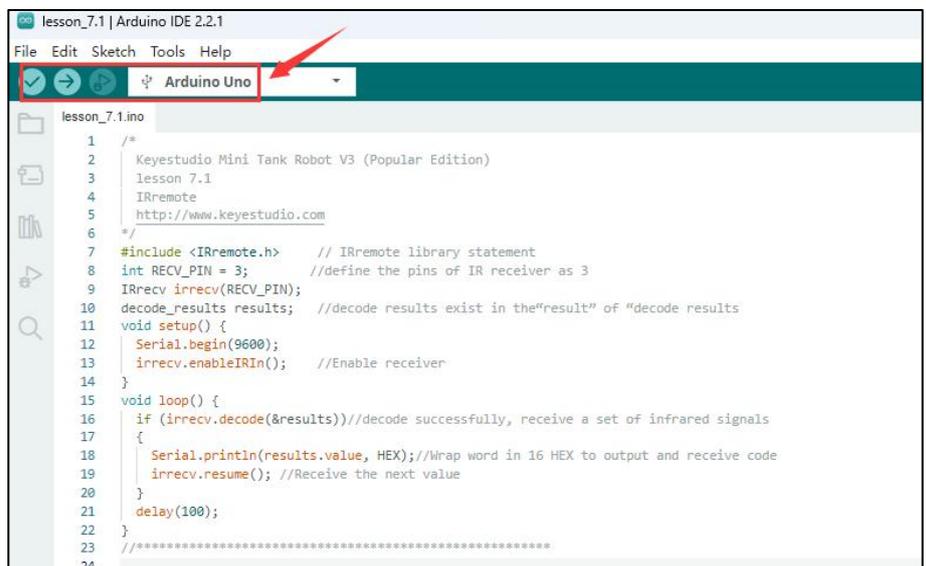
2. Before uploading the code, please make sure you have installed the **IRremote** library file in the arduino IDE.



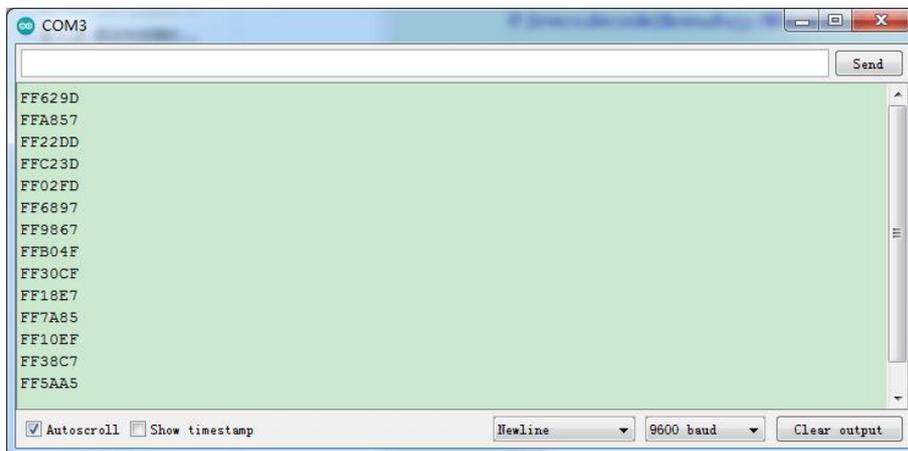
3. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

4. Click compile >>>done compiling. Click upload >>>done uploading.



**Test Results:** Whichever key you press, the serial monitor will display the value of that key. If you press and hold the button, the serial monitor will display a series of error codes.



### Code Explanation:

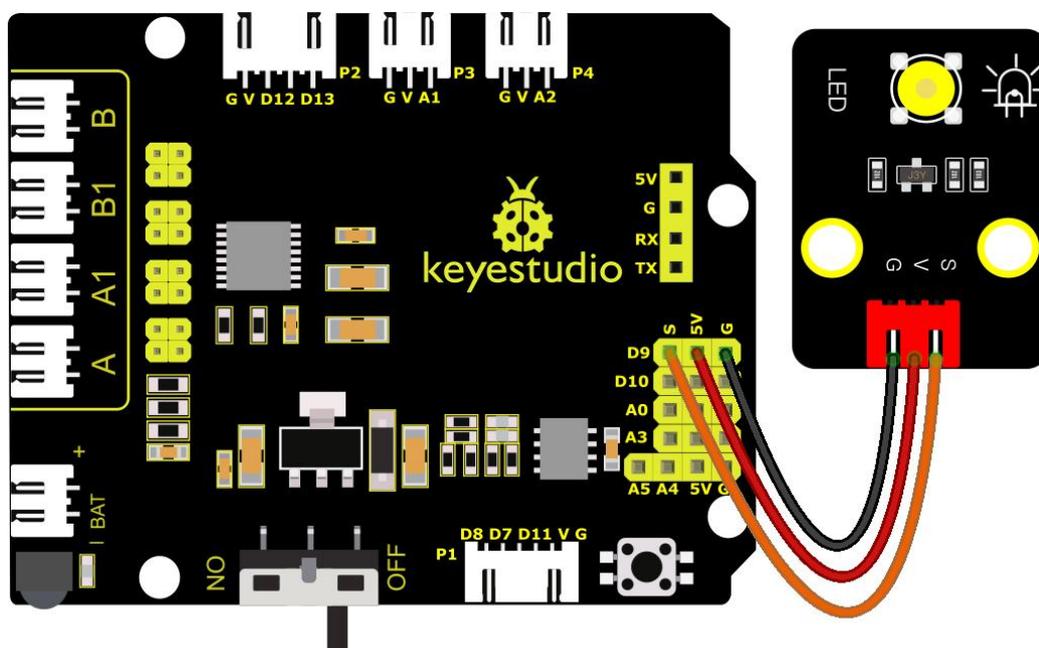
`irrecv.enableIRIn()`: after enabling IR decoding, the IR signals will be received, then function “`decode()`” will check continuously if decode successfully.

`irrecv.decode(&results)`: after decoding successfully, this function will come back to “true”, and keep result in “results”. After decoding a IR signals, run the `resume()` function and receive the next signal.

## Lesson 7.2: Use infrared remote control to control LED

>>>>>>>> *This lesson is an expansion of the Lesson 7.1*

In the previous lesson, we decoded the key values of the infrared remote control. In this course, we will connect the LED module to D9 of the control board, and then control the LED light to turn on and off through the remote control.



1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the [lesson\\_7.2](#) folder with Arduino IDE.

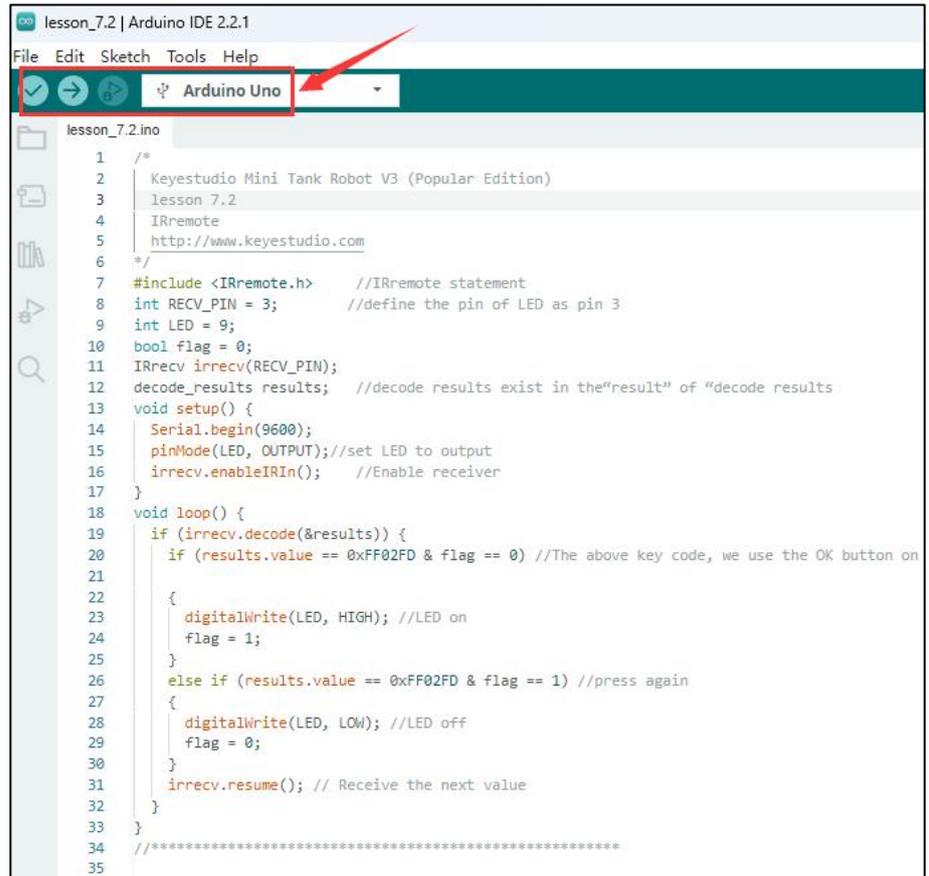


2. Before uploading the code, please make sure you have installed the [IRremote library](#) file in the arduino IDE.

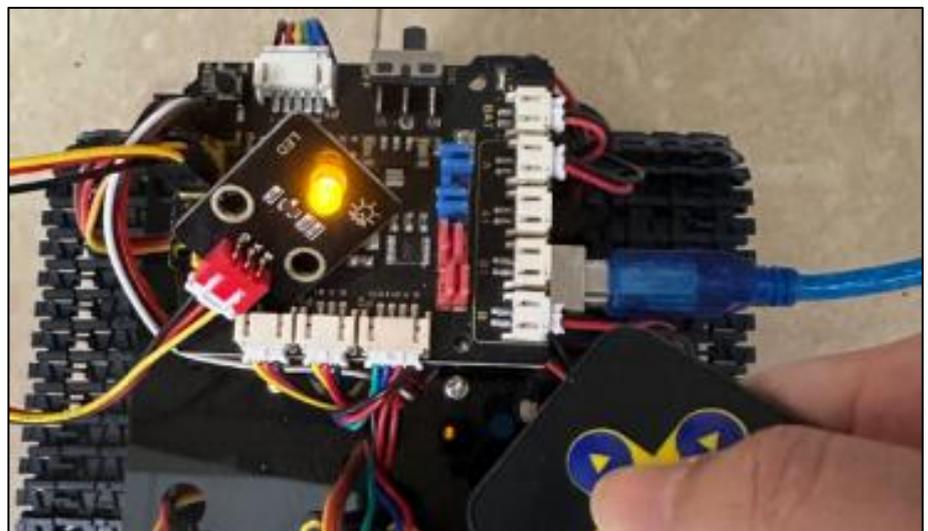
3. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

4. Click compile >>>done compiling.  
Click upload >>>done uploading.



**Test Results:** You can use the OK button of the remote control to control the LED module on/off.

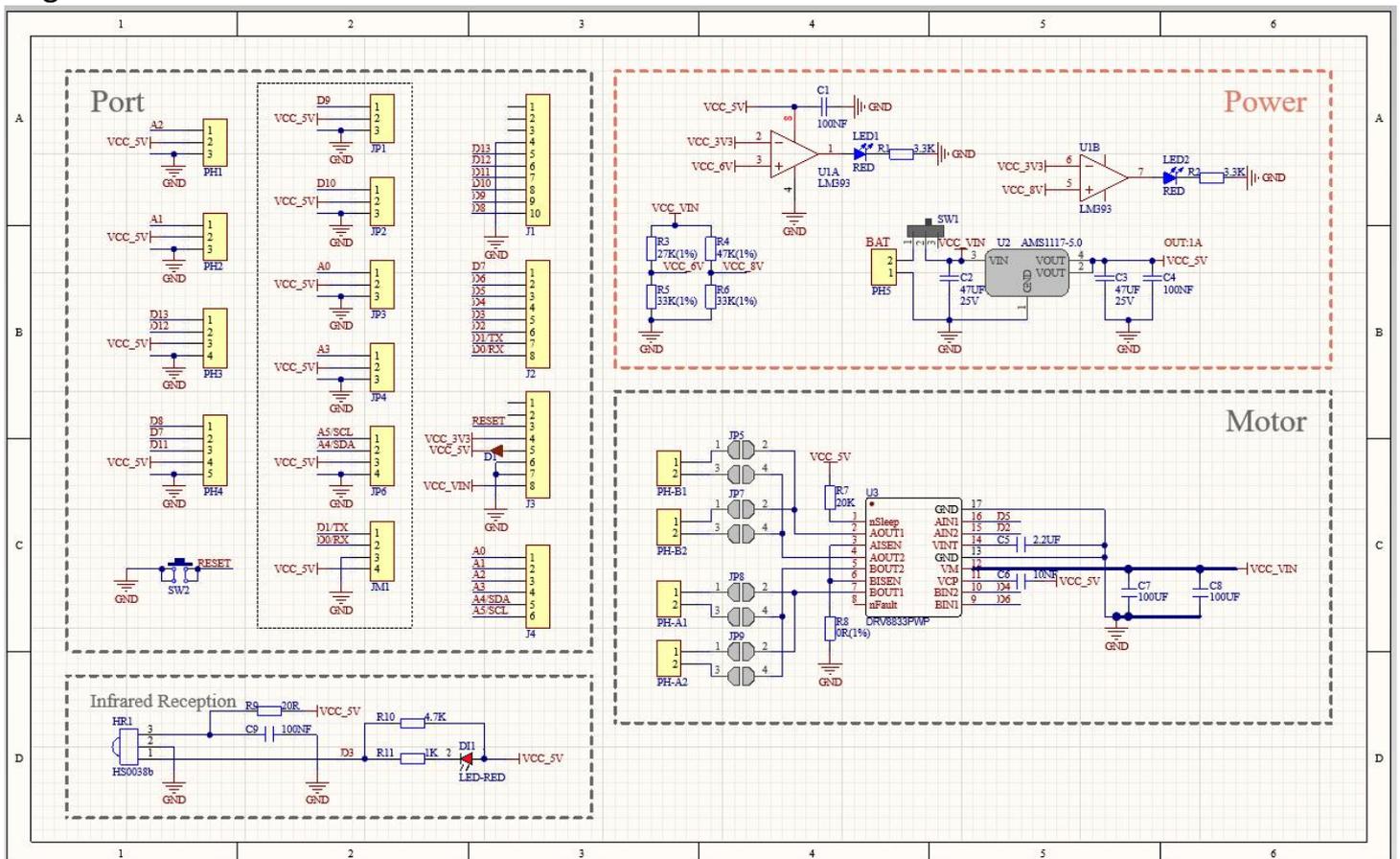


# Lesson 8: Motor Driving and Speed Control

Our smart car uses the most common solution called L298P to directly drive DC motors, two-phase and four-phase motors with the driving current reaching 2A. And the motor's output terminal adopts 8 high-speed Schottky diodes as protection. We have designed an shield based on the L298P circuit of which the laminated design can be directly plugged into the UNO R3 board for use reducing the technical difficulties for users in using and driving the motor.

Stack the it on the control board, power the BAT , turn the DIP switch to the ON end, and power the expansion board and the UNO R3 board at the same time via external power supply. In order to facilitate wiring, the expansion board is equipped with anti-reverse interface (PH2.0 -2P -3P -4P -5P) and thus it can be directly plug with motors, power supply, and sensors /modules. The Bluetooth interface of the drive expansion board is fully compatible with the Keystudio HM-10 Bluetooth module. It also uses 2.54 pin headers to extend out some available digital ports and analog ports, so that you can continue to add other sensors and carry out expansion experiments.

The shield can be connected to 4 DC motors. In the default jumper cap connection mode, the A and A1, B and B1 interface motors are connected in parallel, and their motion pattern is the same. 8 jumper caps can be used to control the rotation direction of the 4 motor interfaces. For example, when the two jumper caps in front of the motor A interface are changed from a horizontal connection to a vertical connection, the rotation direction of the motor A now is opposite to the original rotation direction.



## Parameters:

Logic part input voltage: DC 5V

Driving part input voltage: DC 7-12V

Logic part working current:  $\leq 36\text{mA}$

Driving part working current:  $\leq 2\text{A}$

Maximum dissipation power:  $25\text{W}$  ( $T=75^\circ\text{C}$ )

Control signal input level:

High level:  $2.3\text{V} \leq V_{in} \leq 5\text{V}$

Low level:  $0\text{V} \leq V_{in} \leq 1.5\text{V}$

Working temperature:  $-25^\circ\text{C} \sim +130^\circ\text{C}$

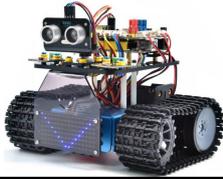
## Drive the robot to move

The direction pin of A motor is D2, the speed control pin is D5; the direction pin of B motor is in D4 and the speed control pin is D6,

According to the table below, we can know how to control the movement of the robot by controlling the rotation of two motors through the digital ports and PWM ports . Among them, the range of PWM value is 0-255. The larger the value is, the faster the motor rotates.

Function	D4	D6 (PWM)	Motor (left) B	D2	D5 (PWM)	Motor (Right) A
Move Forward	HIGH	255-200	Rotate Left	HIGH	255-200 0	Rotate Left
Go Back	LOW	200	Rotate Right	LOW	200	Rotate Right
Rotate Left	LOW	200	Rotate Right	HIGH	255-200 0	Rotate Left
Rotate Right	HIGH	255-200	Rotate Left	LOW	200	Rotate Right
Stop	LOW	0	Stop	LOW	0	Stop

## You need to prepare:

Robot without BT module*1	USB Cable*1	Computer*1	18650 Battery*2
			

In this lesson, you will need to equip the robot with two 18650 batteries of sufficient charge.

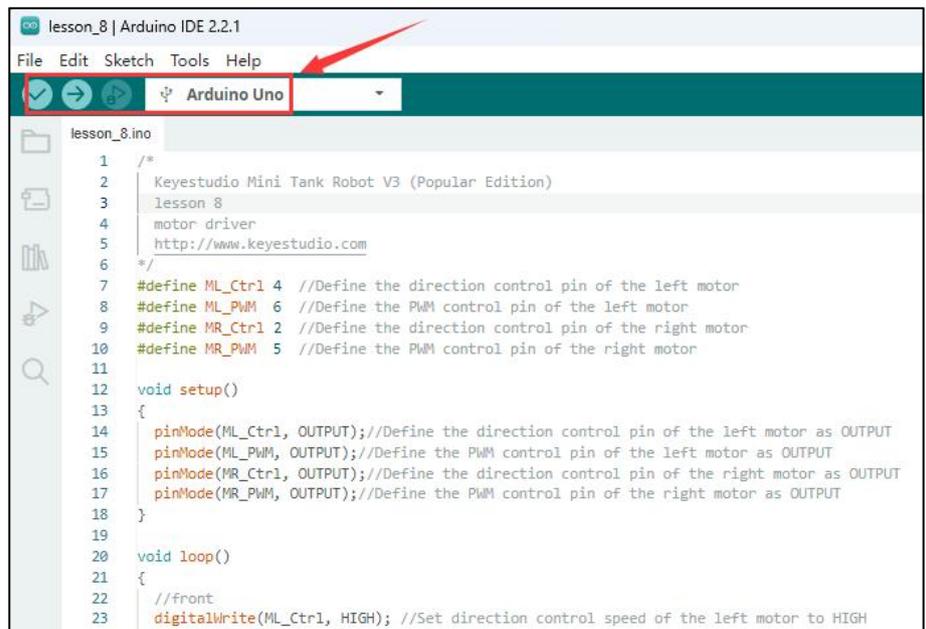
1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the [lesson\\_8](#) folder with Arduino IDE.



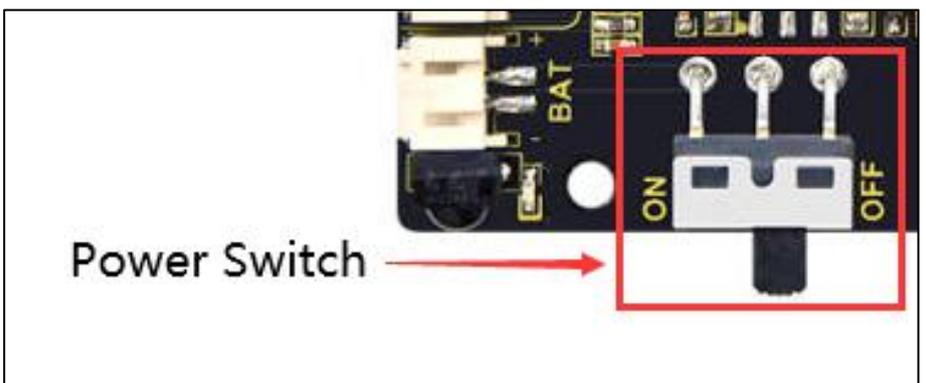
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

The robot will move immediately after uploading this code, so when uploading, please hold the robot with your hands to keep its tracks off the table to prevent it from falling.

3. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield and place the robot on the floor, the smart car moves forward for 2s, steps back for 2s, turns left for 2s, turns right for 2s and stops for 2s and repeats this sequence.



## Lesson 9.1: 8\*16 LED Dot Matrix Displays Smiley Face

With the 8\*16 LED dot matrix, you could design facial expressions, images, patterns and other displays by yourselves.

The 8\*16 LED board comes with 128 LEDs. The data of the microprocessor (Arduino) communicates with the AiP1640 through a two-wire bus interface. Therefore, it can control the on and off of 128 LEDs on the module, so as to make the dot matrix on the module to display the pattern you need. A HX-2.54 4Pin cable is provided for your convenience of wiring.

### Parameters:

Working voltage: DC 3.3-5V

Power loss: 400mW

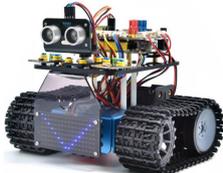
Oscillation frequency: 450KHz

Drive current: 200mA

Working temperature: -40~80°C

Communication mode: two-wire bus

### You need to prepare:

Robot without BT module*1	USB Cable*1	Computer*1	18650 Battery*2
			

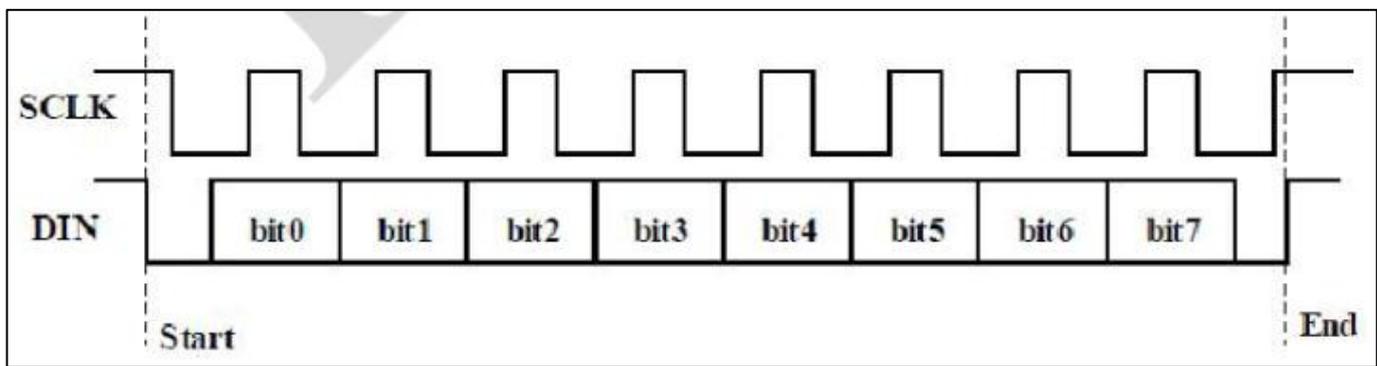
### Principle of the 8\*16 LED dot matrix

How to control each LED of the 8\*16 dot matrix? It is known that each byte has 8 bits and each bit is 0 or 1. when it is 0, LED is off while when it is 1 LED is on. One byte can control one column of the LED, and naturally 16 bytes can control 16 columns of LEDs, that's the 8\*16 dot matrix.

### Pins description and communication protocol

The data of the microprocessor (Arduino) communicates with the AiP1640 through a two-wire bus cable.

The communication protocol diagram is as follows (SCLK) is SCL, (DIN) is SDA



① The starting condition for data input: SCL is high level and SDA changes from high to low.

② For data command setting, there are methods as shown in the figure below

In our sample program, select the way to add 1 to the address automatically, the binary value is 0100 0000 and the corresponding hexadecimal value is 0x40

B7	B6	B5	B4	B3	B2	B1	B0	Description
0	1	Irrelevant choice, fill in 0			0	Irrelevant choice, fill in 0		add 1 to the address automatically Fixed address Universal mode Test mode
0	1				1			
0	1		0					
0	1		1					

③ For address command setting, the address can be selected as shown below.

The first 00H is selected in our sample program, and the binary number 1100 0000 corresponds to the hexadecimal 0xc0

B7	B6	B5	B4	B3	B2	B1	B0	Display address
1	1	Irrelevant choice, fill in 0		0	0	0	0	00H
1	1			0	0	0	1	01H
1	1			0	0	1	0	02H
1	1			0	0	1	1	03H
1	1			0	1	0	0	04H
1	1			0	1	0	1	05H
1	1			0	1	1	0	06H
1	1			0	1	1	1	07H
1	1			1	0	0	0	08H
1	1			1	0	0	1	09H
1	1			1	0	1	0	0AH
1	1			1	0	1	1	0BH
1	1			1	1	0	0	0CH
1	1			1	1	0	1	0DH
1	1			1	1	1	0	0EH
1	1			1	1	1	1	0FH

④ The requirement for data input is that when SCL is at high level when inputting data, the signal on SDA must remain unchanged. Only when the clock signal on SCL is at low level, can the signal on SDA be changed. The input of data is the low bit first, and the high bit later.

⑤ The condition for the end of data transmission is that when SCL is at low level, SDA at low level and SCL at high level, the level of SDA becomes high.

⑥ Display control, set different pulse width, pulse width can be selected as shown in the figure below

In the example, the pulse width is 4/16, and the hexadecimal corresponding to 1000 1010 is 0x8A

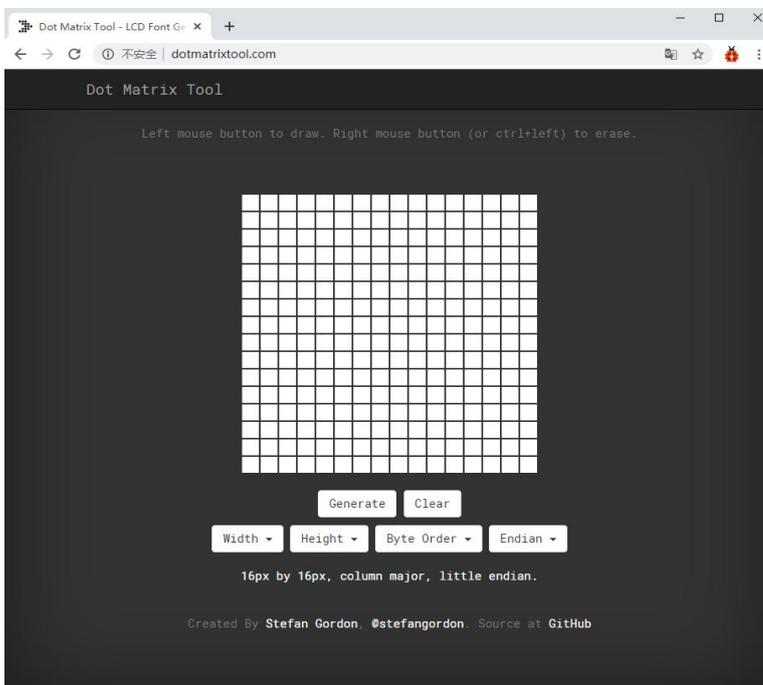
B7	B6	B5	B4	B3	B2	B1	B0	Function	Description
1	0	Irrelevant choice, fill in 0		1	0	0	0	Clear quantity setting  (Brightness setting)	Set pulse width to 1/16
1	0			1	0	0	1		Set pulse width to 2/16
1	0			1	0	1	0		Set pulse width to 4/16
1	0			1	0	1	1		Set pulse width to 10/16
1	0			1	1	0	0		Set pulse width to 11/16
1	0			1	1	0	1		Set pulse width to 12/16
1	0			1	1	1	0		Set pulse width to 13/16
1	0			1	1	1	1		Set pulse width to 14/16
1	0			0	X	X	X	Display switch setting	On
1	0		1	X	X	X	off		

#### 4. Instructions for the use of modulus tool

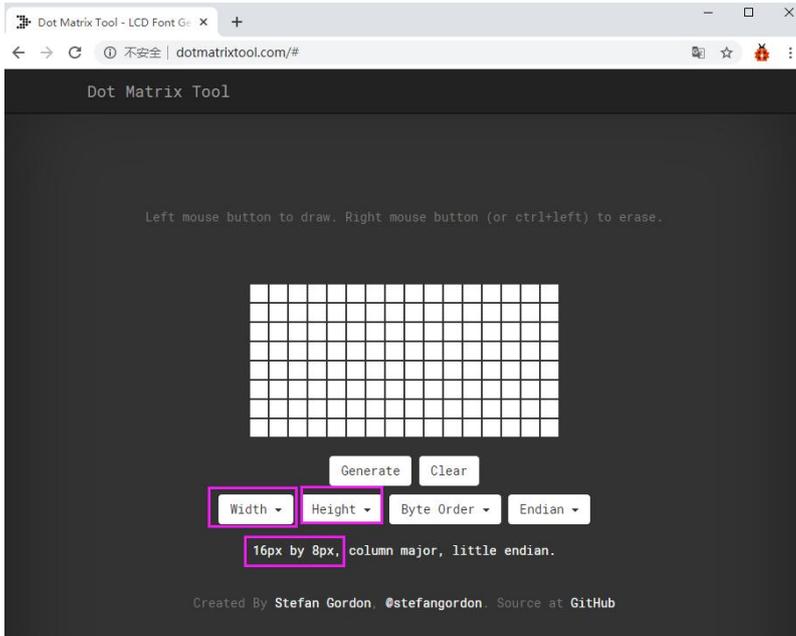
The dot matrix tool uses the online version, and the link is :

<http://dotmatrixtool.com/#>

① Enter the link and the page appears as shown below

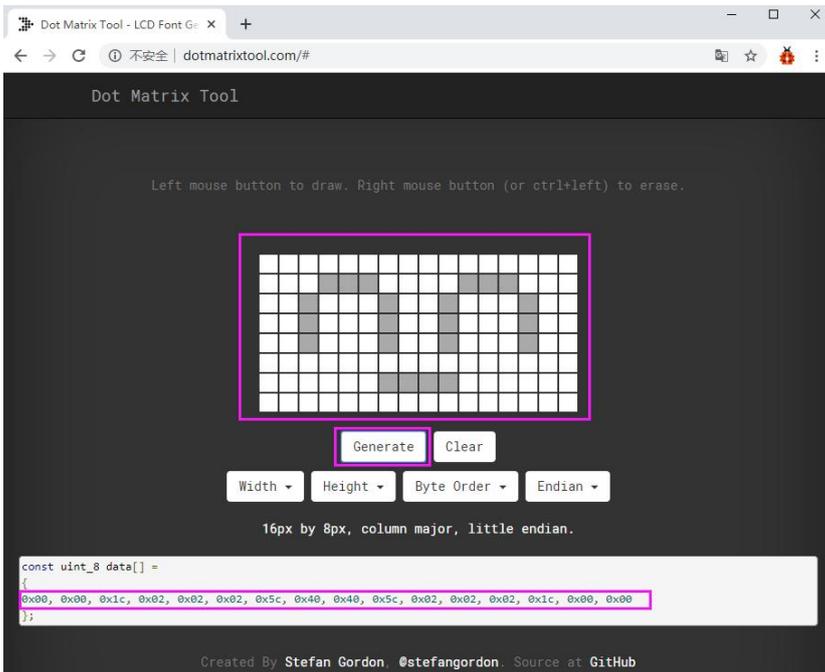


② The dot matrix is 8\*16, so adjust the height to 8 and width to 16, as shown in the figure below



③ Generate hexadecimal data from the pattern

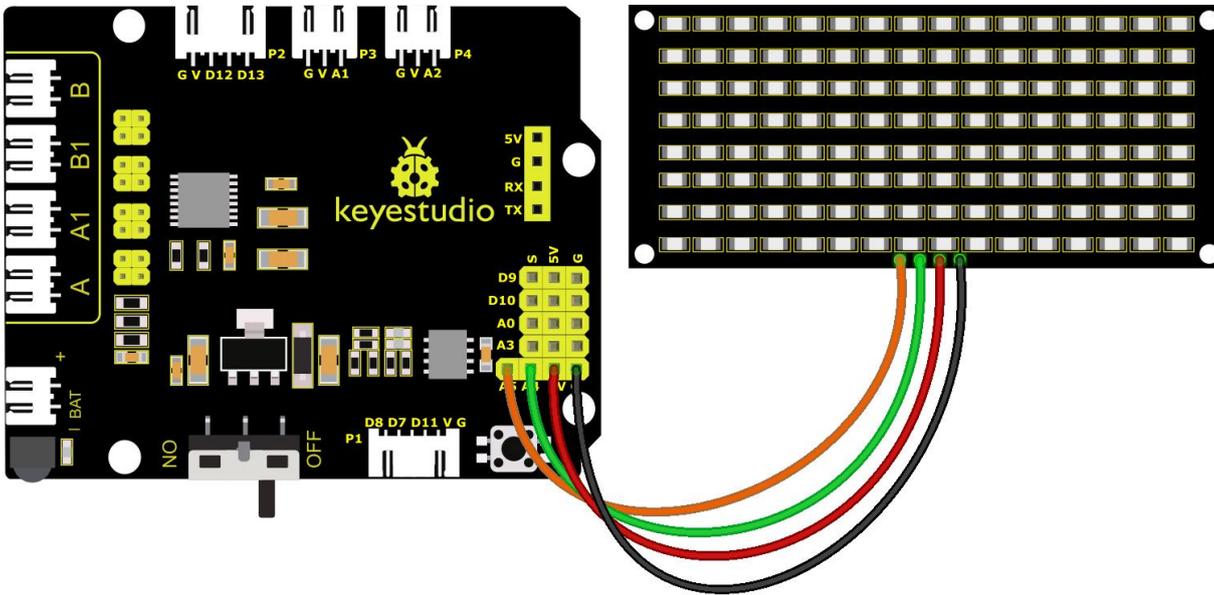
As shown in the figure below, press the left mouse button to select, right click to cancel; draw the pattern you want, click Generate, and the hexadecimal data we need will be generated.



### Connection Diagram:

The GND, VCC, SDA, and SCL of the 8x16 LED matrix are respectively connected to the (GND), + (VCC), A4, A5 of the motor drive shield.

(Note: though it is connected with the IIC pin of Arduino, this module is not for IIC communication. And the IO port here is to simulate I2C communication and can be connected with any two pins )



Note: In the last lesson we made the robot move, before connect it with USB cable, please hold the robot with your hands to keep its tracks off the table to prevent it from falling when connect it to the computer.

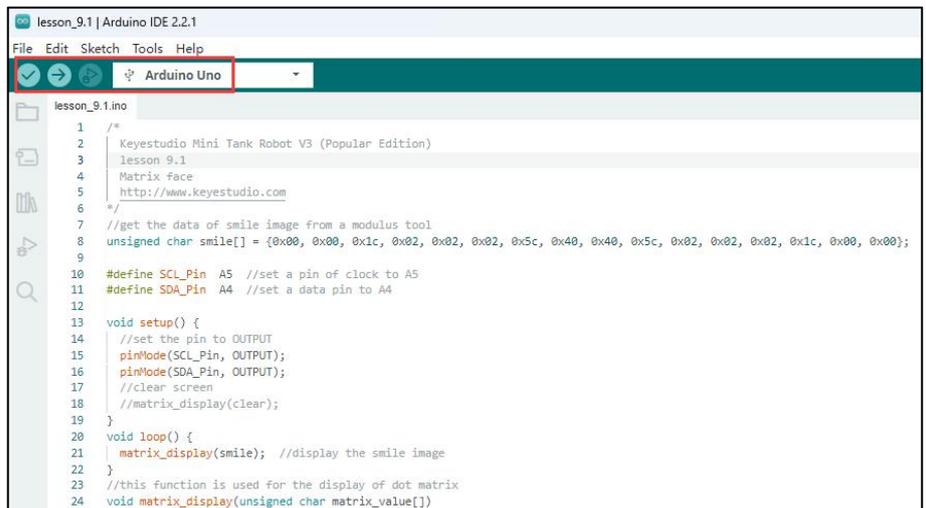


Open the INO file inside the [lesson\\_9.1](#) folder with Arduino IDE.

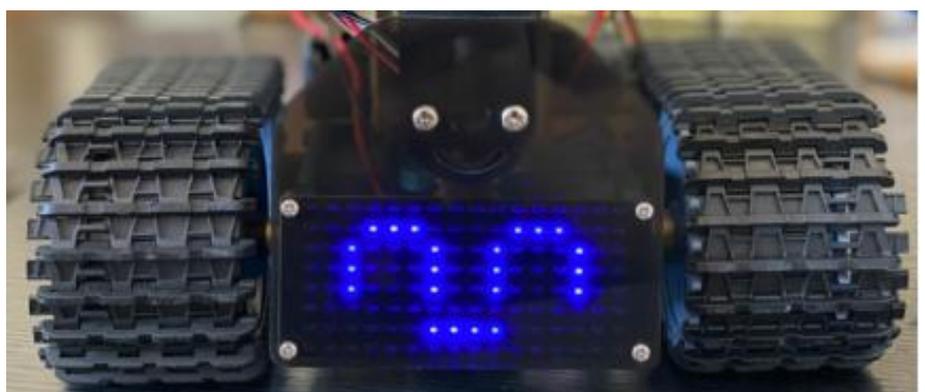
1. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

2. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield, a smile-shaped pattern shows on the dot matrix.



## Lesson 9.2: 8\*16 LED Dot Matrix Displays Direction and Text

>>>>>>>>*This lesson is an expansion of the Lesson 9.1*

In this lesson, we will make the dot matrix display the direction forward and backward, display word stop and then clear the pattern. The time interval is 2000 ms.

Hardware required for this lesson, the configuration of the arduino IDE, and the wiring between the servo motor and the 8833 Motor Driver Shield is in the same way as [Lesson 9.1](#).



**Code obtained from the module tool:**

Code for the pattern start:

```
0x01,0x02,0x04,0x08,0x10,0x20,0x40,0x80,0x80,0x40,0x20,0x10,0x08,0x04,0x02,0x01
```

Code for the pattern going forward:

```
0x00,0x00,0x00,0x00,0x00,0x24,0x12,0x09,0x12,0x24,0x00,0x00,0x00,0x00,0x00,0x00
```

Code for the pattern stepping back:

```
0x00,0x00,0x00,0x00,0x00,0x24,0x48,0x90,0x48,0x24,0x00,0x00,0x00,0x00,0x00,0x00
```

Code for the pattern turning left:

```
0x00,0x00,0x00,0x00,0x00,0x00,0x44,0x28,0x10,0x44,0x28,0x10,0x44,0x28,0x10,0x00
```

Code for the pattern turning right:

```
0x00,0x10,0x28,0x44,0x10,0x28,0x44,0x10,0x28,0x44,0x00,0x00,0x00,0x00,0x00,0x00
```

Code for the pattern stop:

```
0x2E,0x2A,0x3A,0x00,0x02,0x3E,0x02,0x00,0x3E,0x22,0x3E,0x00,0x3E,0x0A,0x0E,0x00
```

Code to clear screen:

```
0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00,0x00
```

1. Connect the V4.0 board to the computer with the usb cable.

Open the INO file inside the

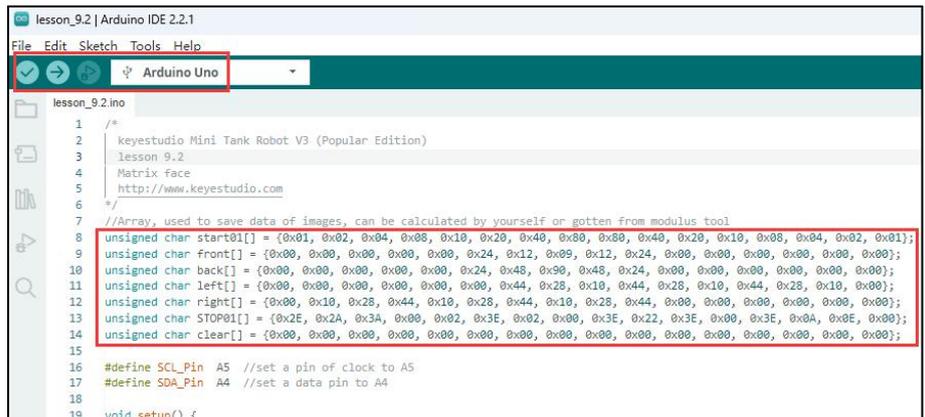
4. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the [lesson\\_9.2](#) folder with Arduino IDE.



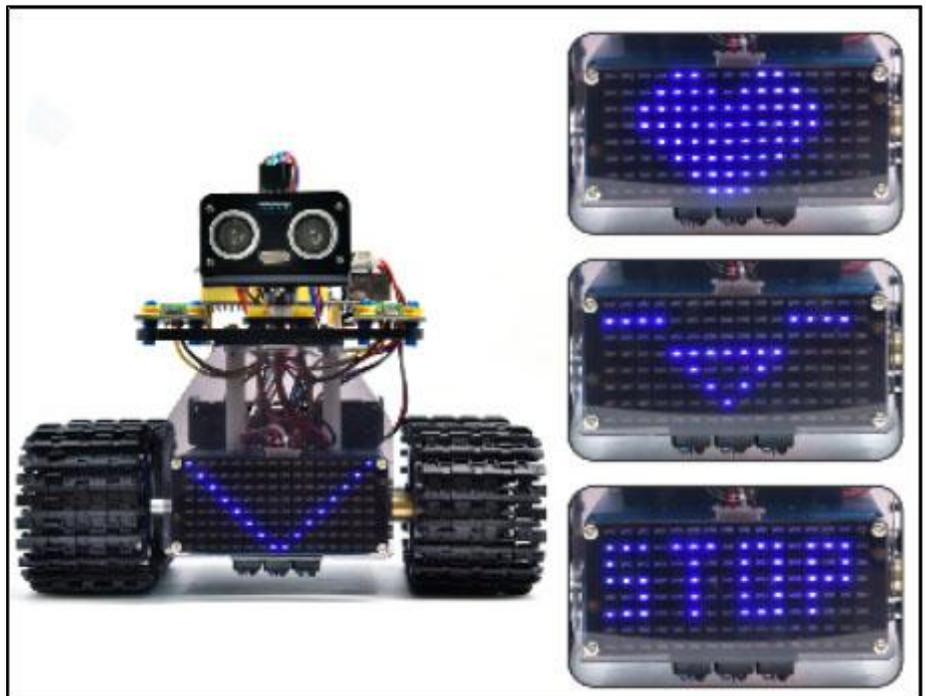
5. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

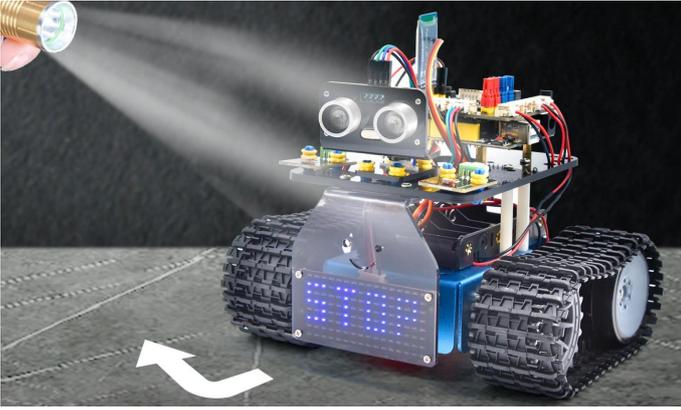
6. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield, the dot matrix display the direction forward and backward, display word stop and then clear the pattern. The time interval is 2000 ms.



## Lesson 10: Light Following Tank



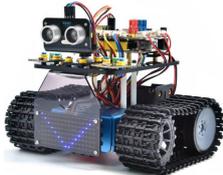
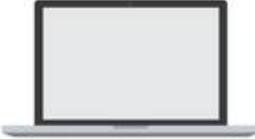
In previous projects, we introduced in detail the use of various sensors, modules, and expansion boards on the smart car. Now let's move to the projects of the smart car. The light-following smart cars, as the name suggests, is a smart car that can follow the light.

We can combine the knowledge from projects photoresistor and motor drive to make a light-seeking smart car. In the project, we use two photoresistor modules to detect the light intensity on the left and right sides of the smart car, read the corresponding analog values, and then control the rotation of the two motors based on these two data so as to control the movements of the smart car.

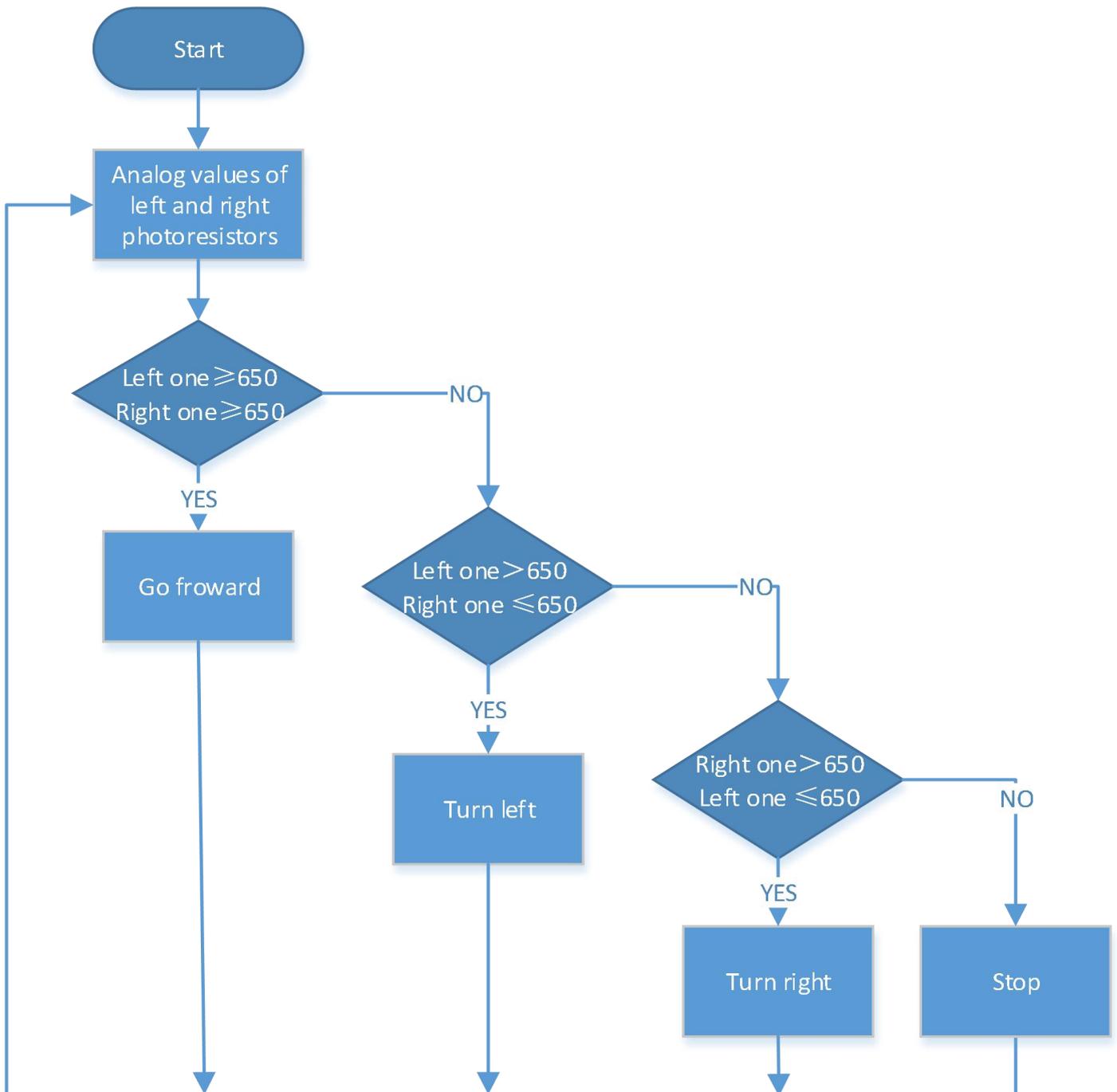
The specific logic of the light-following smart car is shown as below.

Detection (the bigger the brightness, the bigger the value )	Left photoresistor module	left_light
	Right photoresistor module	right_light
Condition	left_light > 650 and right_light > 650	
Movement	Move forward (set PWM to 200)	
Condition	left_light > 650 and right_light ≤ 650	
Movement	Rotate left (set PWM to 200)	
Condition	left_light ≤ 650 and right_light > 650	
Movement	Rotate right (set PWM to 200)	
Condition	left_light ≤ 650 and right_light ≤ 650	
Movement	stop	

**You need to prepare:**

Robot without BT module*1	USB Cable*1	Computer*1	18650 Battery*2
			

## Flow chart



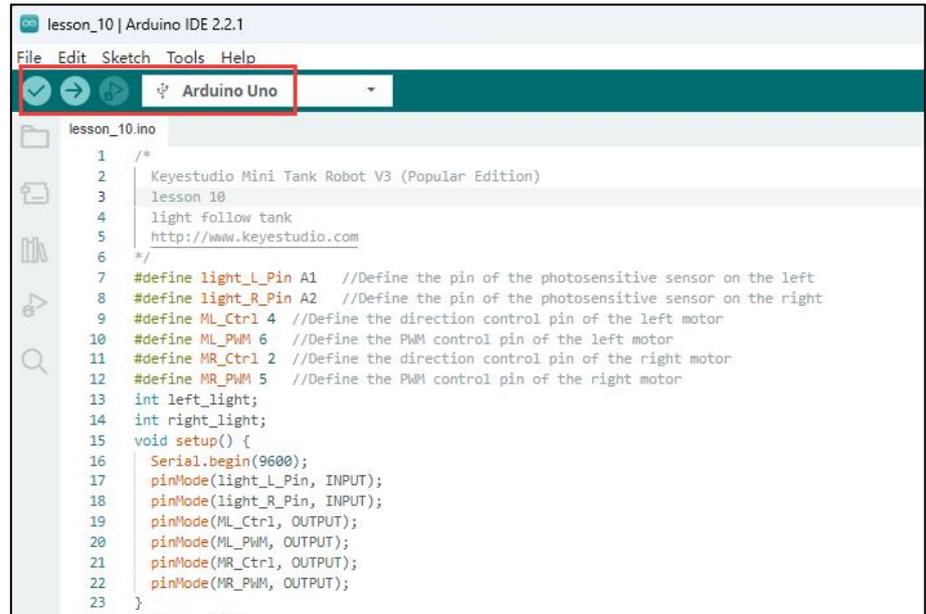
1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_10** folder with Arduino IDE.



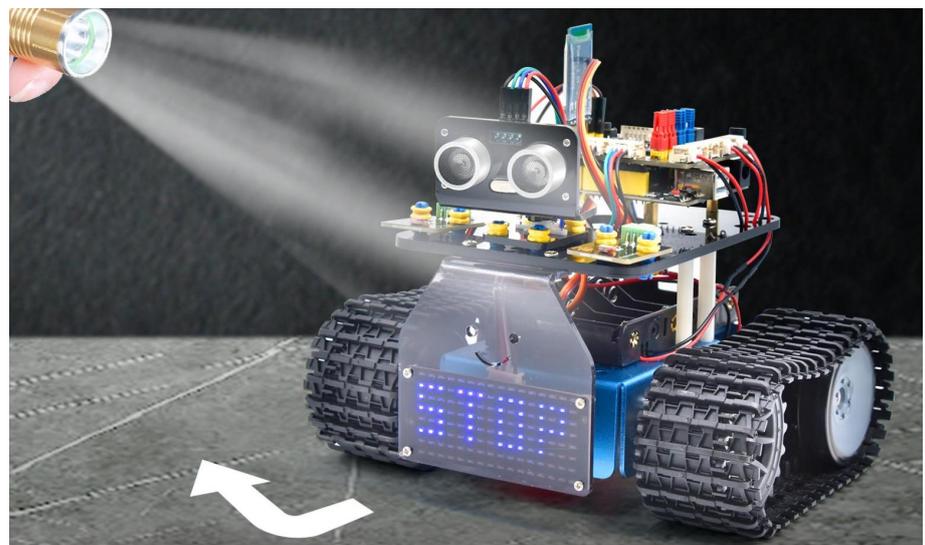
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

The robot may move after uploading this code, so when uploading, please hold the robot with your hands to keep its tracks off the table to prevent it from falling.

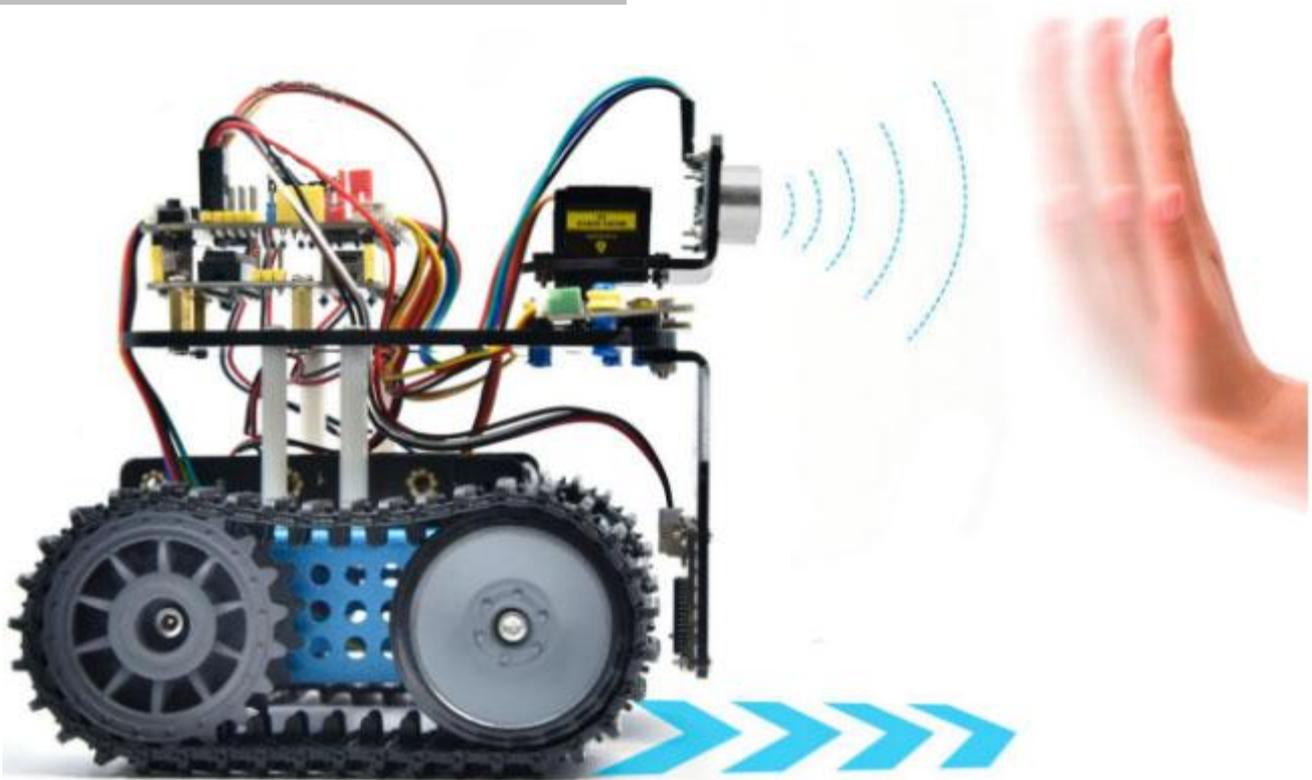
3. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield, place the robot on the floor of a dark room and the robot will be attracted to a bright light such as a flashlight and follow the light source.



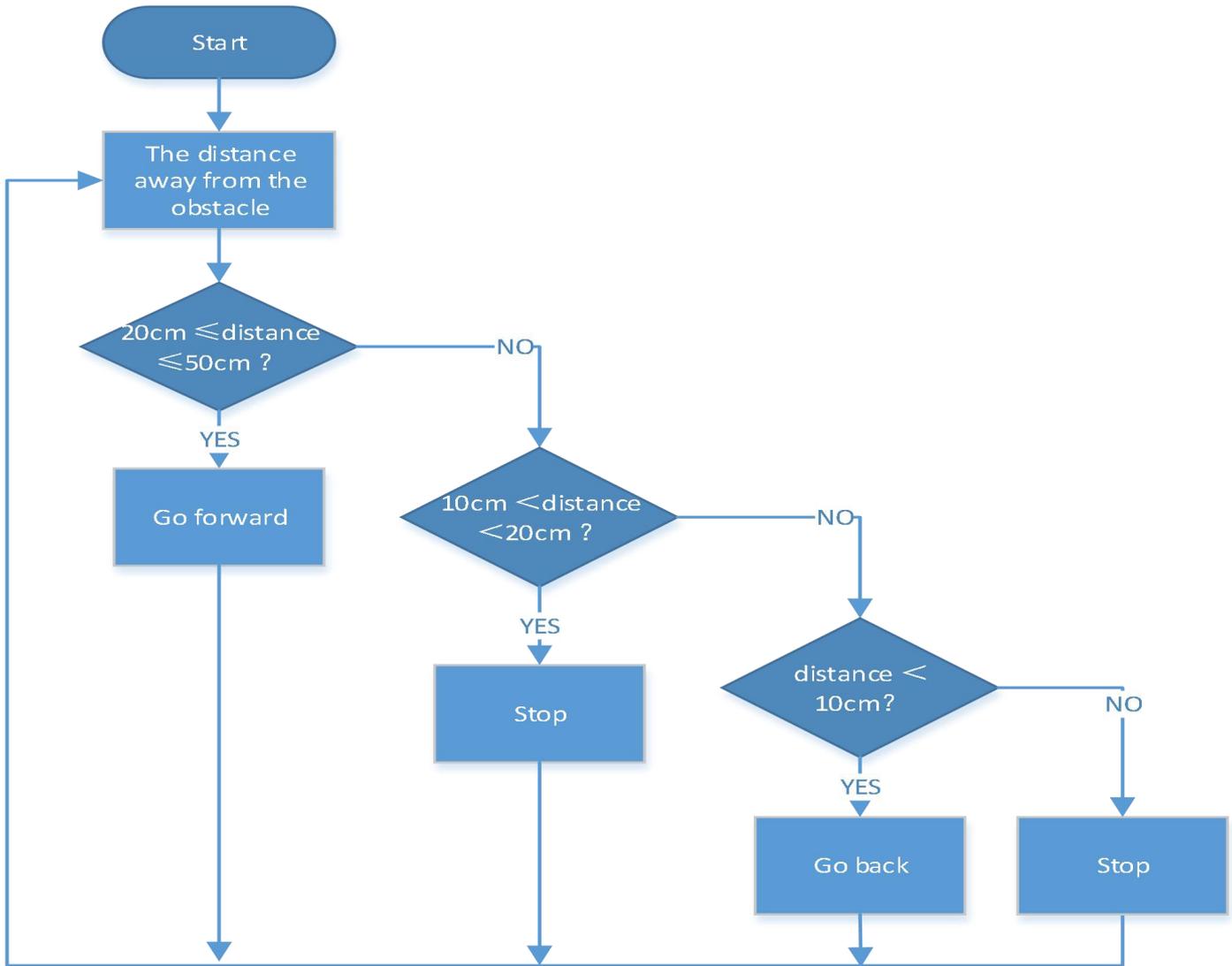
## Lesson 11: Ultrasonic Follow Robot



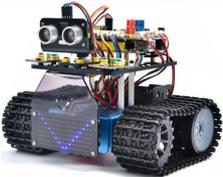
In the previous lesson, we learned about the light-following smart car. And in this lesson, we can combine the knowledge to make an ultrasonic sound-following car. In the project, we use ultrasonic sensors to detect the distance between the car and the obstacle in front, and then control the rotation of the two motors based on this data so as to control the movements of the smart car. The specific logic of the ultrasonic sound- following smart car is shown in the table below:

Detection	The distance between the car and the obstacle front	Distance (unit: cm)
Setting	Set the angle of the servo to 90°	
Condition	distance $\geq$ 20 and distance $\leq$ 50	
Movement	go front	
Condition	distance > 10 and distance < 20	
	distance > 50	
Movement	Stop	
Condition	distance $\leq$ 10	
Movement	go back	

## (2)Flow chart



### You need to prepare:

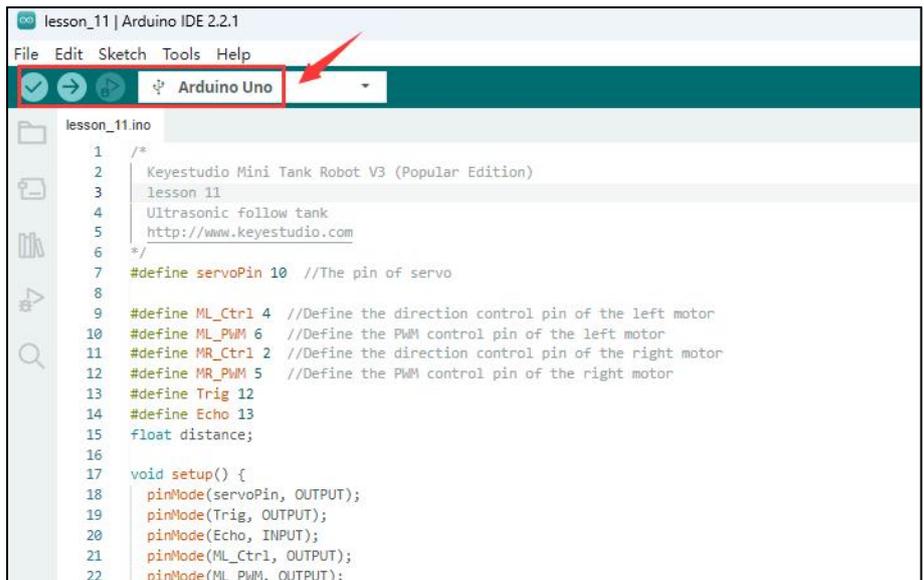
Robot without BT module*1	USB Cable*1	Computer*1	18650 Battery*2
			

1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_11** folder with Arduino IDE.



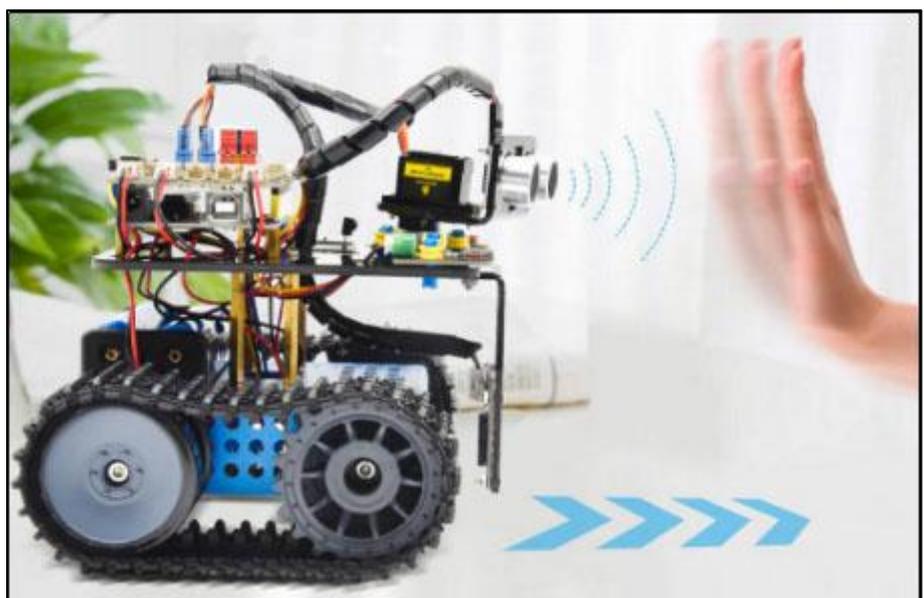
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

The robot may move immediately after uploading this code, so when uploading, please hold the robot with your hands to keep its tracks off the table to prevent it from falling.



3. Click upload >>>done uploading.

**Test Results:** After uploading the code, turn on the power switch of the motor drive shield. Place the robot on the floor and place your hand in front of the robot. You will find that it will keep a certain distance from your hand and follow your hand forward or backward.



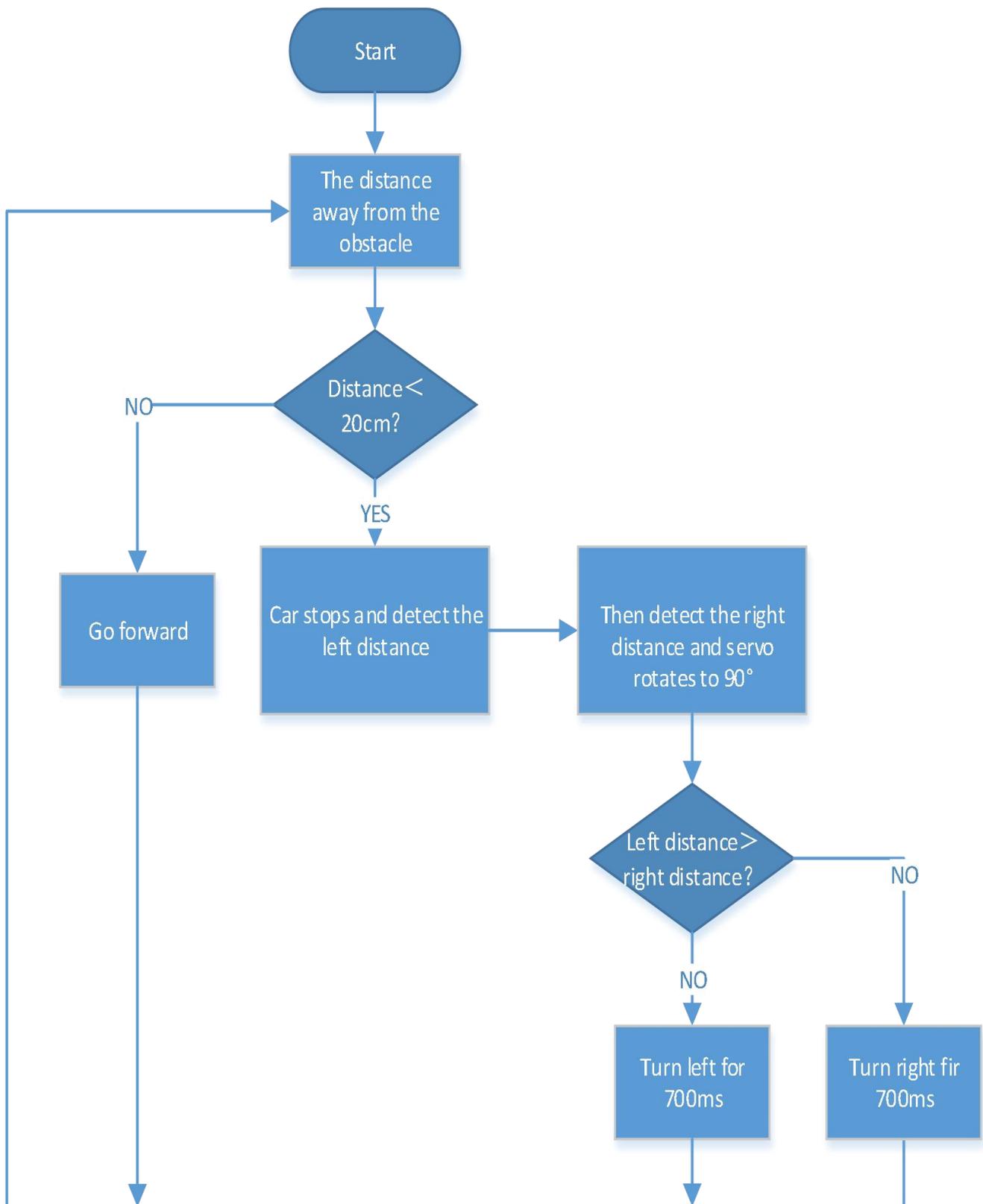
## Lesson 12: Ultrasonic Obstacle Avoidance Robot

In the previous project, we made an ultrasonic sound-following smart car. Using the same components and the same wiring method, we only need to change the test code to turn it into an ultrasonic obstacle avoidance smart car. We use ultrasonic sensors to detect the distance between the smart car and the obstacle in front, and then control the rotation of the two motors based on this data to control the movements of the smart car.

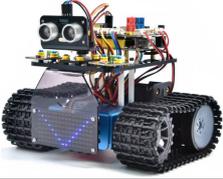
Detection	Distance measured by the ultrasonic sensor between the car and the obstacle in front (set the angle of the servo to 90°)		a (unit: cm)
	Distance measured by the ultrasonic sensor between the car and the obstacle on the right (set the angle of the servo to 20°)		a2 (unit: cm)
	Distance measured by the ultrasonic sensor between the car and the obstacle on the left (set the angle of the servo to 160°)		a1 (unit: cm)
Setting	set the starting angle of the servo to 90°		
Condition 1	Movement		
a < 20	Stop for 1000 ms; set the angle of the servo to 160°, read a1, delay in 500ms; set the angle of the servo to 20°, read a2, delay in 500ms.		
	Condition 2	Movement	
	a1 < 50 or a2 < 50	Compare a1 with a2	
		Condition 3	Movement
		a1 > a2	Set the angle of the servo to 90°, rotate left for 500ms (set PWM to 255) , and move forward (set PWM to 200) .
	a1 < a2	Set the angle of the servo to 90°, rotate right for 500ms (set PWM to 255) , and move forward (set PWM to 200) .	
	Condition 2	Movement	
a1 ≥ 50 and a2 ≥ 50	Random	set the angle of the servo to 90°, rotate left for 500ms (set PWM to 255) , and move forward (set PWM to 200) . set the angle of the servo to 90°, rotate right for 500ms (set PWM to 255) , and move forward (set PWM to 200) .	
Condition 2	Movement		
a ≥ 20	move forward (set PWM to 200)		

The specific logic of the ultrasonic obstacle avoidance smart car is shown in the table below:

## Flow chart



**You need to prepare:**

Robot without BT module*1	USB Cable*1	Computer*1	18650 Battery*2
			

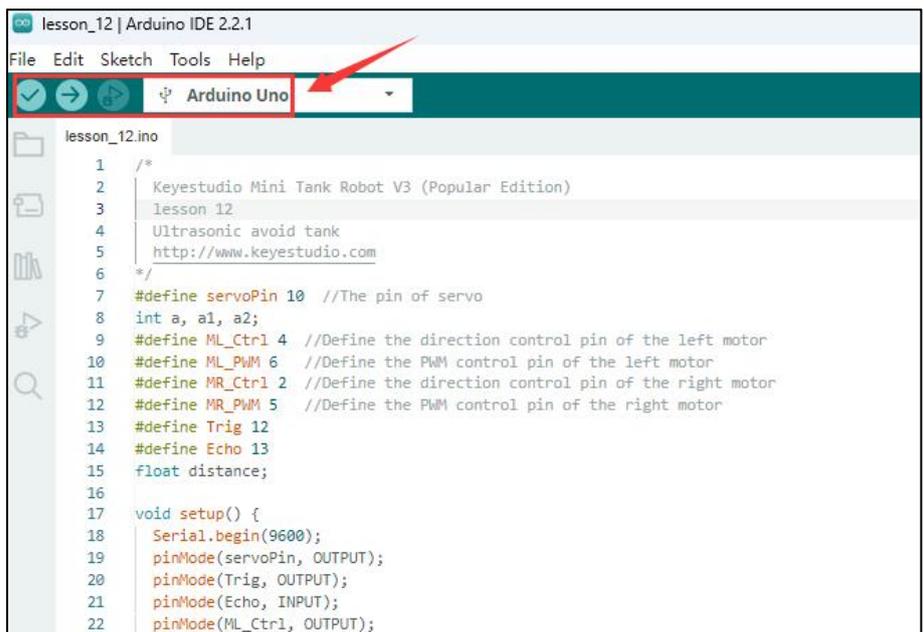
1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_12** folder with Arduino IDE.



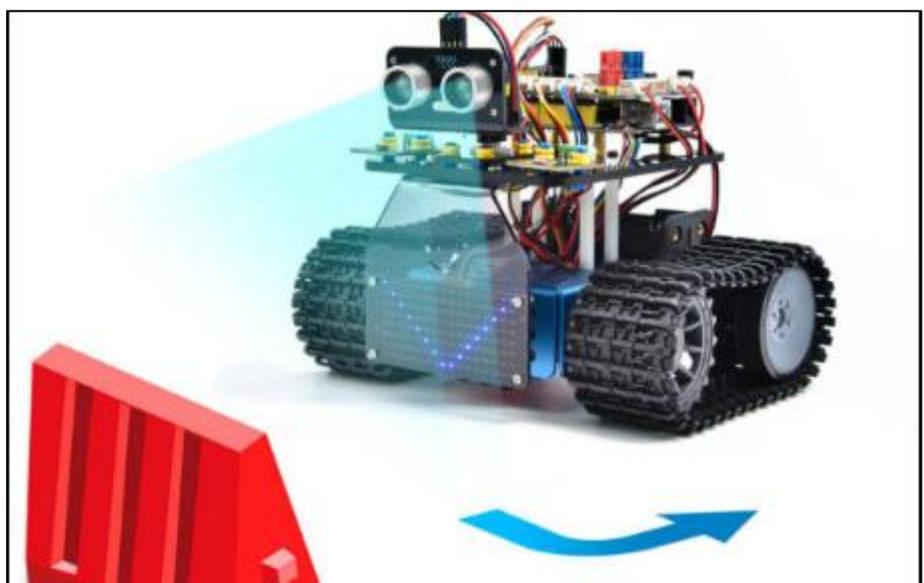
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

The robot will move immediately after uploading this code, so when uploading, please hold the robot with your hands to keep its tracks off the table to prevent it from falling.

3. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield. Place the robot on the floor and the smart car moves forward and automatically avoids obstacles.

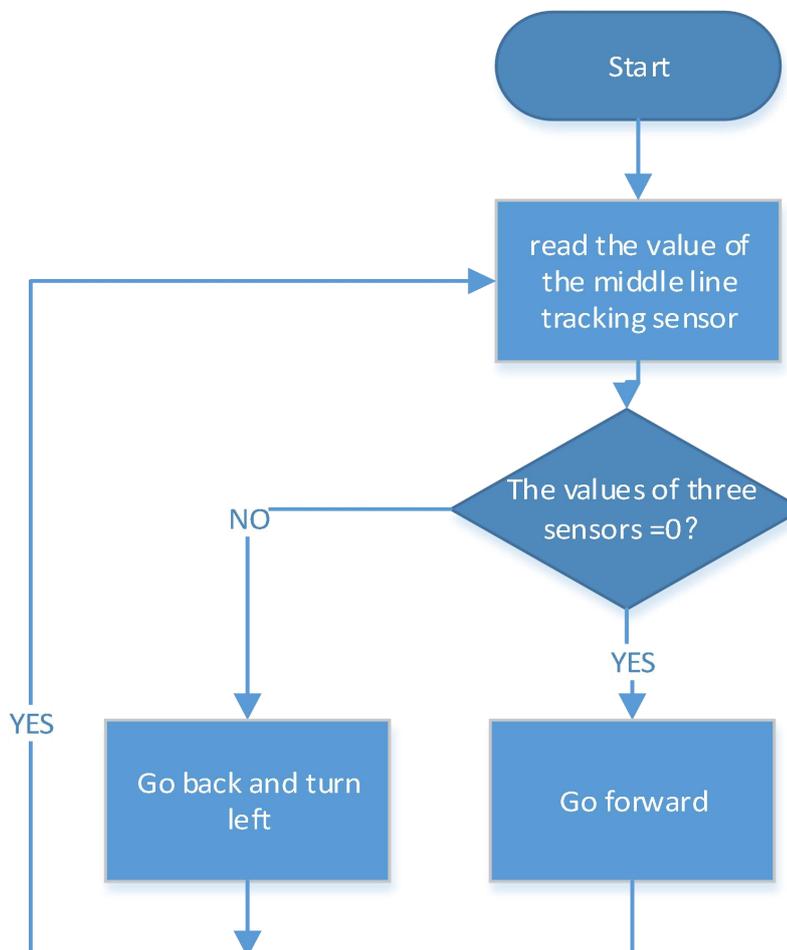


## Lesson 13: Robot Restricted by a circle

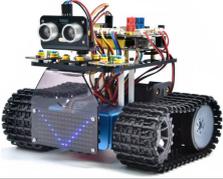
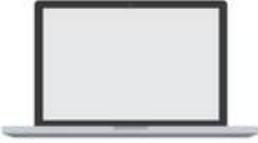
In this lesson, we use the line-tracking sensor to detect whether there is a black line around the smart car, and then control the rotation of the two motors according to the detection results, so as to lock the smart car in a circle drawn in black line.

Detection	Line-tracking sensor in the middle	Black line detected: in high level
		White line detected: in low level
	Line-tracking sensor on the left	Black line detected: in high level
		White line detected: in low level
	Line-tracking sensor on the right	Black line detected: in high level
		White line detected: in low level
Condition		Movement
All the three line-tracking sensors detect no black lines		go front
Any of the three line-tracking sensors detects black lines		go back then rotate to left

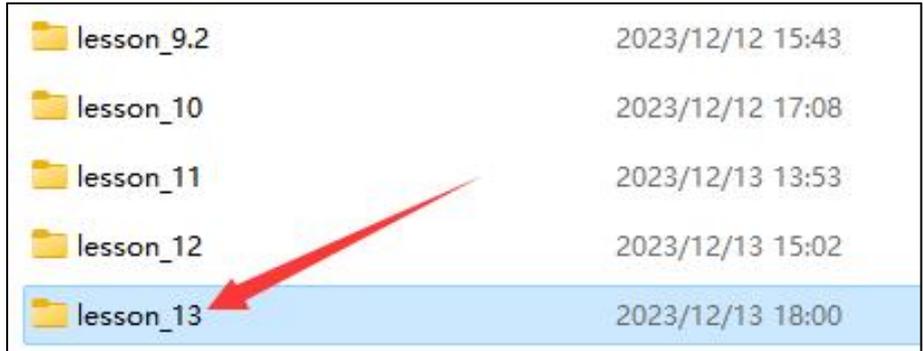
### Flow chart



**You need to prepare:**

Robot without BT module*1	USB Cable*1	Computer*1	18650 Battery*2
			

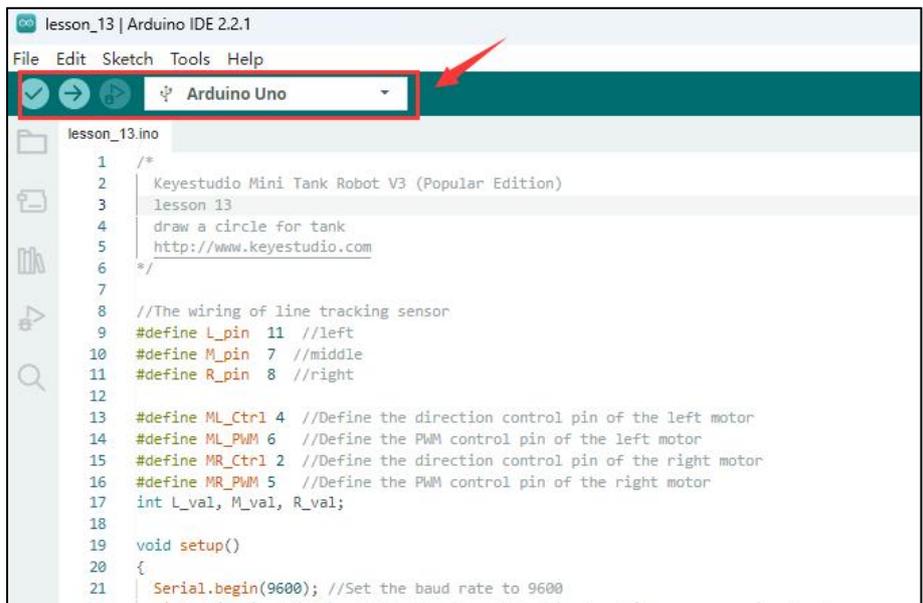
1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_13** folder with Arduino IDE.



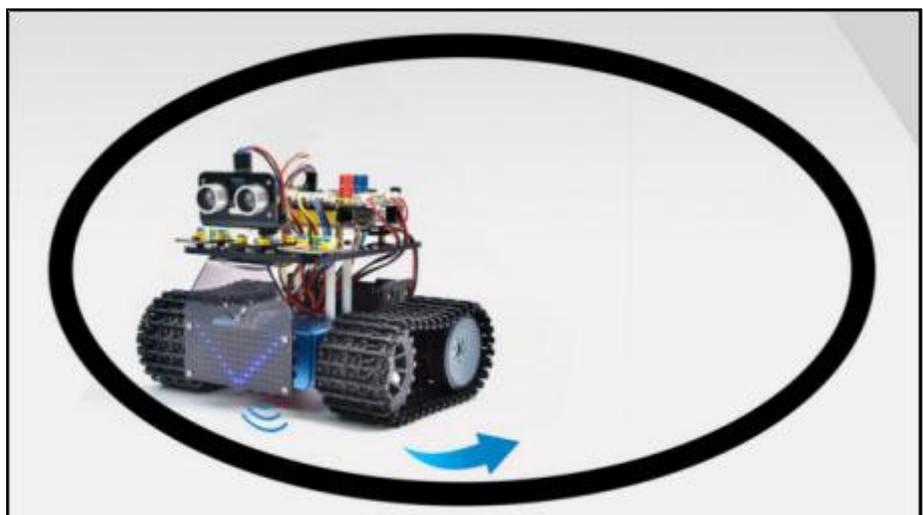
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

The robot will move immediately after uploading this code, so when uploading, please hold the robot with your hands to keep its tracks off the table to prevent it from falling.

3. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield. Draw a black circle on the white paper and place it on the floor. Place the robot inside and the robot will be restricted to run inside the black circle.



## Lesson 14: Line-tracking Tank

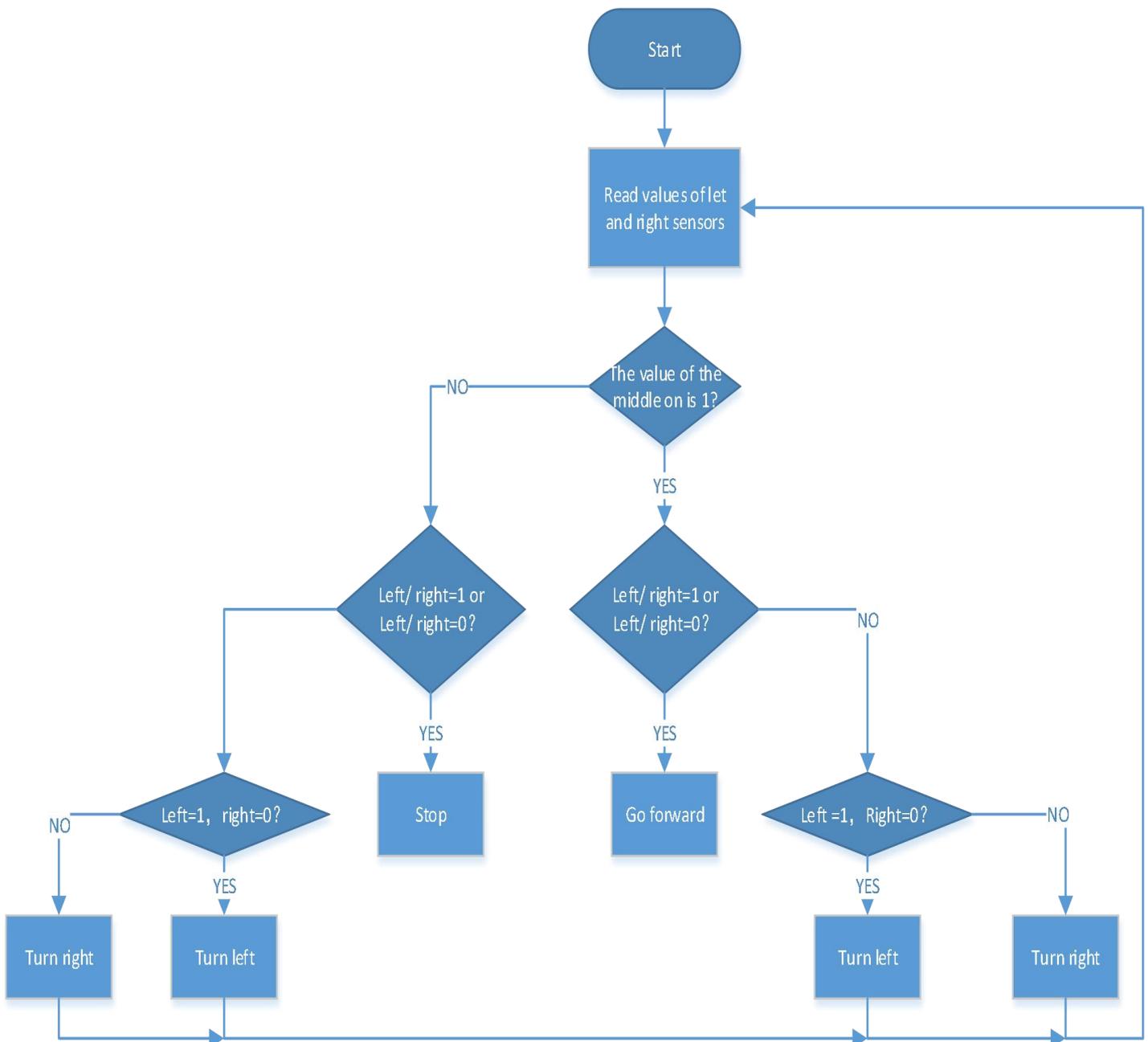
The previous project has introduced how to confine the smart car to move in a certain space. In this project, we could use the knowledge learned before to make it a line-tracking smart car. In the experiment, we use the line-tracking sensor to detect whether there is a black line around the smart car, and then control the rotation of the two motors according to the detection results, so as to make the smart car to move along the black line.

The specific logic of the line-tracking smart car is shown in the table below

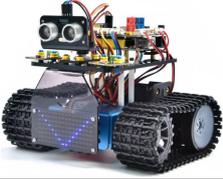
Detection	Line-tracking sensor in the middle	Black line detected: in high level
		White line detected: in low level
	Line-tracking sensor on the left	Black line detected: in high level
		White line detected: in low level
	Line-tracking sensor on the right	Black line detected: in high level
		White line detected: in low level
Condition		Movement
Line-tracking sensor in the middle detects the black line	Line-tracking sensor on the left detects the black line and the one on the right detects white lines	Rotate left (set PWM to 200)
	Line-tracking sensor on the left detects white lines and the one on the right detects the black line	Rotate right (set PWM to 200)
	Both the left and right line-tracking sensors detect white lines	Move forward
	Both the left and right line-tracking sensors detect the black line	
Line-tracking sensor in the middle detects white lines	Line-tracking sensor on the left detects the black line and the one on the right detects white lines	Rotate left (set PWM to 200)

	Line-tracking sensor on the left detects white lines and the one on the right detects the black line	Rotate right (set PWM to 200)
	Both the left and right line-tracking sensors detect white lines	Stop
	Both the left and right line-tracking sensors detect the black line	

## Flow chart



**You need to prepare:**

Robot without BT module*1	USB Cable*1	Computer*1	18650 Battery*2
			

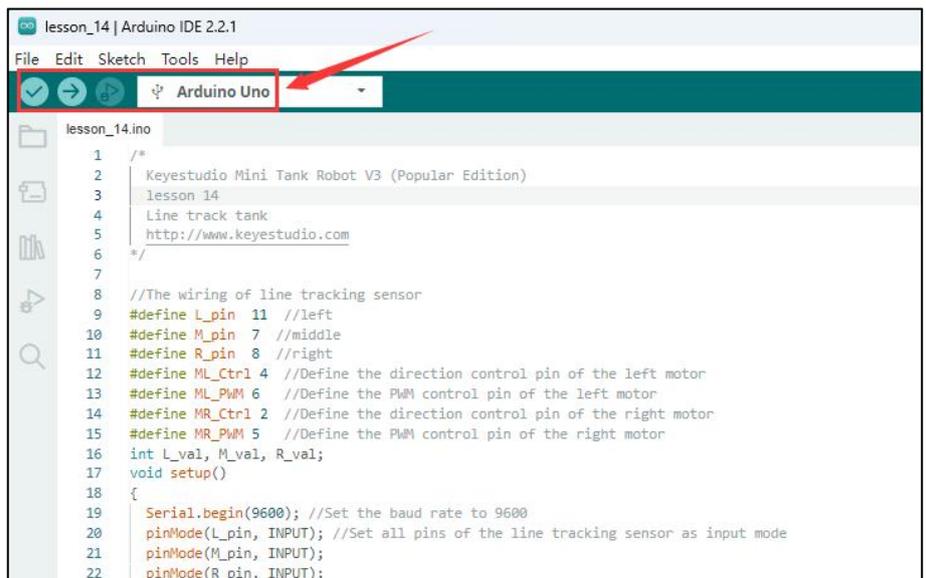
1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_14** folder with Arduino IDE.



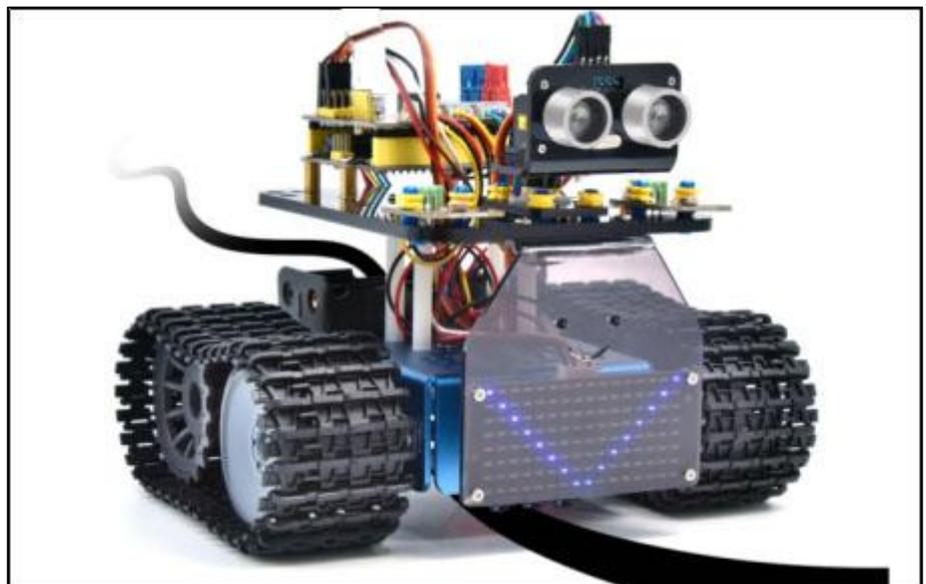
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield. Draw a black line of about 2-3cm on the white paper, place the robot on the floor and point the robot's line tracking sensor towards the black line, and the robot will follow the black line.



## Lesson 15: IR Remote Control Tank

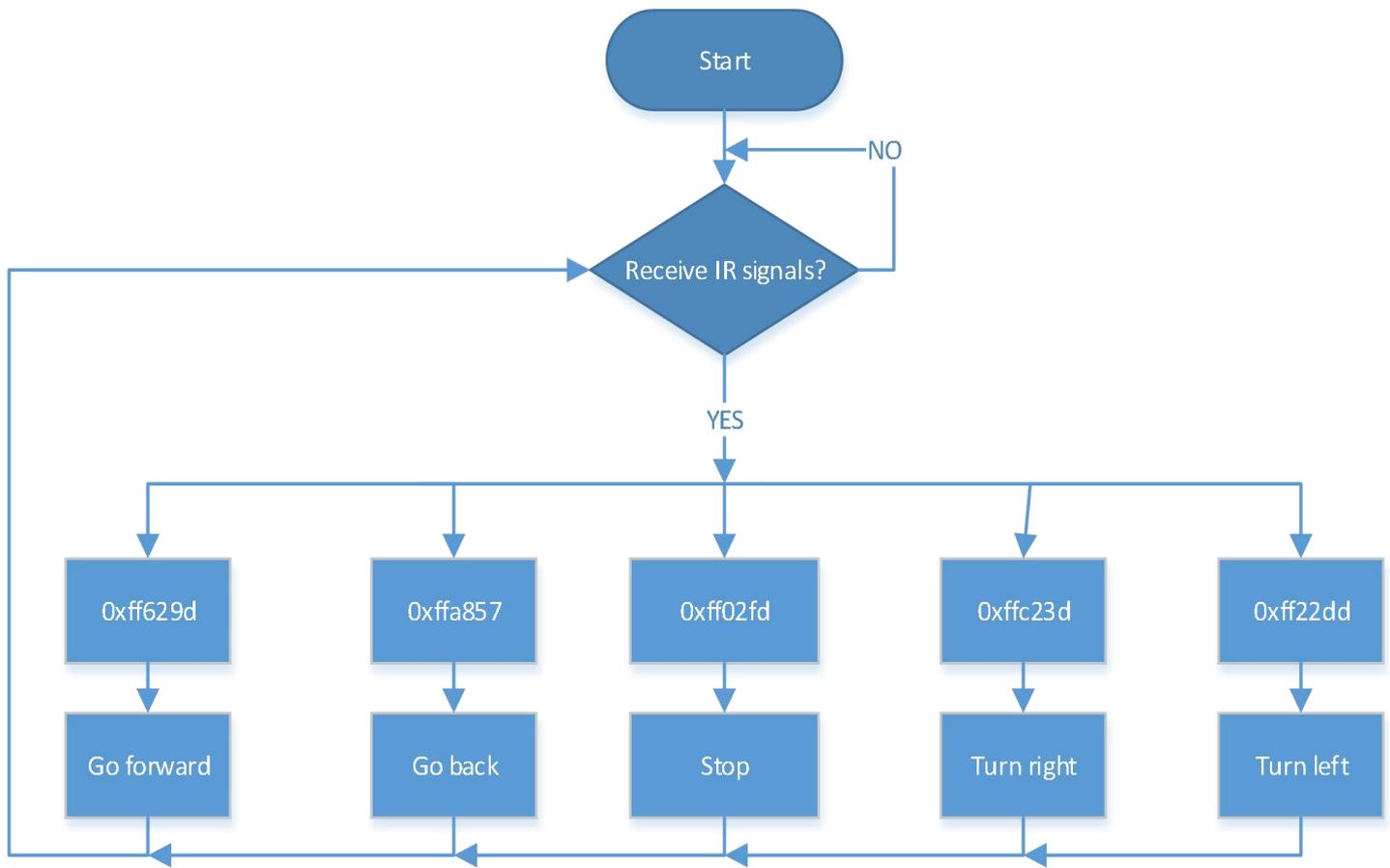
Infrared remote control is one of the most common remote control found applications in electric motors, electric fans, and many other household appliances.

We have tested the corresponding key value of each key of the infrared remote control before. In this lesson, we will set the key value of the remote control in the code to make the remote control to control the movements of the smart car, and display the movement direction of the car on the 8X16 LED dot matrix.

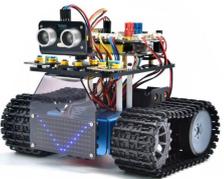
The specific logic of the IR remote control smart car is shown in the table:

Initial setting	Set the angel of the servo to 90°	
	8X16 LED dot matrix shows the pattern“V”	
Ultrasonic key	Key value	Instructions from keys
	FF629D	Move forward
		show the image to go front
	FFA857	Go back
		Go back
	FF22DD	Turn left
		display the pattern of turning left
	FFC23D	Turn right
		display the pattern of turning right
	FF02FD	Stop
		display the pattern“STOP”
	FF30CF	Turn left
		display the pattern“STOP”
	FF7A85	Rotate to right
		Show the pattern to rotate to right

# Flow chart



## You need to prepare:

Robot without BT module*1	USB Cable*1	Computer*1	Remote Control*1	18650 Battery*2
				

1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_15** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

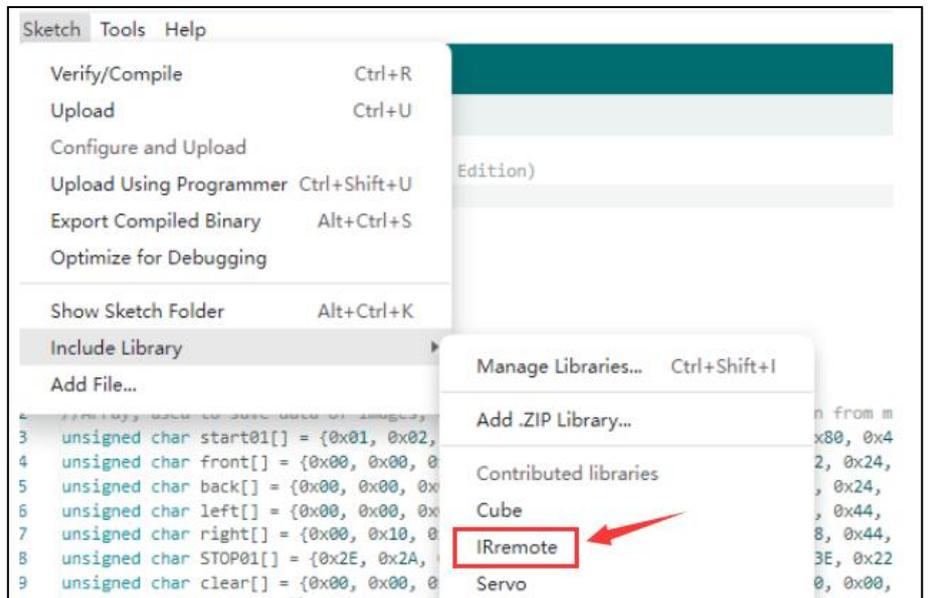
```

1 /*
2 Keystudio Mini Tank Robot V3 (Popular Edition)
3 lesson_15
4 IRremote
5 http://www.keystudio.com
6 */
7 #include <IRremote.h>
8 IRrecv irrecv(3); //
9 decode_results results;
10 long ir_rec; //Used to store the received infrared values
11
12 //Array, used to save data of images, can be calculated by yourself or gotten from modulus tool
13 unsigned char start01[] = {0x01, 0x02, 0x04, 0x08, 0x10, 0x20, 0x40, 0x80, 0x40, 0x20, 0x10, 0x08, 0x04, 0x02, 0x01};
14 unsigned char front[] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x24, 0x12, 0x09, 0x12, 0x24, 0x00, 0x00, 0x00, 0x00, 0x00};
15 unsigned char back[] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x24, 0x48, 0x90, 0x48, 0x24, 0x00, 0x00, 0x00, 0x00, 0x00};
16 unsigned char left[] = {0x00, 0x00, 0x00, 0x00, 0x00, 0x24, 0x44, 0x28, 0x10, 0x44, 0x28, 0x10, 0x44, 0x28, 0x10};
17 unsigned char right[] = {0x00, 0x10, 0x28, 0x44, 0x10, 0x28, 0x44, 0x10, 0x28, 0x44, 0x00, 0x00, 0x00, 0x00, 0x00};
18 unsigned char STOP01[] = {0x2E, 0x2A, 0x3A, 0x00, 0x02, 0x3E, 0x02, 0x00, 0x3E, 0x22, 0x3E, 0x00, 0x3E, 0x0A, 0x0E, 0x00};
19 unsigned char clear[] = {0x00, 0x00, 0x00};
20 #define SCL_Pin A5 //Set the clock pin as A5
21 #define SDA_Pin A4 //Set the data pin as A4
22

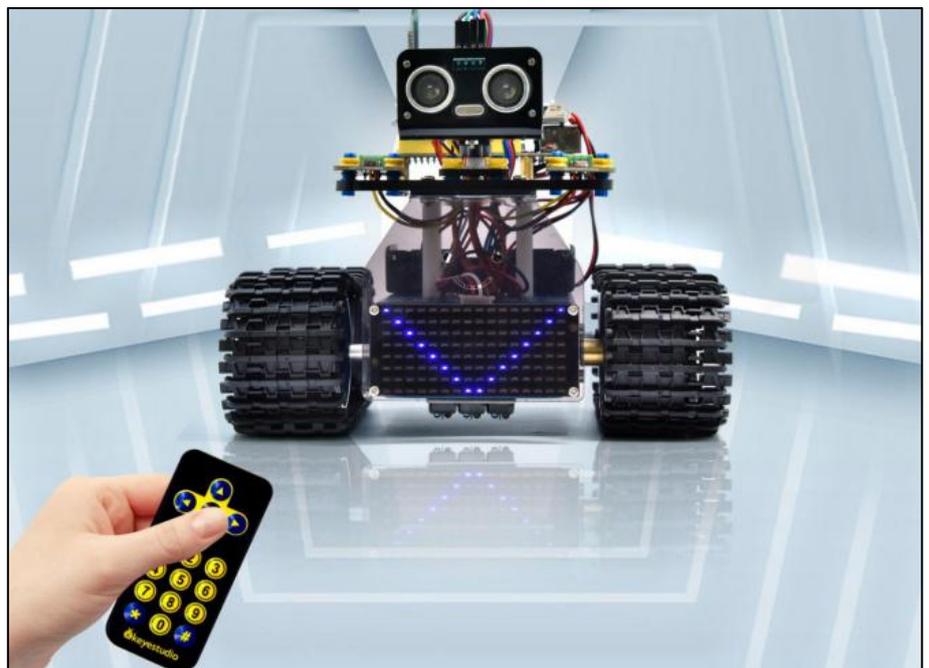
```

Don't connect to the Bluetooth module when uploading code  
Please make sure you have installed the IRremote library

3. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield. Place the robot on the floor, refer to the table above and press different buttons, the robot will move in the corresponding preset direction.



## Lesson 16: Install and Test the Bluetooth APP

The robot tank comes with a DX-BT24 Bluetooth module.



DX-BT24 5.1 Bluetooth module is built for intelligent wireless data transmission. It uses the DAILOG 14531 chip, configured with 256Kb space and follows the V5.1 BLE Bluetooth specification. Support AT instruction, users can change the serial port port rate, device name and other parameters as needed, flexible use.

### (2) Parameter

Bluetooth Protocol: Bluetooth Specification V5.1 BLE

Serial port sending and receiving without byte limit

Communication distance: 40m (open environment)

Operating frequency: 2.4GHz ISM band

Modulation method: GFSK (Gaussian Frequency Shift Keying)

Security Features: Authentication and Encryption

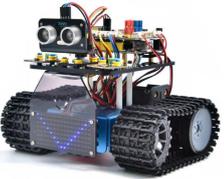
Support Services: Central and Peripheral UUIDs FFE0, FFE1, FFE2

Power consumption: automatic sleep mode, standby current 400uA~800uA, 8.5mA during transmission.

Power supply: 5V

Operating temperature: -10 to +65 degrees Celsius

### You need to prepare:

Robot tank*1	USB Cable*1	Computer*1	Bluetooth module*1	18650 Battery*2
				

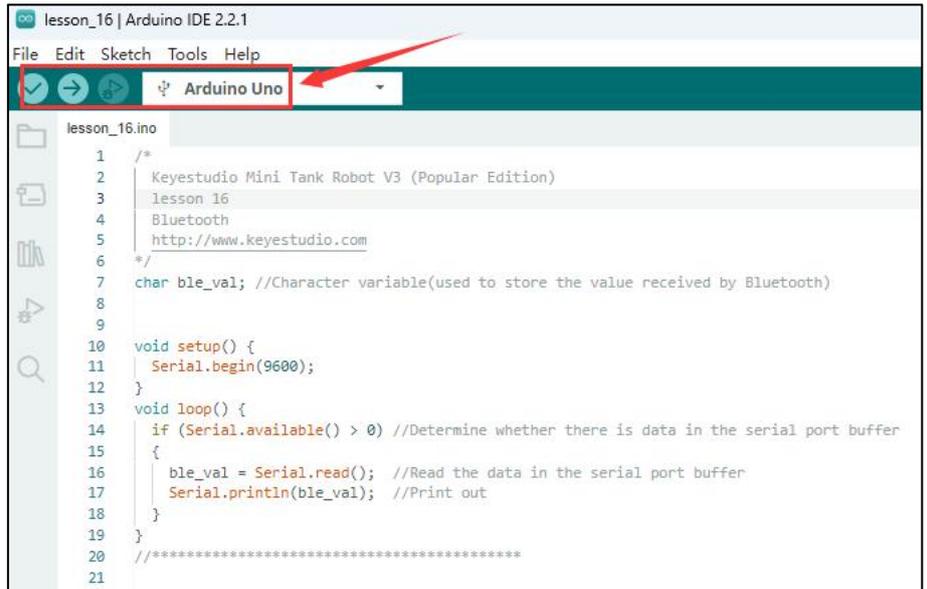
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_16** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Please upload the code before connecting the Bluetooth module.**

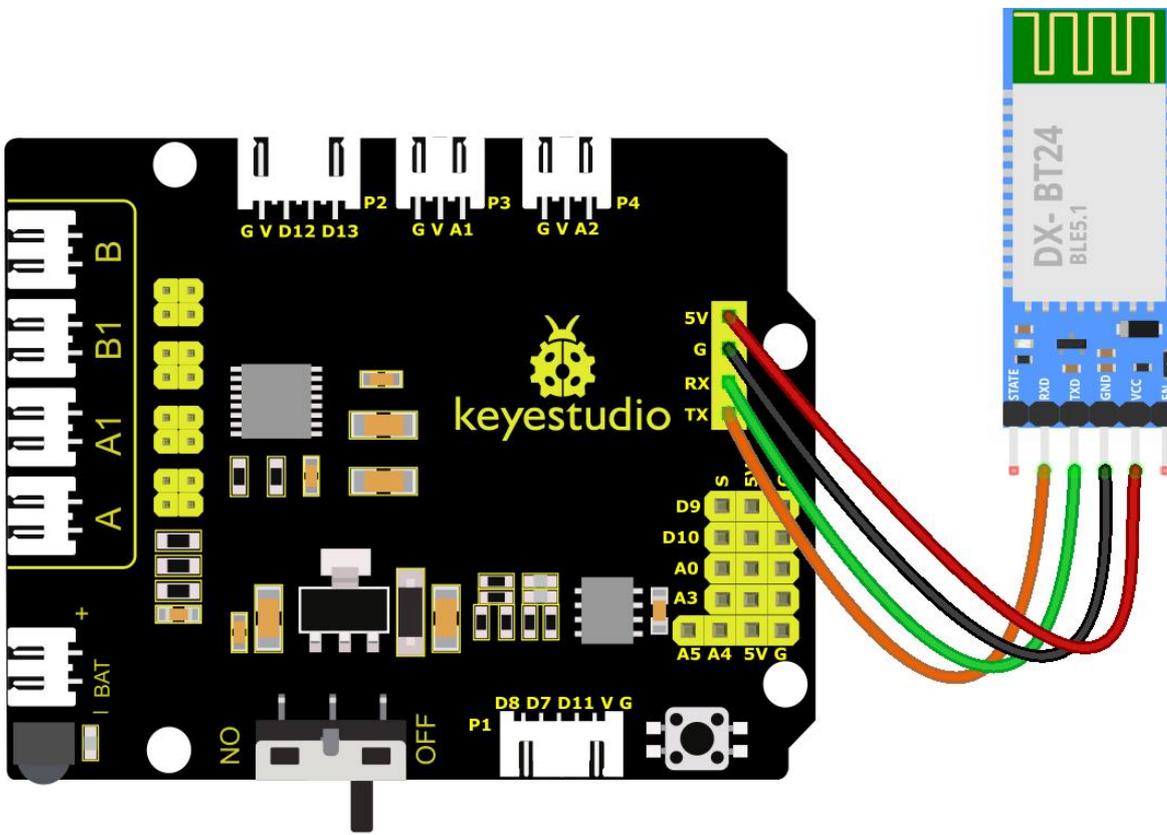
3. Click upload >>>done uploading.



## Connection Diagram:

- 1.STATE is the status test pin connected to the internal light-emitting diode and usually remains unconnected.
- 2.RXD is the serial port interface for receiving terminal.
- 3.TXD is the serial port interface for sending terminal.
- 4.GND is for ground.
- 5.VCC is the positive pole.
- 6.EN/BRK: the disconnection of it represents the disconnection of the Bluetooth and it usually remains unconnected.

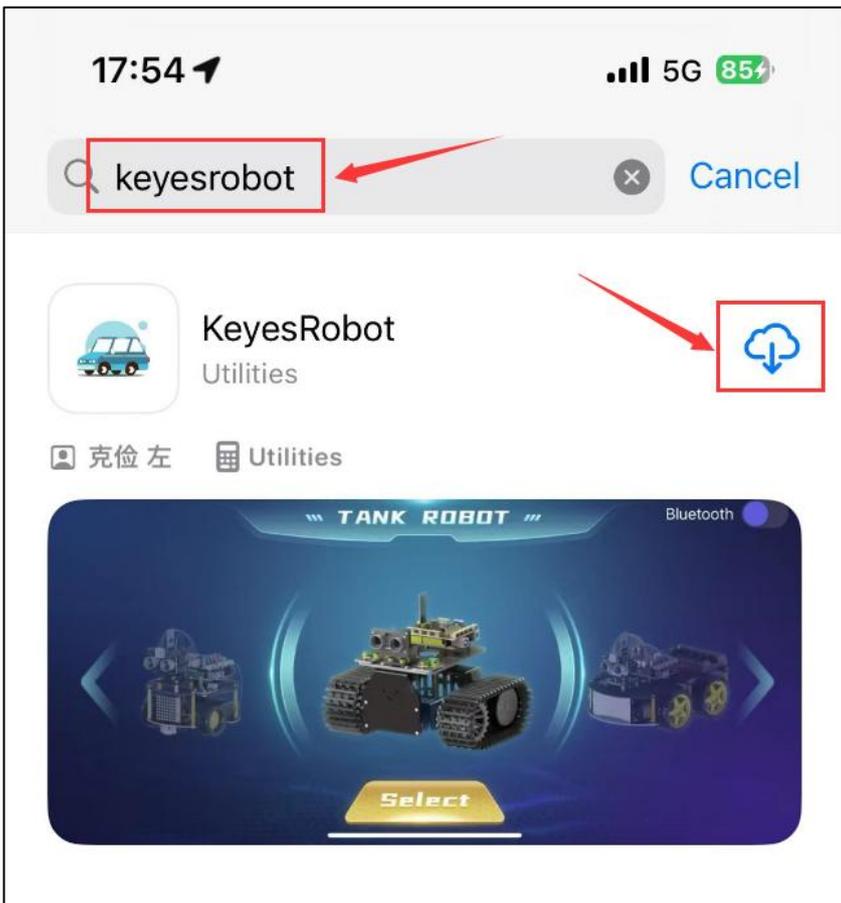
**(Note: No DuPont wire connection is required. The picture is for convenience to show the wiring method. The Bluetooth is directly insert in the motor shield and please pay attention to the direction)**



Download APP and connect the bluetooth:

**For iOS system**

1. Open App Store
2. Search **KeyesRobot** in the Apple Store and click download



3. After the app is installed, you will see the following icon on your phone desktop

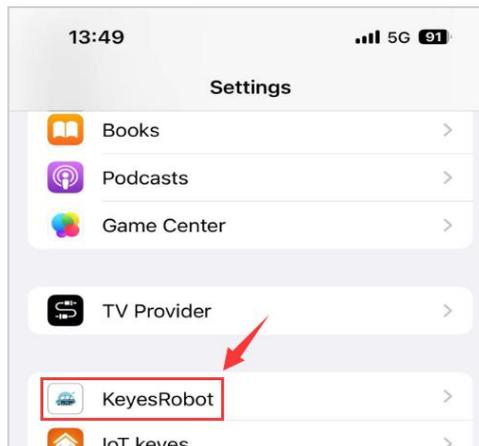


## How to connect APP to Bluetooth module on iOS system:

1. Turn on the Bluetooth and location services on phone through settings.



2. Allow KeyesRobot APP to access Bluetooth through settings.



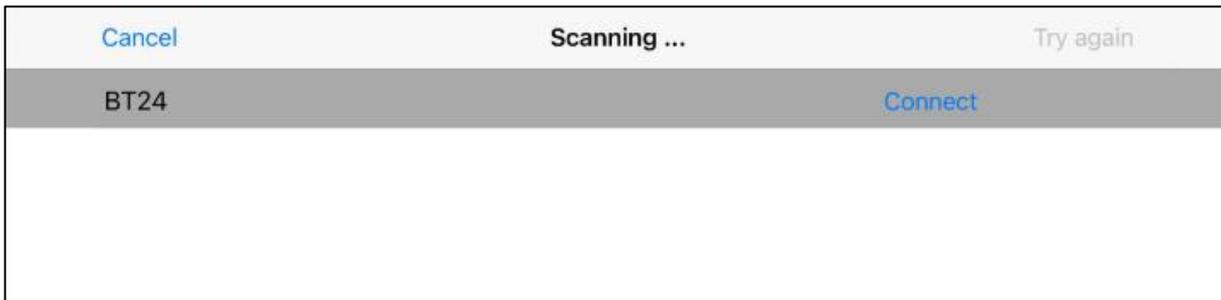
3. Click to open KeyesRobot App



4. KeyesRobot App is a universal APP, which is applied to multiple keystudio robots. If the interface does not display "TANK ROBOT", you can click the left and right buttons to find "TANK ROBOT"
5. Click the **Bluetooth button** Bluetooth  in the upper right corner to scan the bluetooth



6. You will see a Bluetooth named **BT24**, click the connect button



7. If the onboard LED on the Bluetooth module stops flashing and stays on, it means your phone is successfully connected to the Bluetooth module.



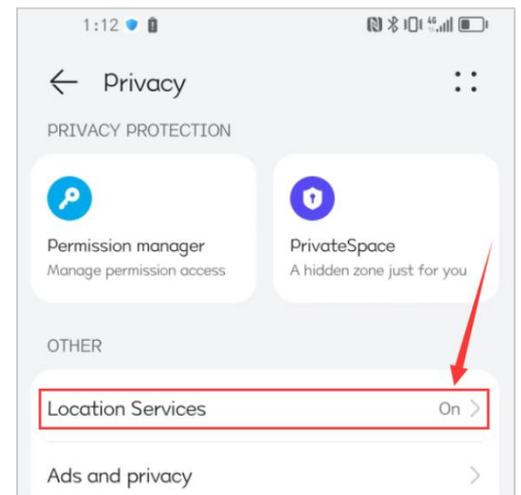
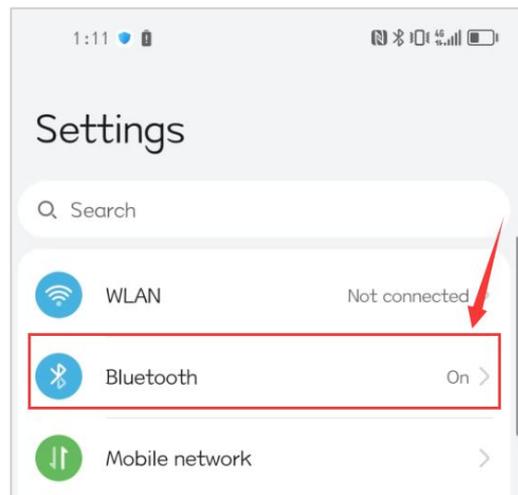
## For Android System

1. Search **KeyesRobot** in Google Play, or open the following link to download and install the app.

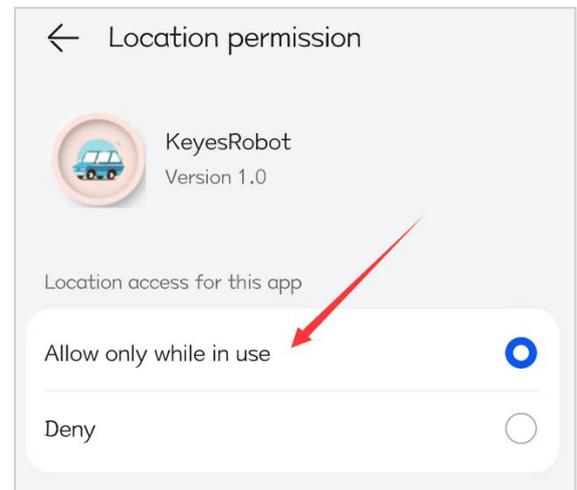
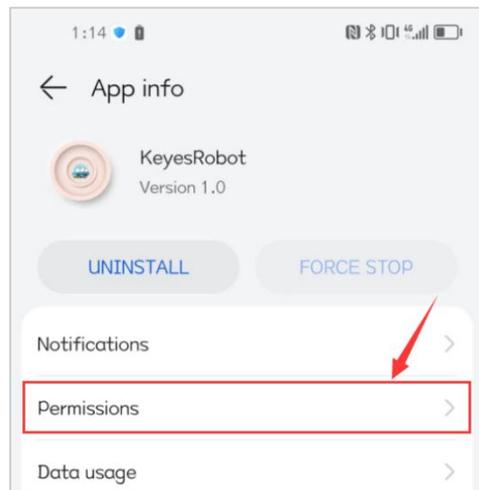
<https://play.google.com/store/apps/details?id=com.keyestudio.keyestudio>



2. Turn on the Bluetooth and the location services of the mobile phone



3. Find the KeyesRobot Bluetooth app from settings, click on the permission options of the app, and enable Location and nearby device permissions. (**Note:** Some mobile phones do not have nearby device permissions function.)



4. Click to open KeyesRobot App



KeyesRobot App is a universal APP, which is applied to multiple keyestudio robots. If the interface does not display "TANK ROBOT", you can click the left and right buttons to find "TANK ROBOT"

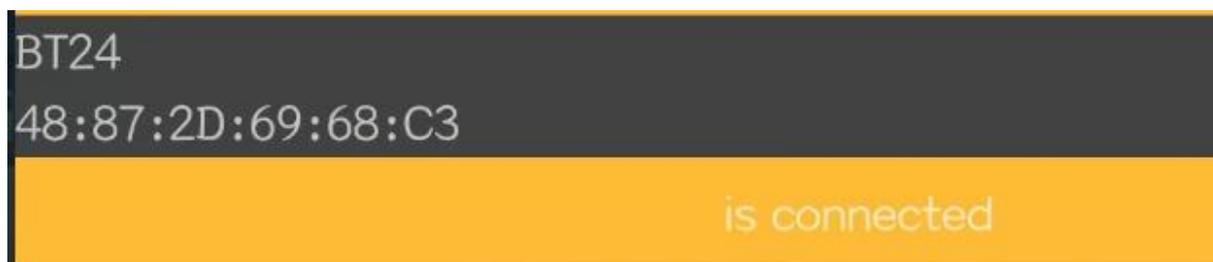


5. Click the Bluetooth button  Bluetooth in the upper right corner to scan the bluetooth

6. Find the Bluetooth named BT24, click the connect button



7. When your phone is successfully connected to the Bluetooth module, the onboard LED on the Bluetooth module will stop flashing and stay on.



## Test the Bluetooth APP:

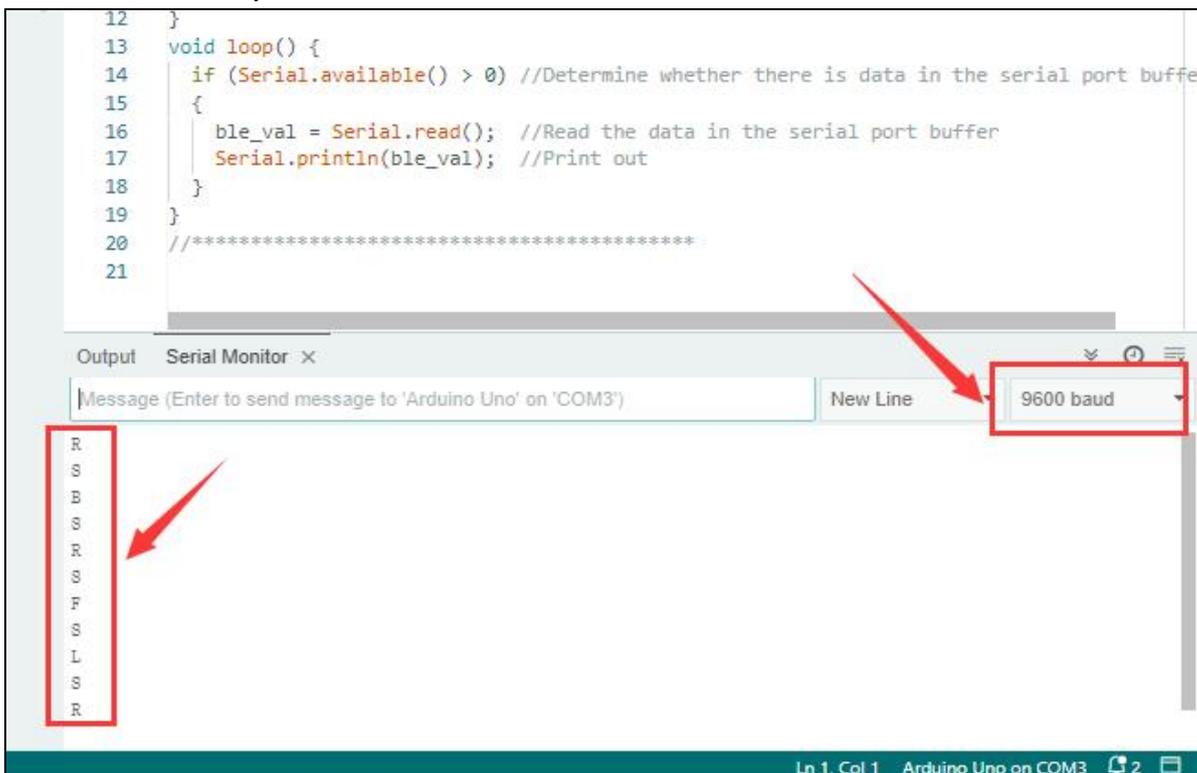
After the mobile phone is successfully connected to the Bluetooth module, click to open the Bluetooth APP and click the **Select** button on the **homepage**.



The main interface of the Bluetooth app is shown in the figure below.

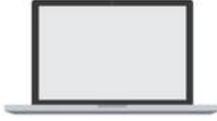
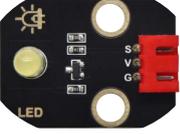


After the code above is successfully uploaded, open the serial monitor of the arduino IDE and set the baud rate to 9600. Click the icon on the APP interface and the serial monitor will display command sent by button.

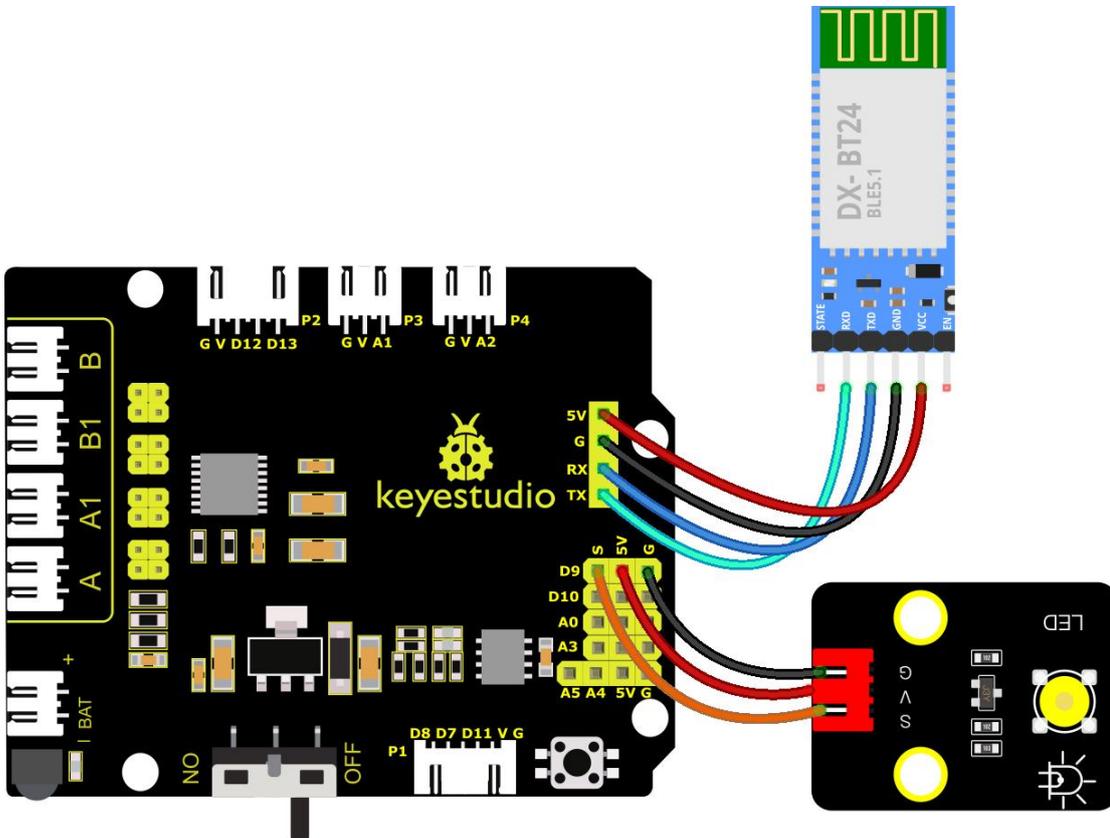


## Lesson 17: Bluetooth controlled LDE

You need to prepare:

Robot tank*1	USB Cable*1	Computer*1	Bluetooth module*1
			
Yellow LED Module*1	3P-3P XH2.54 to 2.54 Dupont Wire	18650 Battery*2	
			

In the last lesson we learned how to install the Bluetooth APP and connect it to the Bluetooth module. In this lesson, we will use Bluetooth APP to control the on and off of the LED module. The LED module is connected to the D9 pin.



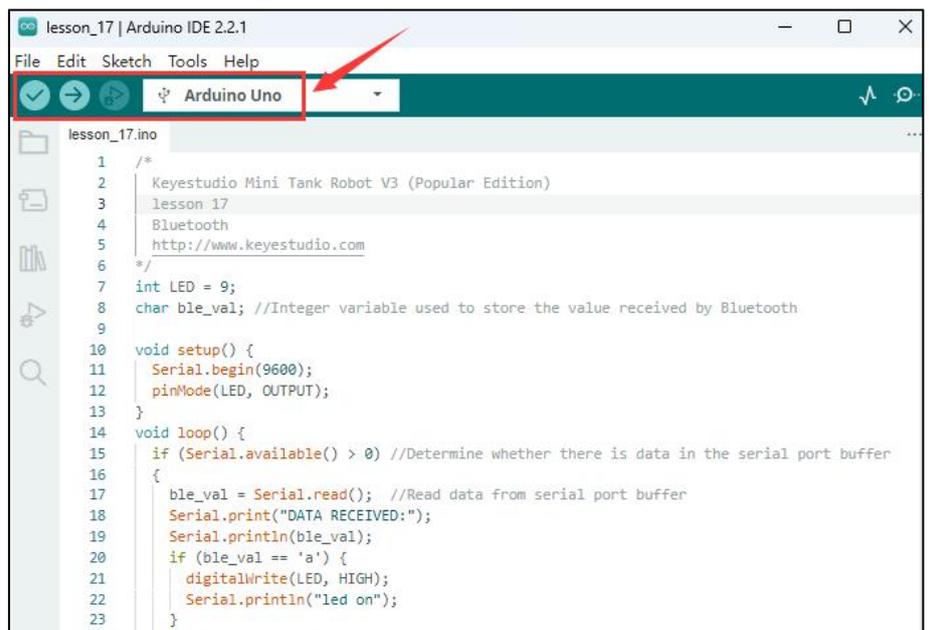
1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_17** folder with Arduino IDE.



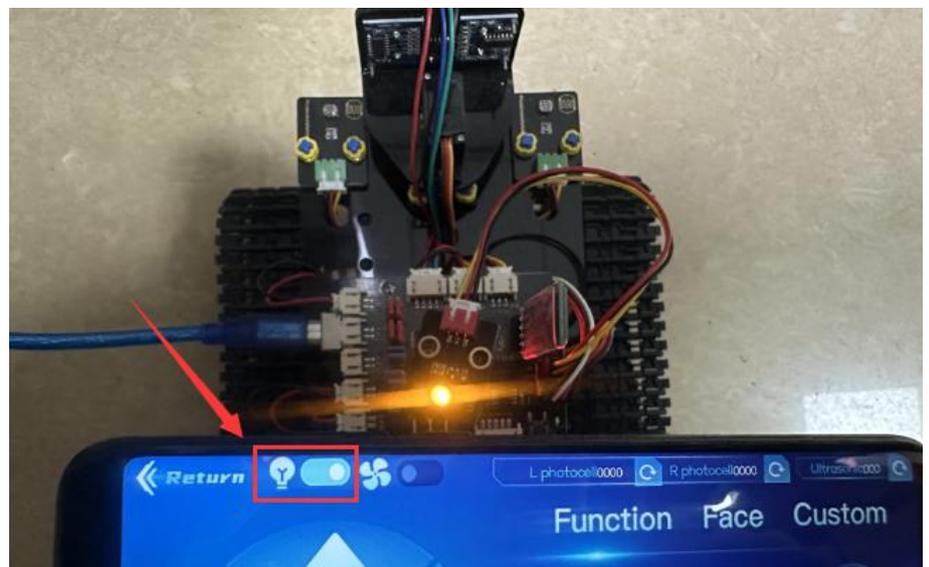
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

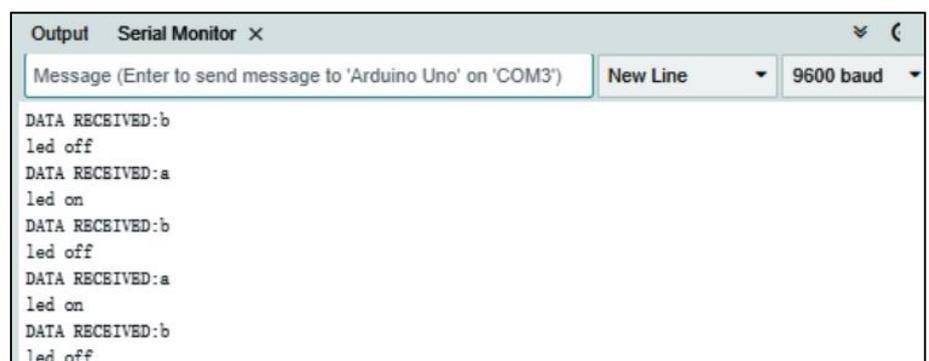
3. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield. Place the robot on the floor and Click  to turn on and off the LED.

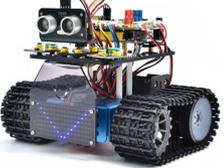
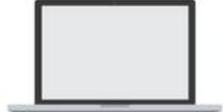


Open the serial monitor of the arduino IDE and set the baud rate to 9600. When you click the light button in the app, the LED switch status will be displayed on the serial monitor.



## Lesson 18: Bluetooth controlled Tank

You need to prepare:

Robot tank*1	USB Cable*1	Computer*1	Bluetooth module*1	18650 Battery*2
				

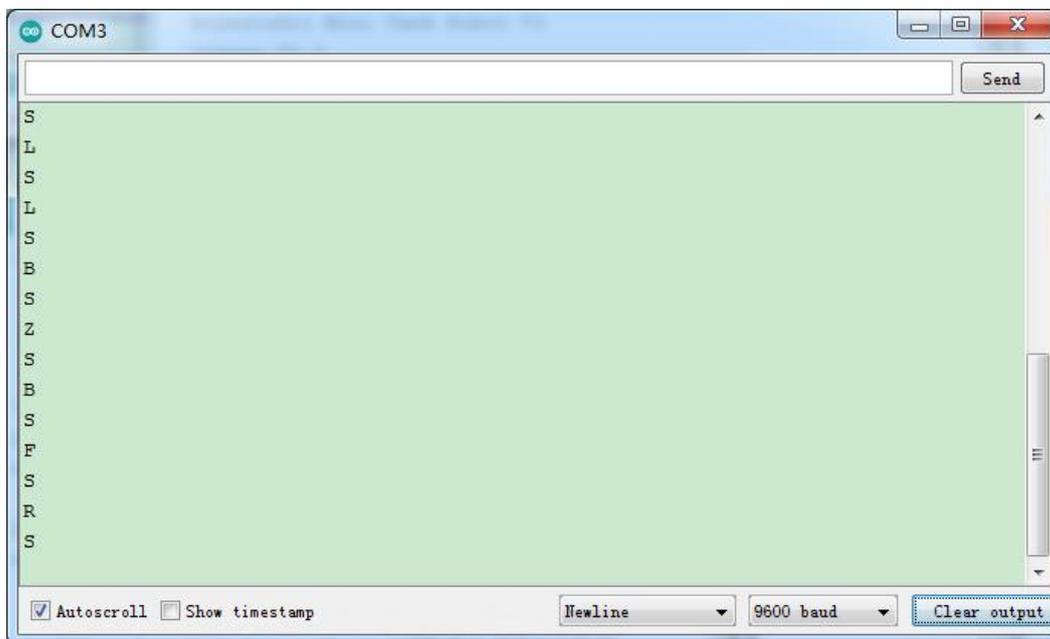
Once the Bluetooth APP is connected to the DX-BT24 V5.1 Bluetooth module, we can send commands to the main control board of the robot to control it by clicking the buttons in the APP.

You can read the corresponding command values of the APP button on the serial monitor by uploading the following code.

```
/*
  Keystudio Mini Tank Robot V3 (Popular Edition)
  bluetooth test
  http://www.keyestudio.com
*/
char ble_val; //Character variable(used to store the value received by Bluetooth)
void setup() {
  Serial.begin(9600);
}
void loop() {
  if(Serial.available() > 0) //Determine whether there is data in the serial port buffer
  {
    ble_val = Serial.read(); //Read the data in the serial port buffer
    Serial.println(ble_val); //Print out
  }
}
//*****
```

**Note: Do not connect the Bluetooth module when uploading the code. You can connect the Bluetooth module after the code is uploaded. Otherwise, the code will not be uploaded successfully.**

Open the serial monitor and set the baud rate to 9600. Click the button in the APP, and the set command value will be displayed on the serial monitor.

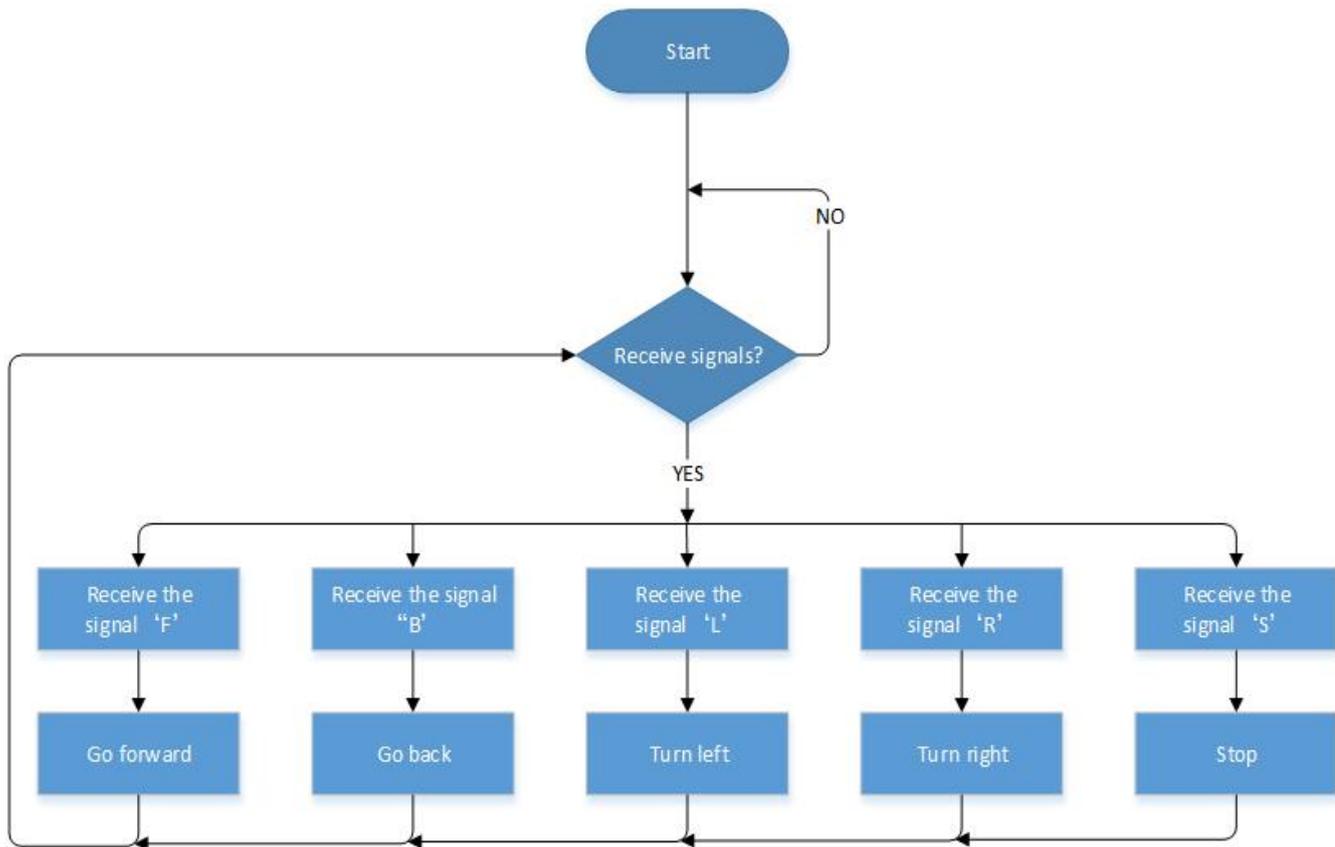


The following table lists the command values and functions corresponding to the App keys:

Keys	Functions	
	Pair and connect DX-BT24 V5.1 Bluetooth module;click again to disconnect	
	select the robot to operate	
	to control the movements of the robot by buttons	
	To control the movements of the robot by joystick	
	To control the movements of the robot by gravity	
	Send “F” when pressed and “S” when released	The car moves forward when it is pressed and stops when released
	Send “L” when pressed and “S” when released	The car turns left when it is pressed tight and stops when released
	Send “R” when pressed and “S” when released	The car turns right when it is pressed tight and stops when released
	Send “B” when pressed and “S” when released	The car turns back when it is pressed tight and stops when released
	Send “u” +digit+ “#” when dragged	Drag to change the speed of the left motor
	Send “v” +digit+ “#” when dragged	Drag to change the speed of the right motor
	Select to enter Function page	
	Send “G” when pressed and “S” when pressed again	Enter obstacle avoidance mode when pressed and exit when pressed again

	Send “h” when pressed and “S” when pressed again	Enter following mode when pressed and exit when pressed again
	Send “e” when pressed and “S” when pressed again	Enter line-tracking mode when pressed and exit when pressed again
	Send “f” when pressed and “S” when pressed again	Enter move-in-confined-space mode when pressed and exit when pressed again
	Send “i” when pressed and “S” when pressed again	Enter light following mode when pressed and exit when pressed again
	Send “j” when pressed and “S” when pressed again	Enter fire extinguishing mode when pressed and exit when pressed again
<b>Face</b>	Select to enter facial expression display mode	
	Send “k” when pressed and “z” when pressed again	Show smiling pattern when clicked and clear expression when clicked again
	Send “l” when pressed and “z” when pressed again	Show disgusting pattern when clicked and clear expression when clicked again
	Send “m” when pressed and “z” when pressed again	Show happy face when clicked and clear expression when clicked again
	Send “n” when pressed and “z” when pressed again	Show sad pattern when clicked and clear expression when clicked again
	Send “o” when pressed and “z” when pressed again	Show disparaging pattern when clicked and clear expression when clicked again
	Send “p” when pressed and “z” when pressed again	Show heart-shaped pattern when clicked and clear expression when clicked again
<b>Custom</b>	Choose to enter the custom function interface; there are six keys 1,2,3,4,5,6; with these keys, you can expand some functions by yourself	
	Click to send “w”	Click to display the analog value detected by the photoresistor on the left
	Click to send “y”	Click to display the analog value detected by the photoresistor on the right
	Click to send “x”	Click to show the distance detected by ultrasonic sensor (unit: cm)
	Click to send “c” Click again to send “d”	Press to turn on the fan and press again to turn off it

## Flow Chart



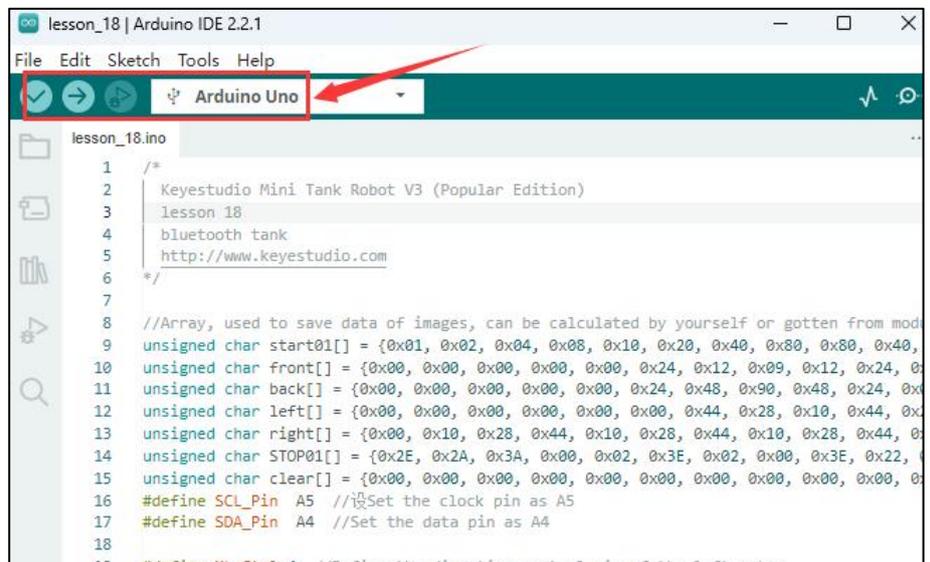
1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_18** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Note: Do not connect the Bluetooth module when uploading the code.**

3. Click upload >>>done uploading.



**Test Results:** After uploading the code, connect the robot to the Bluetooth module and pair the Bluetooth APP. Turn on the power switch of the motor drive shield. Place the robot on the floor, you can use these buttons of the Bluetooth app to control the robot.

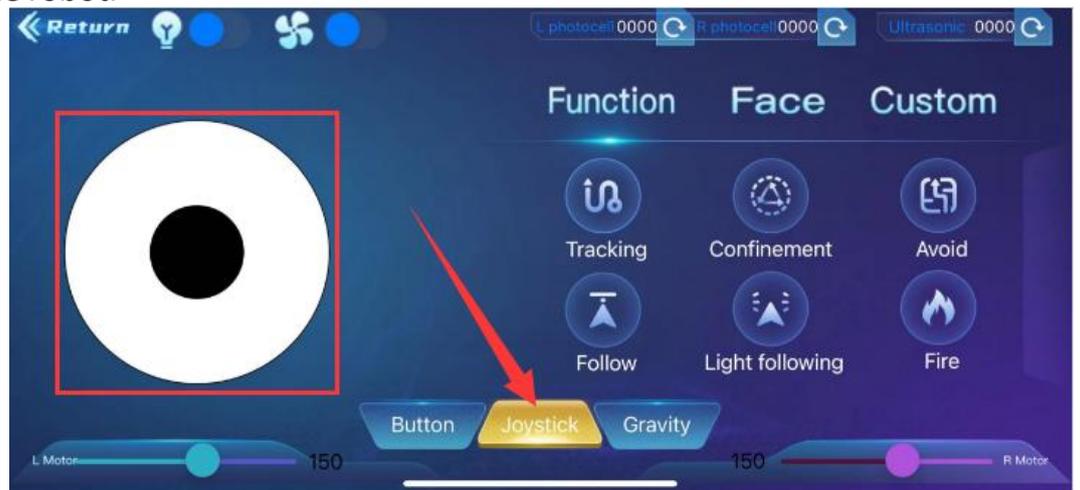
1. The up, down, left and right arrows control the robot to move forward, backward, left and right respectively.

1



2. Click the joystick button and pull the direction of the black point in the white circle to control the movement direction of the robot.

2



3. Click the Gravity button and tilt the phone in the forward, backward, left, and right directions, and the robot will move in the direction in which the phone is tilted.

3



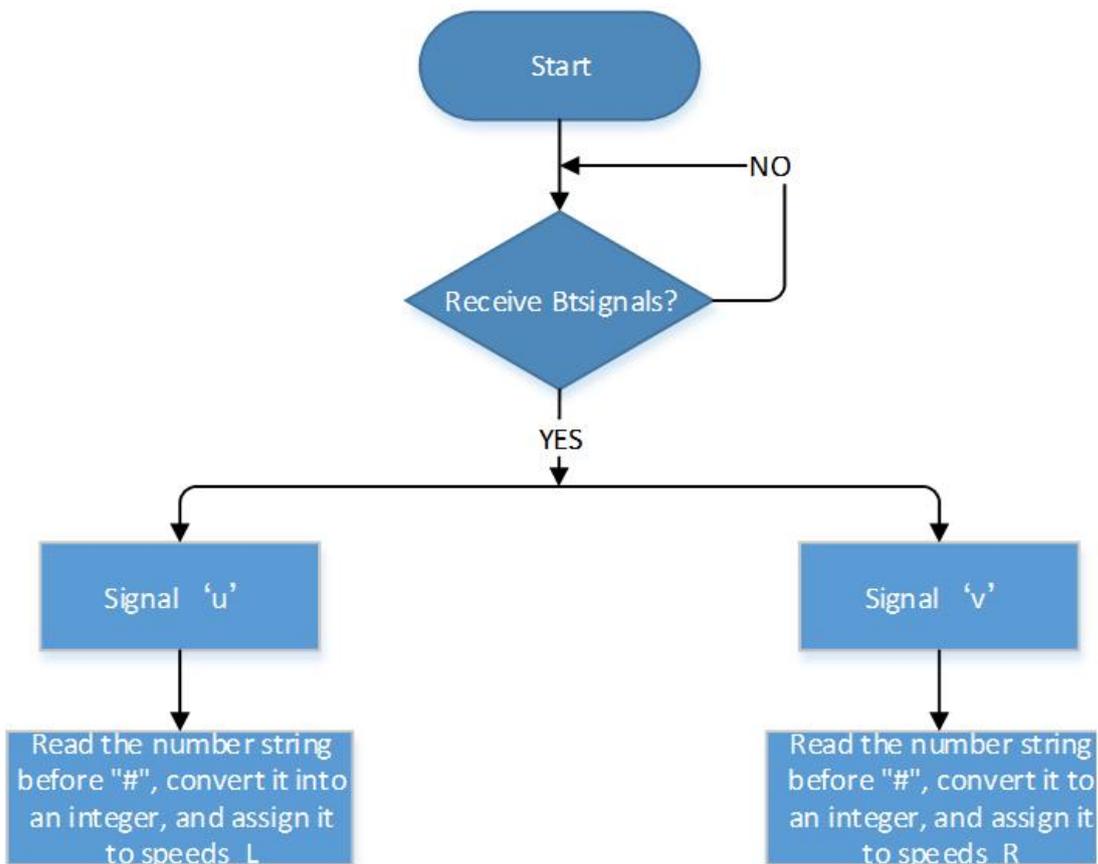
## Lesson 19: Control the Move Speed of the Robot via Bluetooth

>>>>>>> *This lesson is an expansion of the Lesson 18*

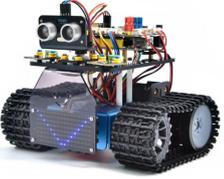


In the previous project, we learned how to control the smart tank with Bluetooth. The PWM value of the motor we used in front of us is 200 (the speed is 200). In this lesson, we will use Bluetooth to adjust the running speed of the smart car. It is not limited to Fixed speed of 200. We define two variables to store the speed values of the left and right motors respectively, this value can only take 0 to 255.

### Flow chart



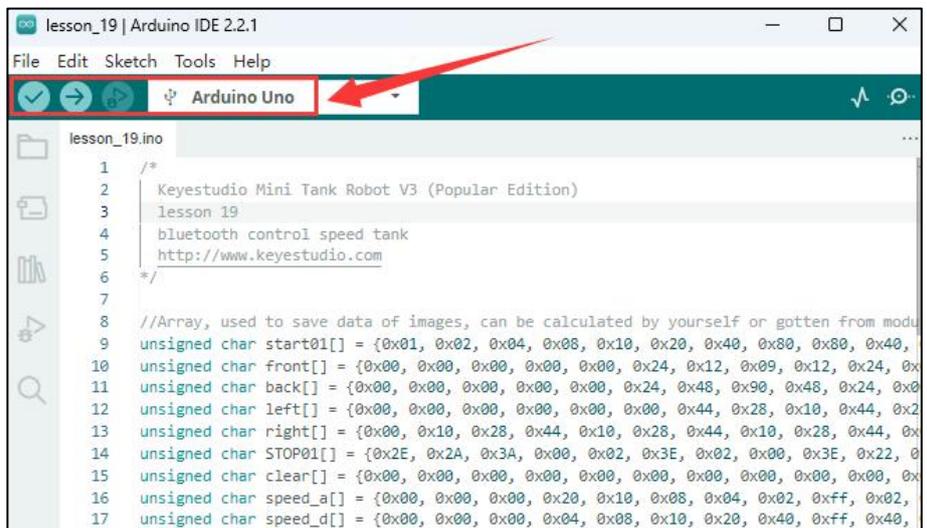
**You need to prepare:**

Robot tank*1	USB Cable*1	Computer*1	Bluetooth module*1	18650 Battery*2
				

1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_19** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.



**Don't connect to the Bluetooth module when uploading code**

3. Click upload >>>done uploading.

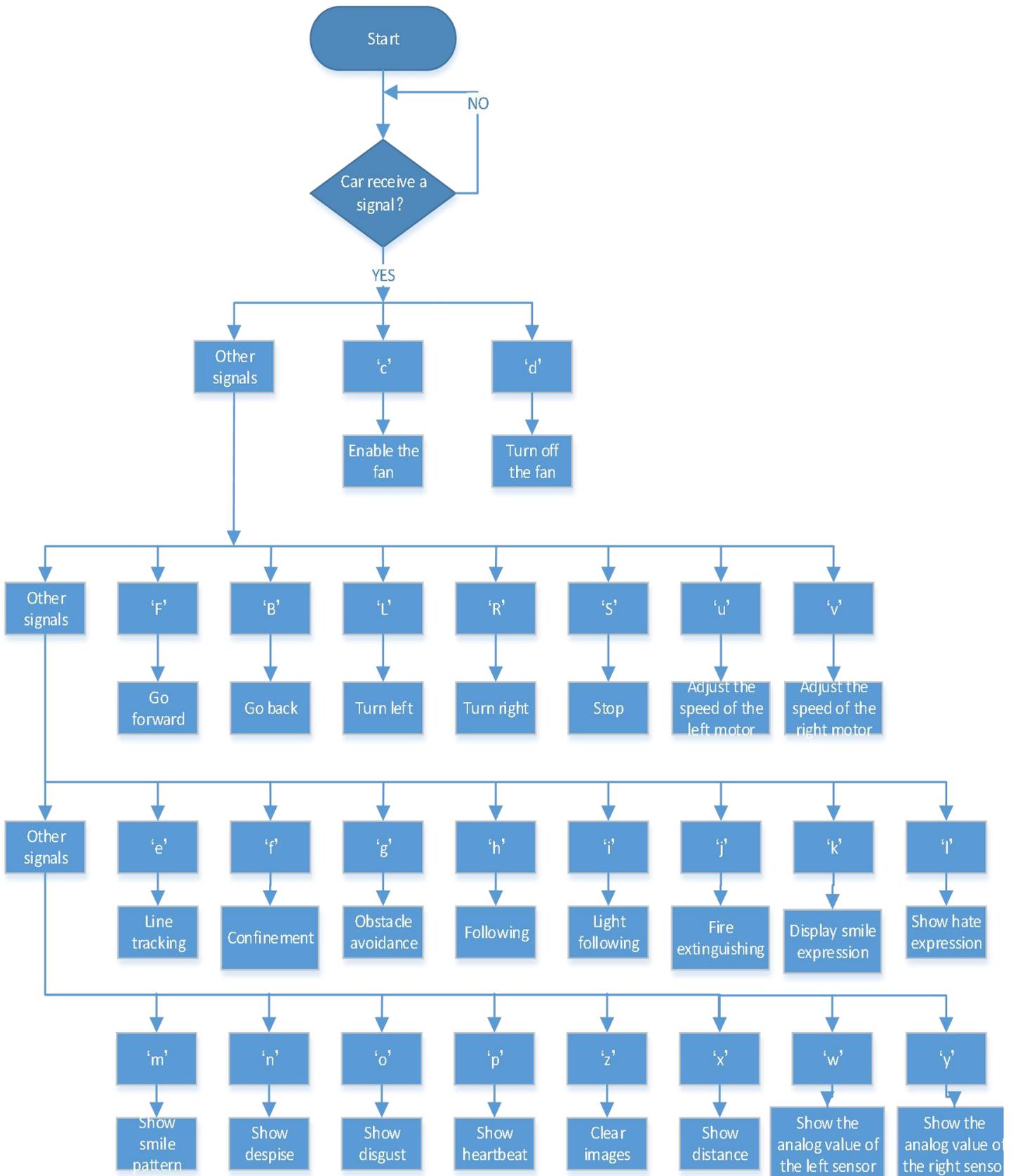
**Test Results:** After uploading the code, connect the robot to the Bluetooth module and pair the Bluetooth APP. Turn on the power switch of the motor drive shield. Place the robot on the floor, you can control its movement through the Button, Joystick, and Gravity buttons of the Bluetooth APP and drag the slider button to change the running speed of the left and right motors.



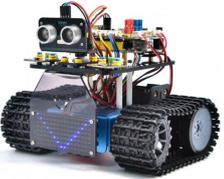
## Lesson 20: Multifunctional Tank

In the previous courses, we could only use Bluetooth APP or remote control to control the robot alone. In this course, we will upload a more complete code that allows us to use Bluetooth APP or remote control to control the robot at the same time.

### Flow chart



**You need to prepare:**

Robot tank*1	USB Cable*1	Computer*1	Bluetooth module*1	18650 Battery*2
				

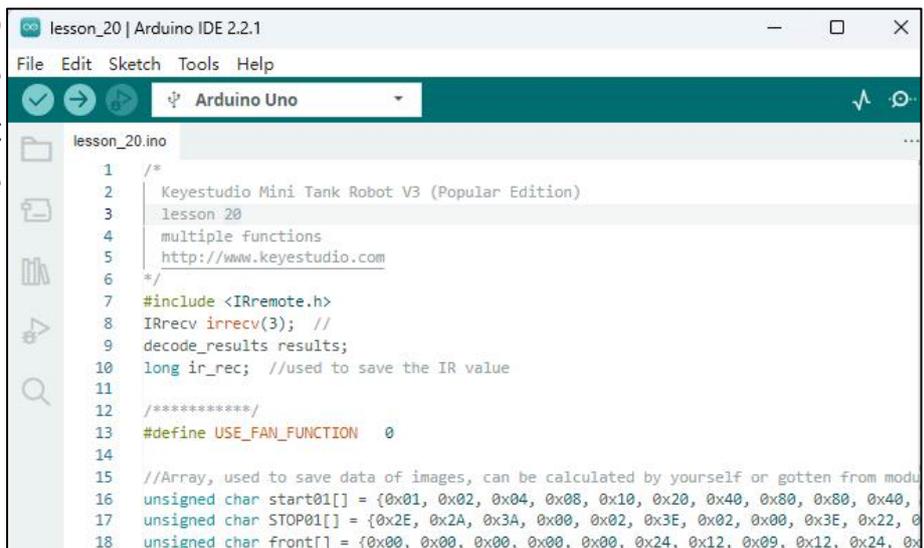
1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_20** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click upload >>>done uploading.

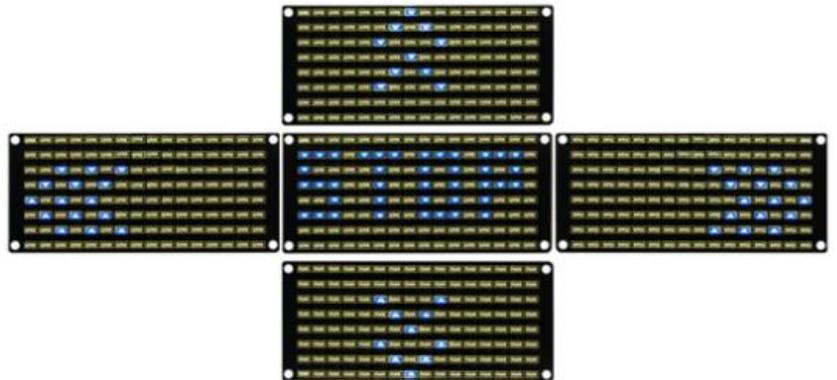
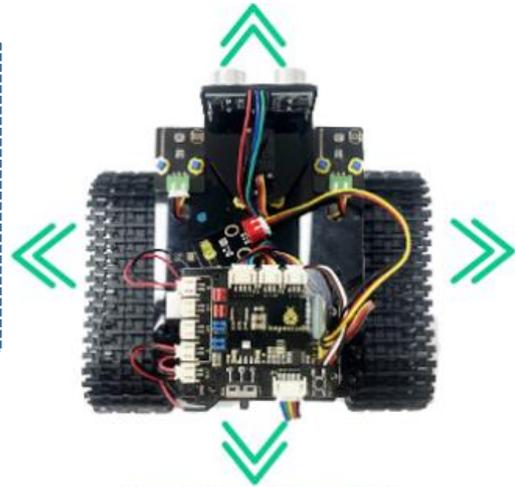


**Test Results:** After uploading the code, connect the robot to the Bluetooth module and pair the Bluetooth APP. Turn on the power switch of the motor drive shield. Place the robot on the floor, you can control the robot by IR Remote and the Bluetooth APP.



Button	Functions
	Move forward 8X16 LED dot matrix displays 
	Move back 8X16 LED dot matrix displays 
	Turn left 8X16 LED dot matrix displays 
	Turn right 8X16 LED dot matrix displays 
	Stop 8X16 LED dot matrix displays "STOP"

You will be able to use the up, down, left, right and stop buttons of the remote control to control the movement of the robot, and the 8X16 LED dot matrix displays will show its movement direction.



After uploading the code and successfully connecting to Bluetooth, you will be able to control the robot using the buttons shown in the picture.

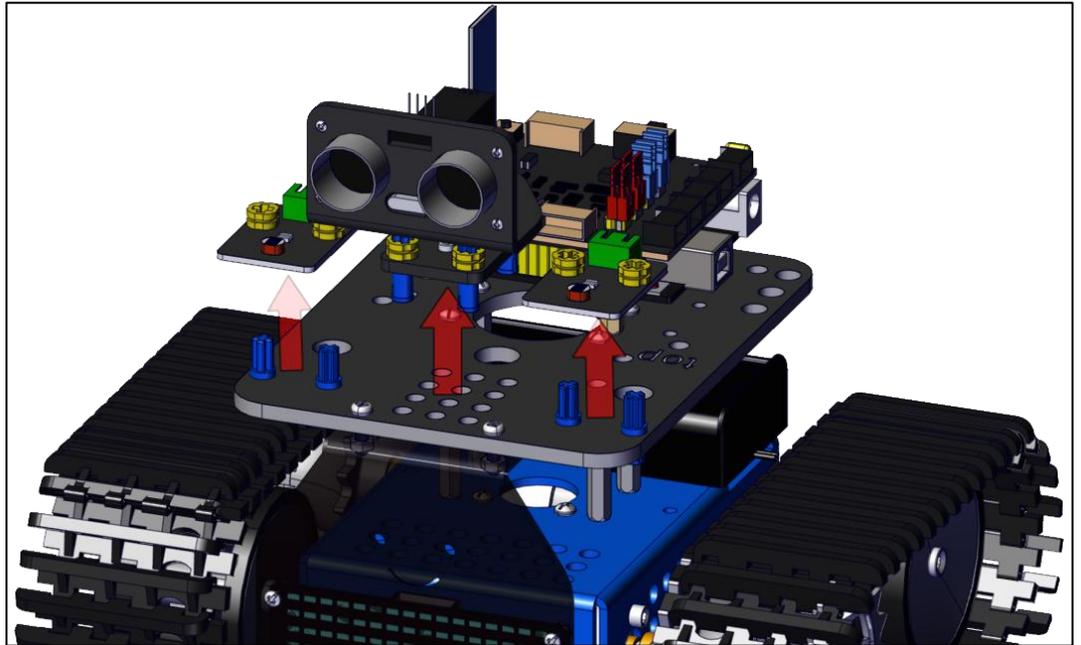


You can also click **Face button** to change the pattern displayed by 8X16 LED dot matrix

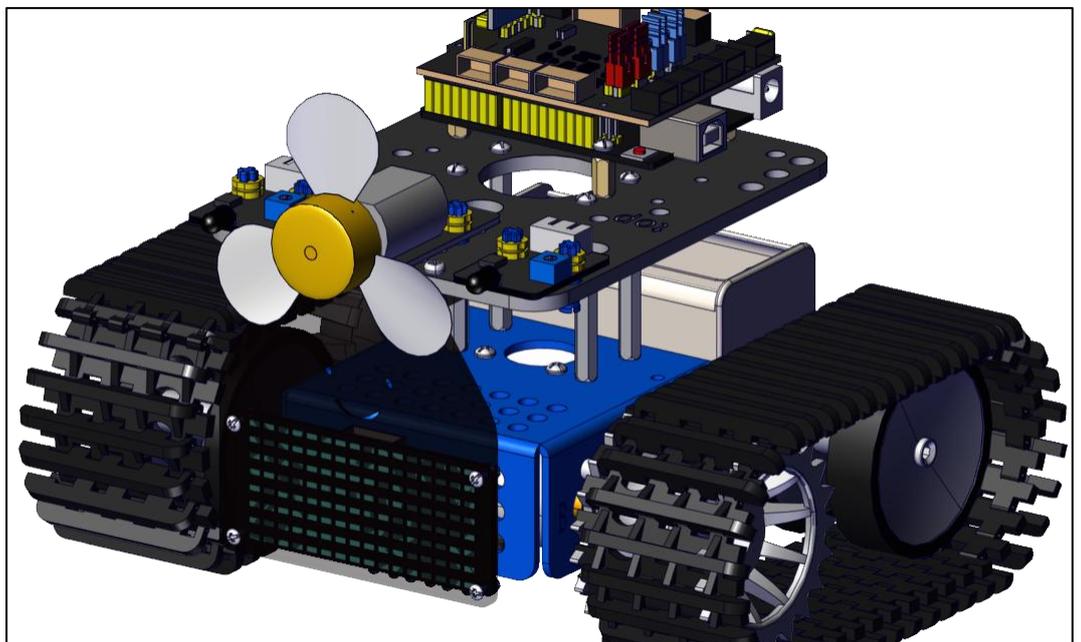
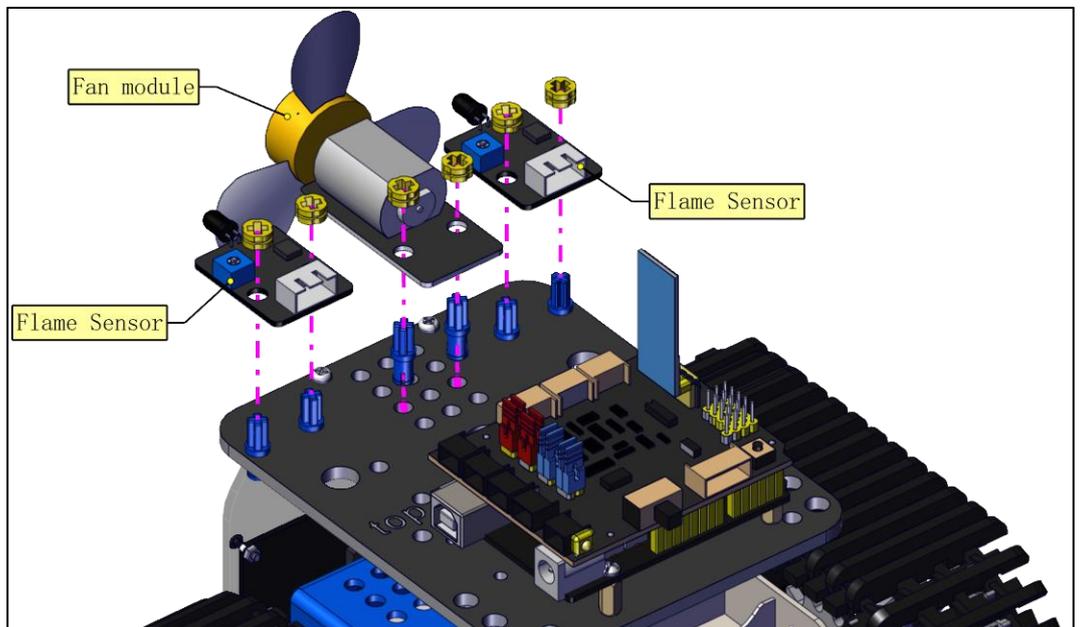


## Expand Experiments: Install flame sensors and a fan on the Robot

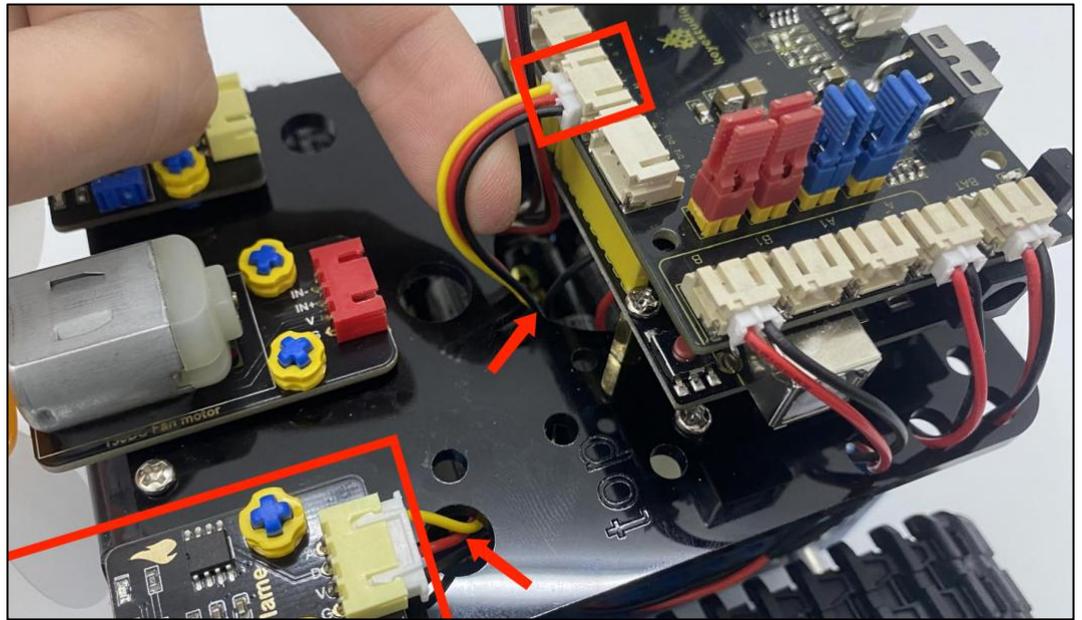
Remove the ultrasonic sensor and two photoresistors from the robot



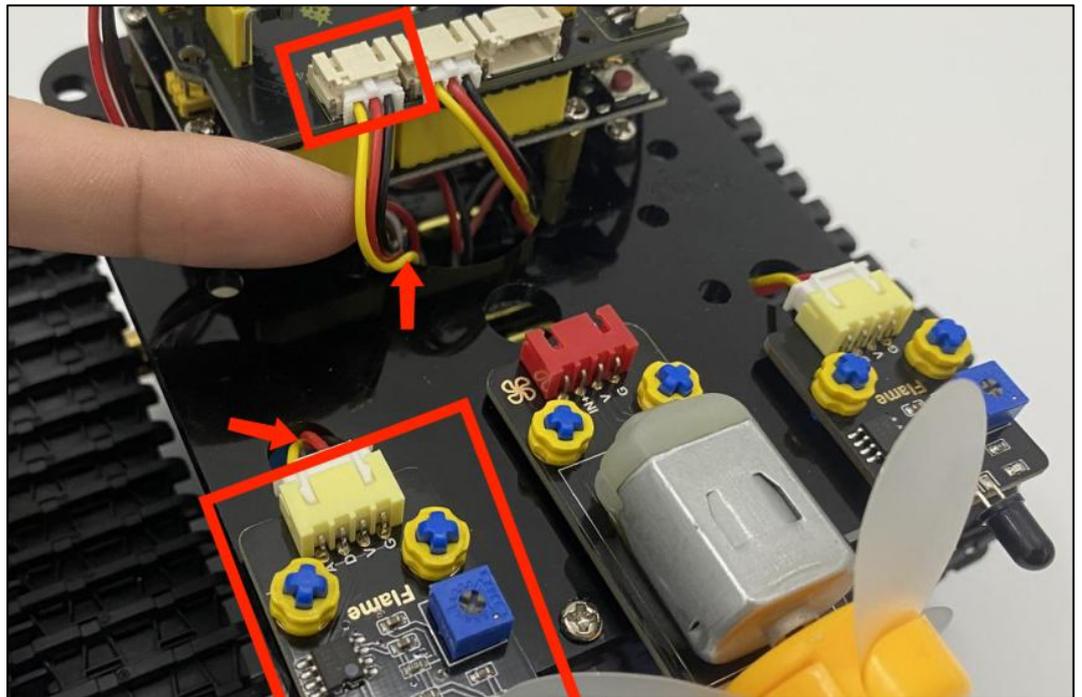
Install a fan module and two flame sensors on the robot



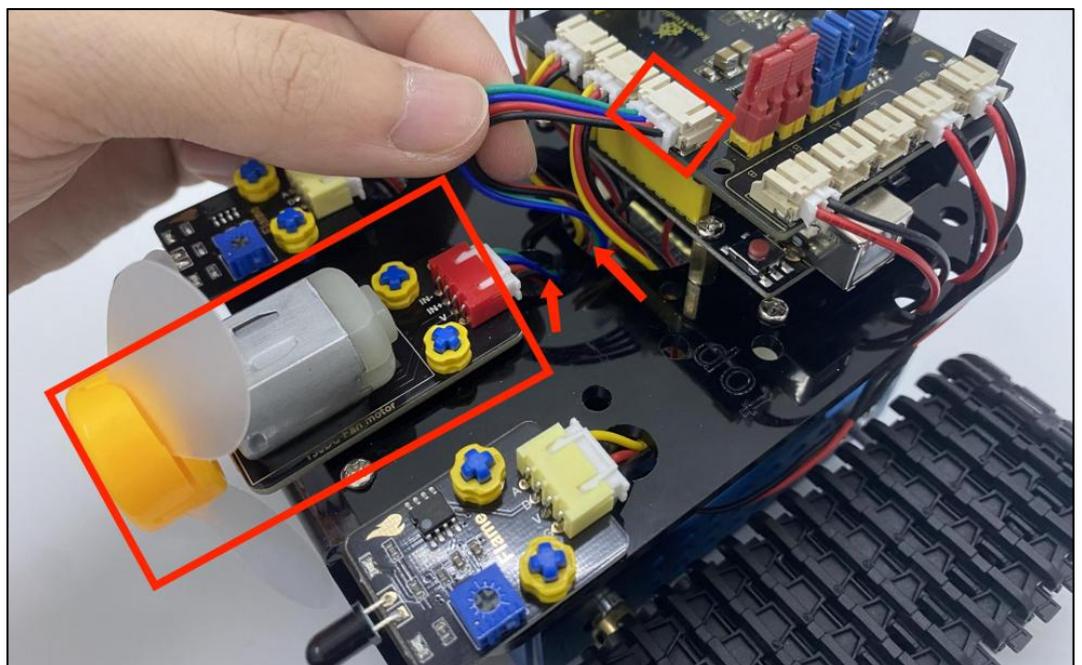
Connect the left flame sensor to the P3 pin of the motor drive shield with 4P-3P XH2.54 to PH2 Wire.



Connect the right flame sensor to the P4 pin of the motor drive shield with 4P-3P XH2.54 to PH2 Wire.



Connect the fan module to the P2 pin of the motor drive shield with 4P XH2.54 to PH2.0 Wire.



## Lesson 21: Read the Value of Flame Sensor

The flame sensor uses IR receiving tube to detect flames, converts the brightness of the flame into signals with high and low levels, input them into the central processor. The corresponding program processing. In both flames close to and without flames, the voltage value of the analog port is varied.

If there is no flame, the analog port is about 0.3V; when there is a flame, the analog port is 1.0V. The closer the flame is, the more the voltage value is. It can be used to detect the fire source or make a smart robot.

Note the probe of flame sensors only bears the temperature between  $-25^{\circ}\text{C}$  and  $\sim 85^{\circ}\text{C}$

In the process of use, pay attention to keep the flame sensor in certain distance to avoid getting damaged.

### Parameters:

Working voltage: 3.3V-5V (DC)

Current: 100mA

Maximum power: 0.5W

Work temperature:  $-10^{\circ}\text{C}$  to  $+50$  degrees Celsius

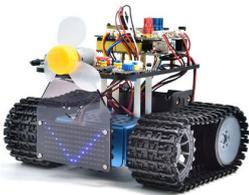
Sensor size: 31.6mmx23.7mm

Interface: 4pin turn 3PIN interface

Output signal: analog signals A0, A1



### You need to prepare:

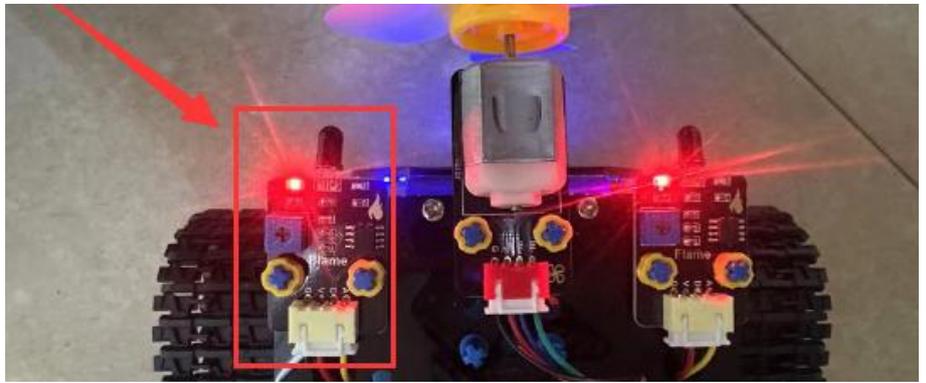
Robot tank without BT module	USB Cable*1	Computer*1	18650 Battery*2	Lighter*1
				

### Note:

1) This experiment requires the use of a fire source. Please make it away from flammable items to prevent fire. Children should experiment under adult supervision. **If you cannot confirm that you are safe, please abandon the experiment.**

2) The flame sensor is not fireproof, please do not burn it directly with flame.

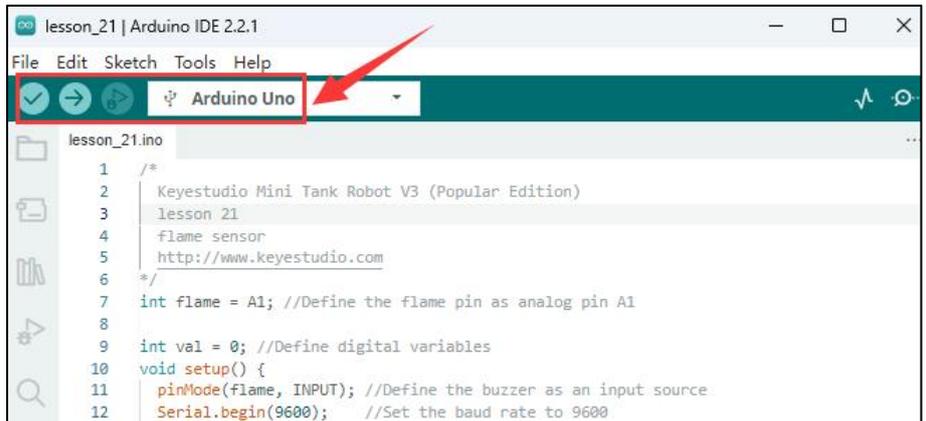
In this lesson, we will test the flame sensor on the left side of the robot and read its analog value.



1. Connect the V4.0 board to the computer with the usb cable. Open the INO file inside the **lesson\_21** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.



**Don't connect to the Bluetooth module when uploading code**

3. Click upload >>>done uploading.

**Note:**

1) Please make it away from flammable items to prevent fire. Children should experiment under adult supervision. **If you cannot confirm that you are safe, please abandon the experiment.**

2) The flame sensor is not fireproof, please do not burn it directly with flame.



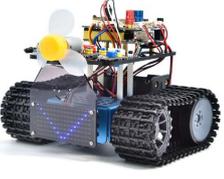
Make the flame slowly approach the flame sensor, and it cannot directly contact the flame sensor to prevent it from burning out.

**Test Results:** You can use the flame of a lighter near the left flame sensor. Open the serial monitor and set the baud rate to 9600. The closer the flame is, the smaller the value read.



## Lesson 22: Flame Warning

You need to prepare:

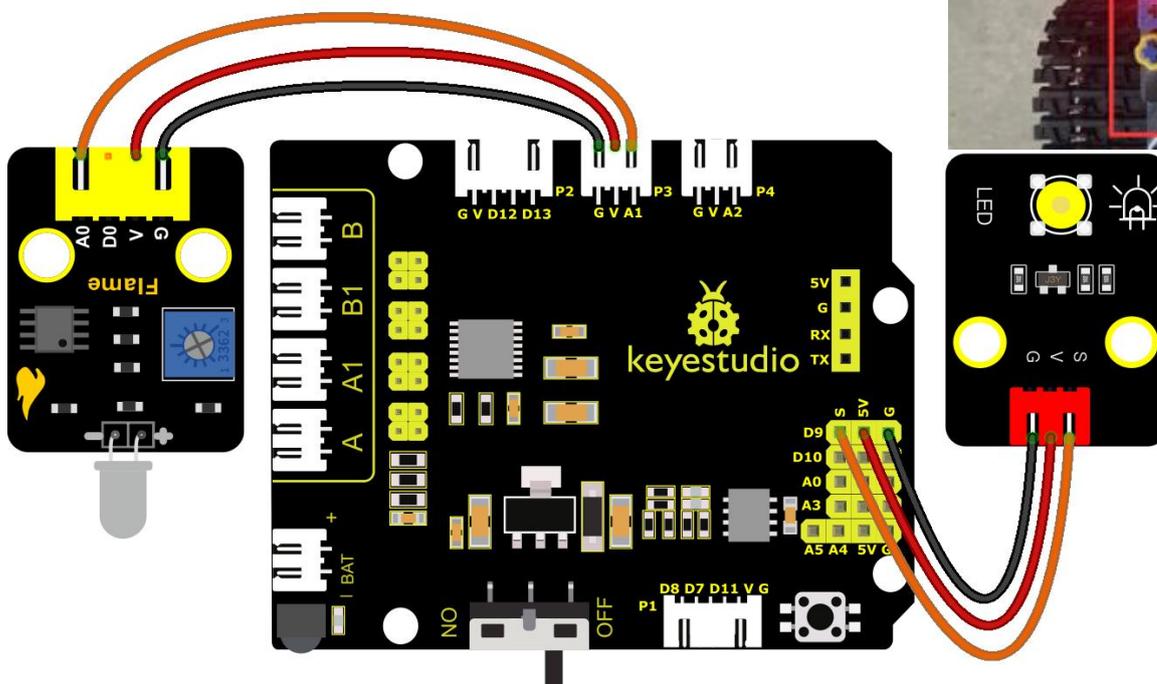
Robot without BT module	USB Cable*1	Computer*1	18650 Battery*2	Lighter*1
				
Yellow LED Module*1	3P-3P XH2.54 to 2.54 Wire			
				

Note:

1) This experiment requires the use of a fire source. Please make it away from flammable items to prevent fire. Children should experiment under adult supervision. **If you cannot confirm that you are safe, please abandon the experiment.**

2) The flame sensor is not fireproof, please do not burn it directly with flame.

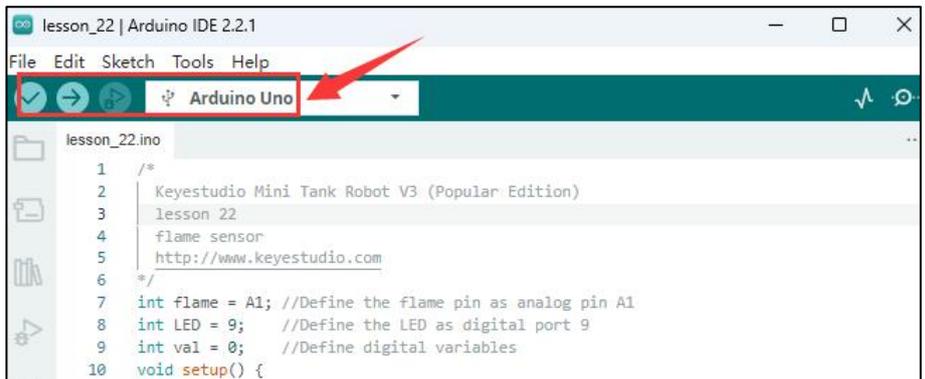
Connect the LED module to pin 9 of the motor drive shield, when the **left flame sensor** detects a flame, the LED module will light up as an alarm.



1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_22** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.



3. Click upload >>>done uploading.

**Test Results:** You can use the flame of a lighter near the left flame sensor. When the flame sensor detects a flame, the LED module will light up as an alarm.

**Note:**

3) Please make it away from flammable items to prevent fire. Children should experiment under adult supervision. **If you cannot confirm that you are safe, please abandon the experiment.**

4) The flame sensor is not fireproof, please do not burn it directly with flame.



## Lesson 23: Get the Fan Rotating

This fan module uses a HR1124S motor-controlling chip, a single-channel H-bridge driver chip containing a low-conductivity resistance PMOS and NMOS power tubes. The low-conducting resistance can ease the power consumption, contributing to the safe work of the chip for longer time.

In addition, its low standby current and low static working current makes itself apply to toys. We can control the rotation direction and speed of the fan by outputting IN + and IN- signals and PWM signals.



### Specification:

Working voltage: 5V

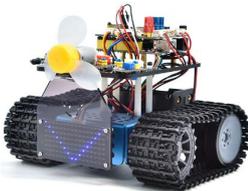
Current: 200MA

Maximum power: 2W

Operating temperature: -10 degrees Celsius to +50 degrees Celsius

Size: 47.6MM \*23.8MM

### You need to prepare:

Robot tank without BT module	USB Cable*1	Computer*1	18650 Battery*2
			

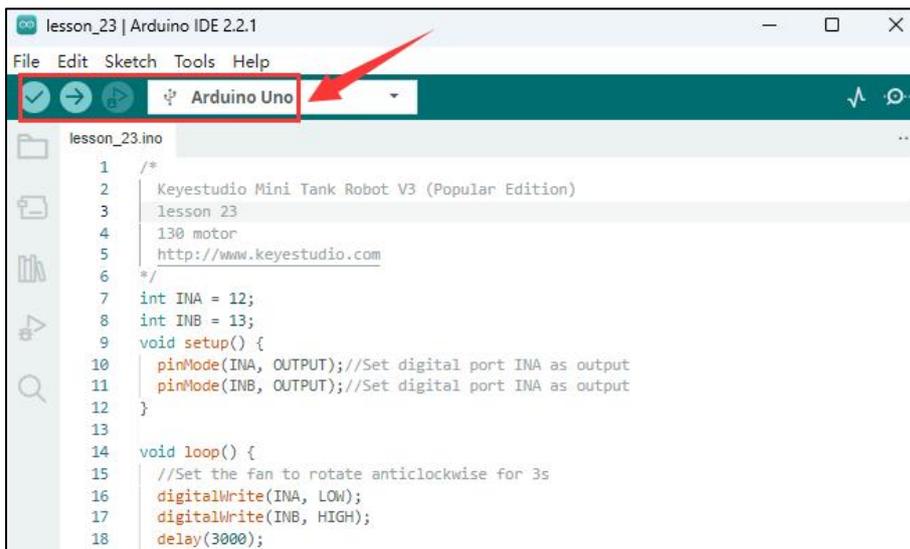
1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_23** folder with Arduino IDE.



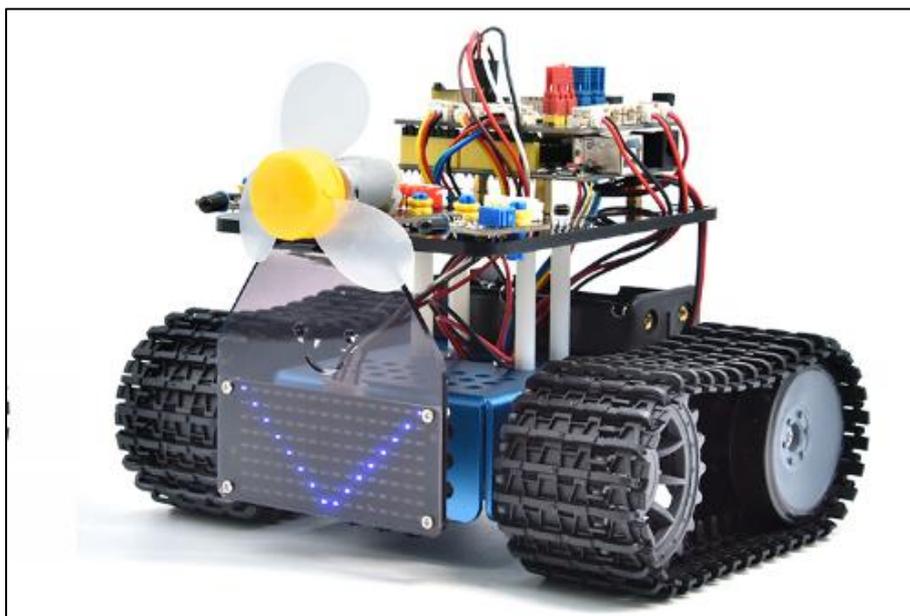
2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

**Don't connect to the Bluetooth module when uploading code**

3. Click upload >>>done uploading.

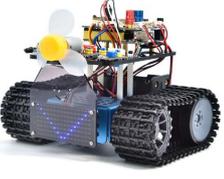


**Test Results:** After uploading the code, turn on the power switch of the motor drive shield, you will see the small fan turn anticlockwise for 3000ms, stop for 1000ms, and clockwise for 300ms.



## Lesson 24: Turn on the fan when Robot detect a flame.

You need to prepare:

Robot without BT module	USB Cable*1	Computer*1	18650 Battery*2	Lighter*1
				

Note:

3) This experiment requires the use of a fire source. Please make it away from flammable items to prevent fire. Children should experiment under adult supervision. **If you cannot confirm that you are safe, please abandon the experiment.**

4) The flame sensor is not fireproof, please do not burn it directly with flame.

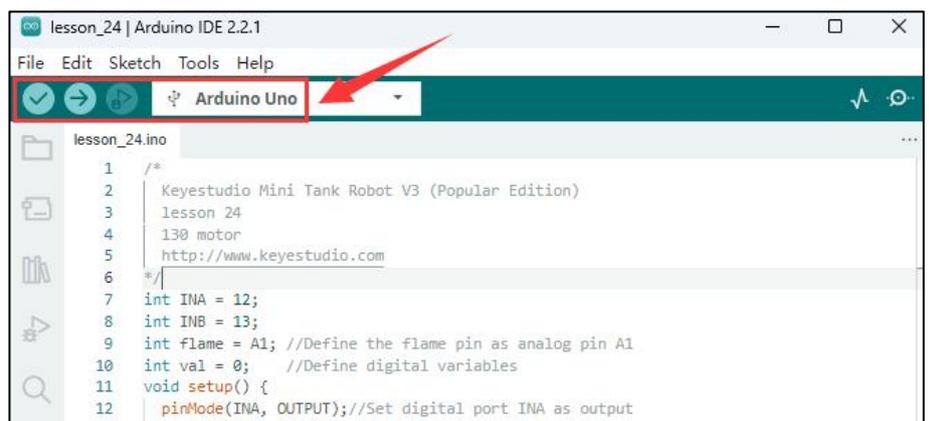
1. Connect the V4.0 board to the computer with the usb cable.

Open the INO file inside the **lesson\_24** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.

3. Click upload >>>done uploading.



**Test Results:** After uploading the code, turn on the power switch of the motor drive shield, you can turn on the fan when flame is detected from the **left flame sensor** of the robot.

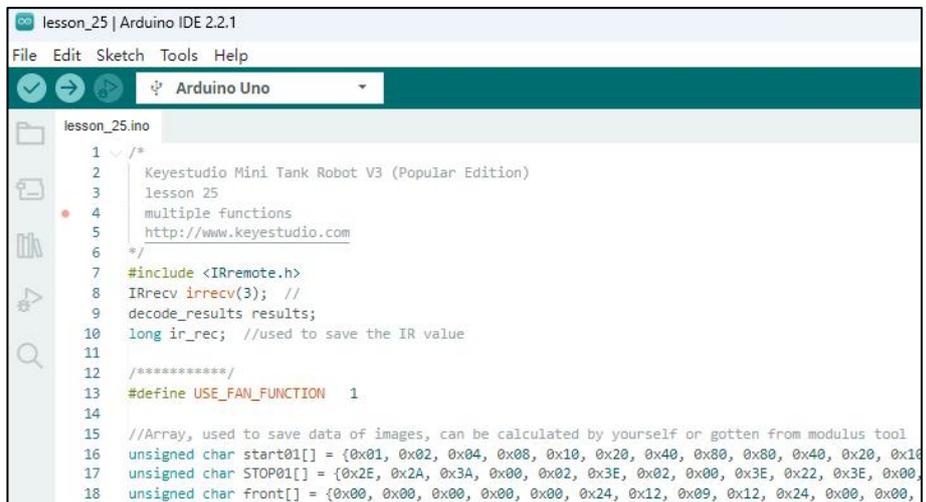




1. Connect the V4.0 board to the computer with the usb cable.  
Open the INO file inside the **lesson\_25** folder with Arduino IDE.



2. Click on Tools, select "Arduino UNO" for the board type in the drop-down menu bar, and select COM-XX for Port as shown in the Device Manager.



**Don't connect to the Bluetooth module when uploading code**

3. Click upload >>>done uploading.

**Test Results:** After uploading the code, connect the robot to the Bluetooth module and pair the Bluetooth APP. Turn on the power switch of the motor drive shield. Place the robot on the floor, you can control the robot via the Bluetooth APP using the buttons shown on the right.



: This button can control the fan on and off.

You can also use these five buttons on the remote control to control the movement of the robot.



Button	Functions
	Move forward 8X16 LED dot matrix displays
	Move back 8X16 LED dot matrix displays
	Turn left 8X16 LED dot matrix displays
	Turn right 8X16 LED dot matrix displays
	Stop 8X16 LED dot matrix displays "STOP"