

LAFVIN

LAFVIN

4WD Smart Robot Car

LAFVIN



Packing List



with Cable



V5 Expansion Board 1PCS



IR Receiver Module 1PCS



Line Tracking Module 3PCS



Ultrasonic Sensor 1PCS



Bluetooth Module 1PCS



Black Tape 1PCS



Acrylic Chassis 2PCS



Servo Motor(SG90) 1PCS



Cell Box 1PCS



F-F Dupont Wire 1PCS



Remote Control 1PCS



Ultrasonic Holder 1PCS



Tire 4PCS



DC Motor 4PCS



L298N Motor Board



5 Bags Screw Kit



Screwdriver 1PCS



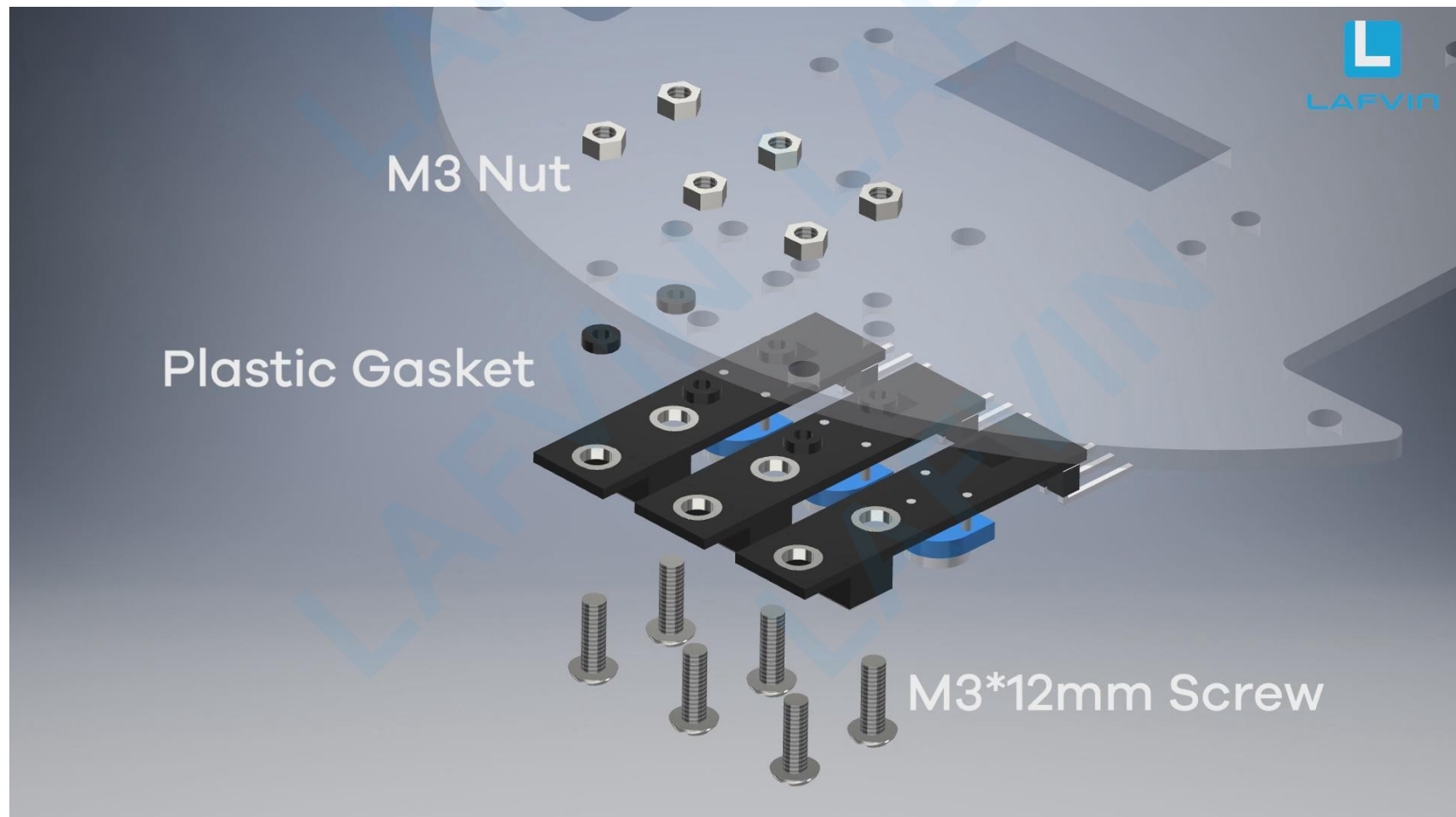
Bundling Belt 4PCS

Content

◆ Step 1:Assembly Tutorial	1
◆ Step 2:Install Arduino IDE	33
◆ Step 3:Install Arduino UNO Driver	34
◆ Step 4:Upload Arduino UNO Main Code	41
◆ Step 5:Infrared Remote Control Robot Car	49
◆ Step 6:APP connects to the Bluetooth module	54
◆ Step 7:APP Remote Control Multi-function Mode	61
Line Tracking Sensor	66
L298N Motor Driver	70
Ultrasonic Sensor Module	75

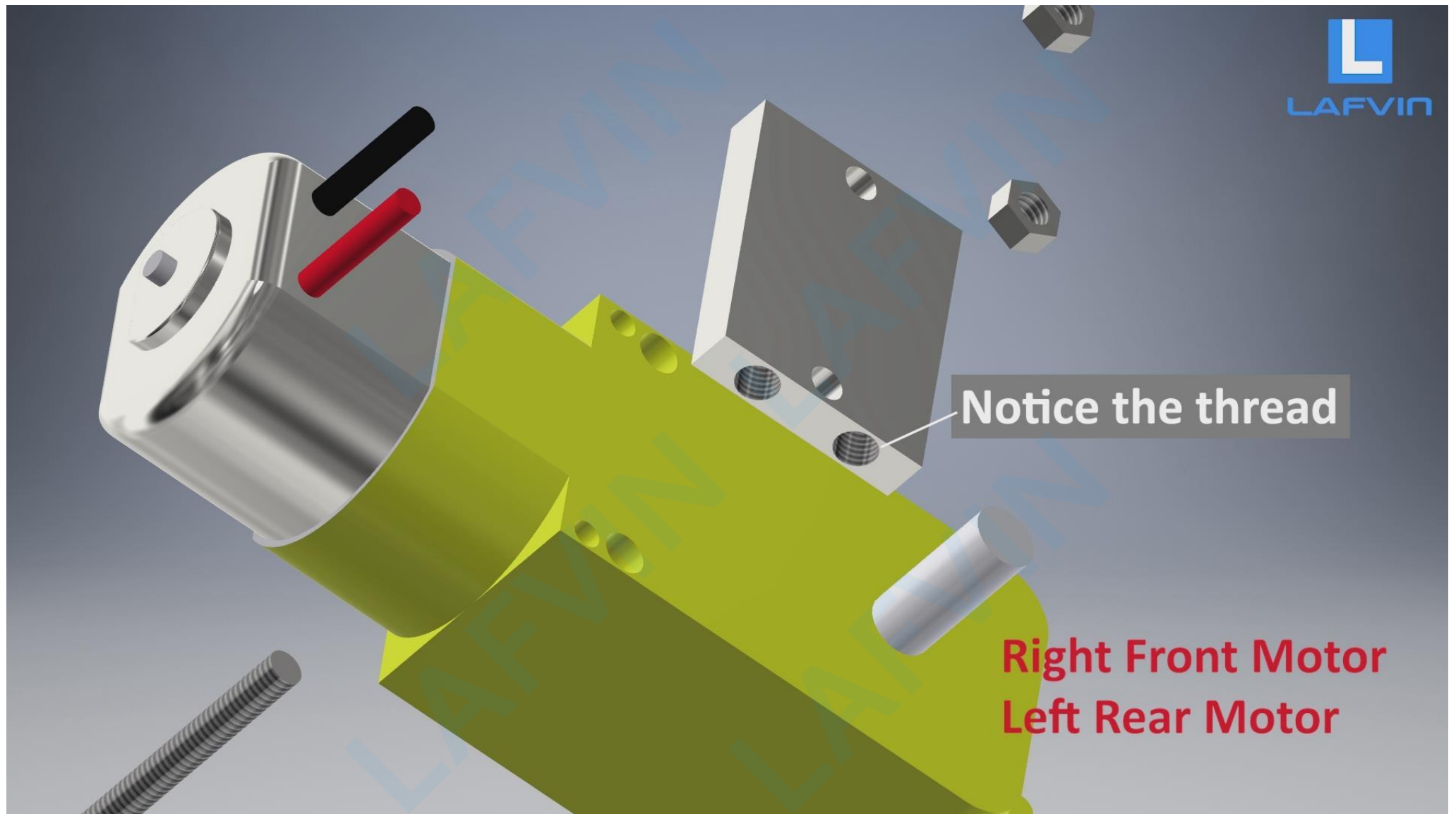
◆ Step 1: Assembly Tutorial

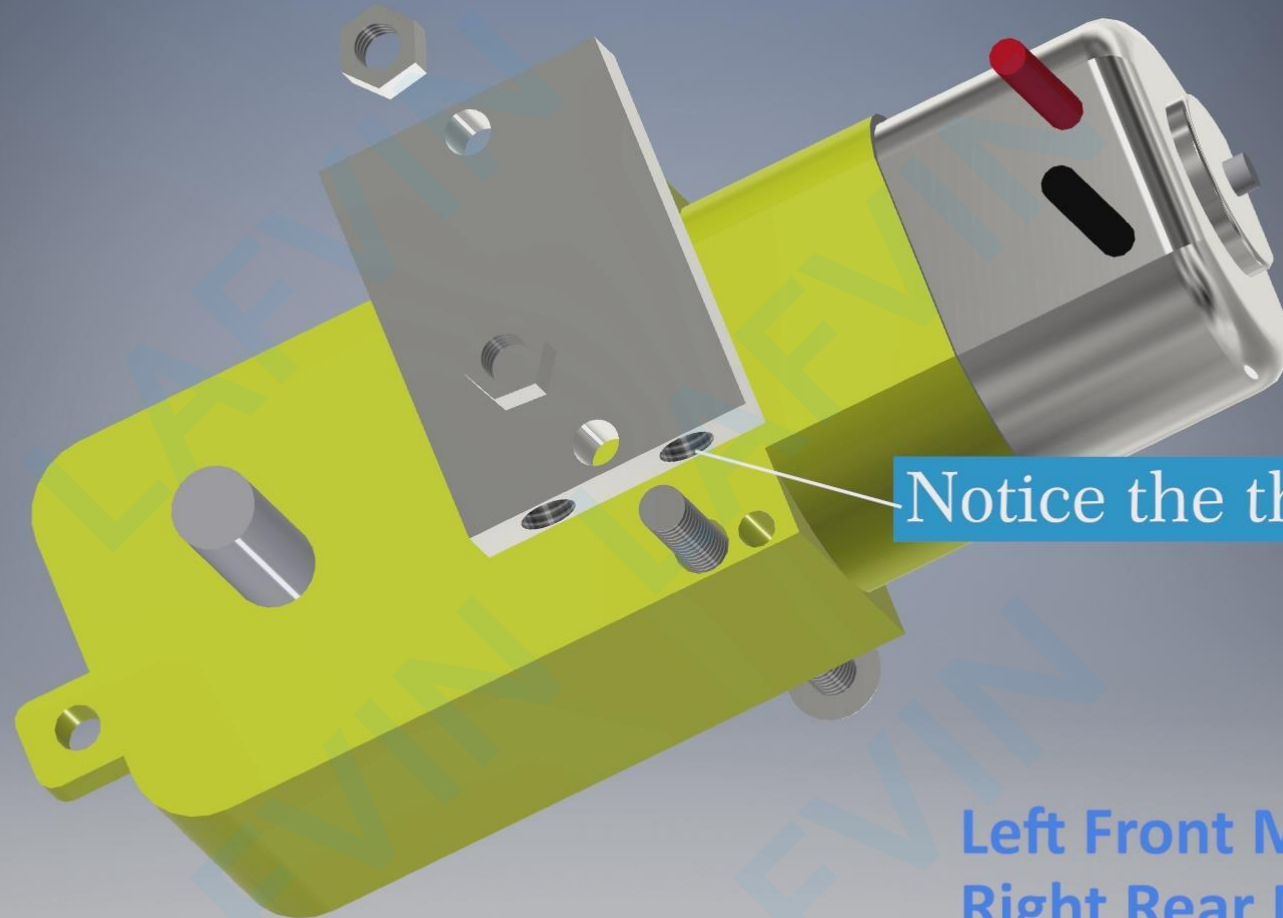
For more details refer to video



M3*40mm Copper Cylinder

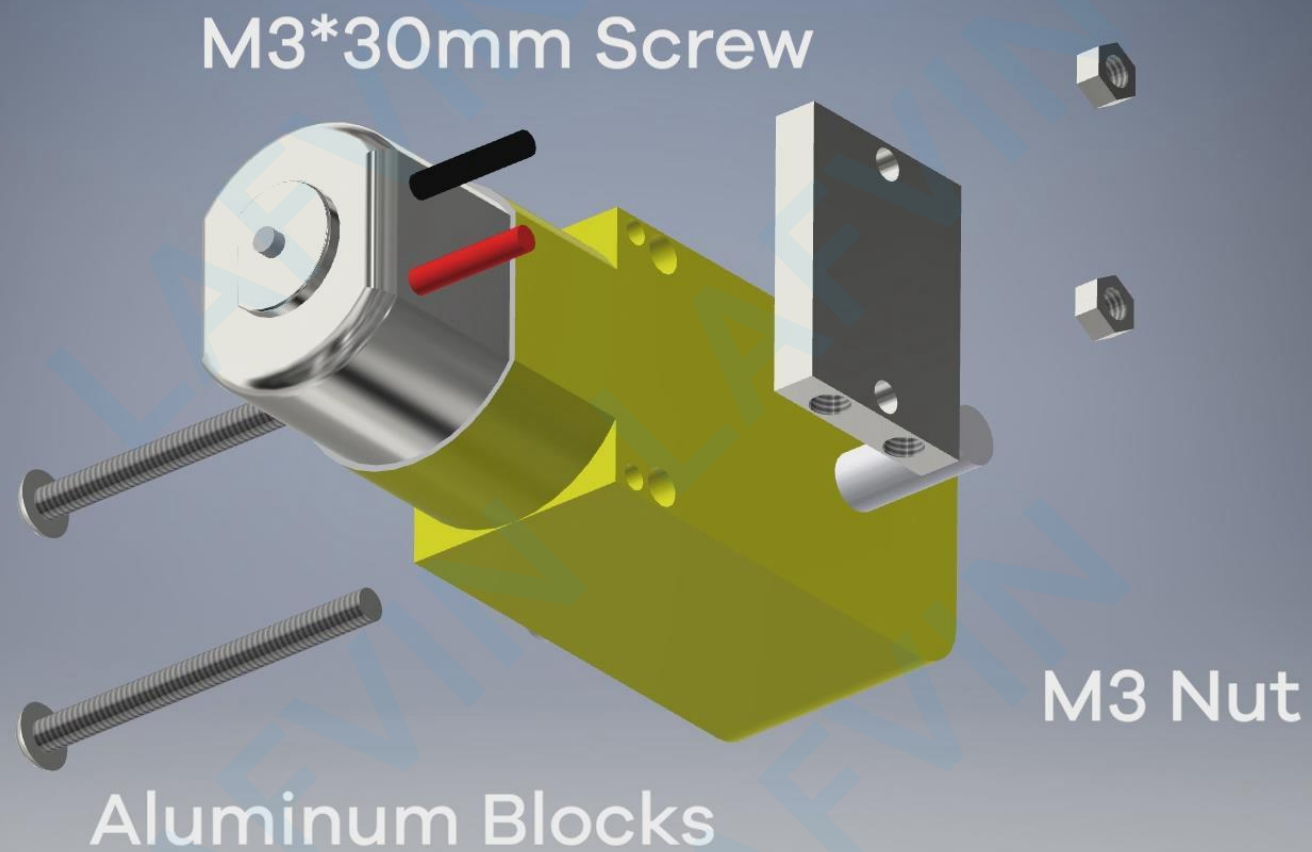
M3*10mm Screw

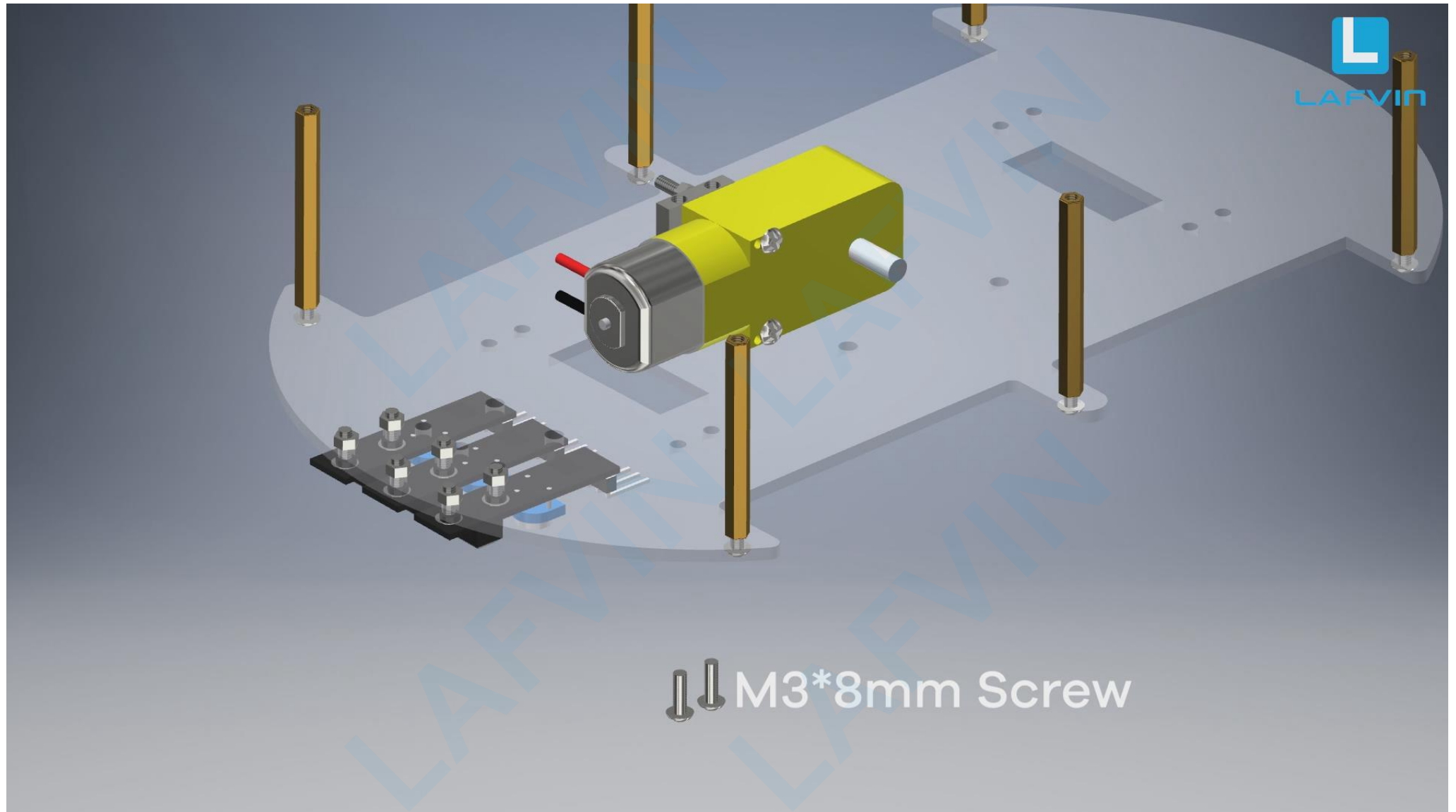


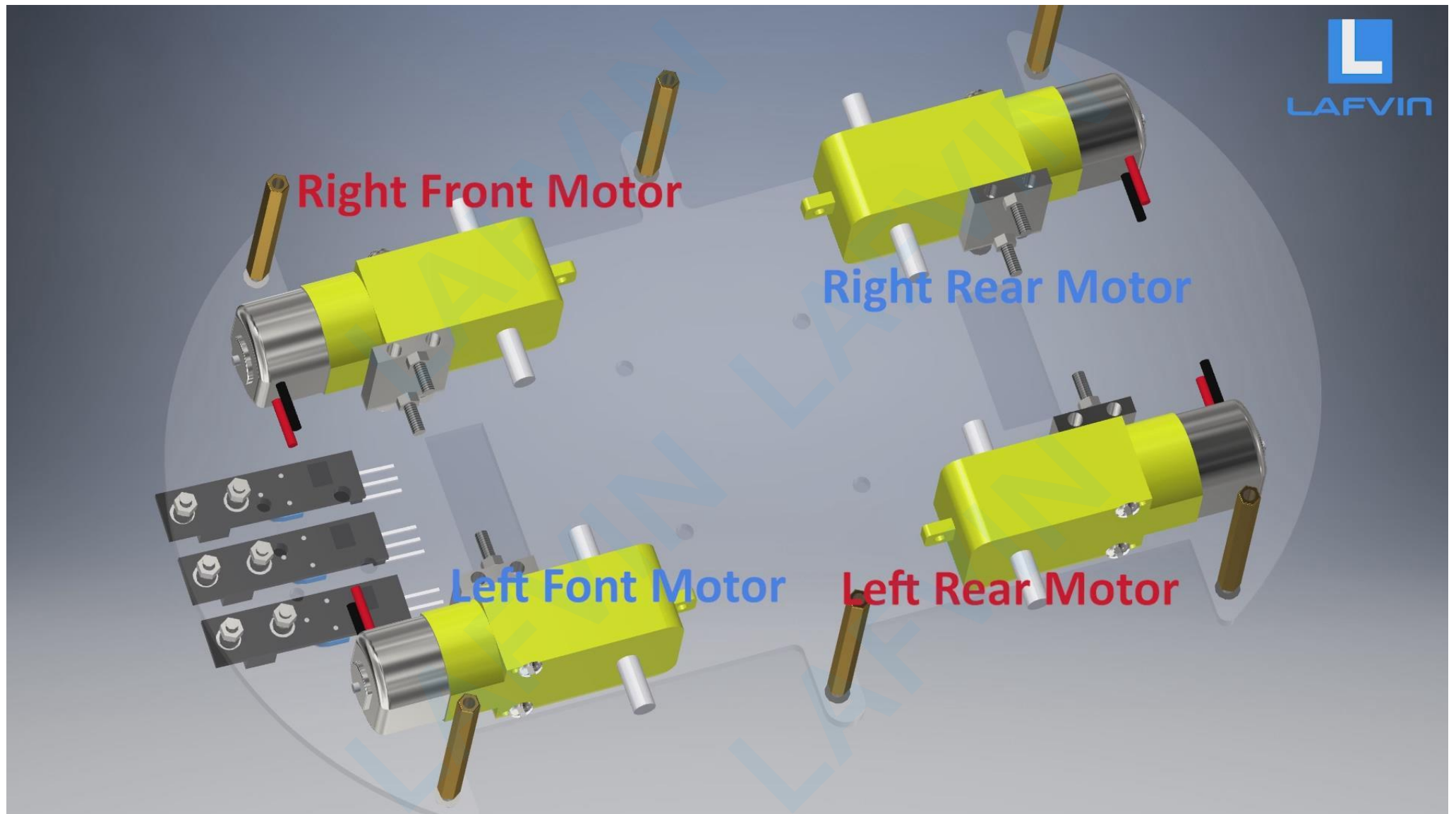


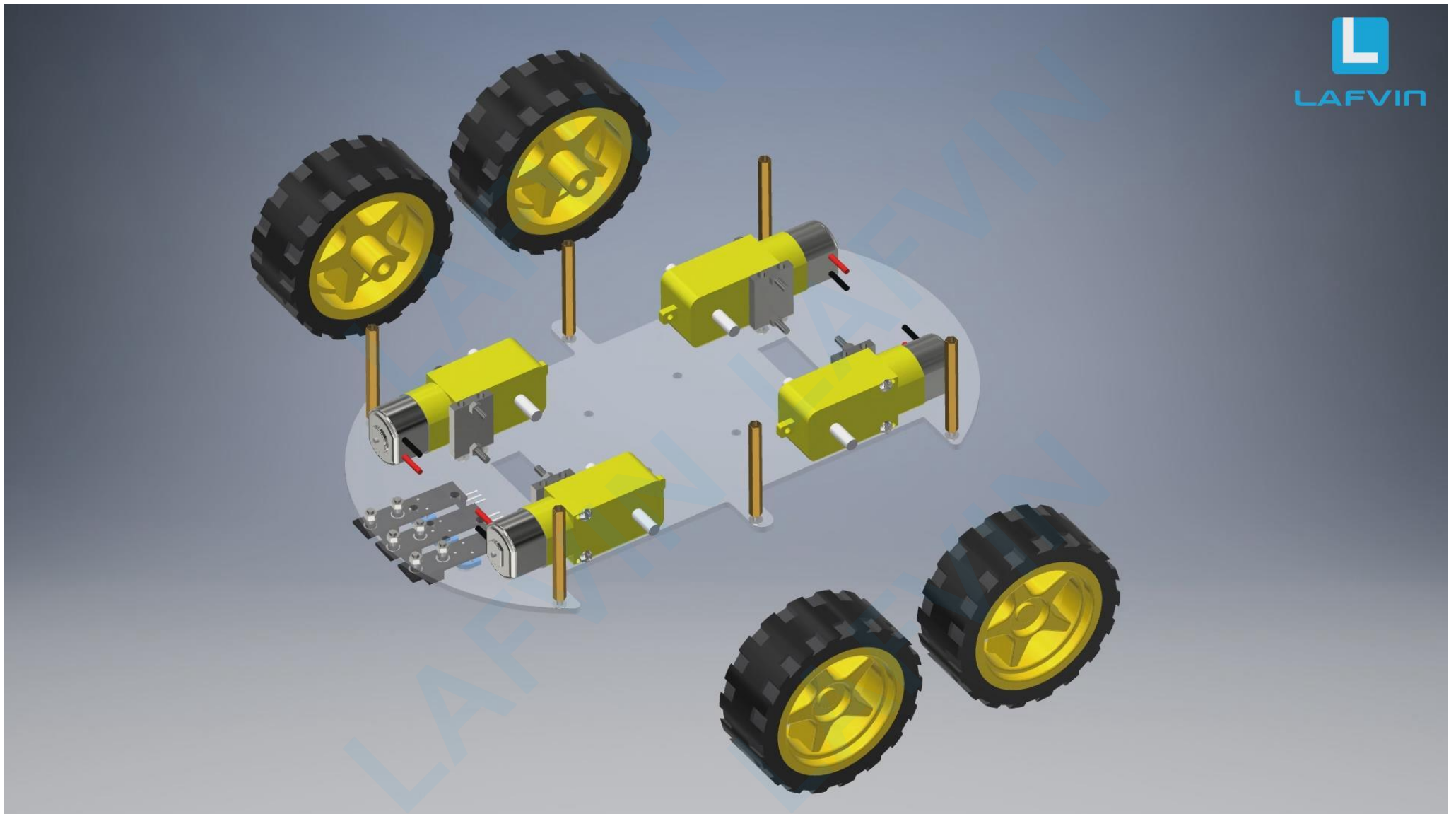
Notice the thread

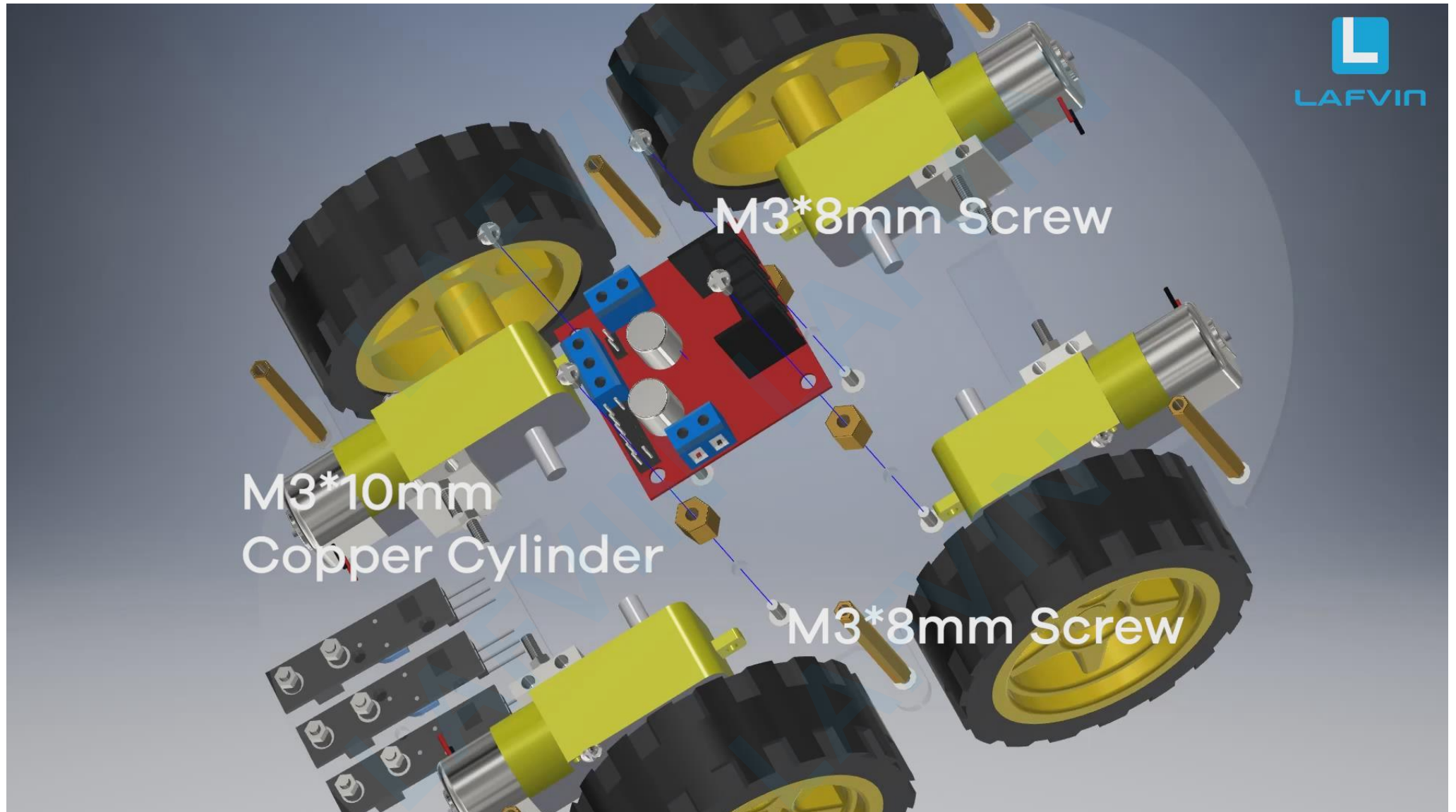
Left Front Motor
Right Rear Motor

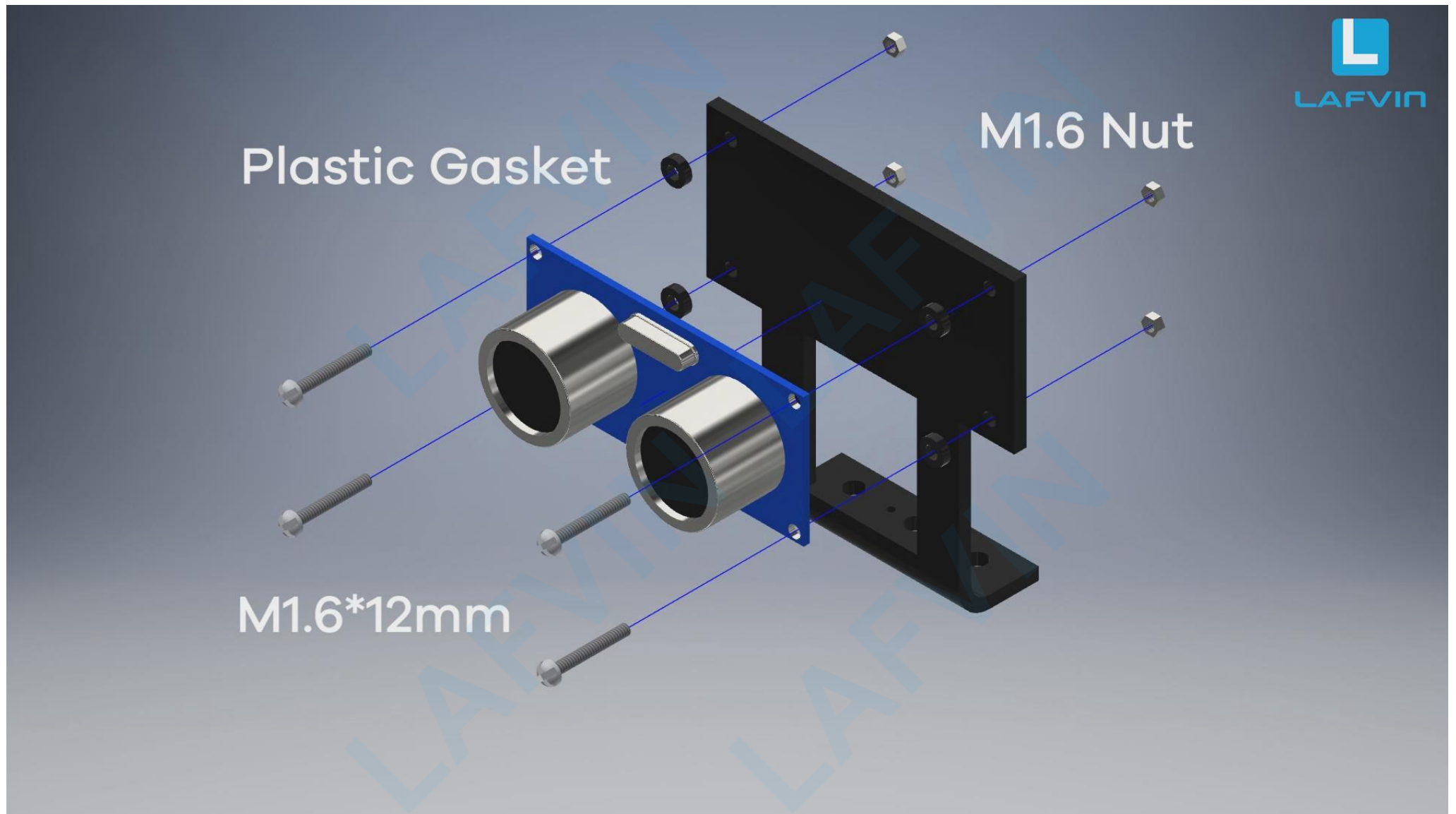




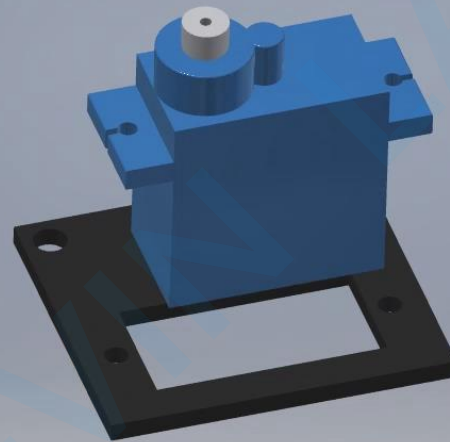








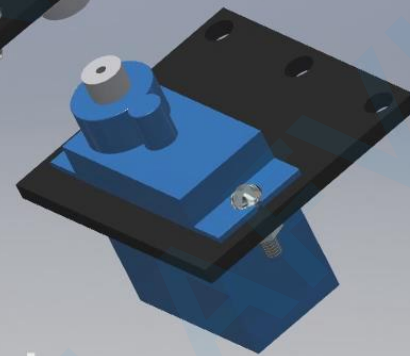
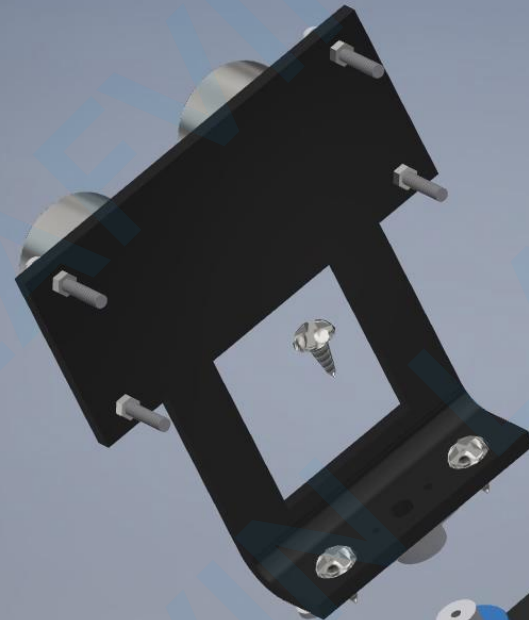
M2*12mm



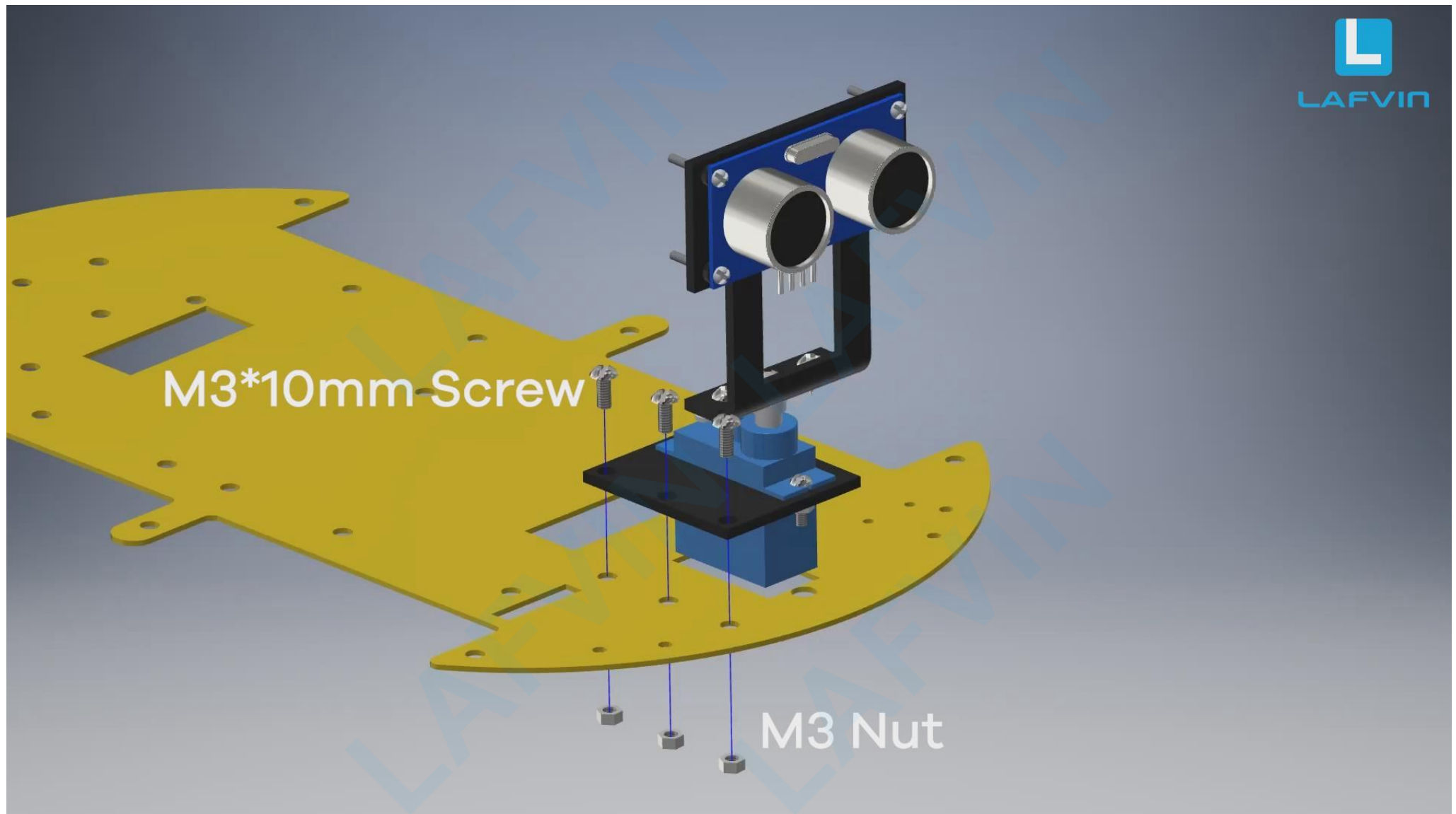
M2 Nut

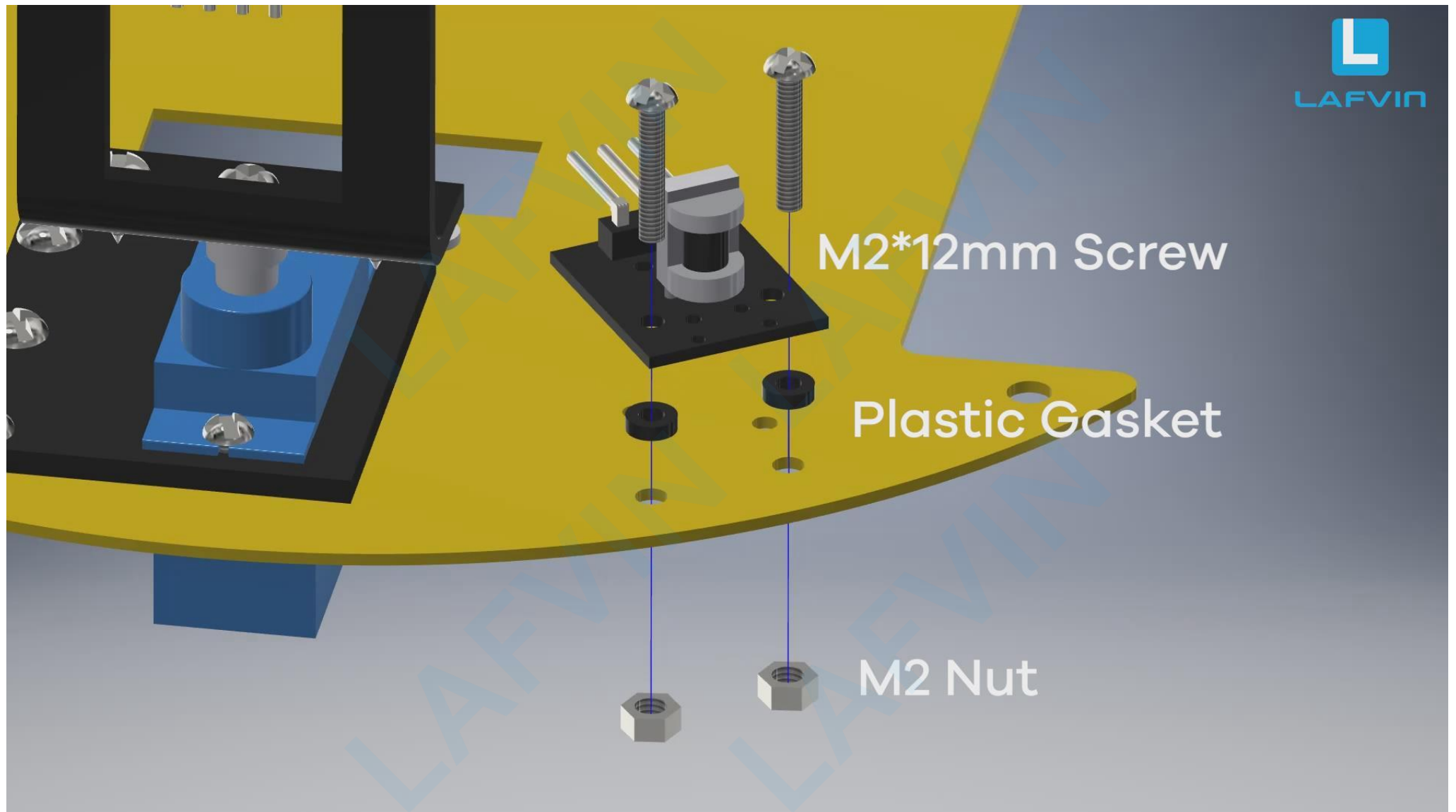
Get Screw from Servo Package

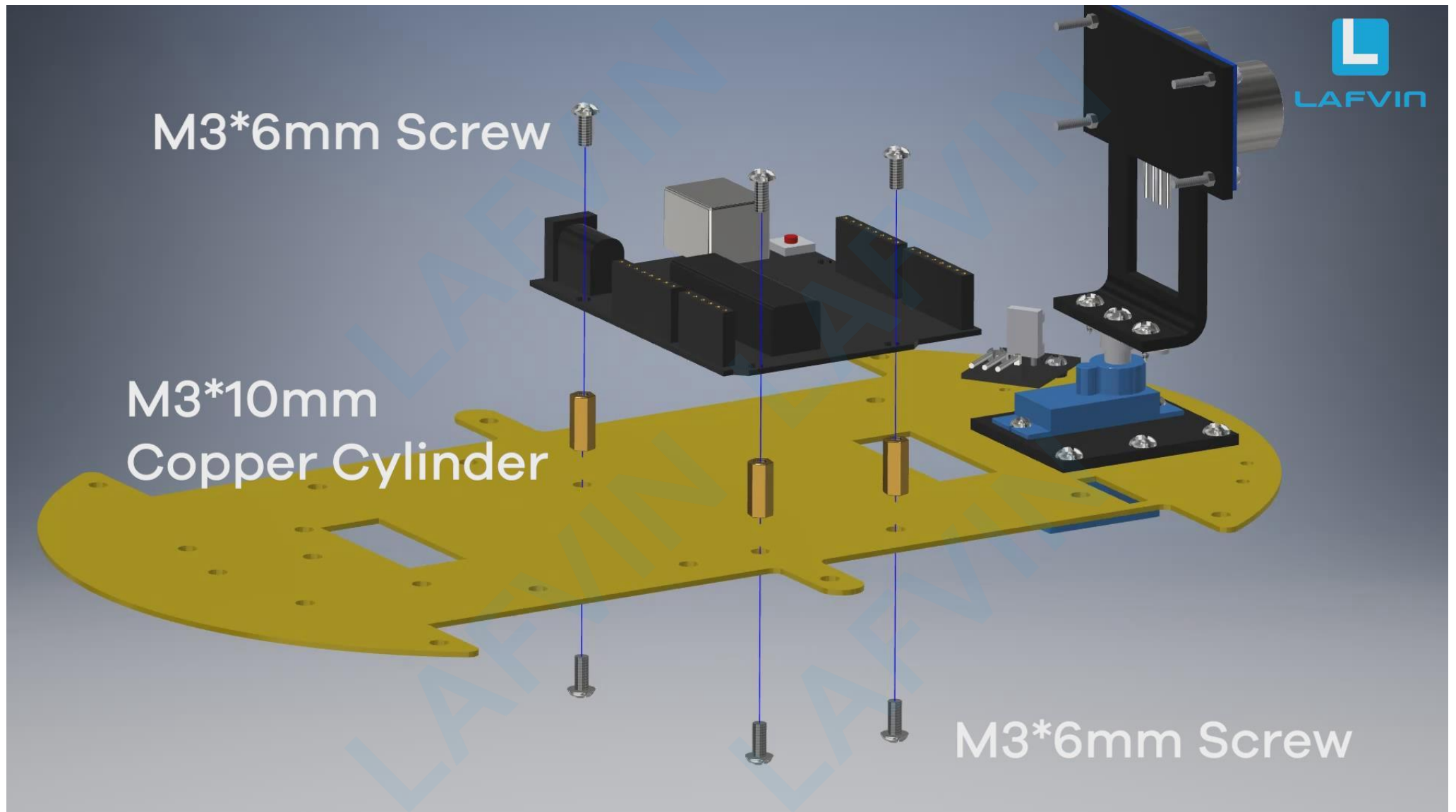


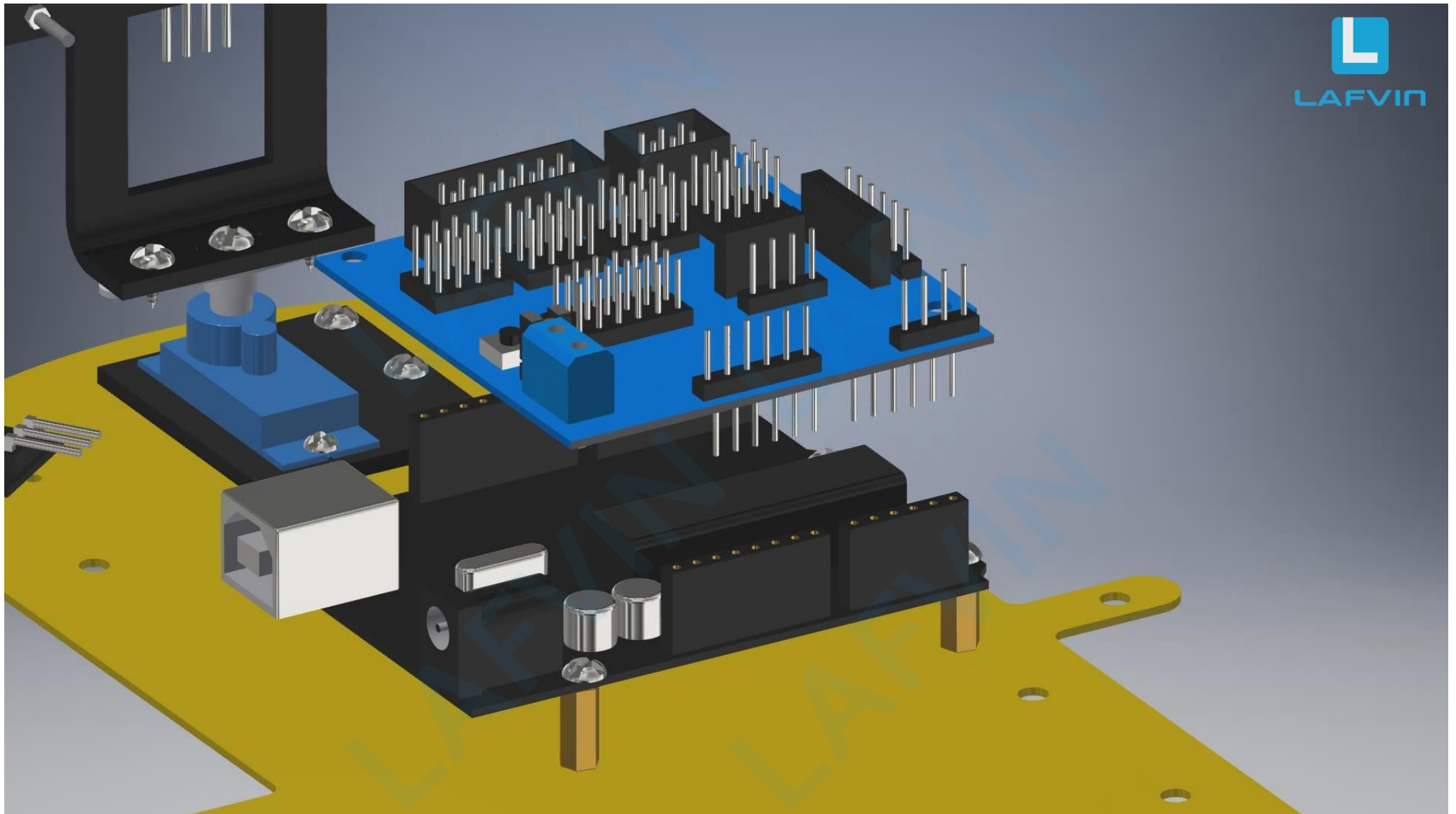


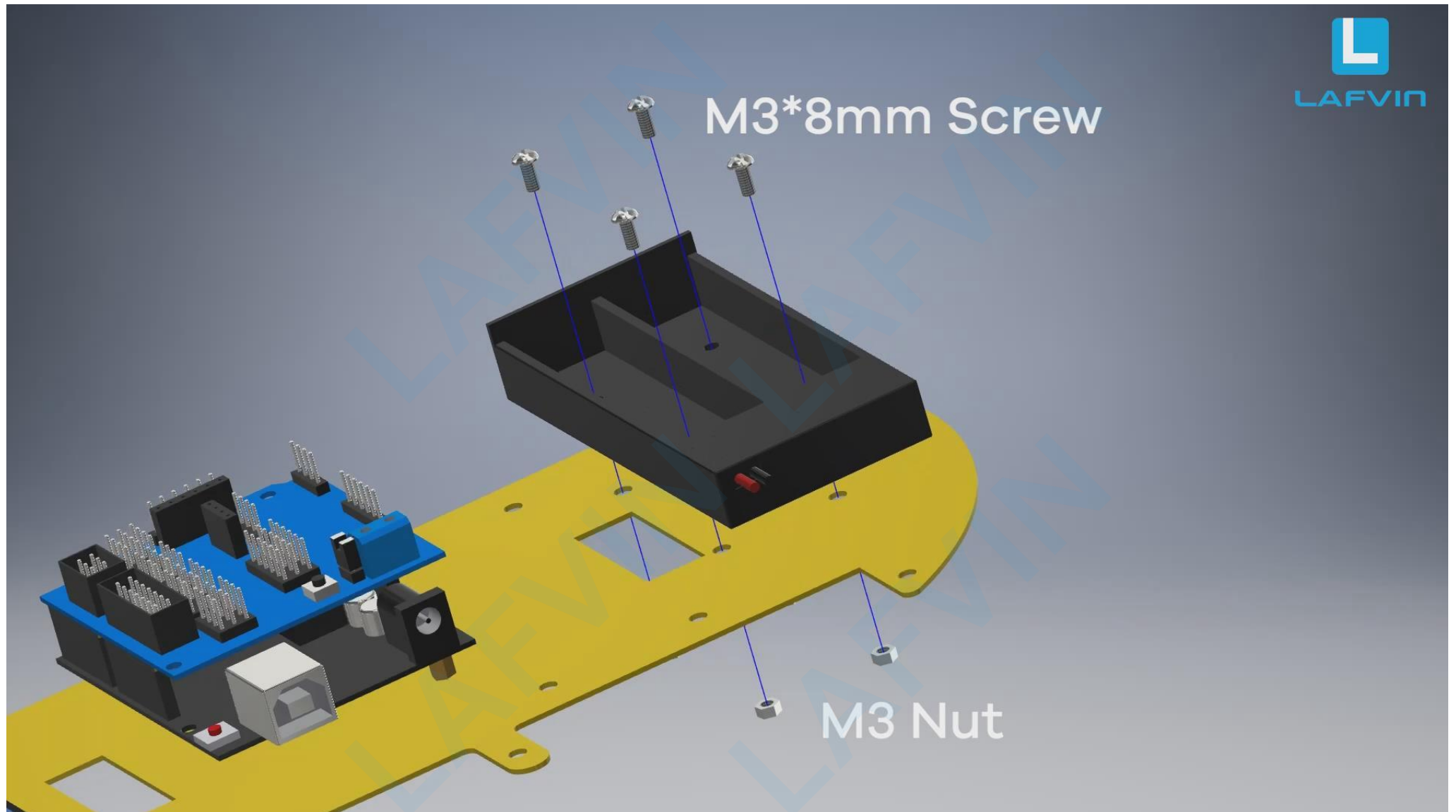
Get Screw from Servo Package

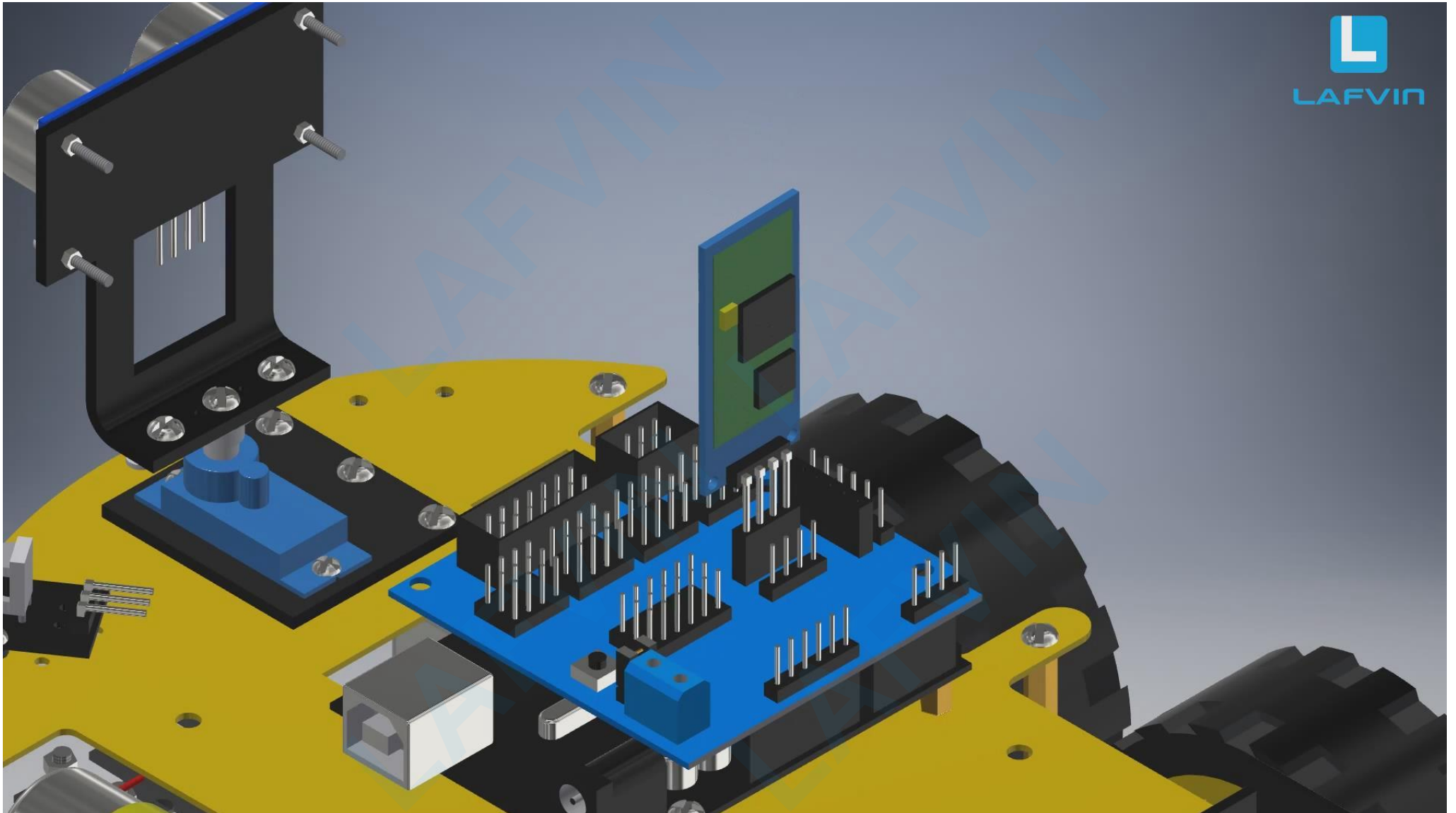




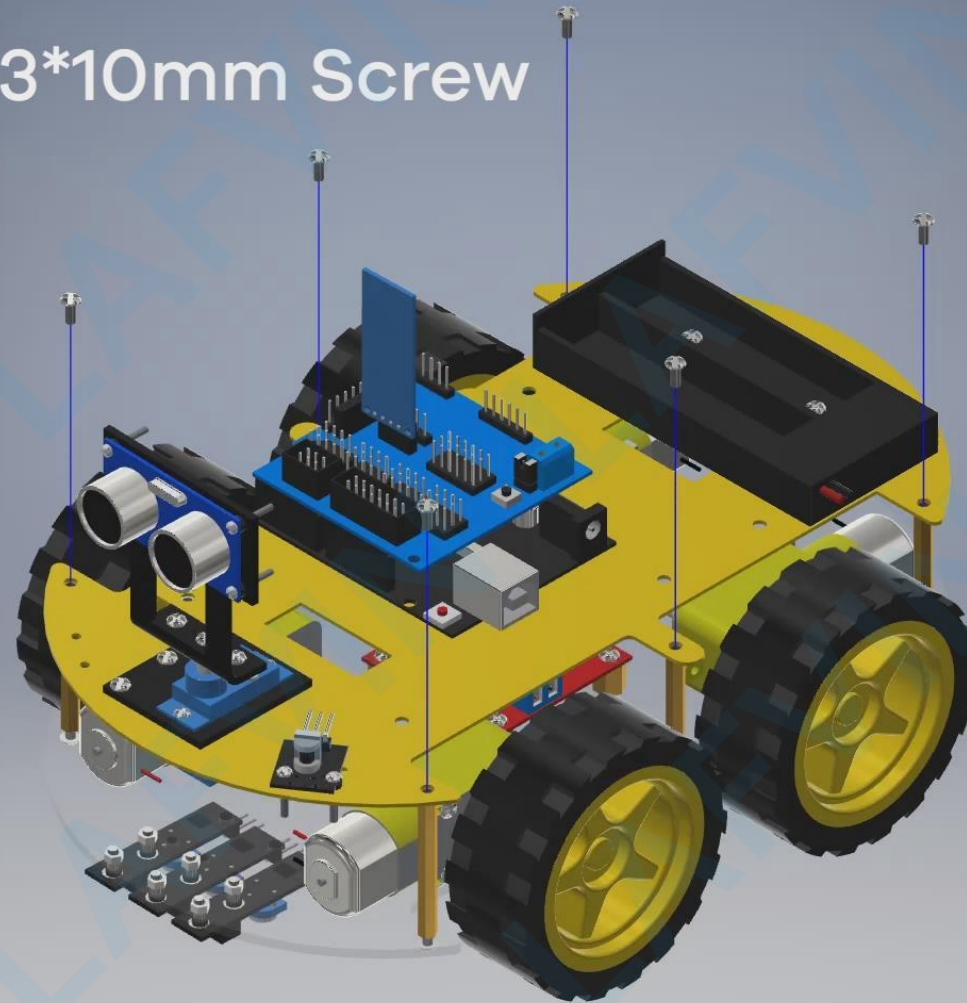


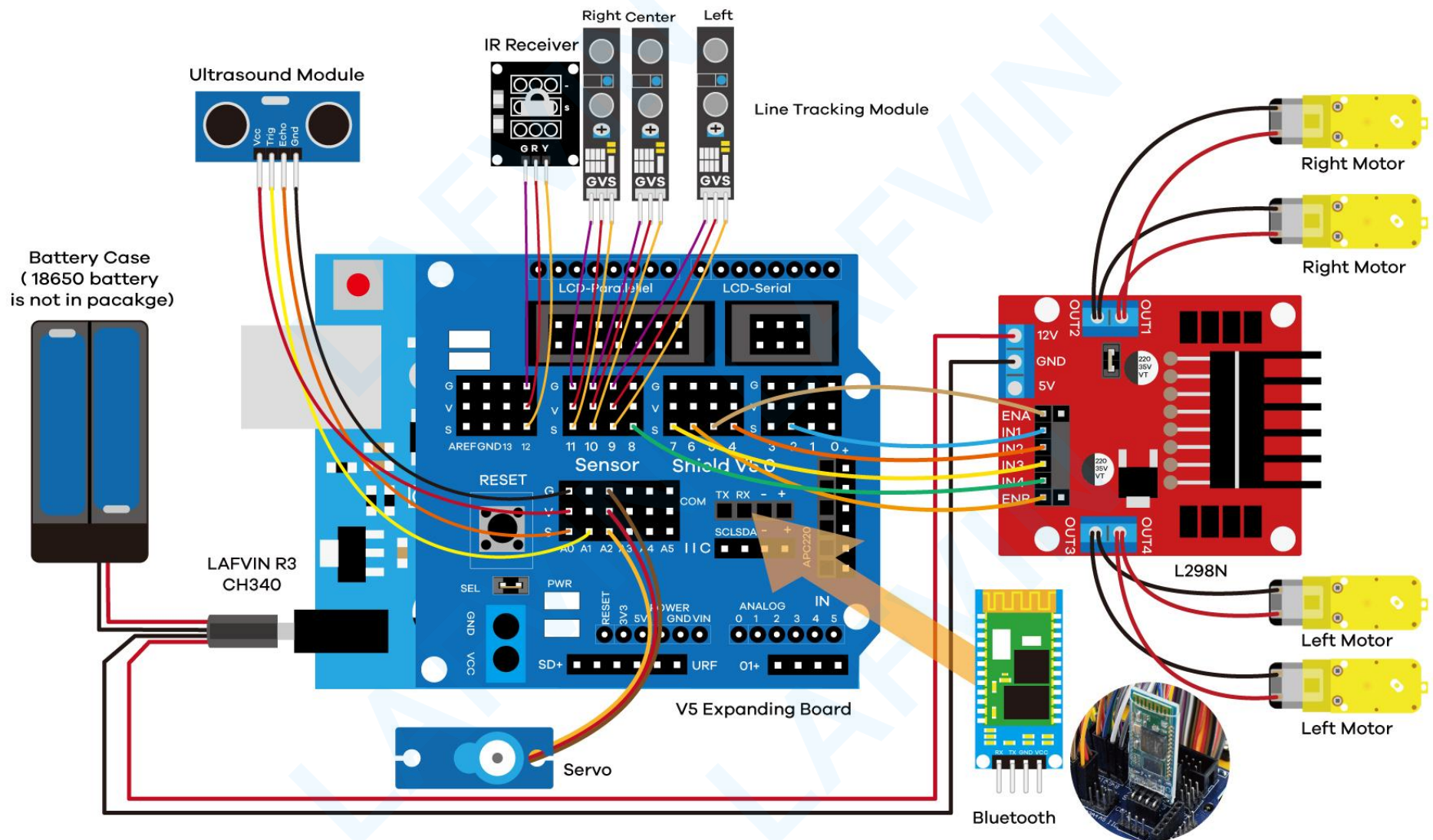


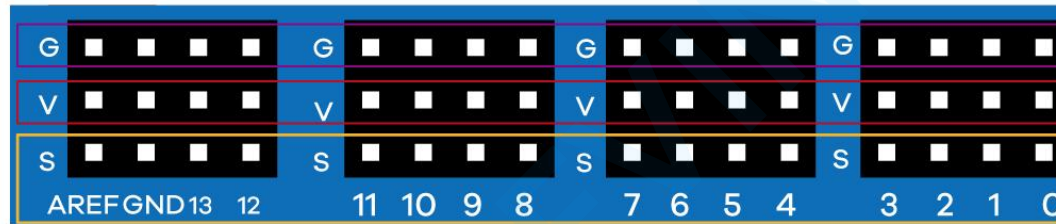




M3*10mm Screw

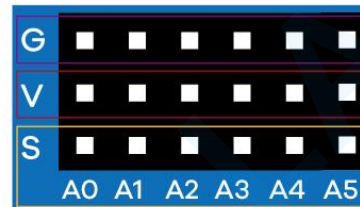
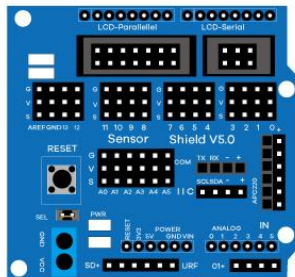






G: GND
V: VCC
S: Signal (Digital IO Ports D0-D13)

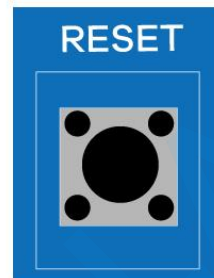
V5 Expanding Board



G: GND
V: VCC
S: Signal (Analog IO Ports A0-A5)



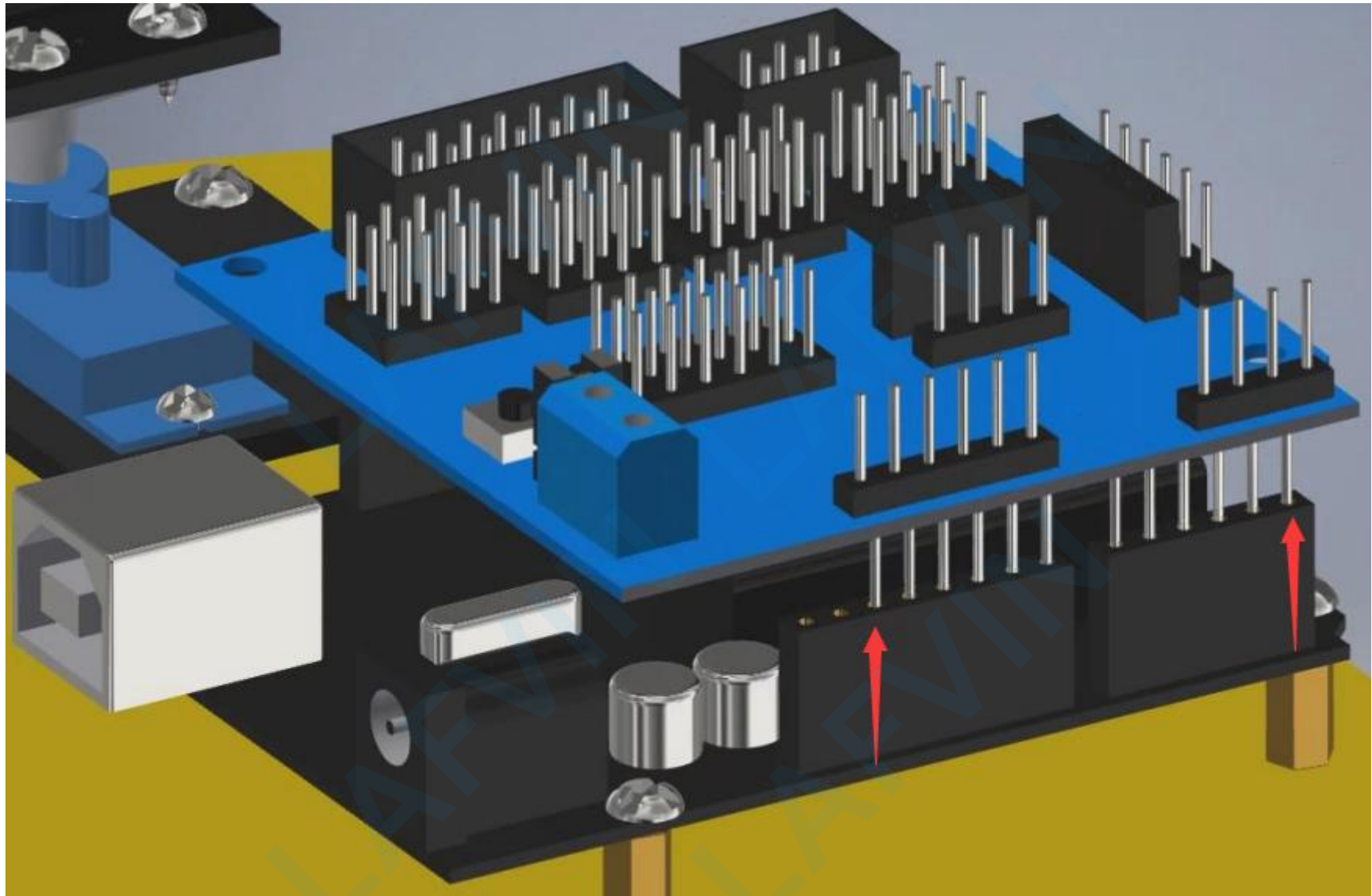
Bluetooth Port

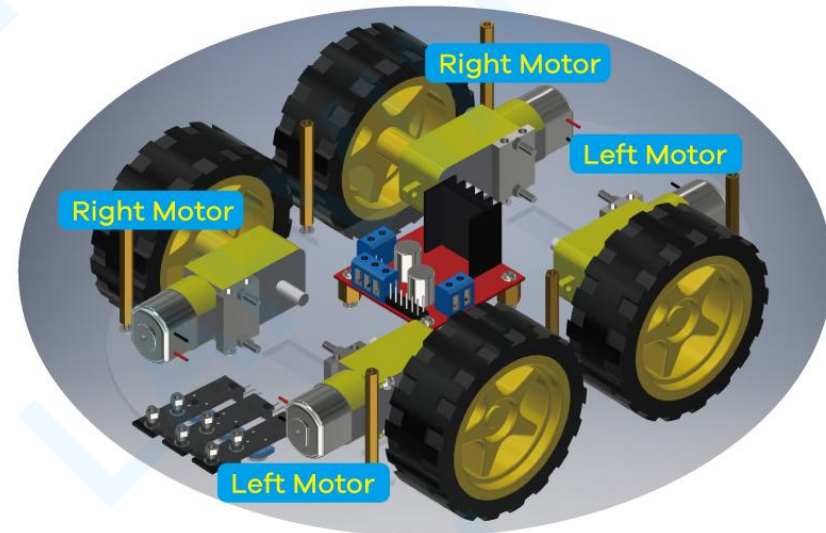
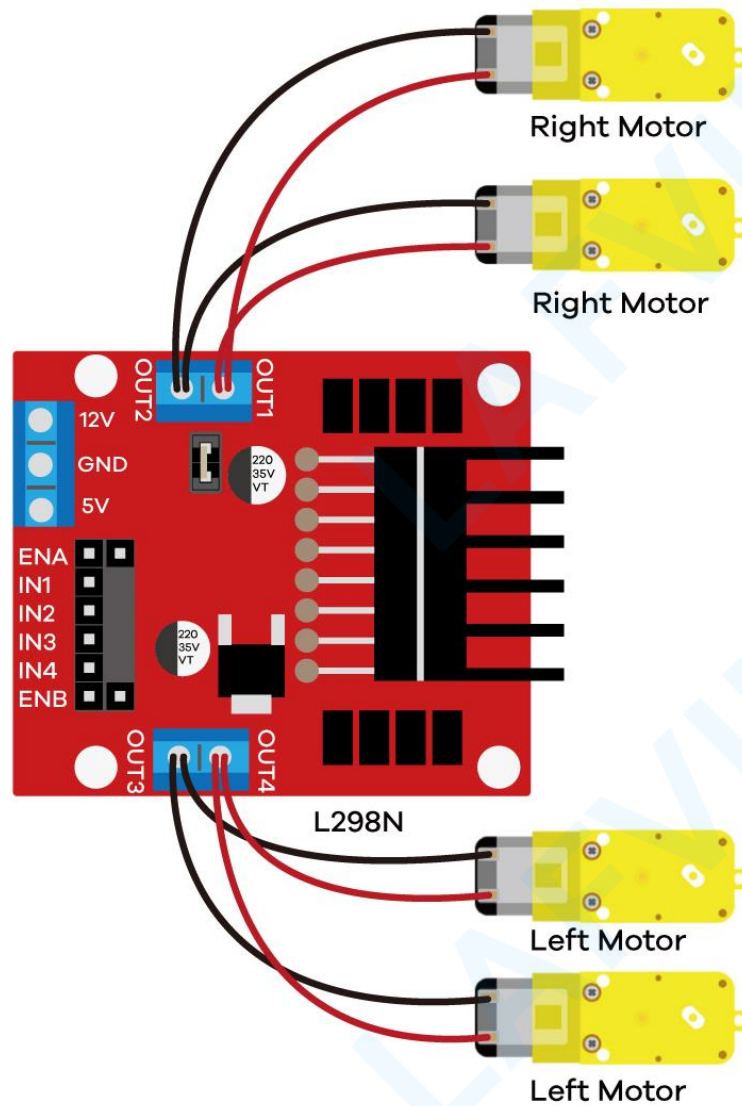


Reset Button



Power LED





Left Motor

Cable
Red Black

OUT 4 OUT 3

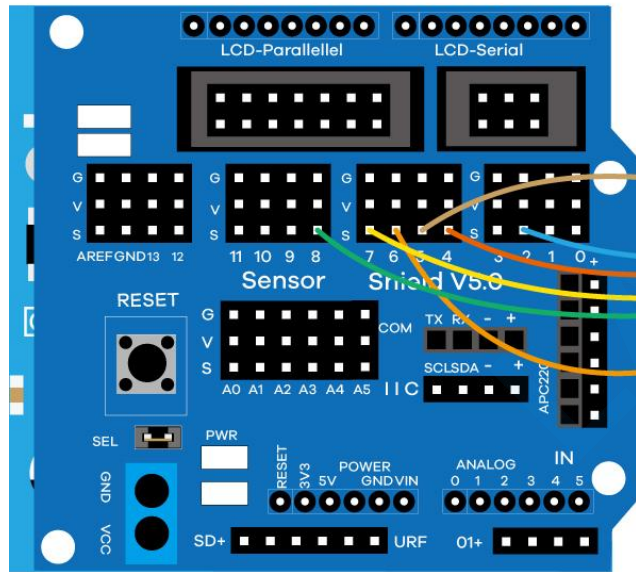
L298N

Right Motor

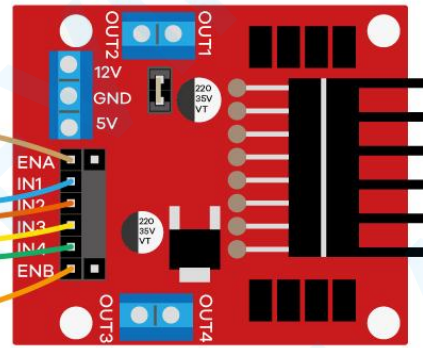
Cable
Red Black

OUT 1 OUT 2

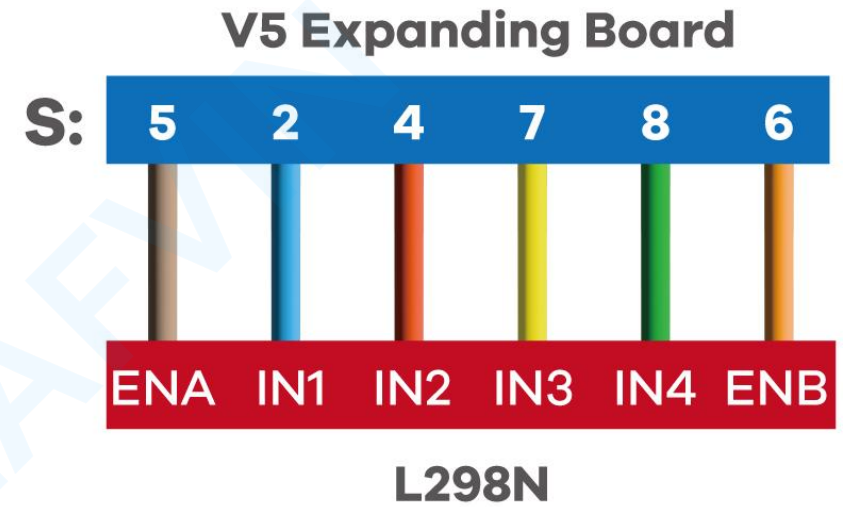
L298N



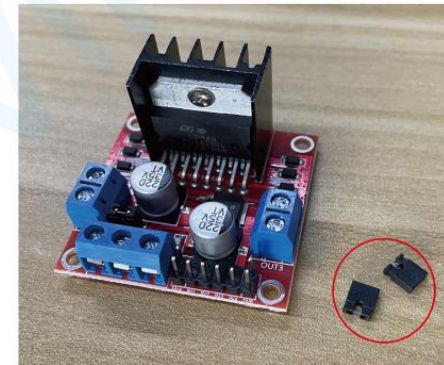
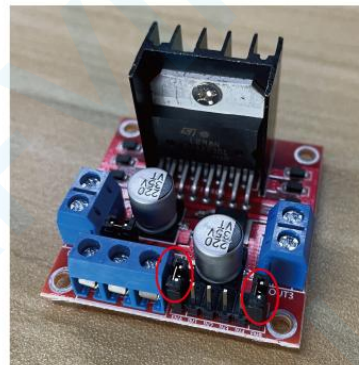
V5 Expanding Board



L298N

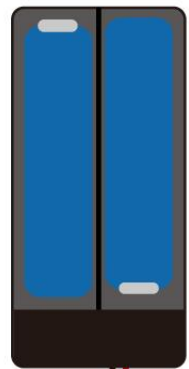


Please unplug the shorting cap

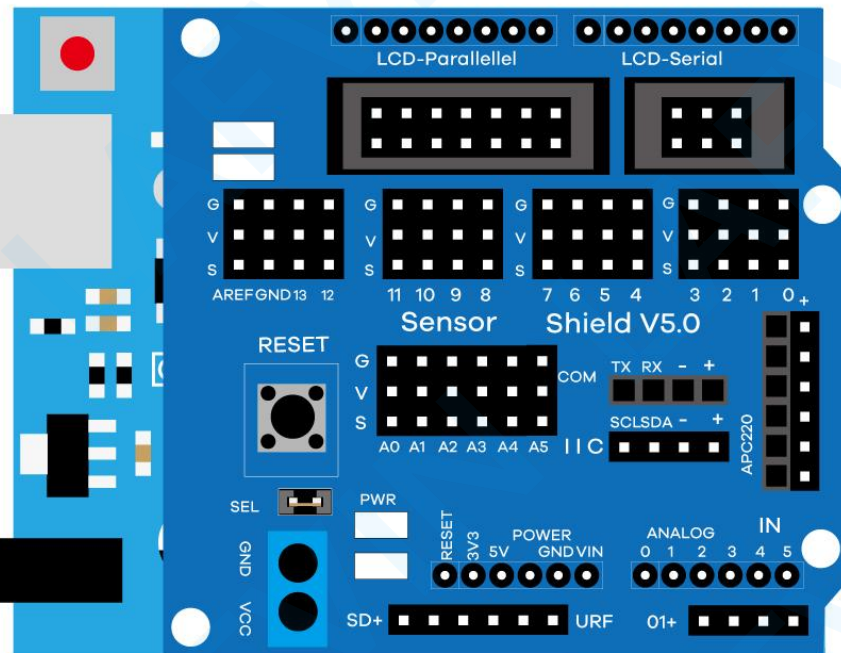


LAFVIN

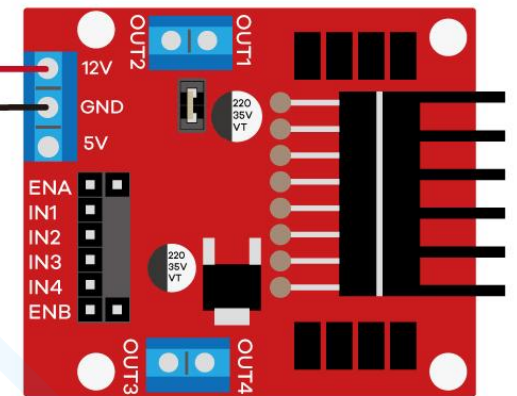
Battery Case
(18650 battery
is not in package)



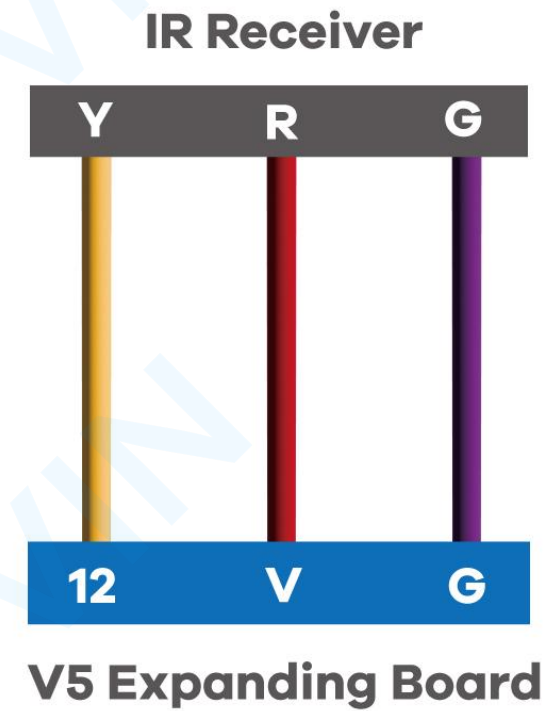
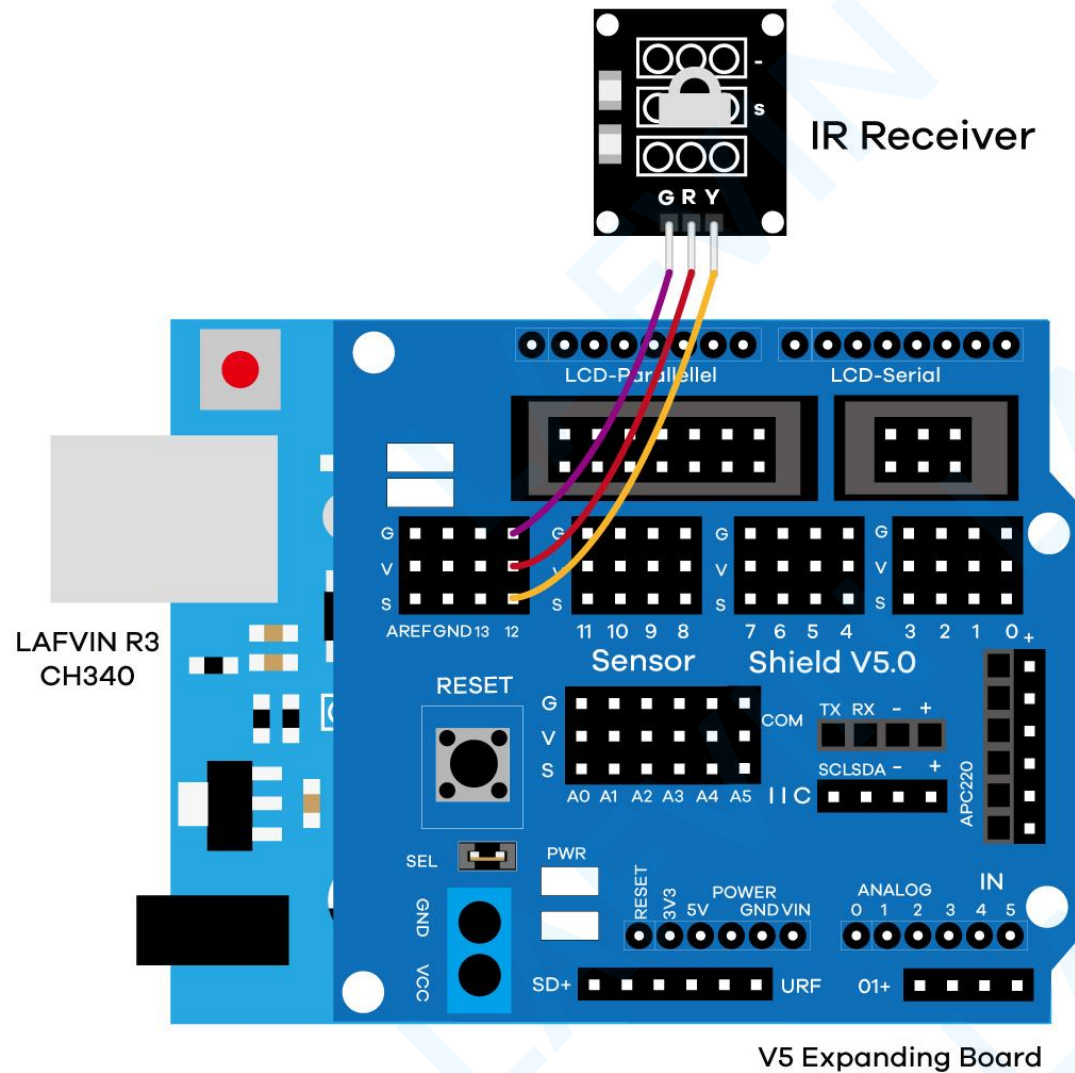
LAFVIN R3
CH340



V5 Expanding Board



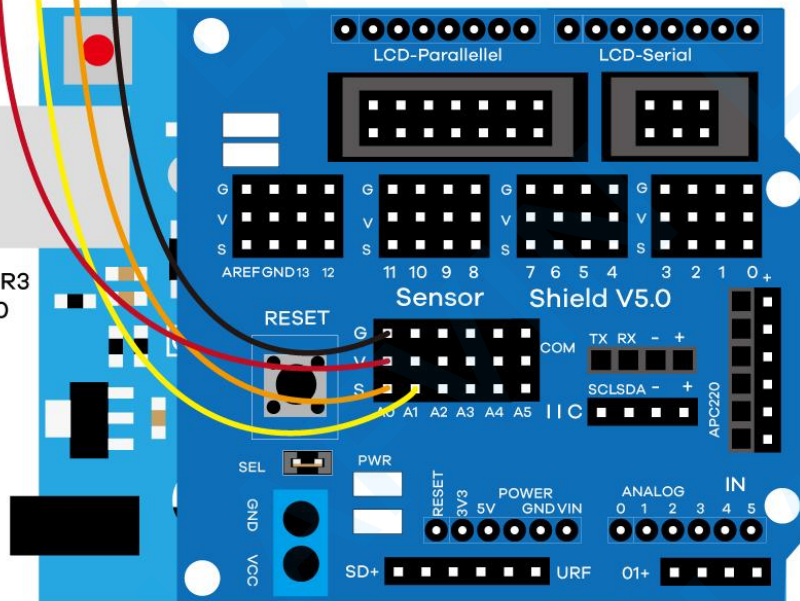
L298N



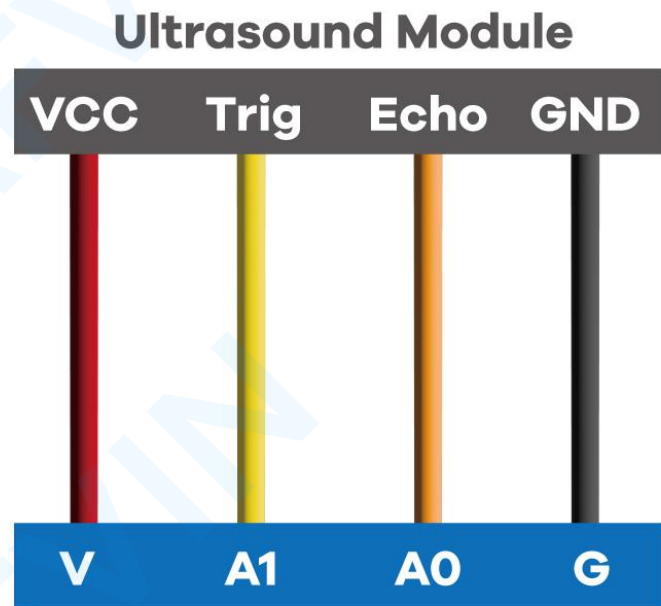


Ultrasound Module

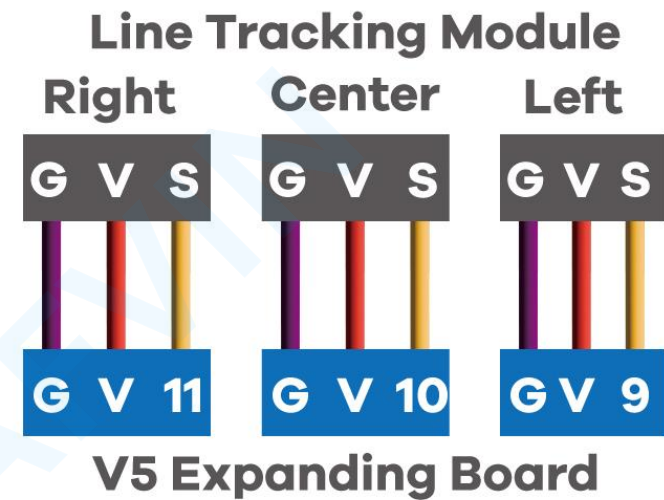
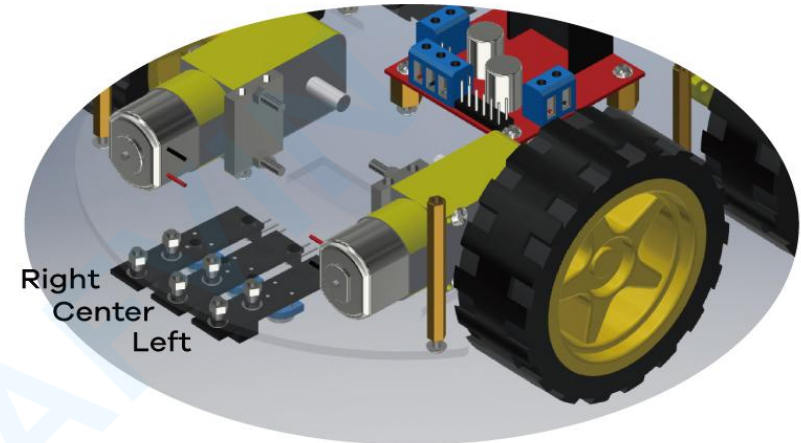
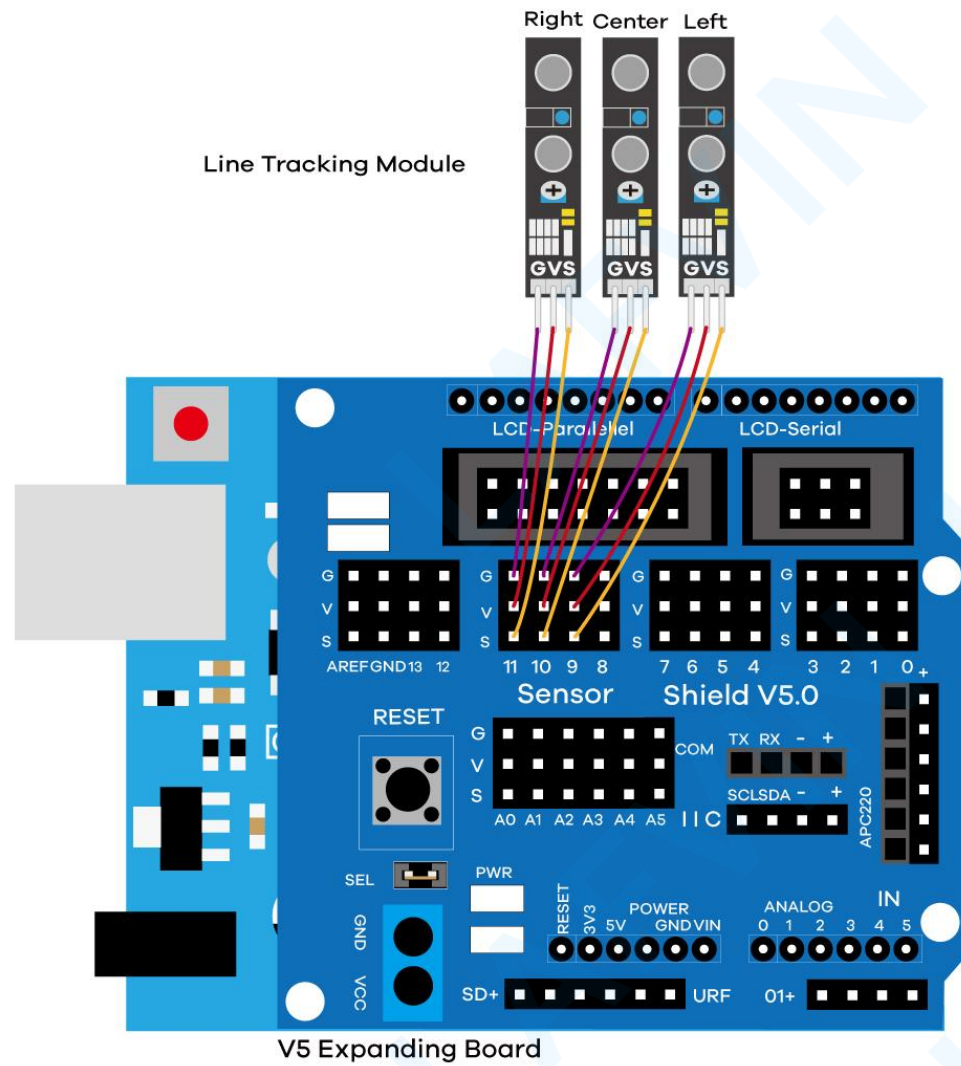
LAFVIN R3
CH340

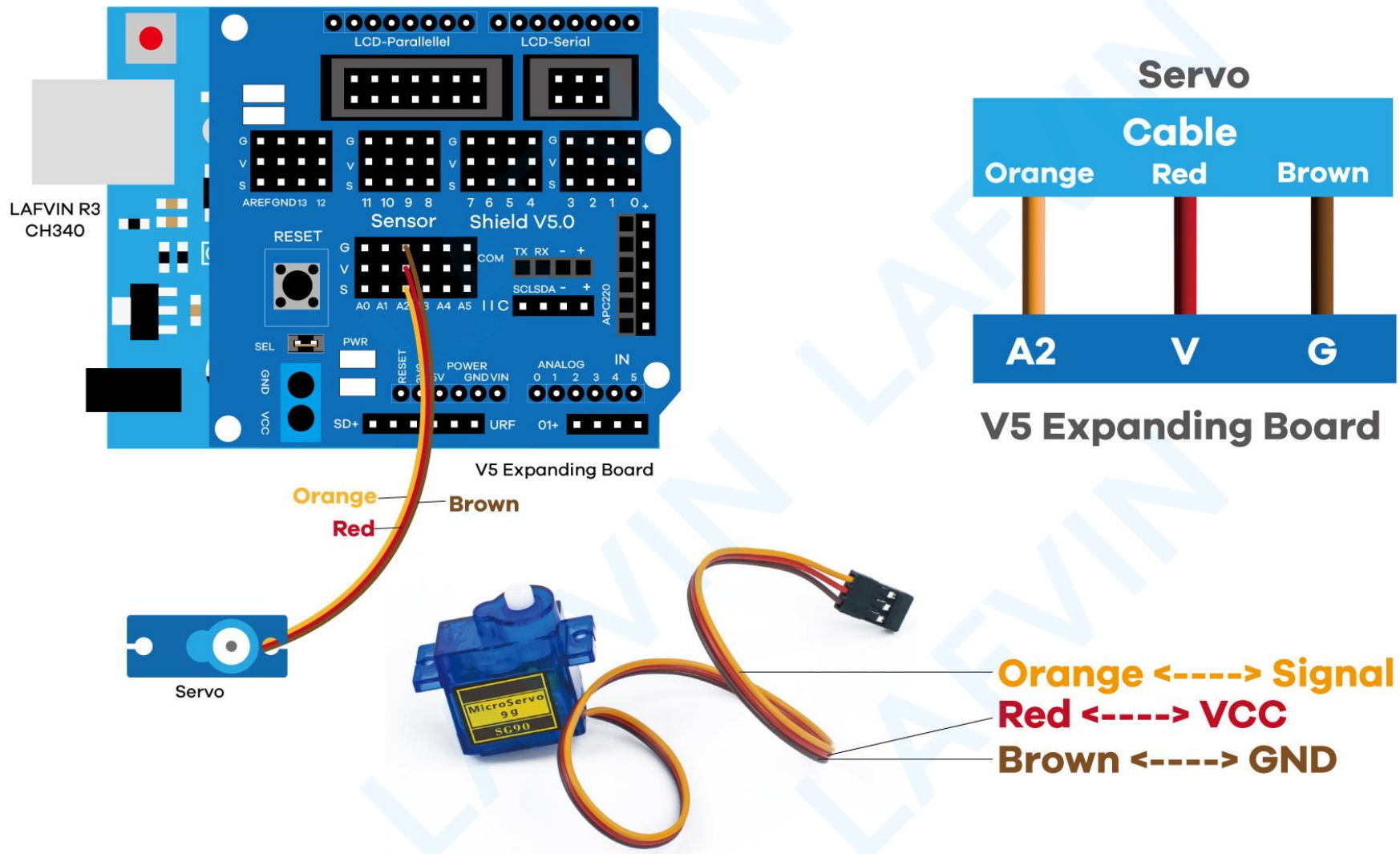


V5 Expanding Board

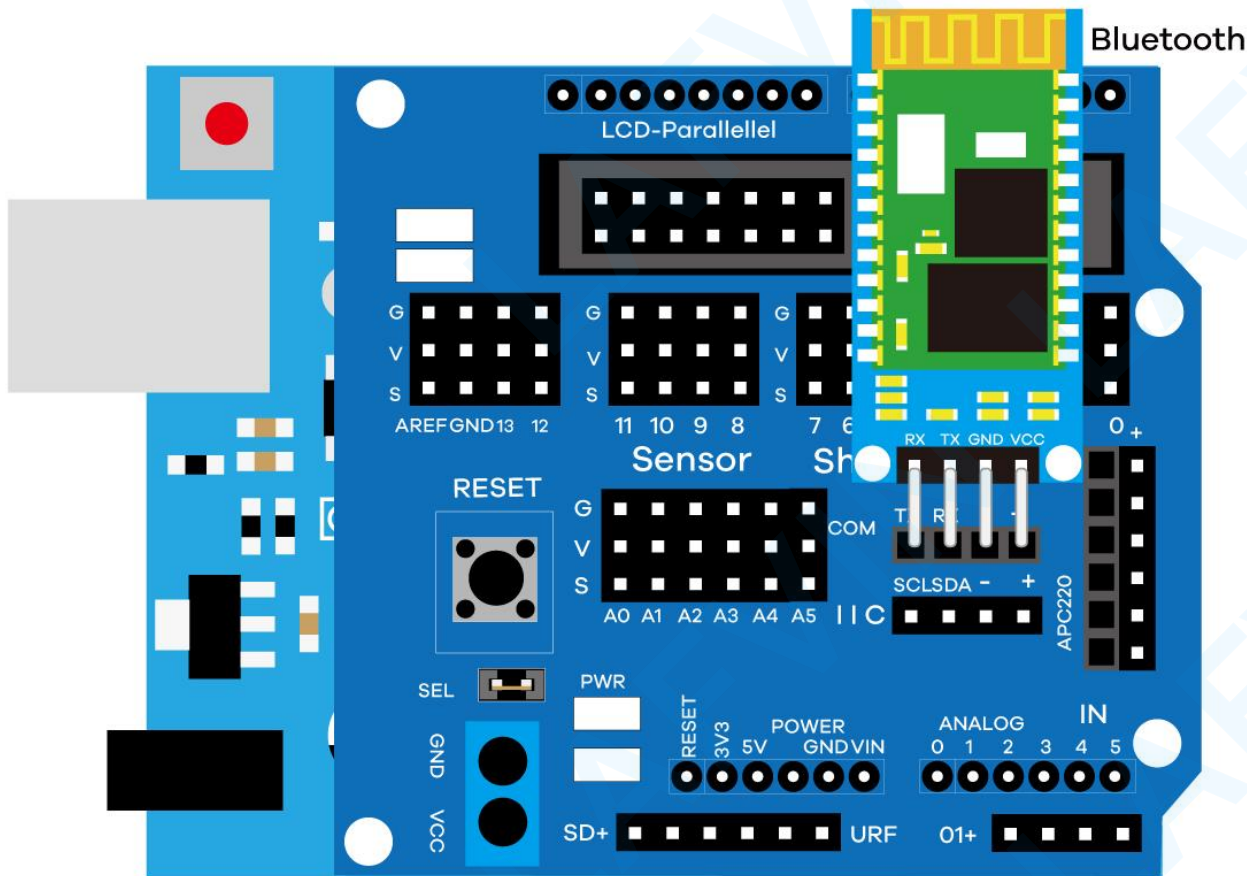


V5 Expanding Board





Insert directly, no wires required



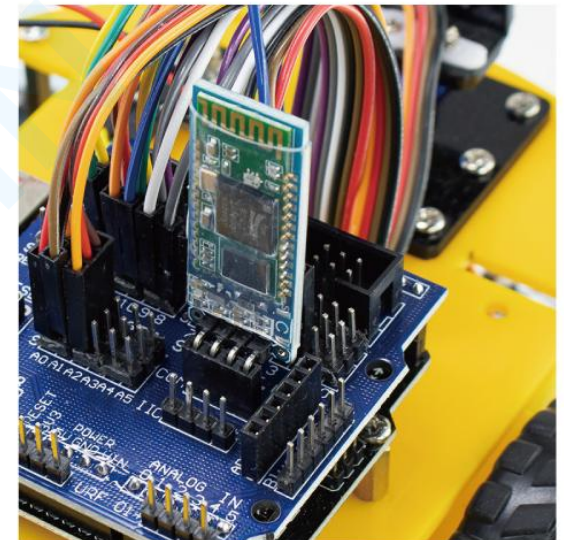
V5 Expanding Board

Bluetooth

RX TX GND VCC

TX RX - +

V5 Expanding Board



LAFVIN



◆ Step 2: Install Arduino IDE

Go to <https://www.arduino.cc/en/Main/Software>. If you have questions about the installation of Arduino IDE, you can refer to [Getting Started with Arduino products](#).

Before starting this installation procedure, make sure you have the latest version of the Arduino IDE installed in your computer. If you don't, uninstall it and install it again. Otherwise, it may not work.

Downloads



Arduino IDE 1.8.19

The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. This software can be used with any Arduino board.

Refer to the [Getting Started](#) page for Installation instructions.

SOURCE CODE

Active development of the Arduino software is [hosted by GitHub](#). See the instructions for [building the code](#). Latest release source code archives are available [here](#). The archives are PGP-signed so they can be verified using [this](#) gpg key.

DOWNLOAD OPTIONS

Windows Win 7 and newer
Windows ZIP file

Windows app Win 8.1 or 10 [Get](#) 

Linux 32 bits
Linux 64 bits
Linux ARM 32 bits
Linux ARM 64 bits

Mac OS X 10.10 or newer

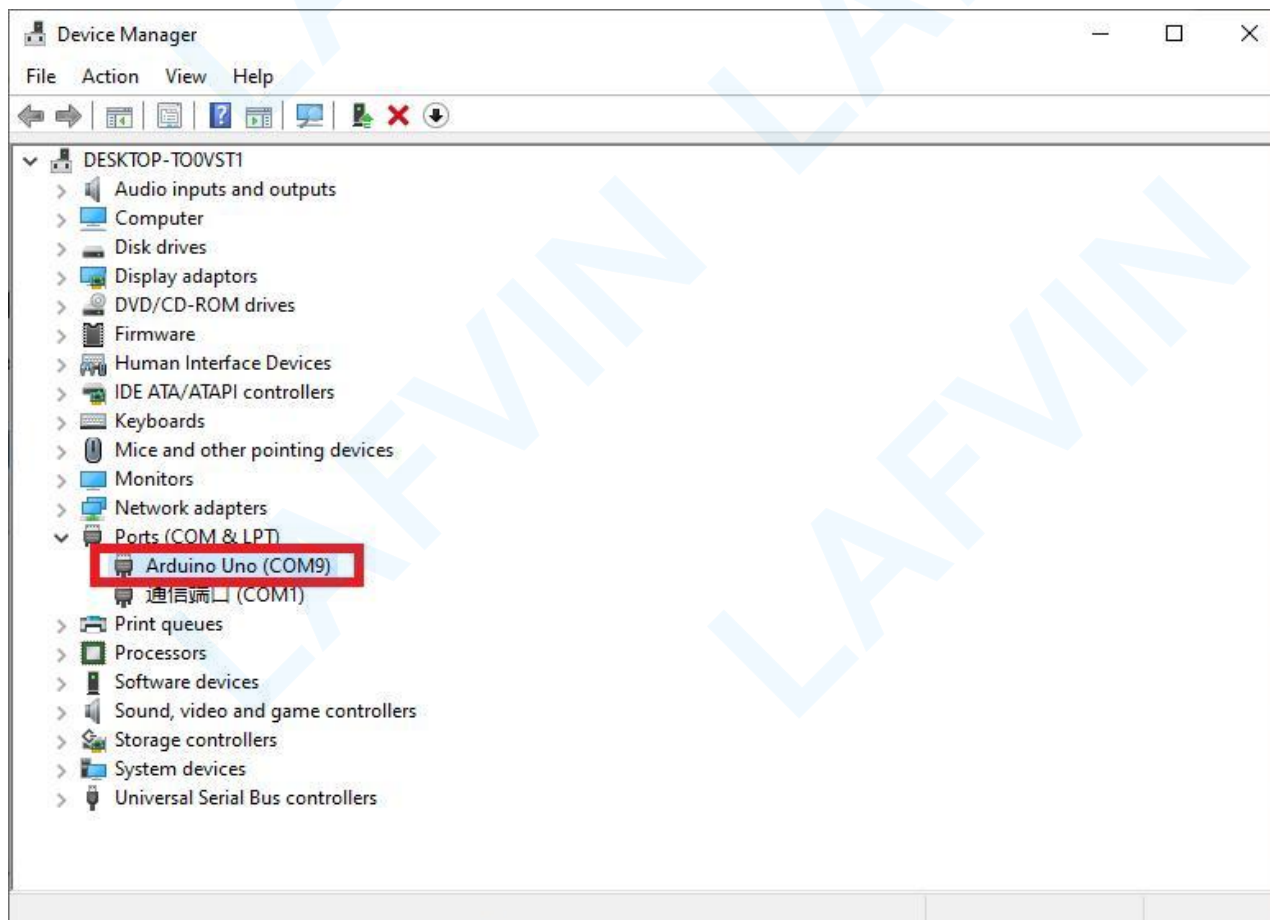
[Release Notes](#)

[Checksums \(sha512\)](#)

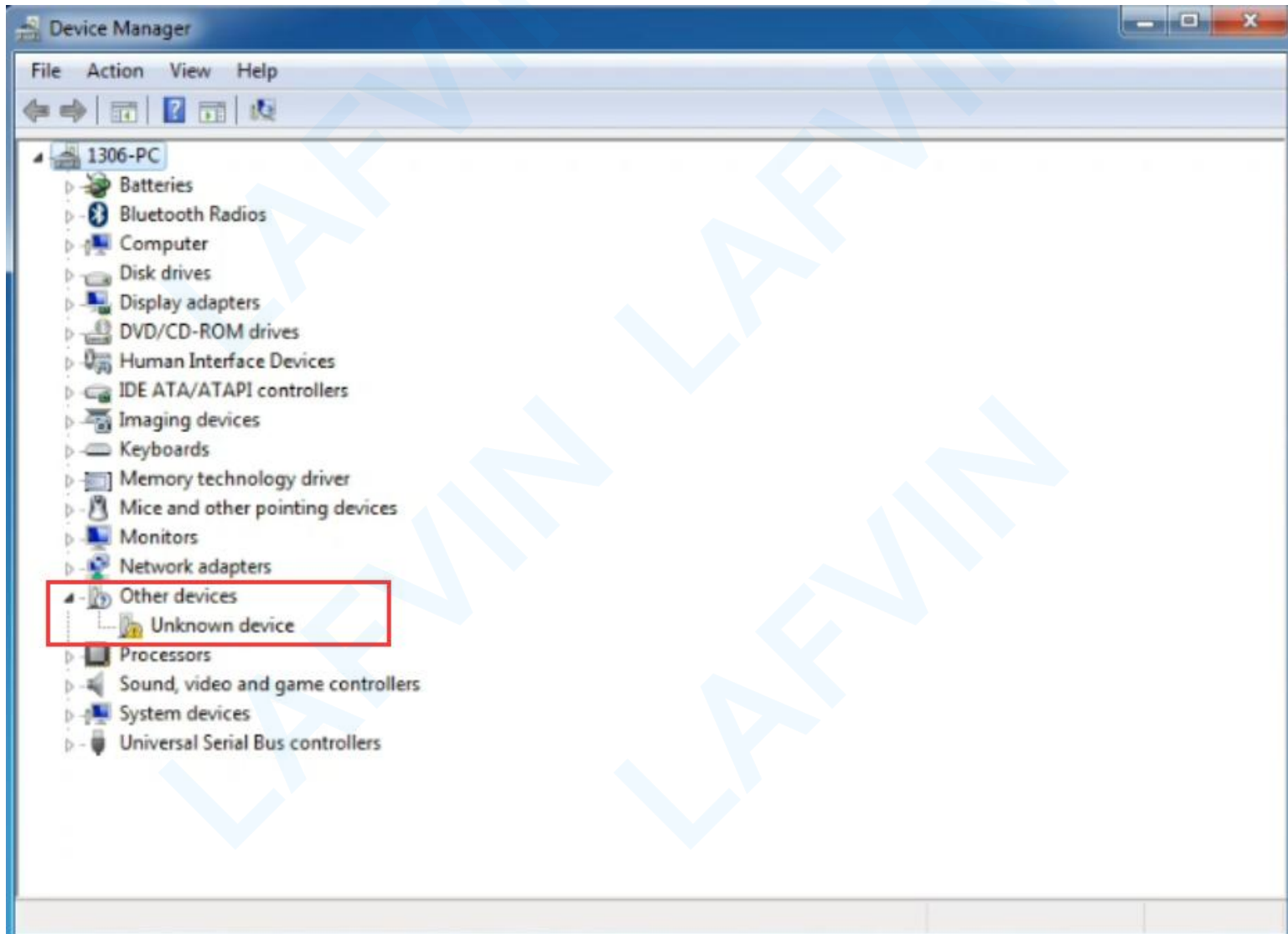
◆ Step 3: Install Arduino UNO Driver

① Plug in your Arduino UNO board. Navigate to the Windows Device Manager (Start > type 'device manager' to search > double click the first result to launch the Device Manager).

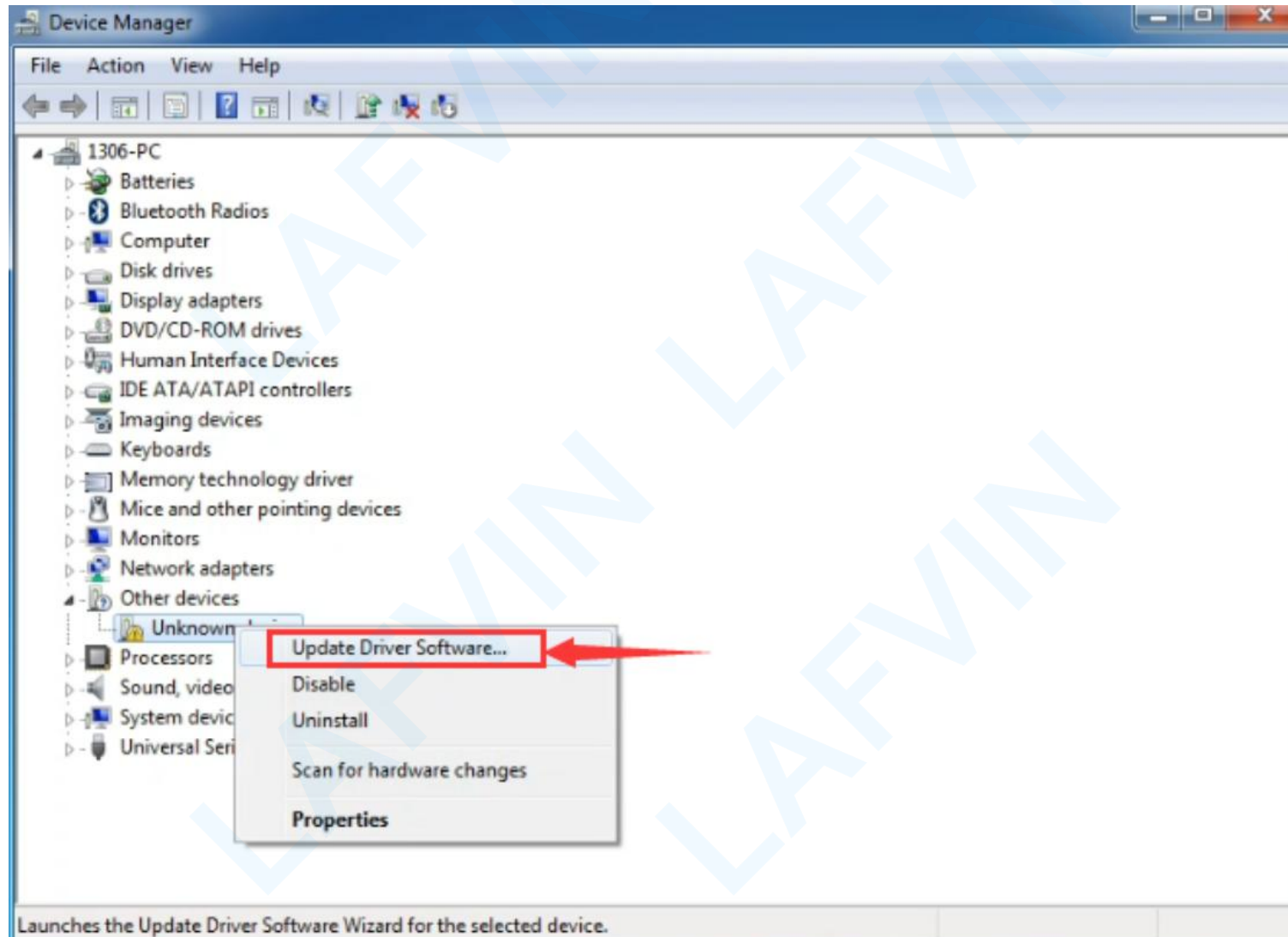
Go to the port, if "**Arduino UNO (COMX)**" appears in the port list, it means that the Arduino Uno driver has been successfully installed. you can skip this step.



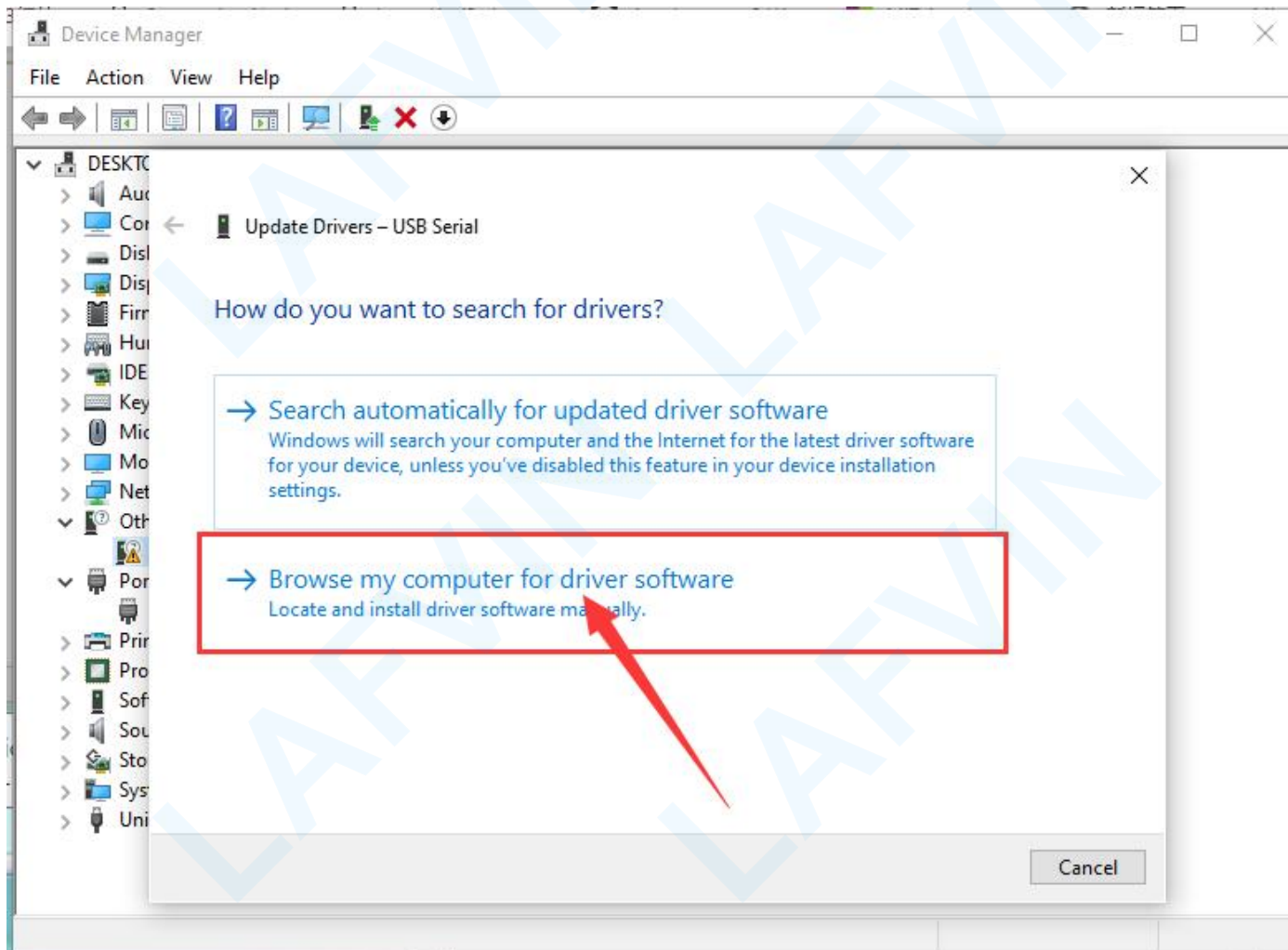
②If you can't find the Arduino UNO (COMX) port, go to "Other Devices" and find an **“Unknown device”** or an **“Arduino UNO”** list item with a yellow warning



③Then right-click on the device and select the top menu option (Update Driver Software...) shown as the figure below.

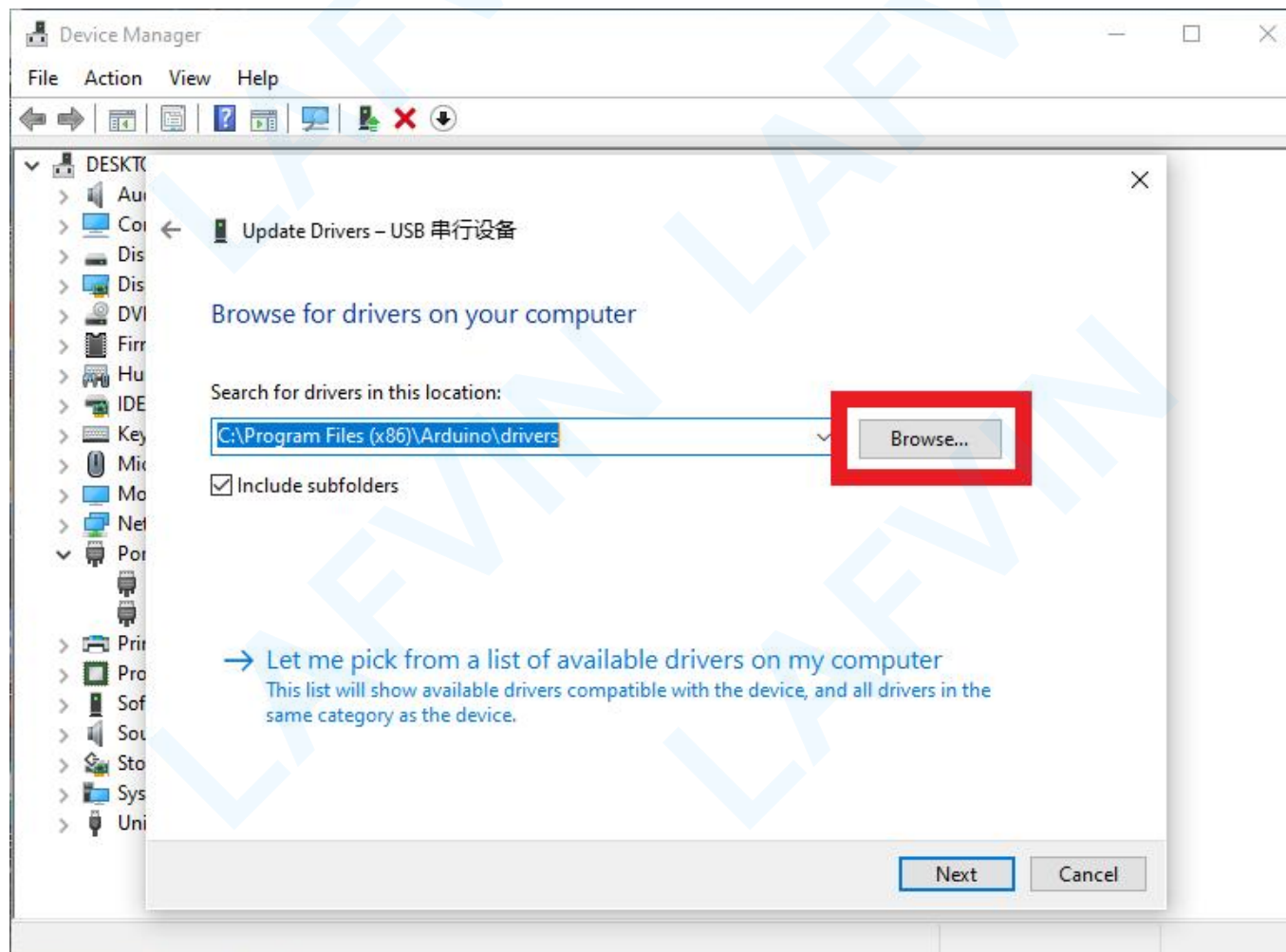


④Then it will be prompted to either “Search Automatically for updated driver software” or “Browse my computer for driver software”. Shown as below. In this page, select “Browse my computer for driver software”.



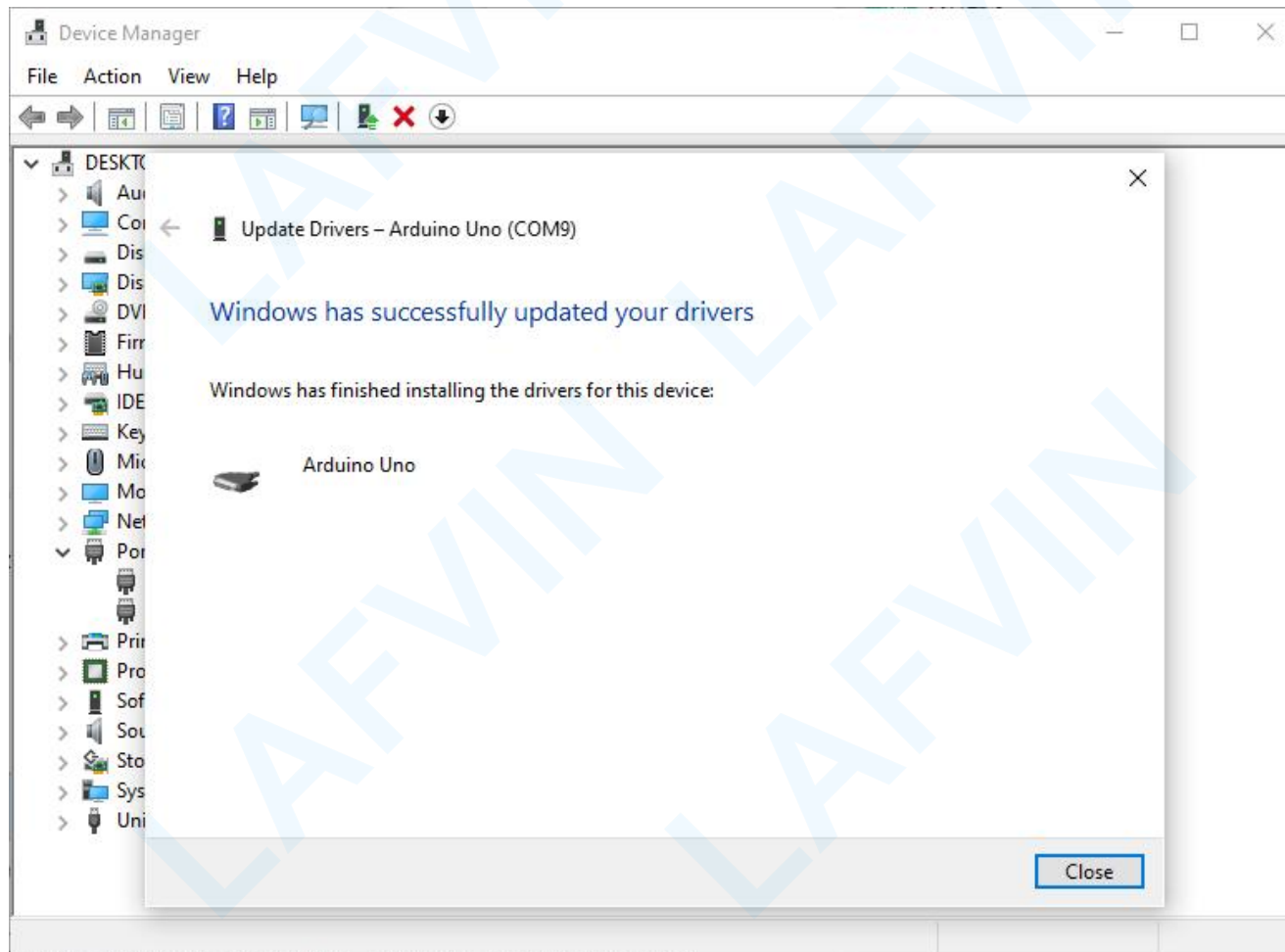
⑤Right-click on the device and select the top menu option (Update Driver Software...).

Select the option to browse and navigate to the :**C:\Program Files(x86)\Arduino\drivers**.(Note: Here is the path you choose to install Arduino IDE. Since I installed the Arduino IDE on the C disk,so the location path I chose is **C:\Program Files(x86)\Arduino\drivers**)

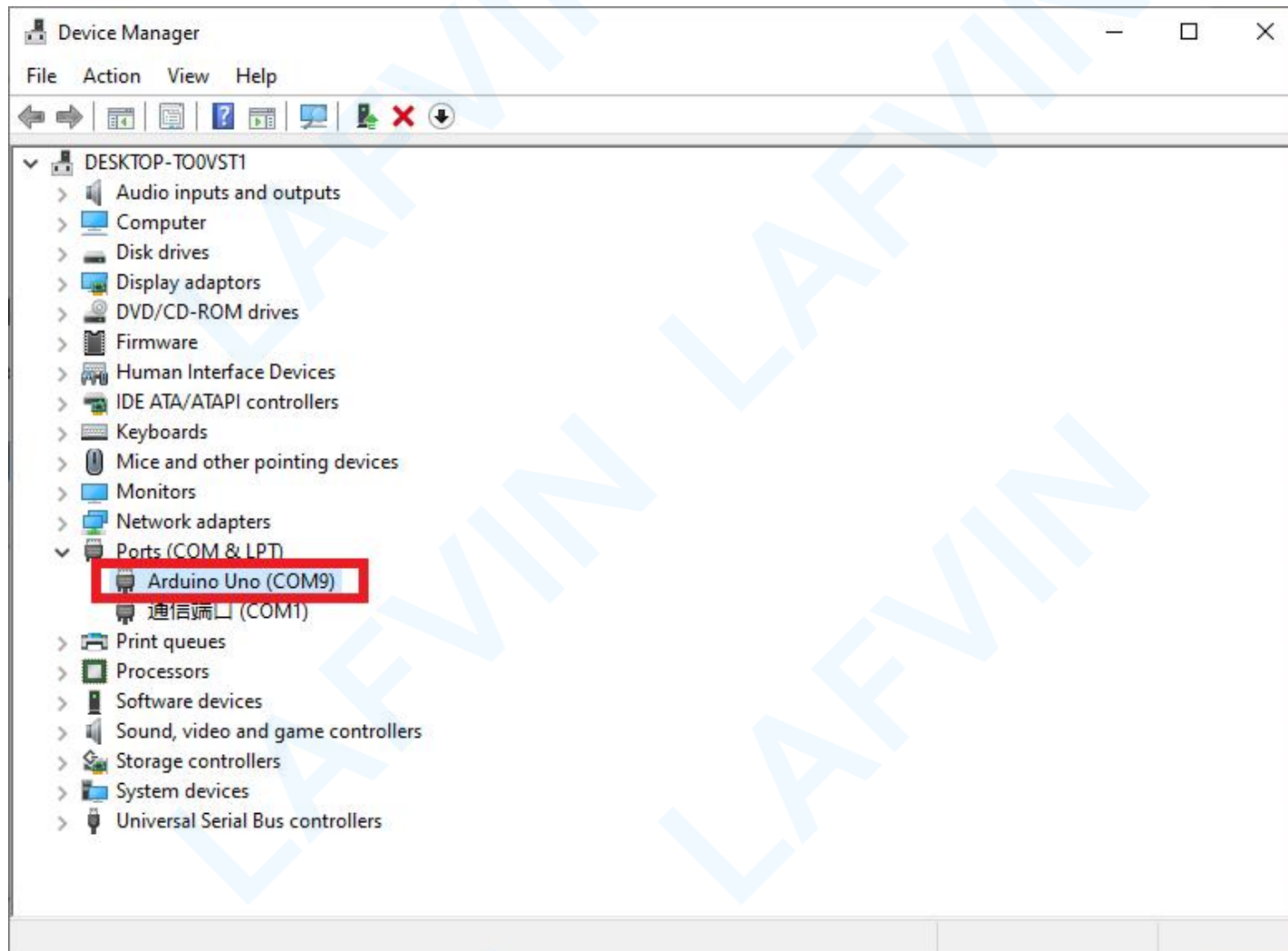


⑥Click “Next” and you may get a security warning, if so, allow the software to be installed.

Once the software has been installed, you will get a confirmation message. Installation completed,click “Close”.



Up to now, the driver is installed well. Then you can right click “Computer”—>“Properties”—>“Device manager”, you should see the device as the figure shown below.

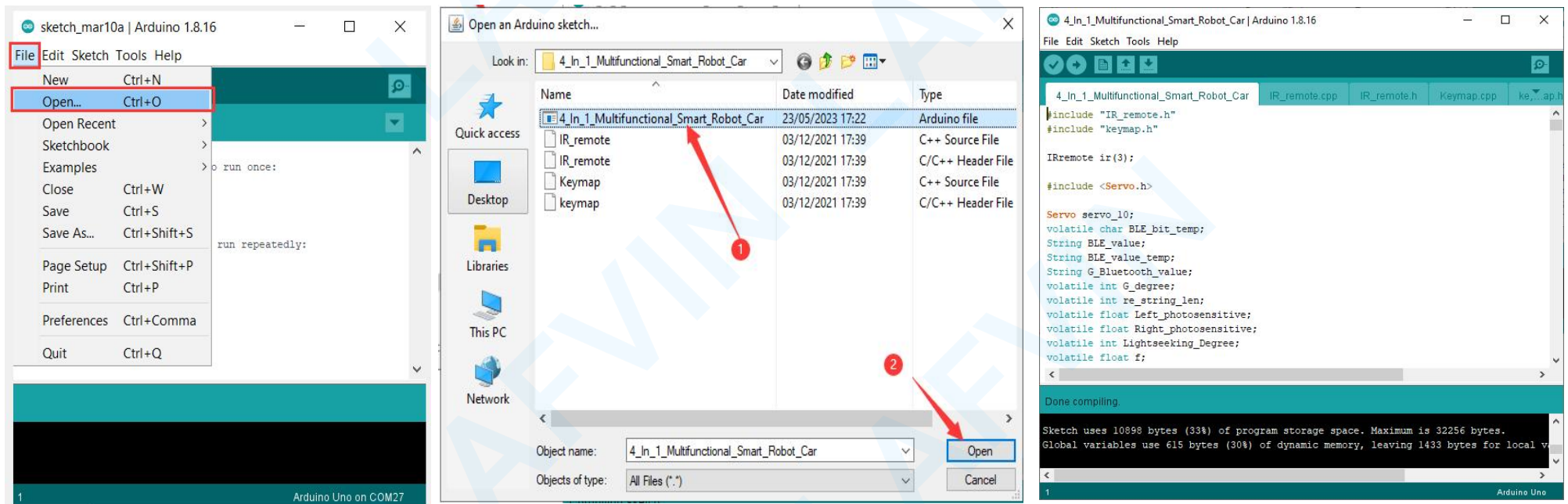


◆ Step 4: Upload Arduino UNO Main Code

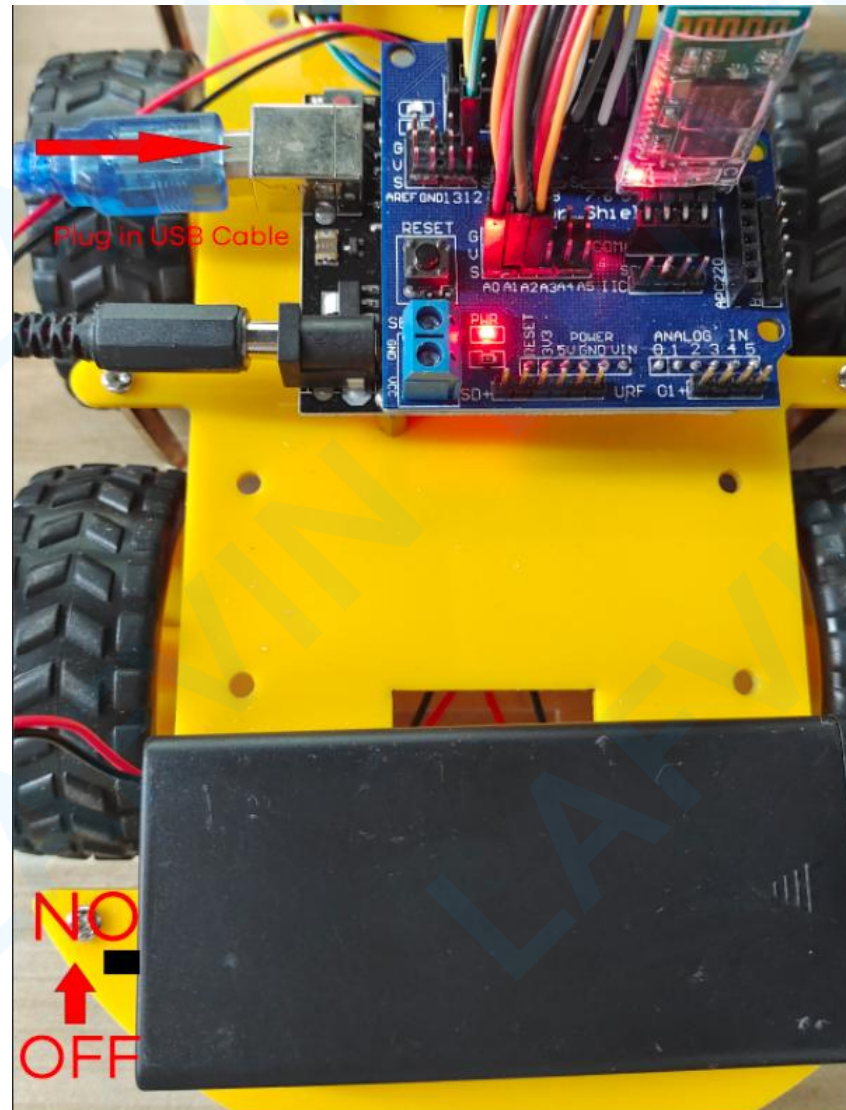
① Start Arduino IDE, open the code in

File->Open...->LAFVIN 4WD Smart Robot Car V1>Main Code>4_In_1_Multifunctional_Smart_Robot_Car.ino

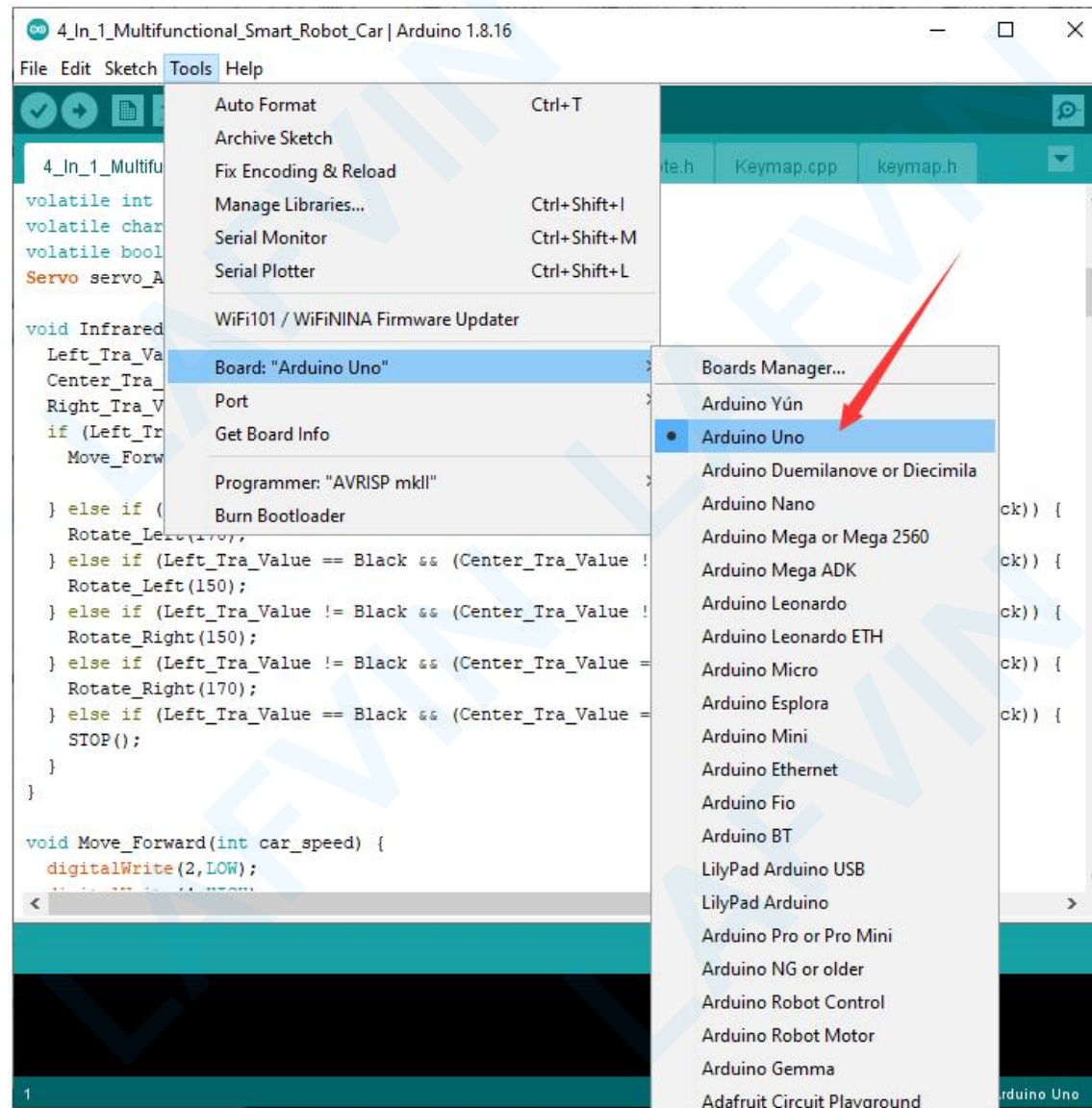
Important: Keymap.c Keymap.h IR_remote.c IR_remote.h file and .ino Arduino file are saved in the same folder



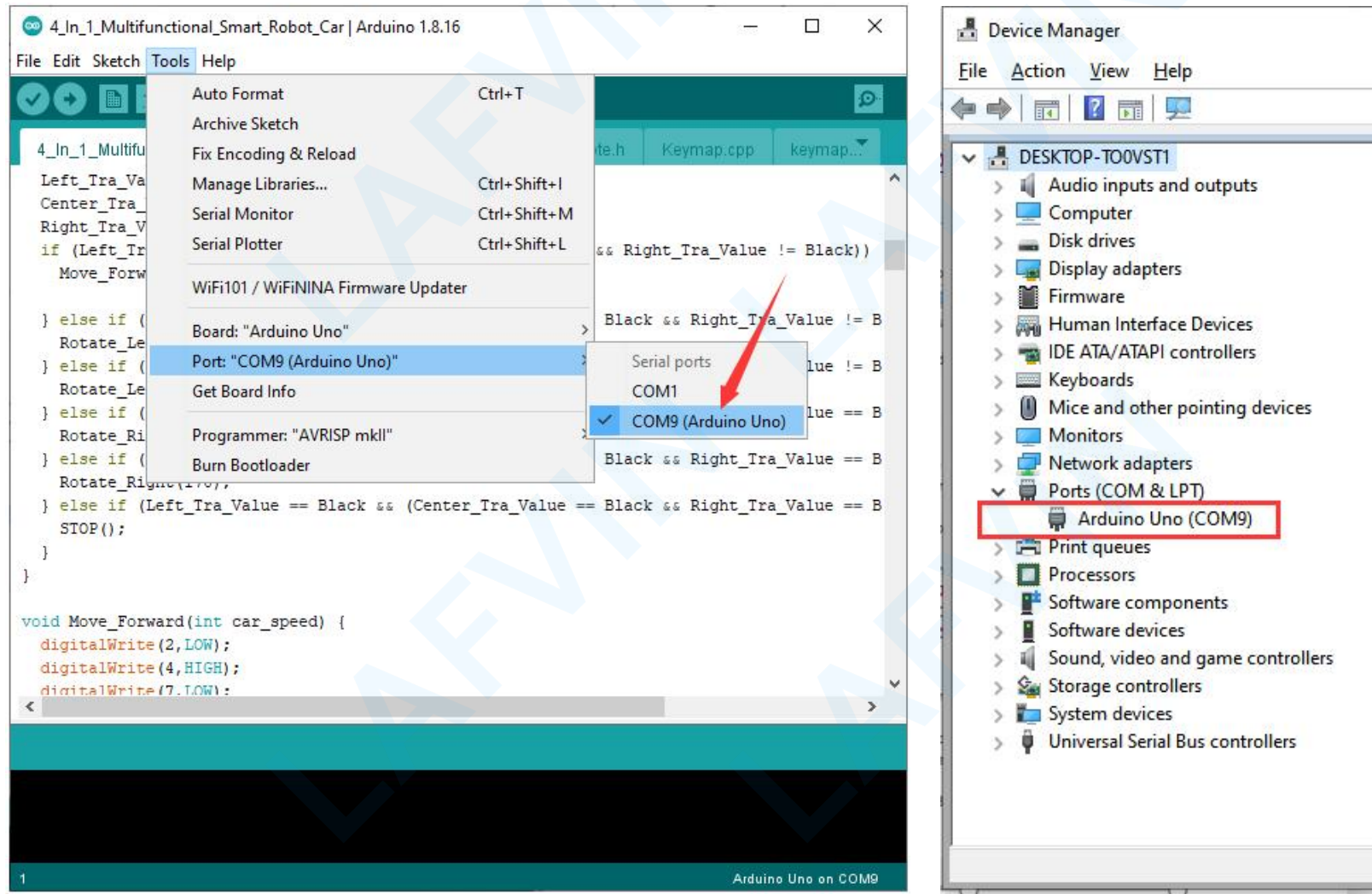
② Plug the Arduino UNO board to your computer. (It may be that the power supply of the USB interface of the computer is insufficient, you can turn on the power switch of the robot car at the same time.)



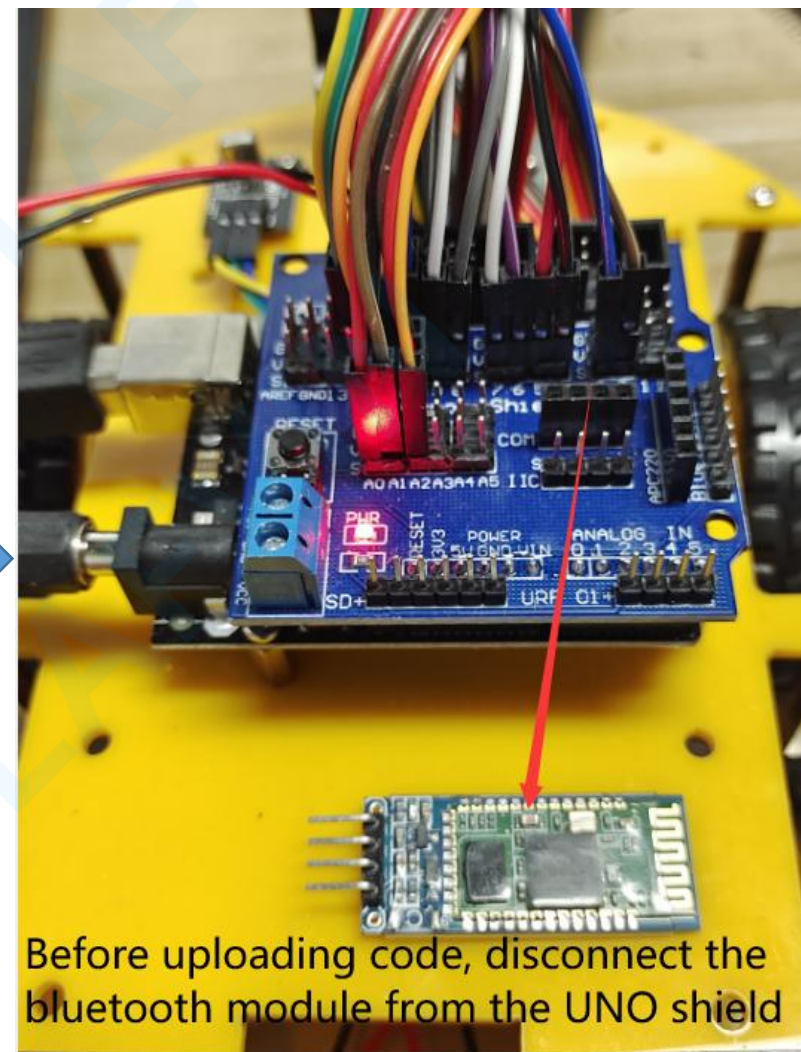
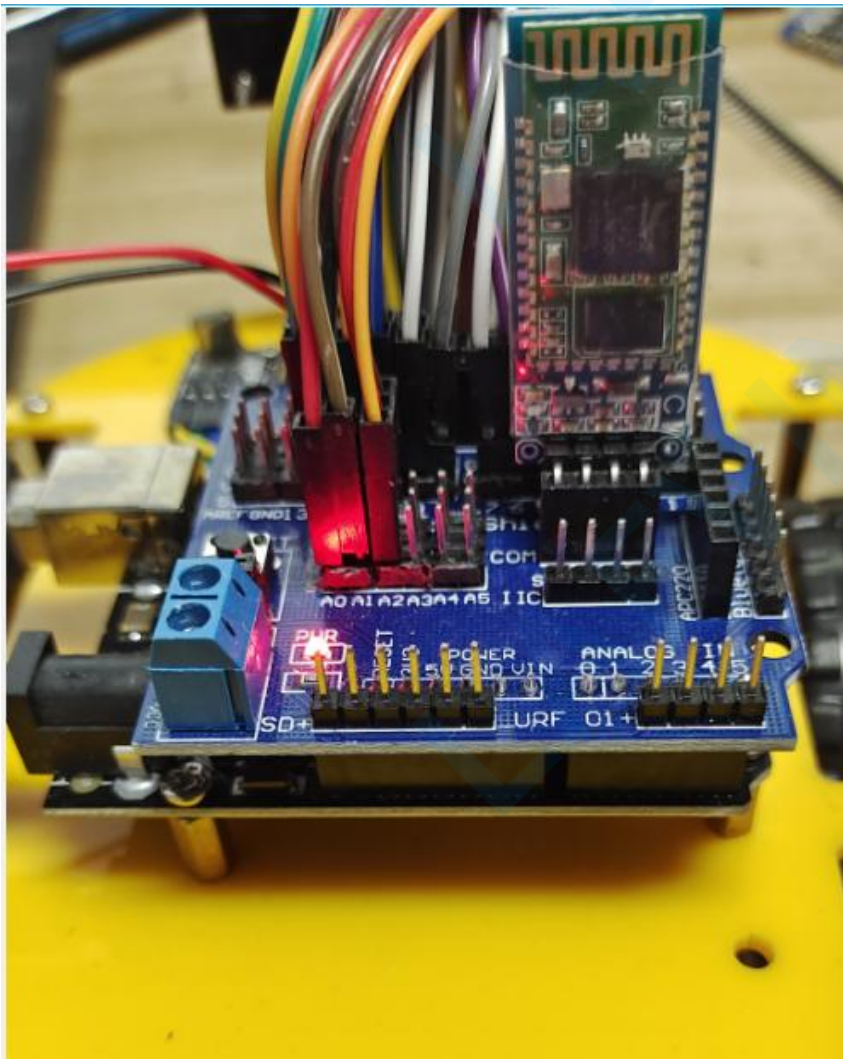
③Select your Board in **Tools > Board menu>Arduino UNO**



④Select the Port (if you don't see the COM Port in your Arduino IDE, you need to [Install the Arduino UNO Drivers](#))

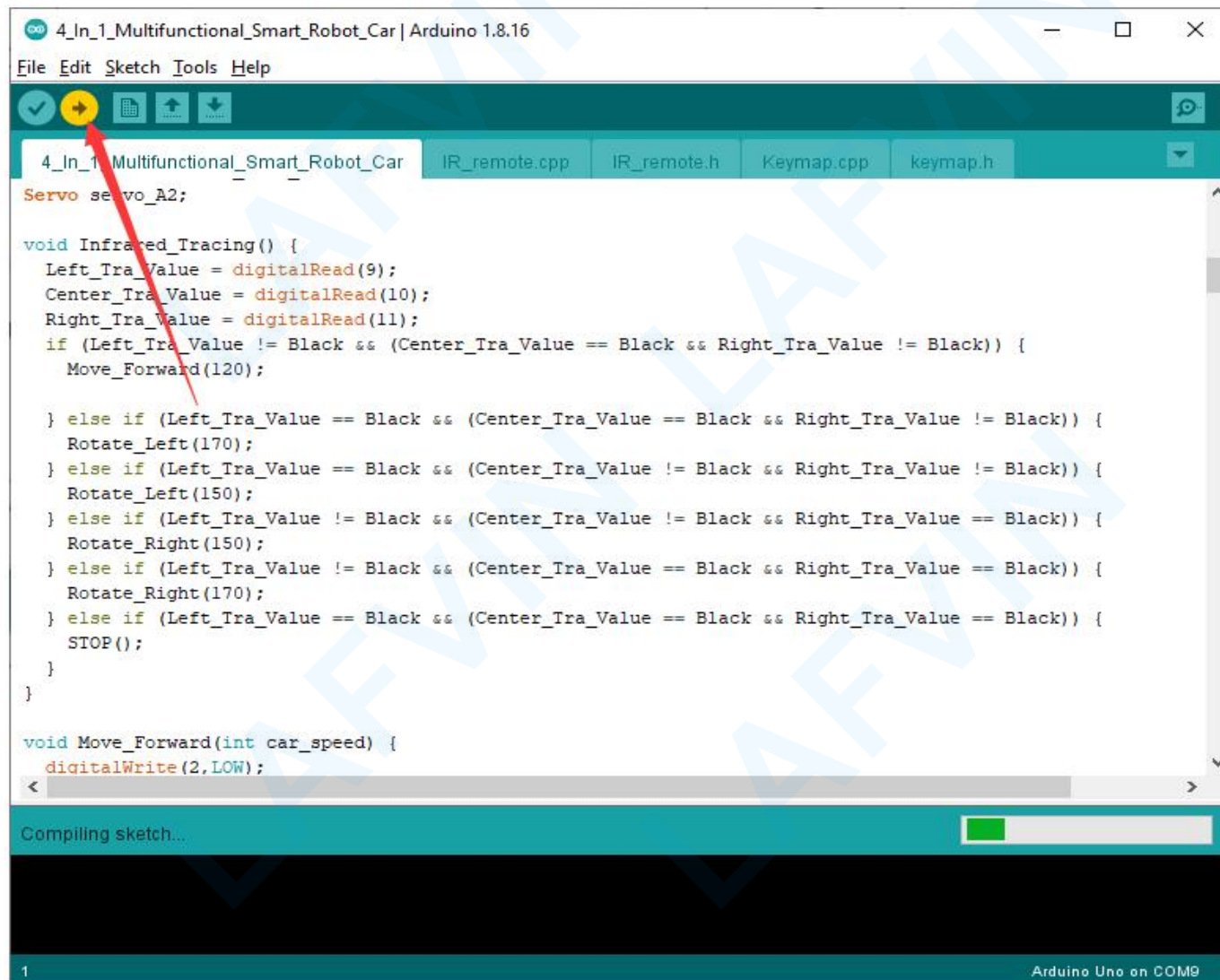


- ⑤ Important: Before uploading code, disconnect the Bluetooth module from the V5 shield. After successfully uploading the code, install the Bluetooth module to the V5 expansion board. Otherwise, the Bluetooth module occupies the communication interface(RX TX) for uploading code, and Arduino IDE will report an error that the upload failed.



Before uploading code, disconnect the bluetooth module from the UNO shield

⑥Click the **Upload** button in the Arduino IDE. Wait a few seconds while the code compiles and uploads to your board.

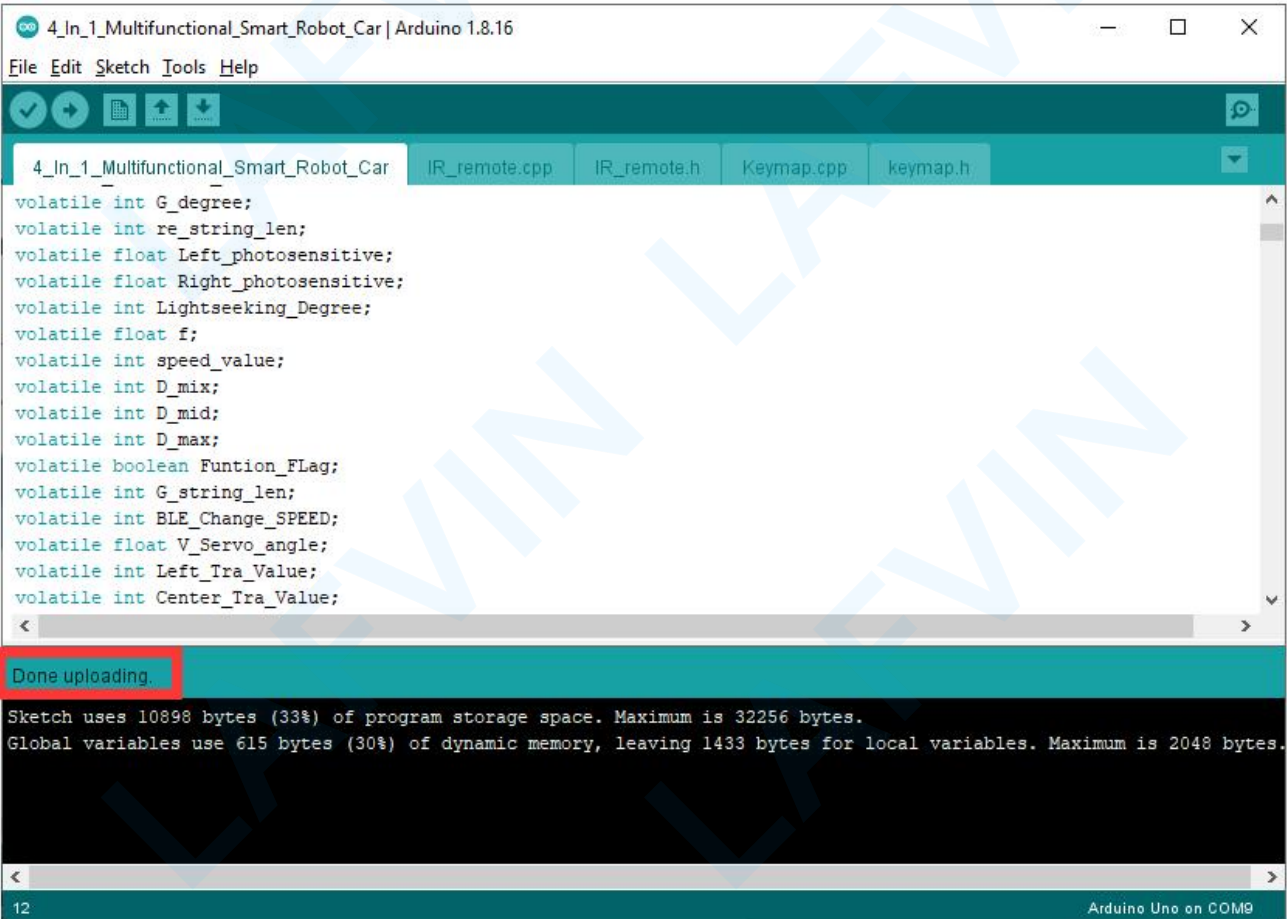


If everything went as expected, you should see a “**Done uploading.**” message. If the Arduino IDE reports errors maybe you missed some steps. Arduino getting started guide is as follows

[Upload a sketch in Arduino IDE](#)

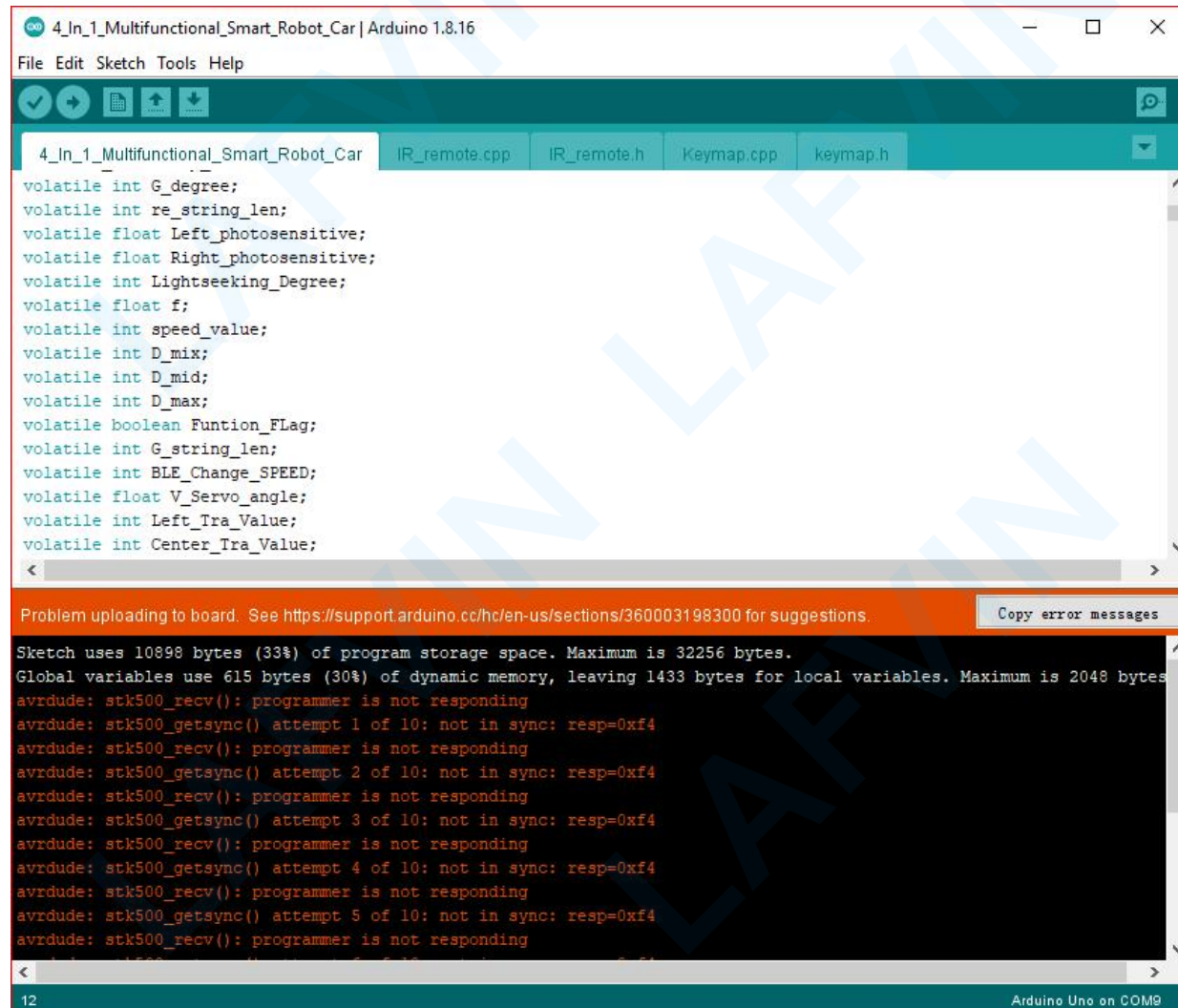
[Errors when uploading a sketch](#)

[Getting Started with Arduino products](#)



```
4_In_1_Multifunctional_Smart_Robot_Car | Arduino 1.8.16
File Edit Sketch Tools Help
4_In_1_Multifunctional_Smart_Robot_Car IR_remote.cpp IR_remote.h Keypad.cpp keypad.h
volatile int G_degree;
volatile int re_string_len;
volatile float Left_photosensitive;
volatile float Right_photosensitive;
volatile int Lightseeking_Degree;
volatile float f;
volatile int speed_value;
volatile int D_mix;
volatile int D_mid;
volatile int D_max;
volatile boolean Funtion_Flag;
volatile int G_string_len;
volatile int BLE_Change_SPEED;
volatile float V_Servo_angle;
volatile int Left_Tra_Value;
volatile int Center_Tra_Value;
Done uploading.
Sketch uses 10898 bytes (33%) of program storage space. Maximum is 32256 bytes.
Global variables use 615 bytes (30%) of dynamic memory, leaving 1433 bytes for local variables. Maximum is 2048 bytes.
12 Arduino Uno on COM9
```


If the error `avrdude: stk500_recv(): programmer is not responding` is reported. The frequent reason is that the RX TX of the UNO device port is occupied, you need to perform [steps ⑤](#).



```
4_In_1_Multifunctional_Smart_Robot_Car | Arduino 1.8.16
File Edit Sketch Tools Help
4_In_1_Multifunctional_Smart_Robot_Car IR_remote.cpp IR_remote.h Keymap.cpp keymap.h
volatile int G_degree;
volatile int re_string_len;
volatile float Left_photosensitive;
volatile float Right_photosensitive;
volatile int Lightseeking_Degree;
volatile float f;
volatile int speed_value;
volatile int D_mix;
volatile int D_mid;
volatile int D_max;
volatile boolean Funtion_Flag;
volatile int G_string_len;
volatile int BLE_Change_SPEED;
volatile float V_Servo_angle;
volatile int Left_Tra_Value;
volatile int Center_Tra_Value;

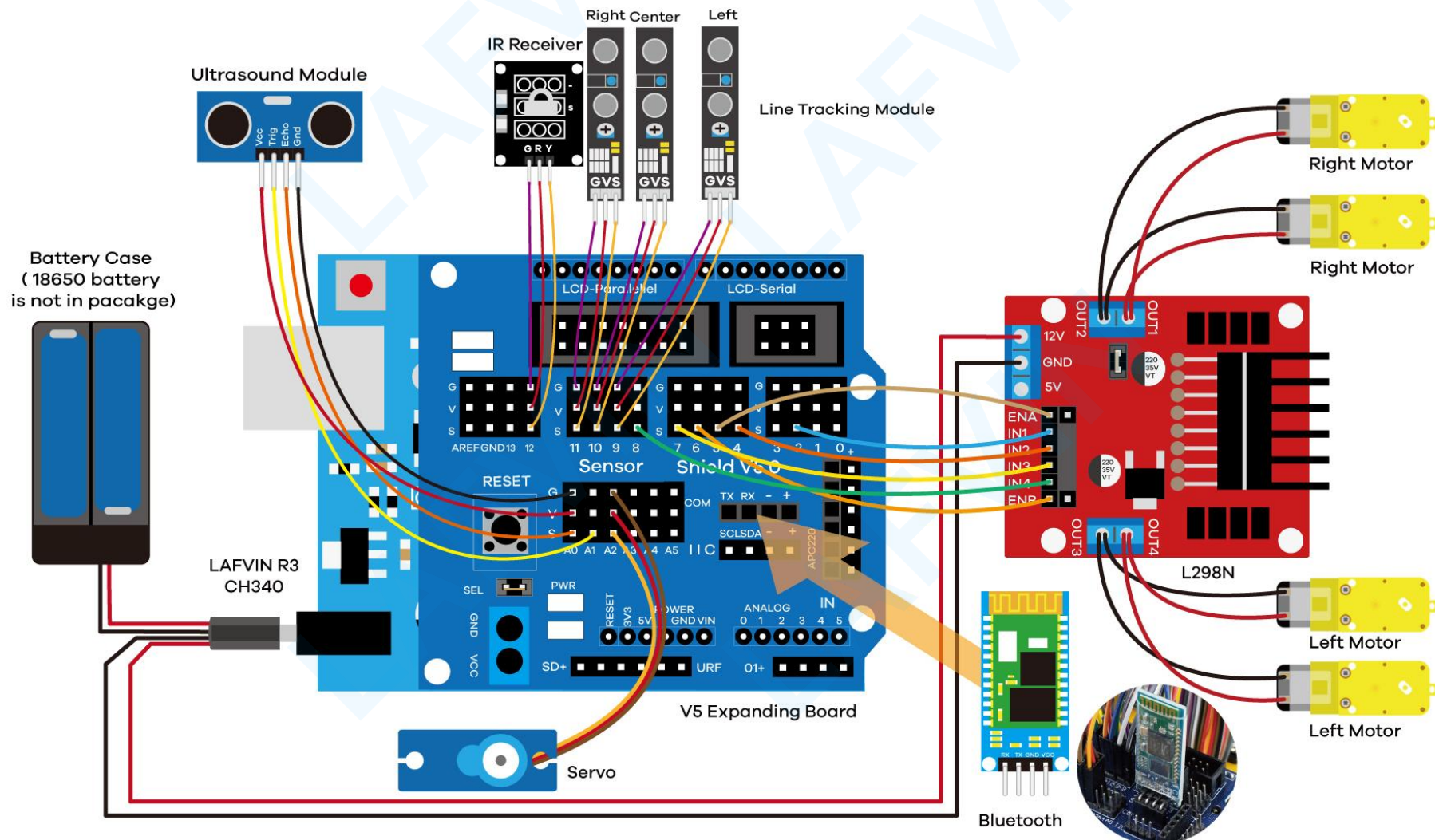
Problem uploading to board. See https://support.arduino.cc/hc/en-us/sections/360003198300 for suggestions. Copy error messages

Sketch uses 10898 bytes (33%) of program storage space. Maximum is 32256 bytes.
Global variables use 615 bytes (30%) of dynamic memory, leaving 1433 bytes for local variables. Maximum is 2048 bytes
avrdude: stk500_recv(): programmer is not responding
avrdude: stk500_getsync() attempt 1 of 10: not in sync: resp=0xf4
avrdude: stk500_recv(): programmer is not responding
avrdude: stk500_getsync() attempt 2 of 10: not in sync: resp=0xf4
avrdude: stk500_recv(): programmer is not responding
avrdude: stk500_getsync() attempt 3 of 10: not in sync: resp=0xf4
avrdude: stk500_recv(): programmer is not responding
avrdude: stk500_getsync() attempt 4 of 10: not in sync: resp=0xf4
avrdude: stk500_recv(): programmer is not responding
avrdude: stk500_getsync() attempt 5 of 10: not in sync: resp=0xf4
avrdude: stk500_recv(): programmer is not responding
12 Arduino Uno on COM9
```

◆ Step 5: Infrared Remote Control Robot Car



Check again whether the circuit wiring is correct, pay attention to the positive and negative marks of each interface. Turn on the power switch.



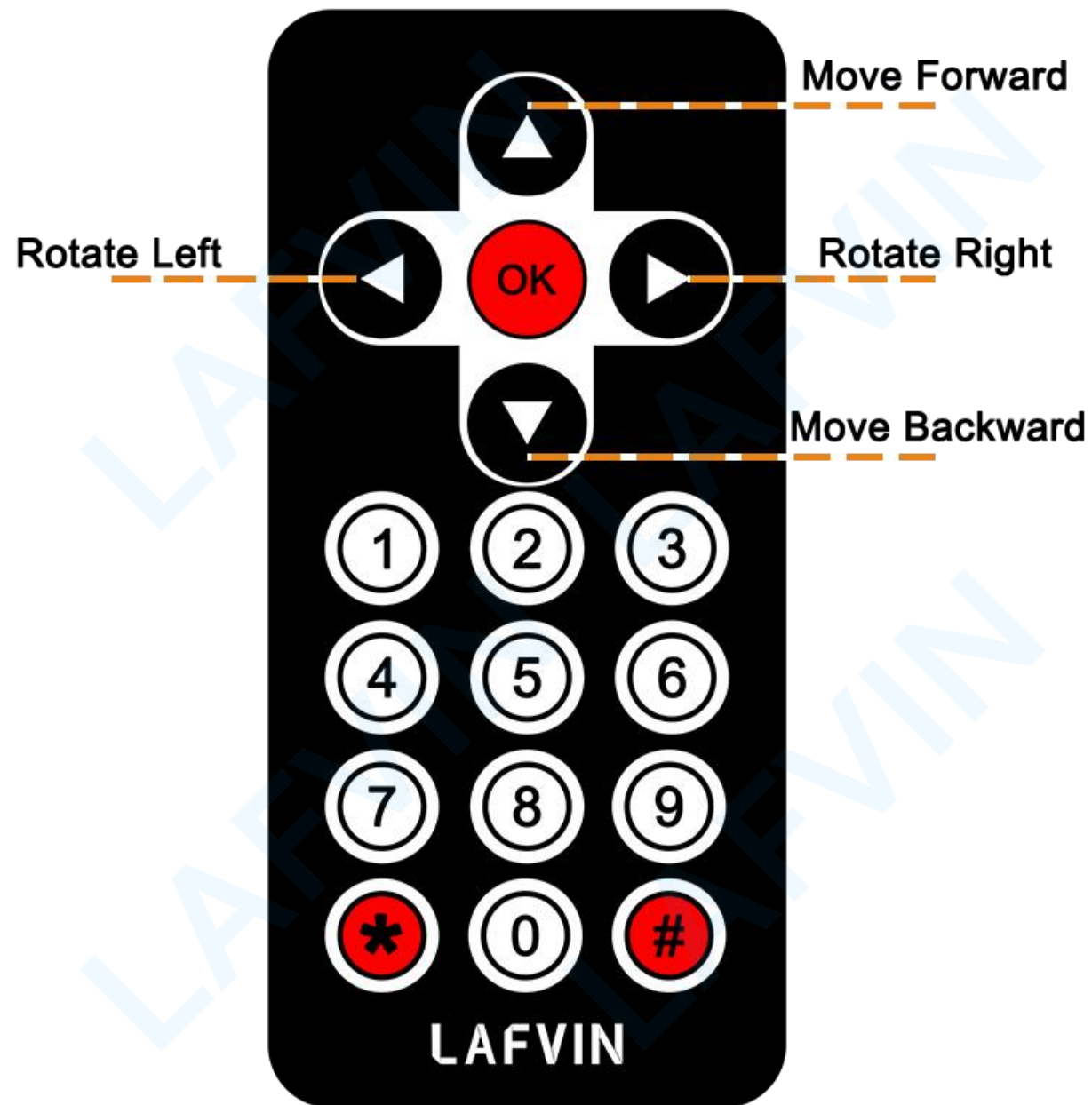
Use the infrared remote control function after successfully uploading the code of Arduino UNO.

Note:

- ① Due to air transportation requirements, maybe the infrared remote control does not contain the battery CR2025 3V. You need to prepare the battery yourself.
- ② The infrared remote control transmitter and receiver are in the same straight line, which will get better control effect.

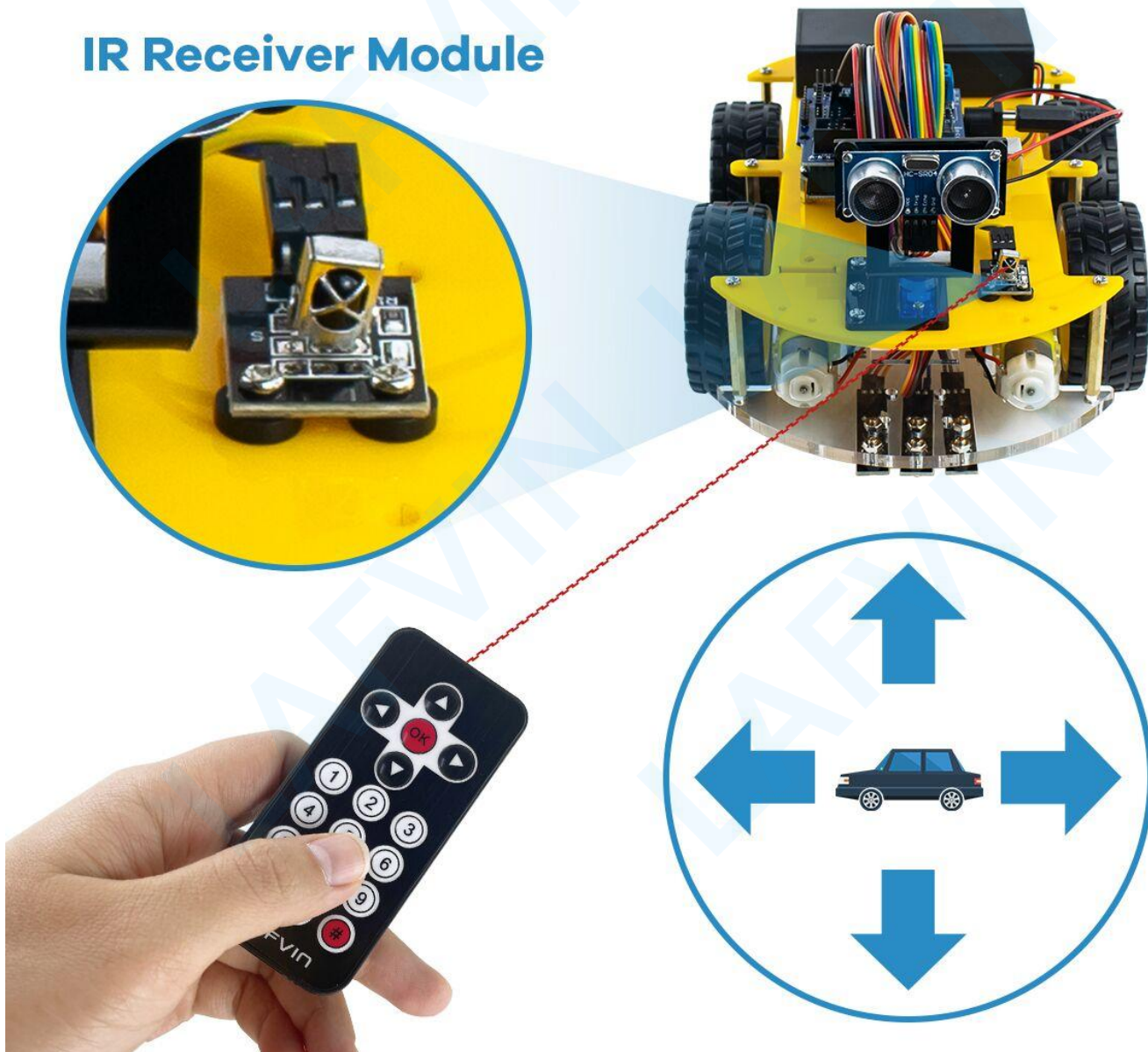


LAFVIN



IR Control

IR Receiver Module



The robot car is not moving? There may be the following reasons:

- ① It must be powered by batteries with a voltage $>7.4V$. Turn on the battery power switch. Only 5V USB power supply cannot drive the motor. Due to air transportation, the battery may not be included in the kit, you need to buy the battery yourself.
- ② When the infrared receiving module receives the instruction, the signal indicator light should flash, if not, the possible reason is that the wiring of the infrared receiving module is wrong.
- ③ The wiring of the motor driver L298N module is wrong or the wiring is loose.
- ④ The code upload failed or an incorrect code was uploaded
- ⑤ Upload the test code to check whether the motor drive circuit part is working normally?
Code path: ... \Test Code \Test_L298N_Motor.ino

LAFVIN

◆ Step 6: APP connects to the Bluetooth module



① Install the .apk file to your phone(Only supports Android system)



LAFVIN_4WD_Smart_Robot_Car_V1.apk

LAFVIN

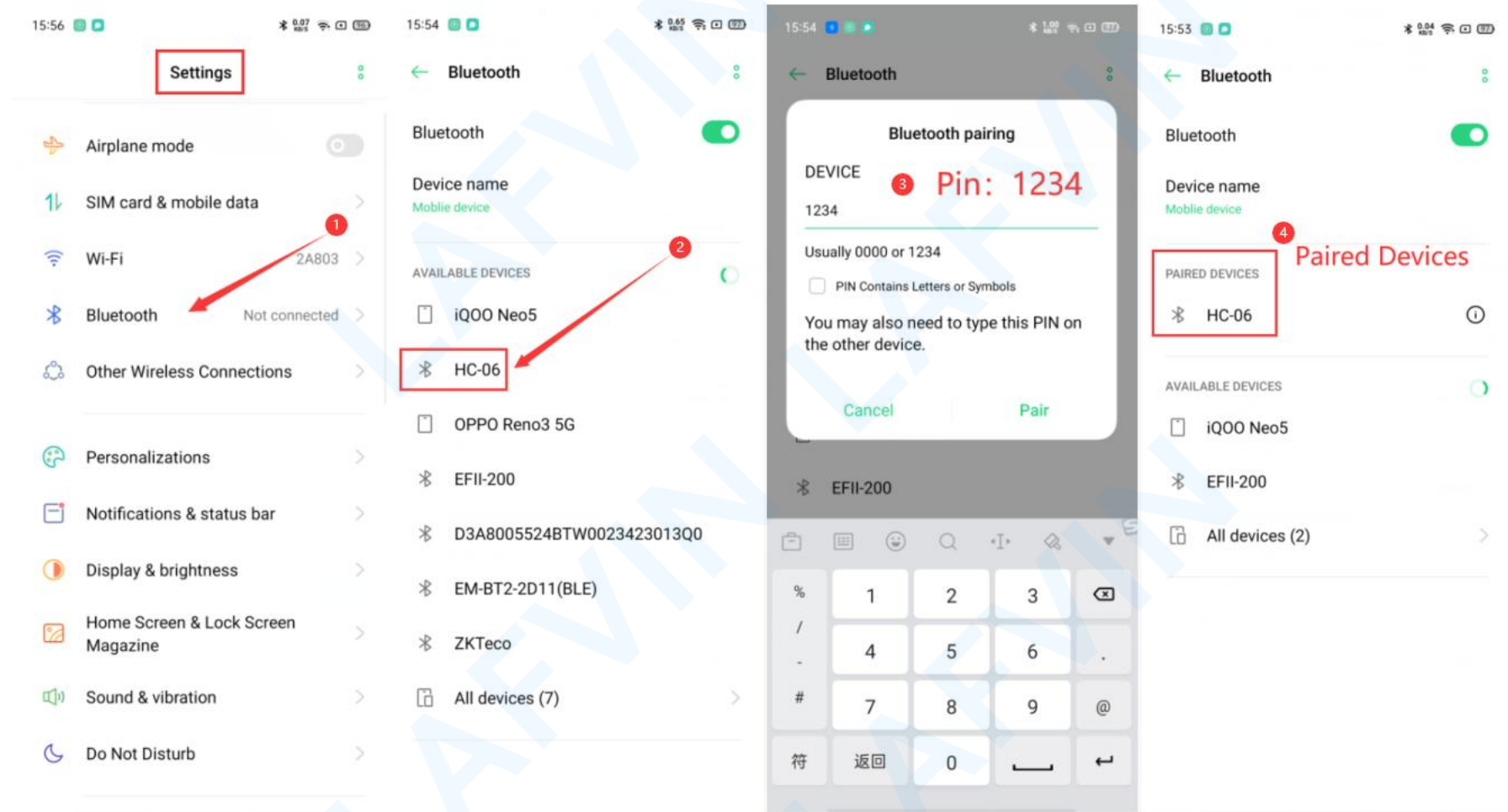


② **Allow** the app to find and connect to nearby devices

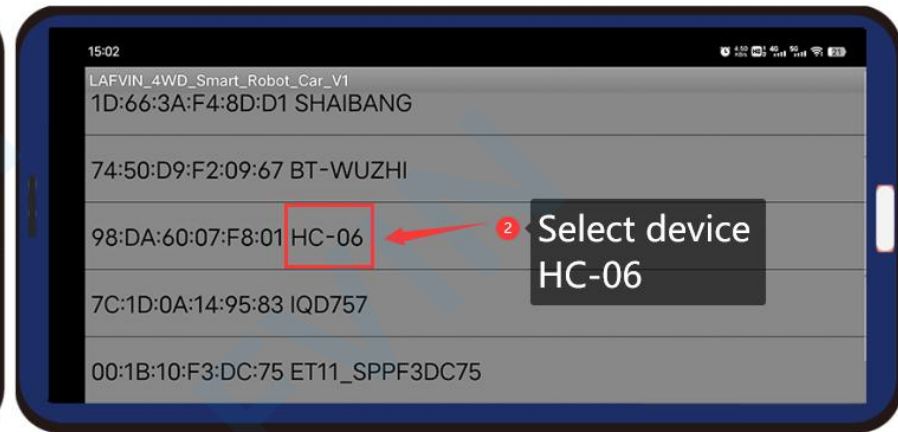
Tip: If the phone's Android version > 12 or Android SDK version > 31, the app needs to obtain permission to access nearby devices, otherwise the Bluetooth communication function cannot be used.



③The app connects to the HC-06 Bluetooth module



LAFVIN

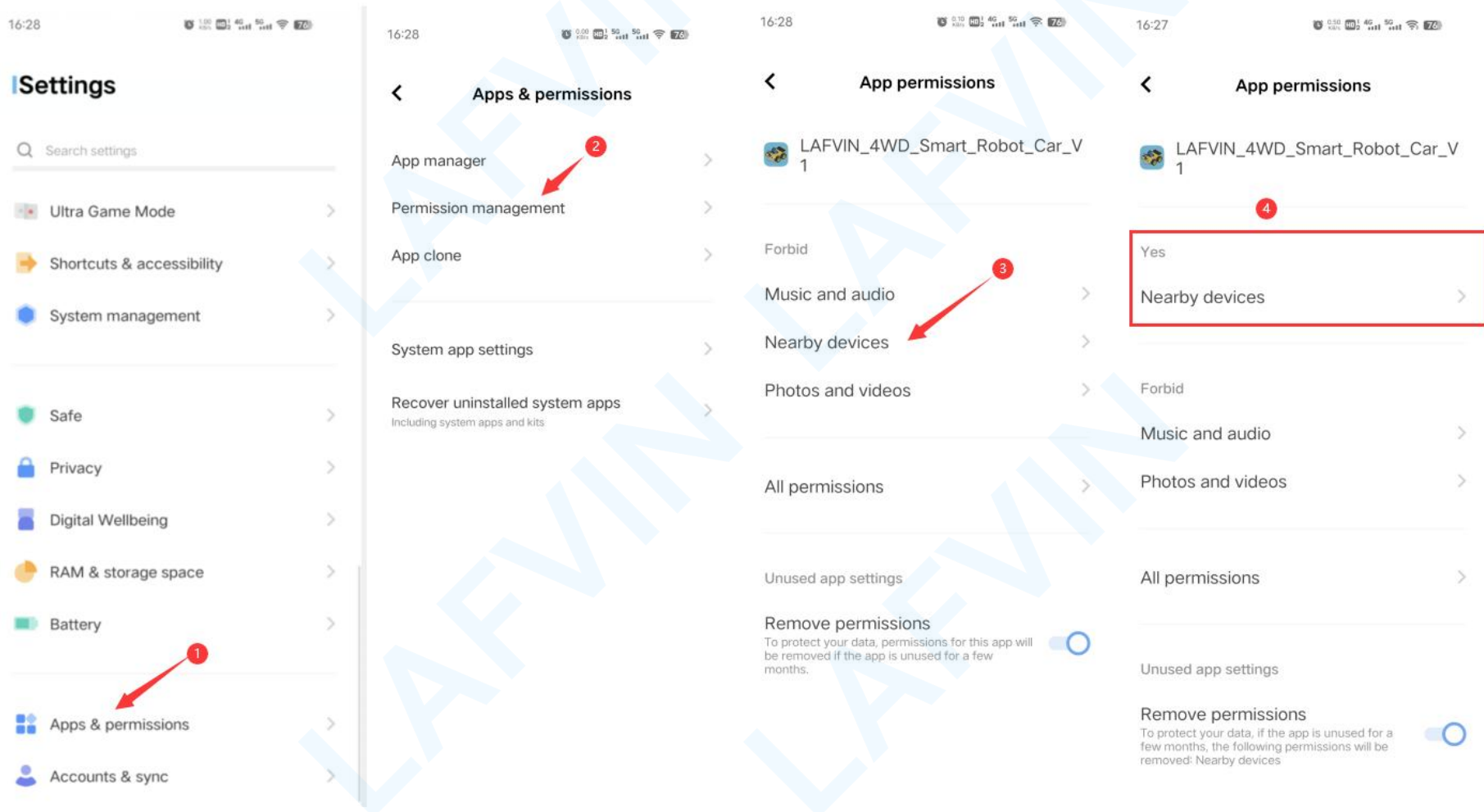


Tip: If it reports Error 507: Unable to connect. Is the device turned on?

The possible reason is that you have selected the wrong bluetooth device, or the HC-06 is not in available status. Try restarting the power supply, when the HC-06 module's onboard LED light flashes rapidly (Available status). [Repeat the above steps.](#)



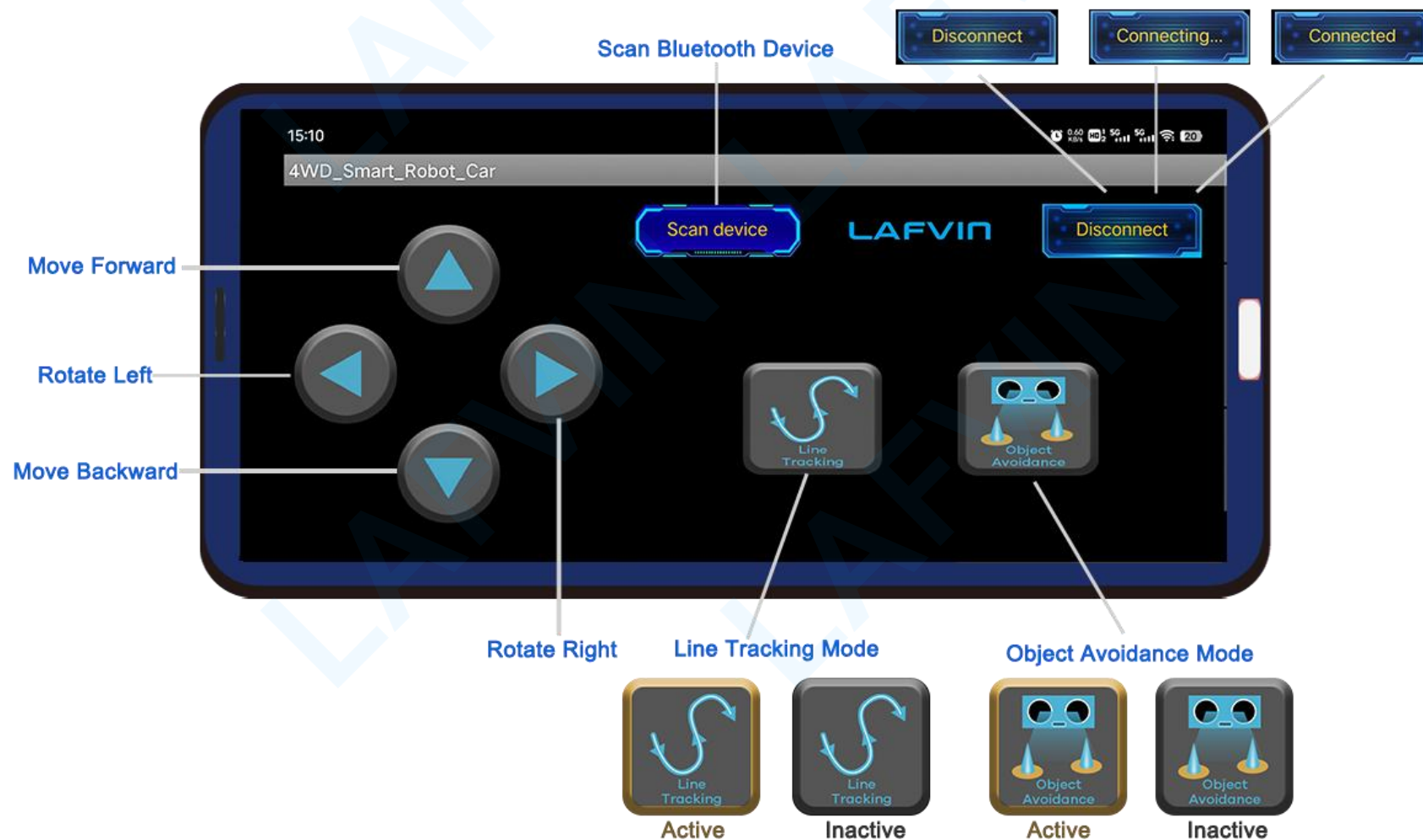
Tip: If you click to search for devices, the device list does not have any selectable devices. The app needs permission to access nearby devices.



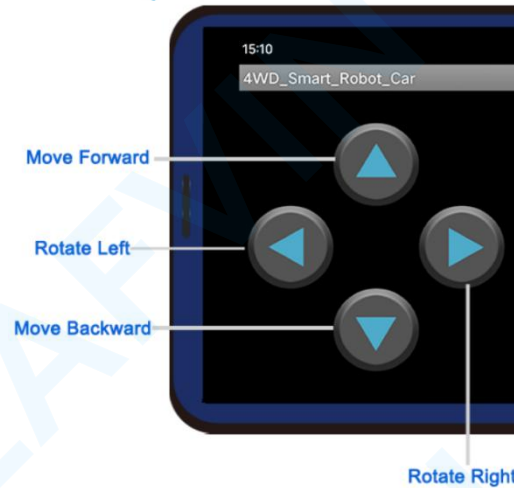
◆ Step 7: APP Remote Control Multi-function Mode



- Direction Control Mode
- Line-Tracking Mode
- Object-Avoidance Mode



➤ Direction Control Mode





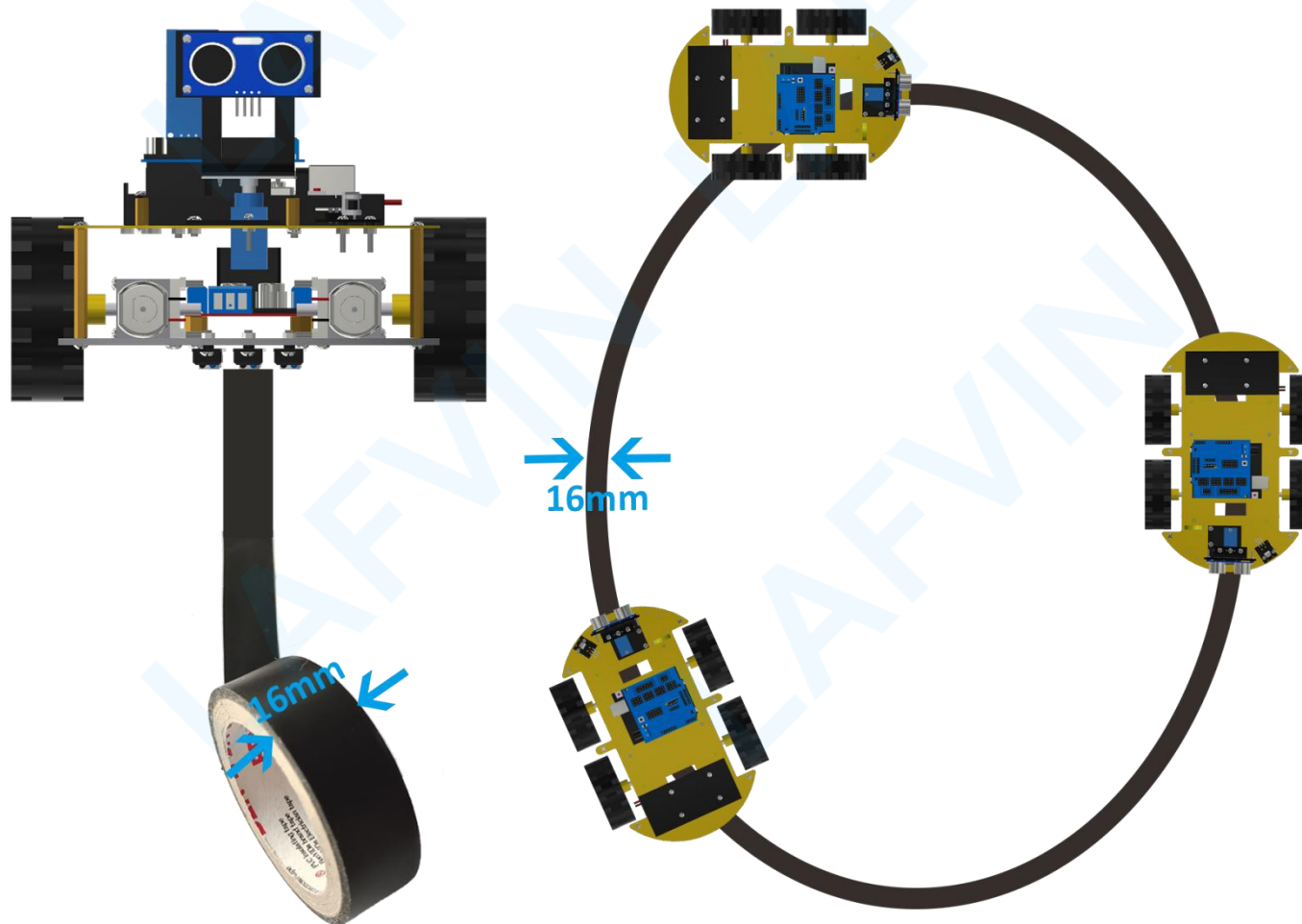
The robot car is not moving? There may be the following reasons:

- ① It must be powered by batteries with a voltage $>7.4V$. Turn on the battery power switch. Only 5V USB power supply cannot drive the motor. Due to air transportation, the battery may not be included in the kit, you need to buy the battery yourself.
- ② The wiring of the motor driver L298N module is wrong or the wiring is loose.
- ③ The code upload failed or an incorrect code was uploaded

➤ Line-Tracking Mode



Put the car on the manual track first. When the  key of the APP remote control is pressed, the car will activate the line-tracking mode. In this mode, the car will drive along the track. When pressing the  key of the APP remote control or other keys, it will exit the line-tracking mode.



Not working perfectly?

①First check if the wiring of the line-tracing sensor is correct?

S(Left)-->D7 S(center)-->D8 S(right)-->D9

②Then check Status LED Lights

Each sensor should work normally as follows:





white detected-->Status LED ON Black detected-->Status LED OFF

③Finally adjust the sensitivity

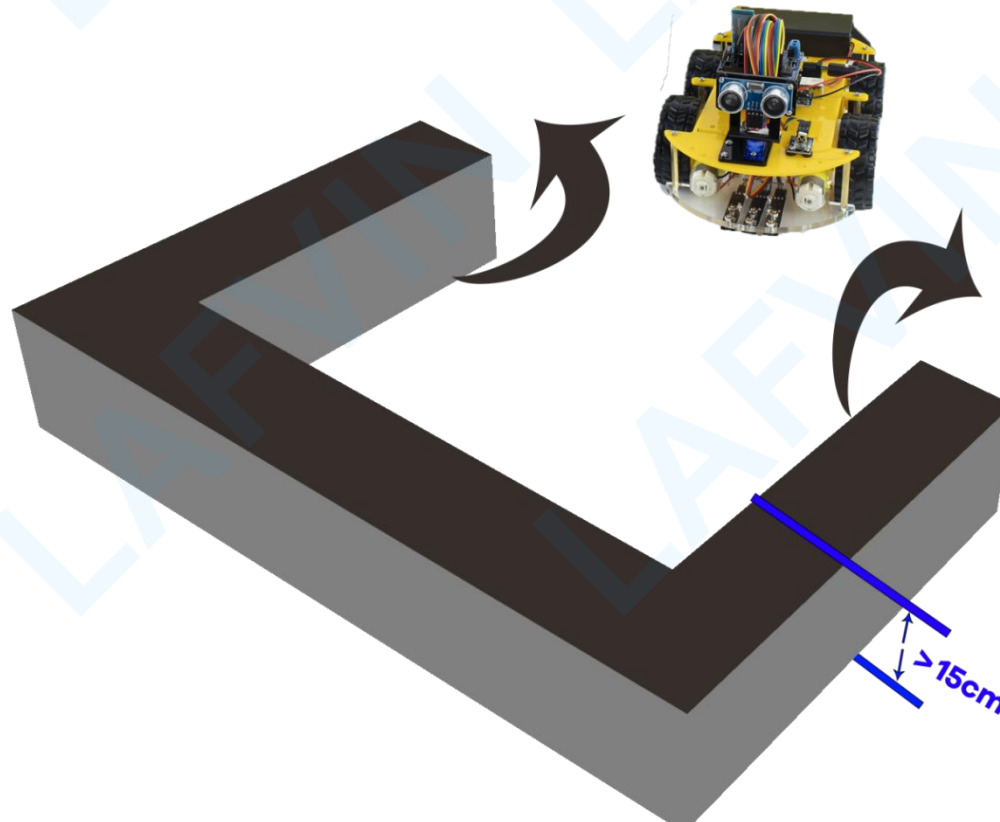
Since the sensors on the line-tracking module are greatly affected by the environment, if the car does not work well under the line-tracking module, you can use the potentiometer to adjust the sensor thresholds so that it can enable the car to operate at its optimum level.

➤ Object-Avoidance Mode

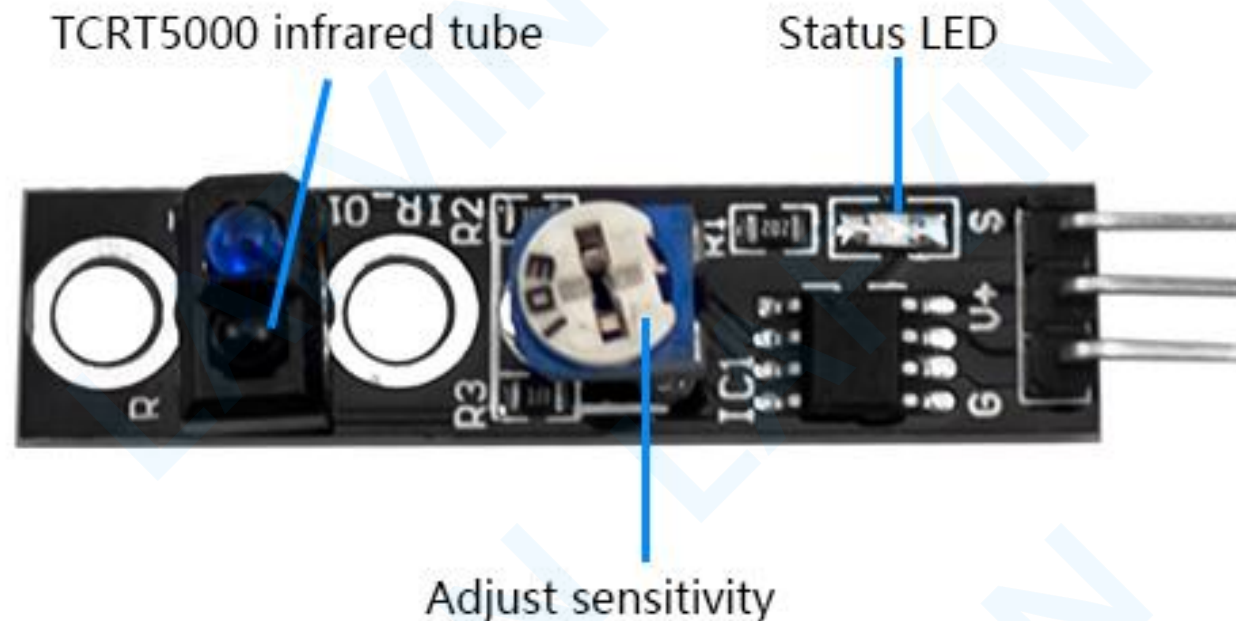


The car will enter the object-avoidance mode when pressing  key of the APP remote control. The car will move forward automatically until encountering obstacles. And it will automatically turn to the direction without obstacles to continue moving forwards when encountering obstacles. When pressing the  key of the APP remote control or other keys, it will exit the object-avoidance mode.

Tip: Can't work correctly? Try to restart the battery power. Or [test the ultrasonic sensor module separately.](#)



Line Tracking Sensor

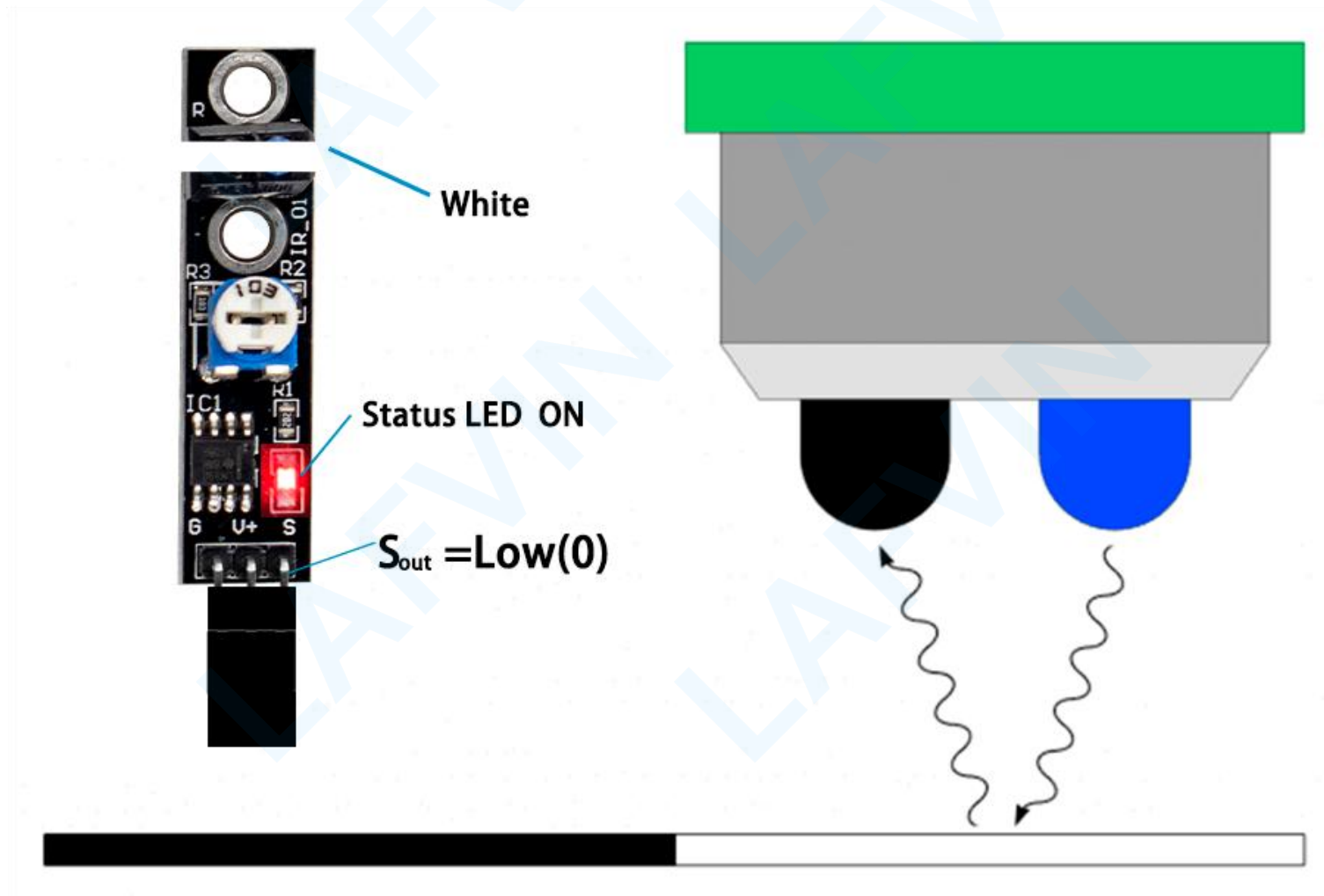


Line Tracking Sensor has three pins – one for ground (G), one for Vcc (V+), and one for the signal output (S). The blue LED is the IR transmitting LED, and the black LED is the IR receiving LED. Line Tracking Sensor has a range from 0.5 cm to 1.5 cm.

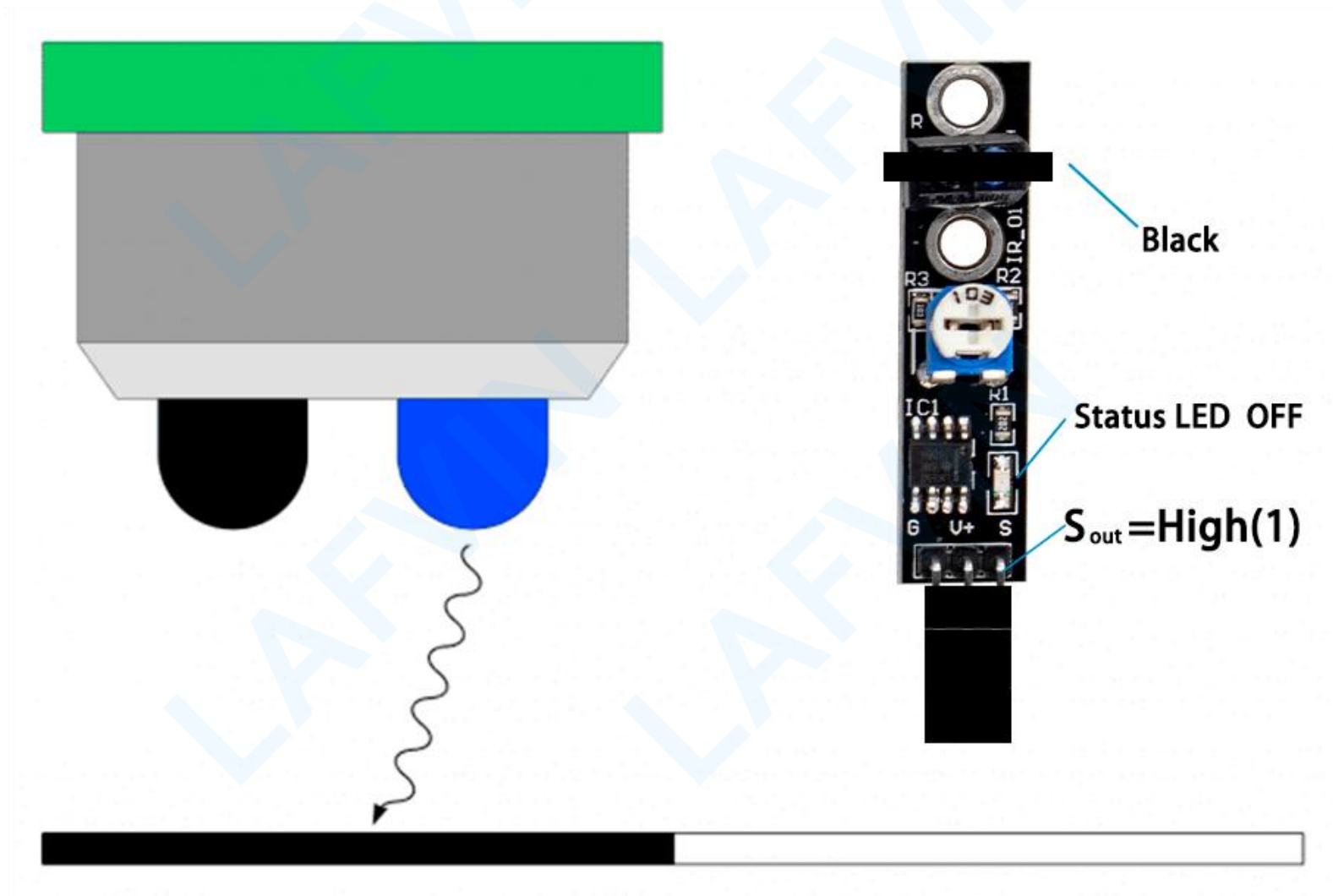
The IC on the board is an LM393 comparator, which converts the analog signal from the receiving LED into a digital signal that is sent to the sensor's signal output pin.

How Tracking Sensors Work

White surfaces reflect IR light and dark surfaces absorb IR light. When the tracking sensor is placed over a white surface, the IR light is reflected off of the surface. The reflected IR light is detected by the receiving LED.

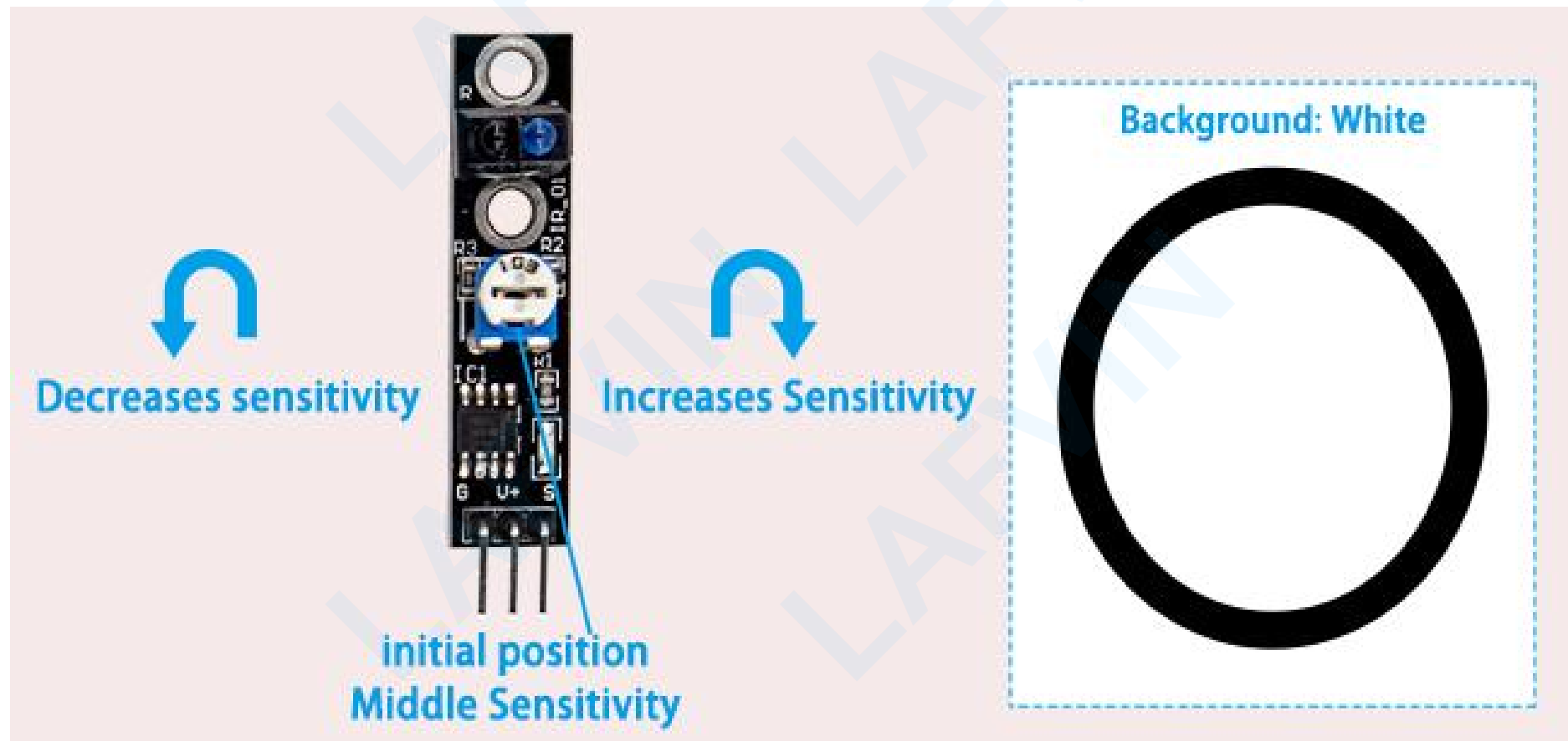


But if the tracking sensor is placed over a darkly colored surface, the IR light gets absorbed by the surface. So the receiving LED won't detect any IR light.



Adjust sensitivity

The initial position of the potentiometer is medium sensitivity, and the black line can achieve the best effect on the white background. If the background color is not white, you may need to increase the sensitivity for better results.

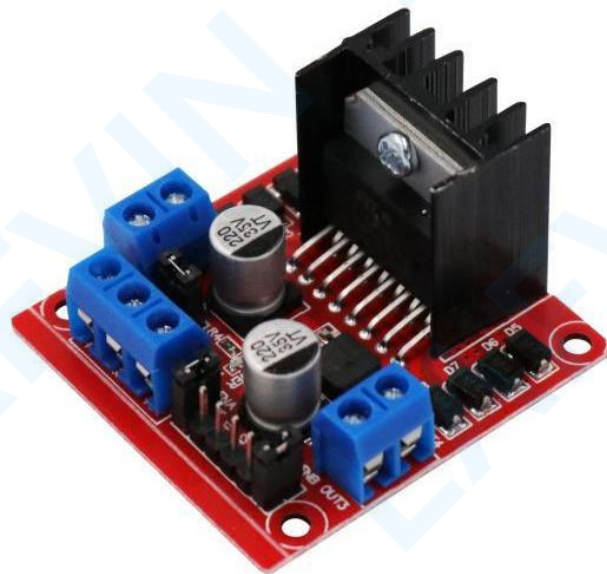


L298N Motor Driver

There are many ways to control a DC motor. The method we'll use here is suitable for most hobbyist motors, that require 6V or 12V to operate.

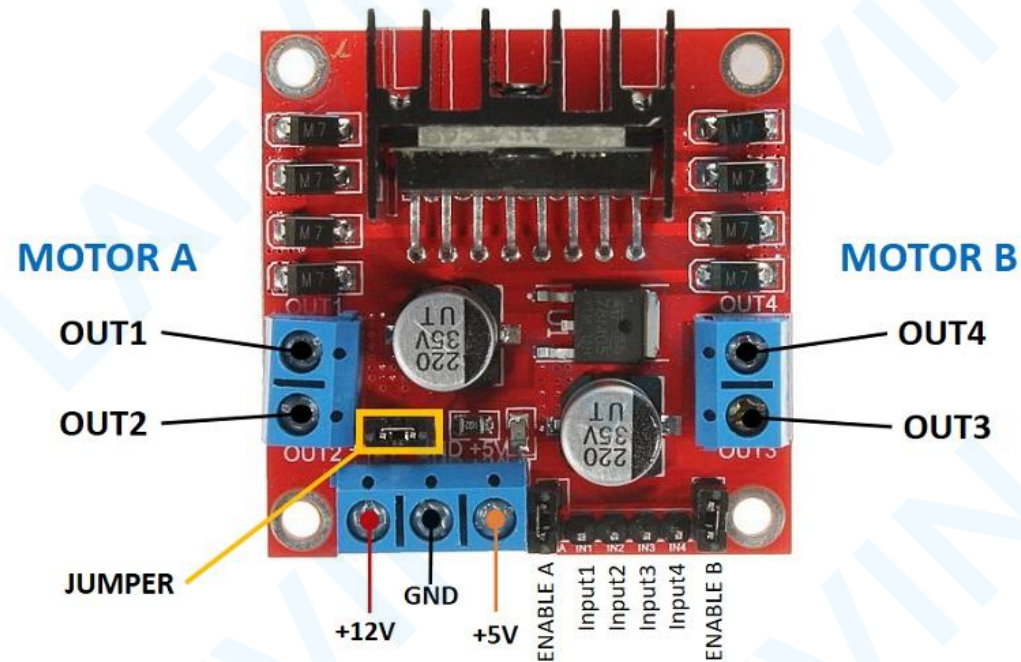
We're going to use the L298N motor driver that can handle up to 3A at 35V. Additionally, it allows us to drive two DC motors simultaneously, which is perfect to build a robot.

The L298N motor driver is shown in the following figure:



L298N Motor Driver pinout

Let's take a look at the L298N motor driver pinout and see how it works.



The motor driver has a two terminal block in each side for each motor. OUT1 and OUT2 at the left and OUT3 and OUT4 at the right.

- **OUT1:** DC motor A + terminal
- **OUT2:** DC motor A – terminal
- **OUT3:** DC motor B + terminal
- **OUT4:** DC motor B – terminal

At the bottom you have a three terminal block with **+12V**, **GND**, and **+5V**. The **+12V** terminal block is used to power up the motors. The **+5V** terminal is used to power up the L298N chip. However, if the jumper is in place, the chip is powered using the motor's power supply and you don't need to supply 5V through the **+5V** terminal.

Note: if you supply more than 12V, you need to remove the jumper and supply 5V to the +5V terminal.

It's important to note that despite the +12V terminal name, with the setup we'll use here (with the jumper in place) you can supply any voltage between 6V and 12V. In this tutorial will be using 4 AA 1.5V batteries that combined output approximately 6V, but you can use any other suitable power supply.

In summary:

- **+12V:** The +12V terminal is where you should connect your power supply
- **GND:** power supply GND
- **+5V:** provide 5V if jumper is removed. Acts as a 5V output if jumper is in place
- **Jumper:** jumper in place – uses the motors power supply to power up the chip. Jumper removed: you need to provide 5V to the +5V terminal. If you supply more than 12V, you should remove the jumper

At the bottom right you have four input pins and two enable terminals. The input pins are used to control the direction of your DC motors, and the enable pins are used to control the speed of each motor.

- **IN1:** Input 1 for Motor A
- **IN2:** Input 2 for Motor A
- **IN3:** Input 1 for Motor B
- **IN4:** Input 2 for Motor B
- **EN1:** Enable pin for Motor A
- **EN2:** Enable pin for Motor B

There are jumper caps on the enable pins by default. You need to remove those jumper caps to control the speed of your motors.

Control DC motors with the L298N

Now that you're familiar with the L298N Motor Driver, let's see how to use it to control your DC motors.

Enable pins

The enable pins are like an ON and OFF switch for your motors. For example:

- If you send a **HIGH signal** to the enable 1 pin, motor A is ready to be controlled and at the maximum speed;
- If you send a **LOW signal** to the enable 1 pin, motor A turns off;
- If you send a **PWM signal**, you can control the speed of the motor. The motor speed is proportional to the duty cycle. However, note that for small duty cycles, the motors might not spin, and make a continuous buzz sound.

SIGNAL ON THE ENABLE PIN	MOTOR STATE
HIGH	Motor enabled
LOW	Motor not enabled
PWM	Motor enabled: speed proportional to duty cycle

Input pins

The input pins control the direction the motors are spinning. Input 1 and input 2 control motor A, and input 3 and 4 control motor B.

- If you apply LOW to input1 and HIGH to input 2, the motor will spin forward;

- If you apply power the other way around: HIGH to input 1 and LOW to input 2, the motor will rotate backwards. Motor B can be controlled using the same method but applying HIGH or LOW to input 3 and input 4.

Controlling 2 DC Motors – ideal to build a robot

If you want to [build a robot car](#) using 2 DC motors, these should be rotating in specific directions to make the robot go left, right, forward or backwards.

For example, if you want your robot to move forward, both motors should be rotating forward. To make it go backwards, both should be rotating backwards.

To turn the robot in one direction, you need to spin the opposite motor faster. For example, to make the robot turn right, enable the motor at the left, and disable the motor at the right. The following table shows the input pins' state combinations for the robot directions.

DIRECTION	IN 1(D2)	IN 2(D4)	IN 3 (D7)	IN 4(D8)
Forward	0	1	0	1
Backward	1	0	1	0
Rotate_Right	1	0	0	1
Rotate_Left	0	1	1	0
Stop	1	1	1	1

Ultrasonic Sensor Module

If the obstacle avoidance function cannot work correctly, you can test the ultrasonic sensor module separately. (In order not to be affected by other factors, you need to disconnect the connection of other modules. For example, infrared module, line tracking module, L298N motor drive module)

Description

The HC-SR04 ultrasonic sensor uses sonar to determine the distance to an object. This sensor reads from 2cm to 400cm (0.8inch to 157inch) with an accuracy of 0.3cm (0.1inches), which is good for most hobbyist projects. In addition, this particular module comes with ultrasonic transmitter and receiver modules.

The following picture shows the HC-SR04 ultrasonic sensor



Features

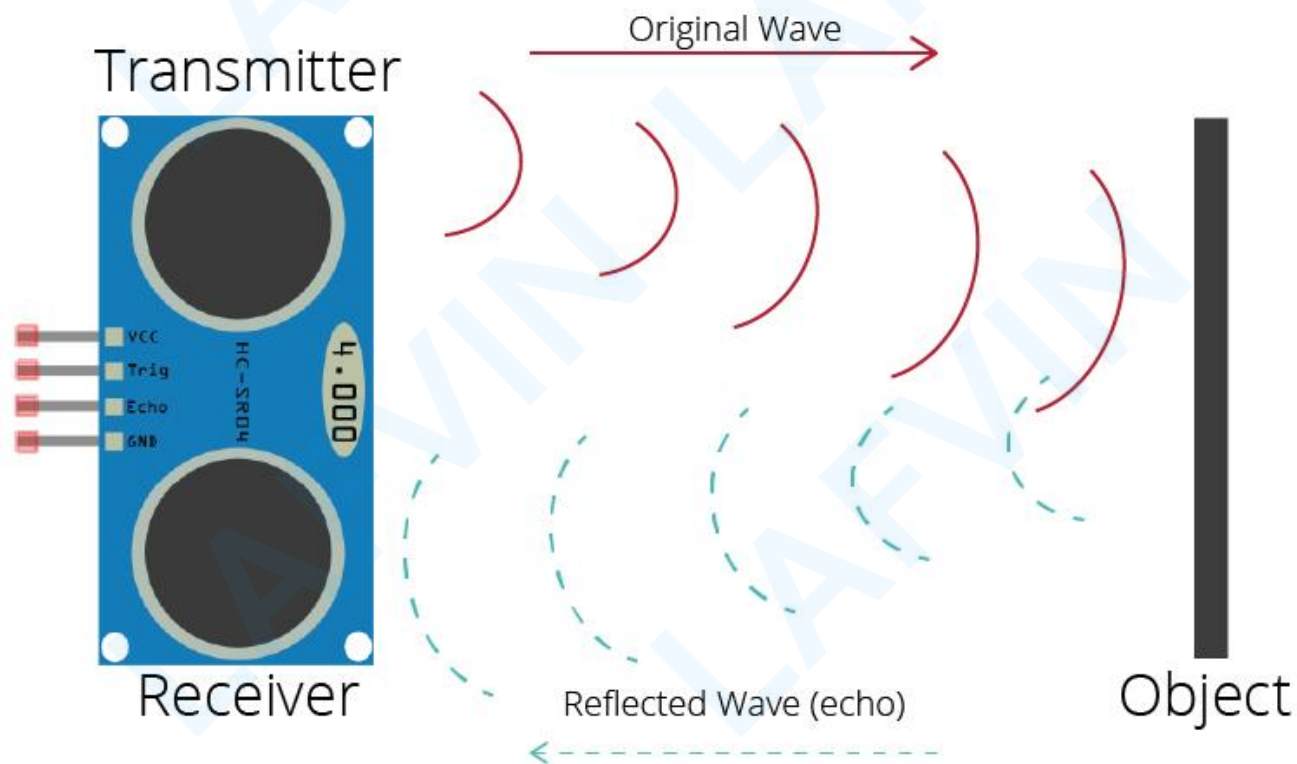
Here's a list of some of the HC-SR04 ultrasonic sensor features and specs:

- Power Supply :+5V DC
- Quiescent Current : <2mA
- Working Current: 15mA
- Effectual Angle: <15°
- Ranging Distance : 2cm – 400 cm/1" – 13ft
- Resolution : 0.3 cm
- Measuring Angle: 30 degree
- Trigger Input Pulse width: 10uS TTL pulse
- Echo Output Signal: TTL pulse proportional to the distance range
- Dimension: 45mm x 20mm x 15mm

How Does it Work?

The ultrasonic sensor uses sonar to determine the distance to an object. Here's what happens:

1. The ultrasound transmitter (trig pin) emits a high-frequency sound (40 kHz).
2. The sound travels through the air. If it finds an object, it bounces back to the module.
3. The ultrasound receiver (echo pin) receives the reflected sound (echo).



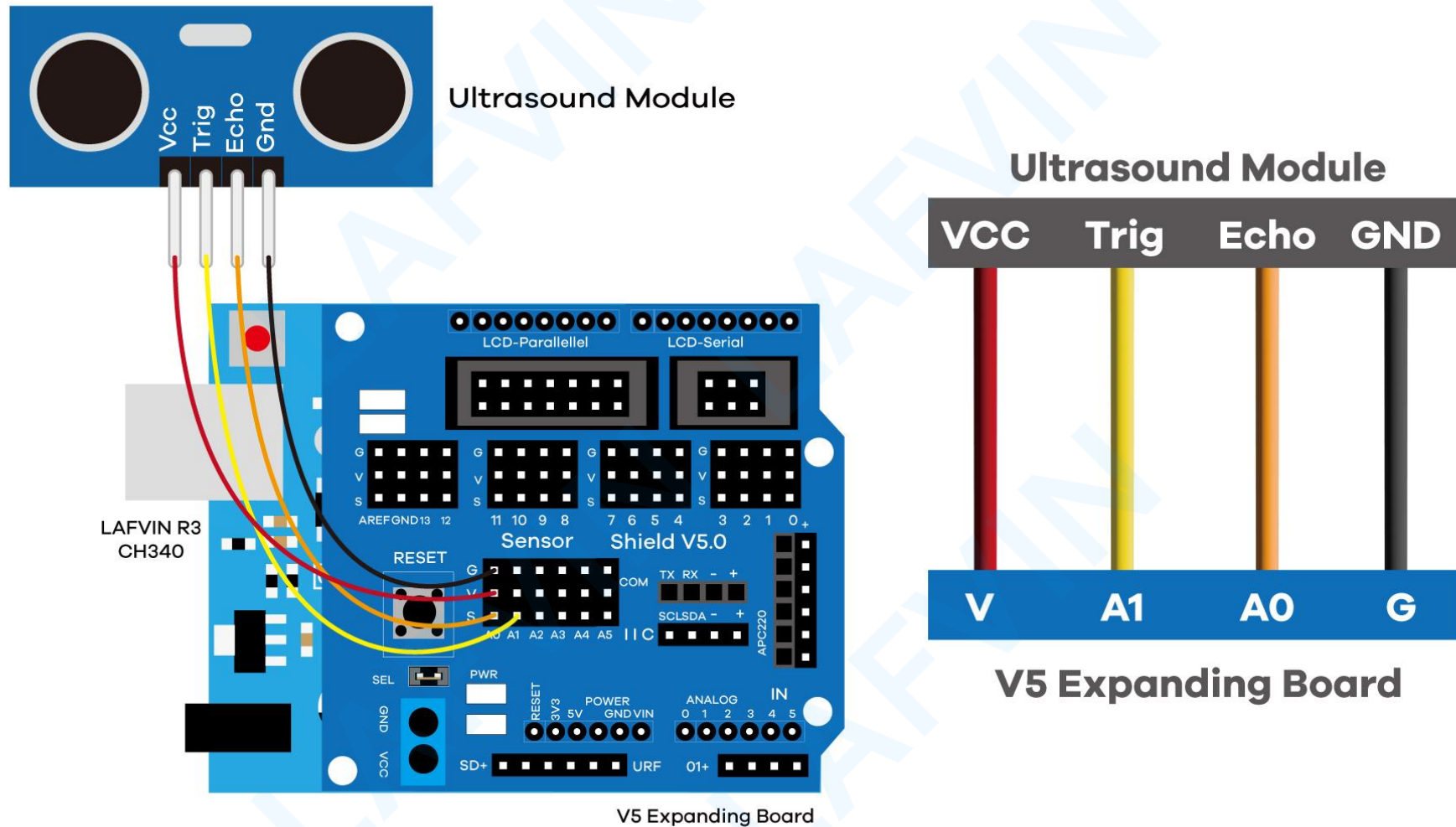
The time between the transmission and reception of the signal allows us to calculate the distance to an object. This is possible because we know the sound's velocity in the air. Here's the formula:

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

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

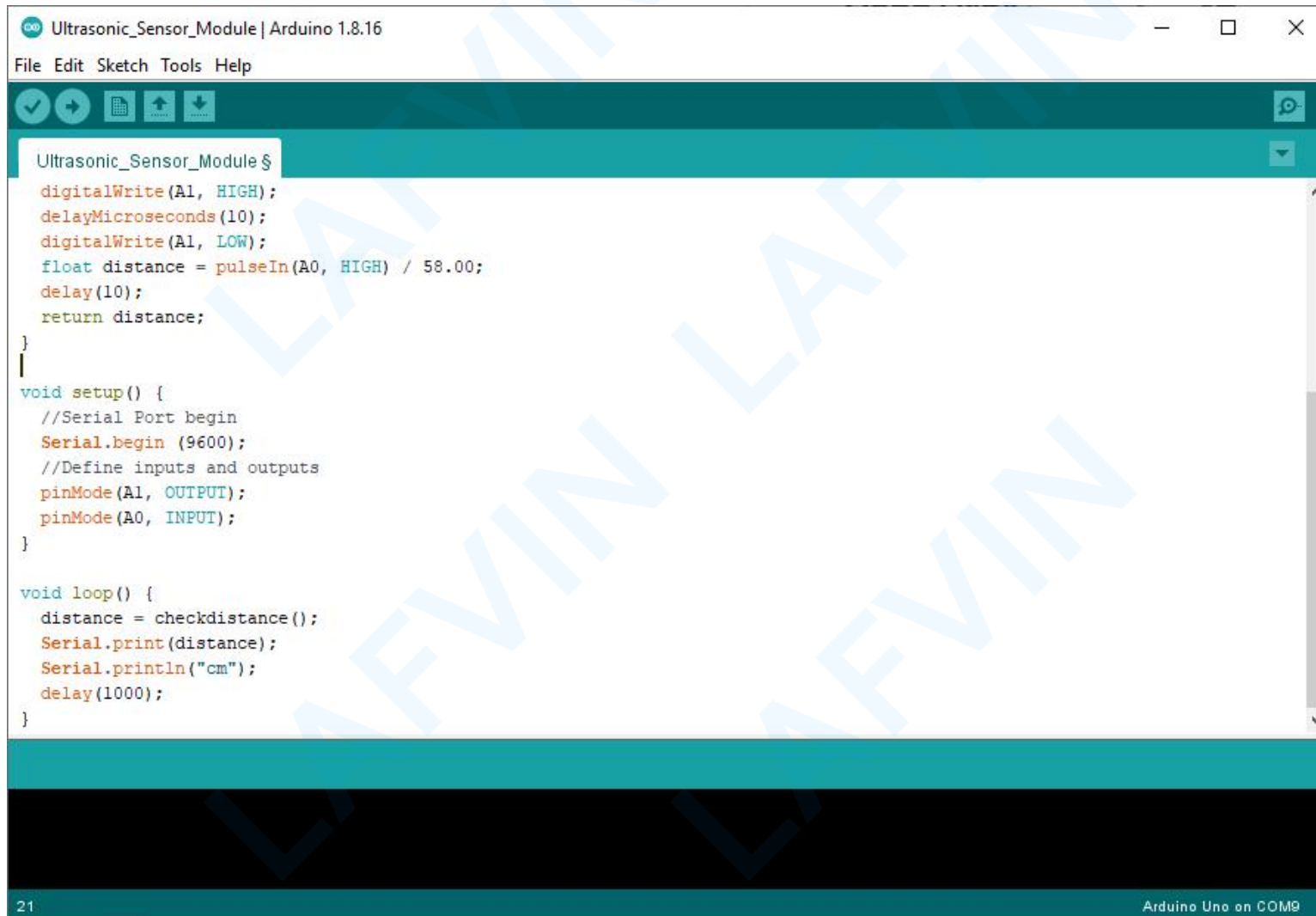
speed of sound in the air at 20°C (68°F) = **343m/s**

Arduino Shield V5.0 with HC-SR04 Sensor Wiring



Code

Click the **File>>Open** icon to open **Test Code\Ultrasonic_Sensor_Module.ino** file.




The screenshot shows the Arduino IDE interface with the file 'Ultrasonic_Sensor_Module.ino' open. The code is as follows:

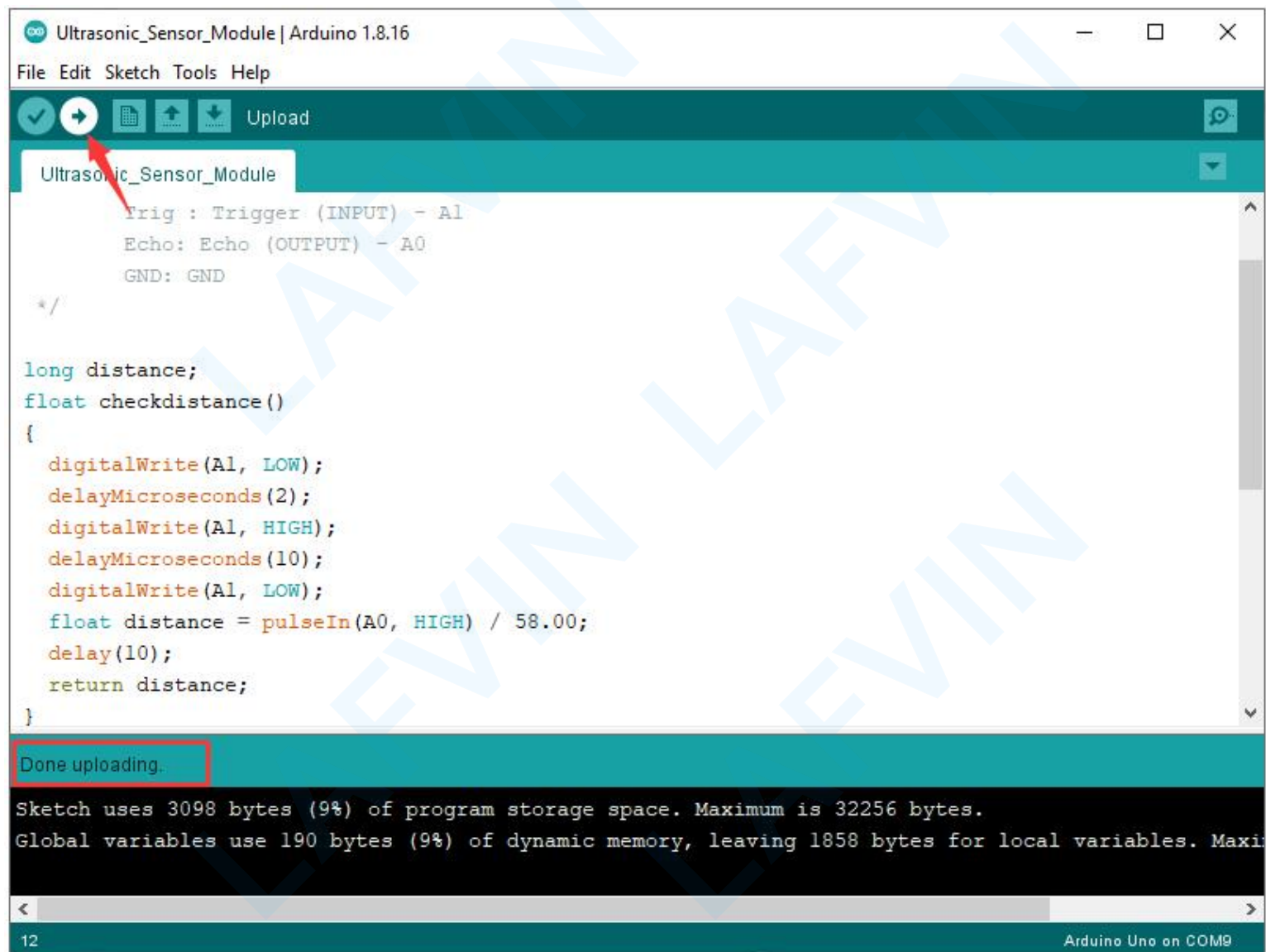
```
Ultrasonic_Sensor_Module $
digitalWrite(A1, HIGH);
delayMicroseconds(10);
digitalWrite(A1, LOW);
float distance = pulseIn(A0, HIGH) / 58.00;
delay(10);
return distance;
}

void setup() {
  //Serial Port begin
  Serial.begin (9600);
  //Define inputs and outputs
  pinMode(A1, OUTPUT);
  pinMode(A0, INPUT);
}

void loop() {
  distance = checkdistance();
  Serial.print(distance);
  Serial.println("cm");
  delay(1000);
}
```

The status bar at the bottom indicates '21' and 'Arduino Uno on COM9'.

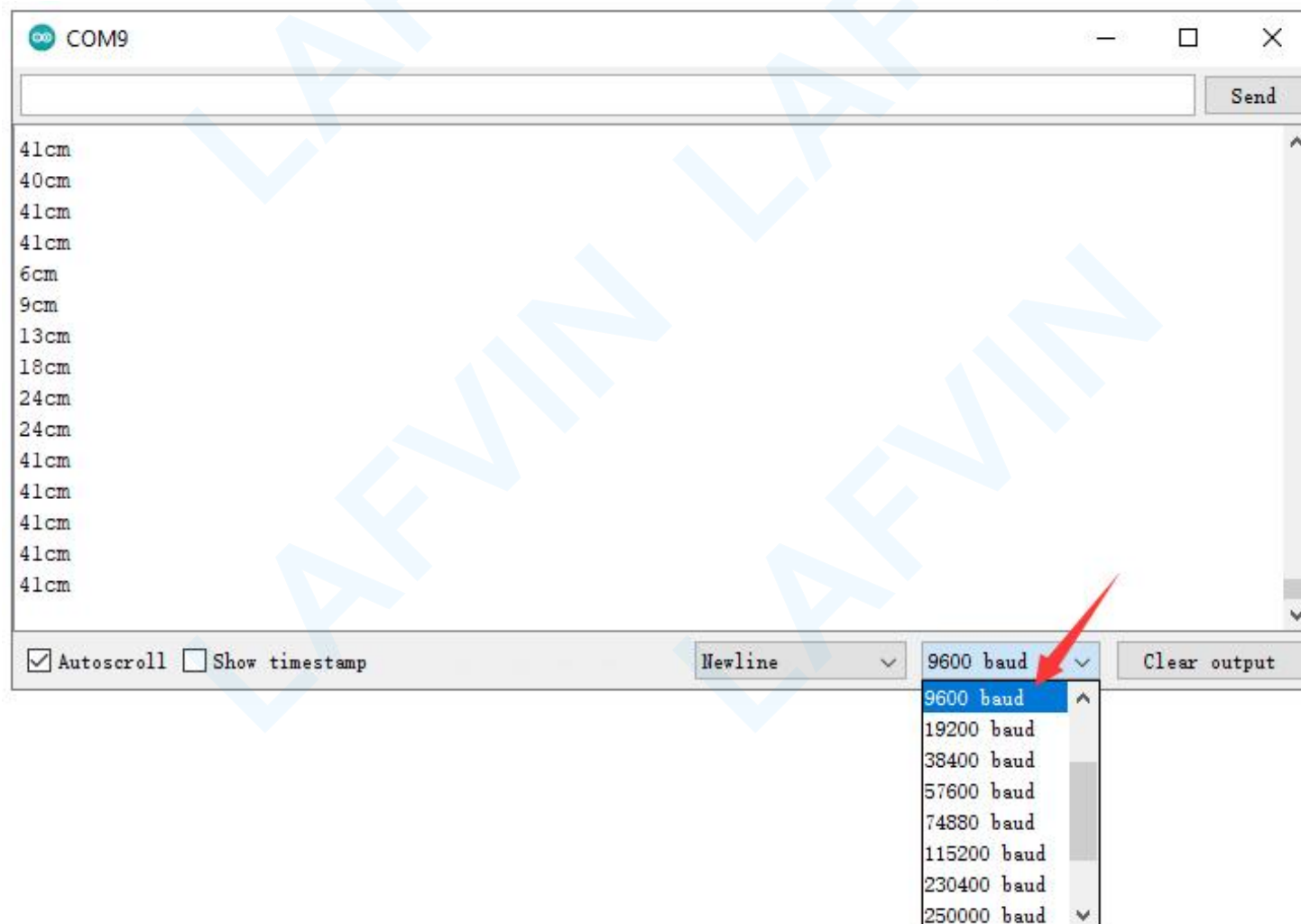
Click , Upload the sketch to Arduino UNO.



Then Click , open the Serial Monitor at a baud rate of 9600.



The distance to the nearest object is printed in the Serial Monitor window. When the object moves, the distance of the monitor printed will change.



Tip: If the printing value is 0cm, the reason for the error may be:

① Wrong wiring.

② The voltage provided by the USB data cable is insufficient, and you need to use the battery to supply power at the same time.

