MotorDriverBoard Graphic programming tutorial_V.4.0

版本修订

日期	版本号	描述	作者
2019-9-20	V.1.0	create documents	Twisted
2019-10-31	V.2.0	Modify sample program screenshot	Twisted
2019-11-11	V.3.0	Modify document pictures and add functions	Twisted
2019-11-30	V.4.0	Modify document picture, add function, add V3, V4, V5 initialization version selection	Twisted

Chapter 1 understanding the motordriverboard and its programming environment

1.1 Summary

The motordriverboard motor drive board can drive 4-way DC motor, 2-way coding motor, 2-way stepping motor, 6-way steering gear (two can be connected with external power supply), and the drive current can reach 2A. The driver board is specially designed for Arduino uno R3 / mega2560 motherboard, which can be directly connected to Arduino uno / mega2560. The motherboard integrates a passive buzzer, two RGB LED lights, and one infrared receiver. In addition, PS2 socket, UART interface, I2C interface, ultrasonic obstacle avoidance module socket and other sensor interfaces are reserved, which is very convenient to connect various sensor modules.

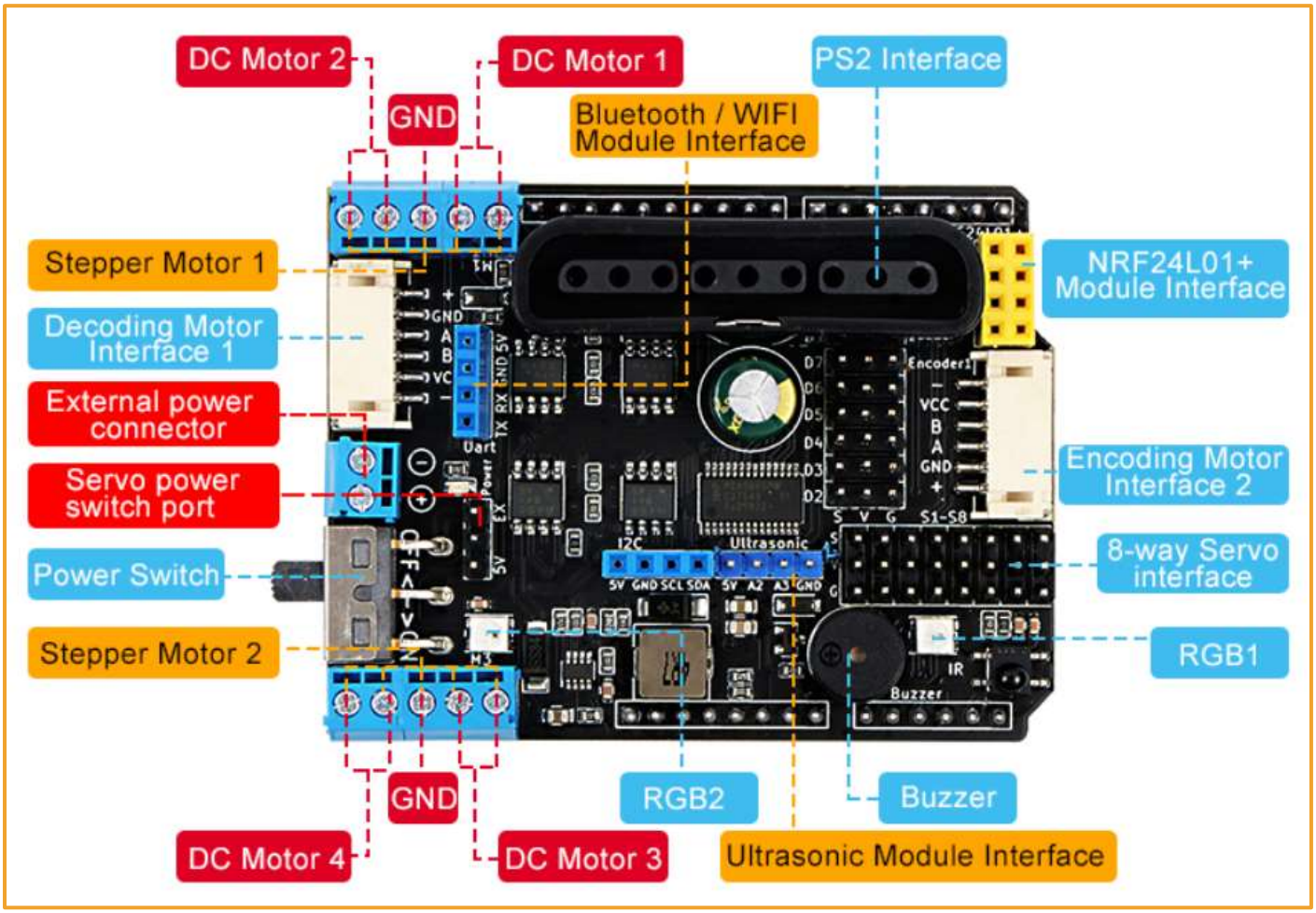


Figure 1-1-1

1.2 MotorDriverBoard Composition frame

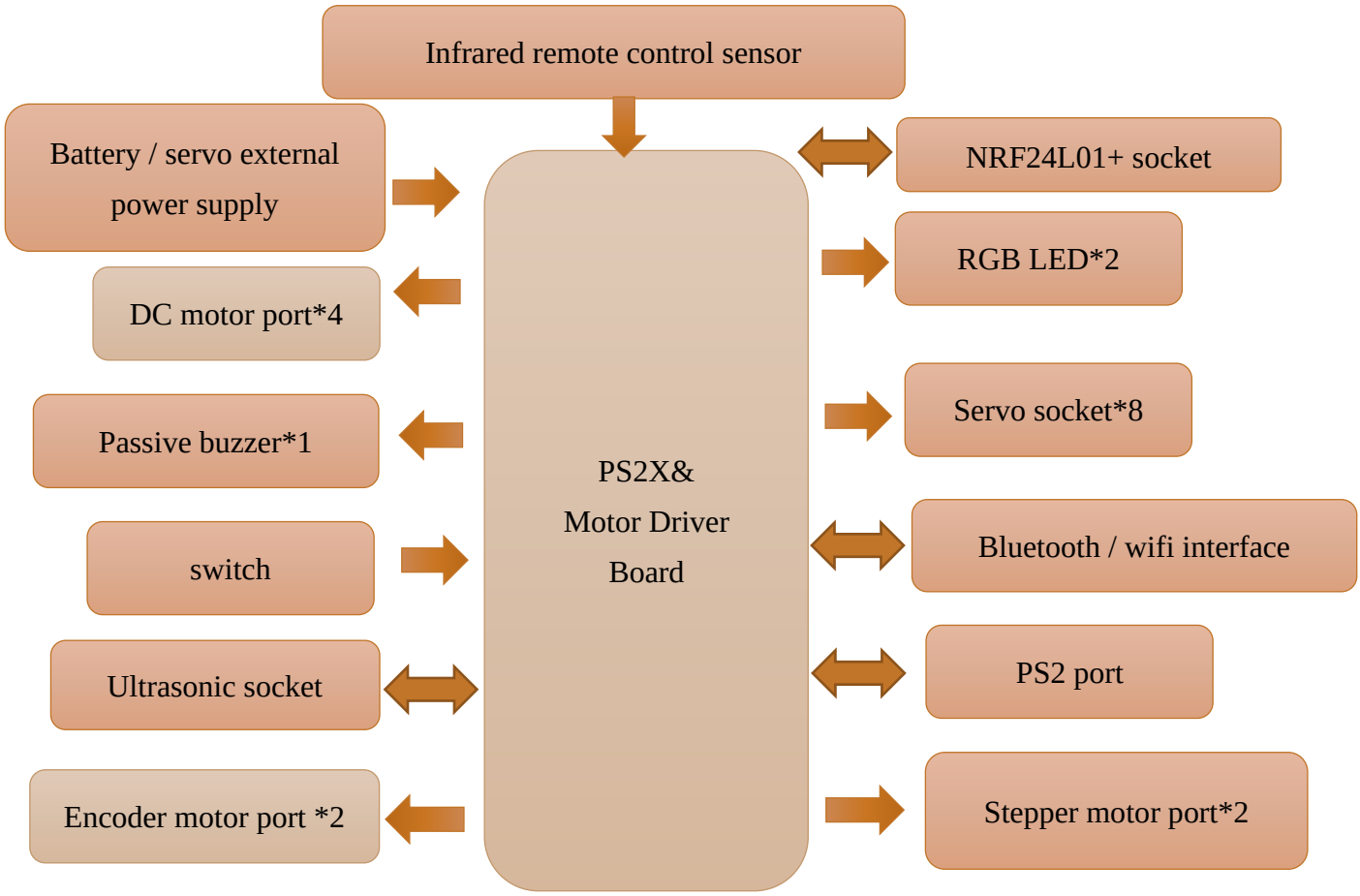


Figure 1-2-1

1.3 Common problem

Q: How to power the motor driver board?

A: It can be directly powered through the DC port of Arduino uno, with a voltage range of 6-12V.

Q: How many motors can the motor driver board drive?

A: The motor driver board can drive 4 DC motors or 2 stepping motors and 2 coding motors.

Q: How does the ps2x & motor driver board connect external power to drive the steering gear?

A: The 8 steering engines of the motor driver board support external power supply to drive the steering engine. The external power supply interface of the steering engine is connected with the power supply, and the skipping pin cap is connected with the ex. the external power supply of the steering engine is good.

Q： I want to upload the sequence to Arduino board, and prompt that the upload fails. What is the reason?

A: Before uploading the sample program to the Arduino board, you need to check whether the board and the computer are connected properly, then install the driver and try again.

Q： I want to upload the sequence to Arduino board, and then turn on the power and the motor does not work. What is the reason?

A: First, check whether the green indicator light on the arduinouno board is on. If it is not on, it means that the power supply is abnormal. Then check whether the battery voltage is above 6V. Then check whether the connection port of the motor is consistent with the port set in the program. After confirming that it is correct, restart the machine.

Q： My Arduino will crash when the motor is running. Is the driver broken?

A: When the motor is running, the power consumption is large, so it needs to ensure that the battery has sufficient power. You can try to charge the battery first, and then start it.

1.4 MotorDriverBoard Graphic programming software： Mixly

1.4.1 Software and driver installation

Windows Installation steps of Mixly software

Figure 1-4-1-1

After downloading and decompressing, double-click mixly.exe to open the mixly software

Figure 1-4-1-2

Figure 1-4-1-3

Mac Installation steps of Mixly software

1. Installation JDK

   Mac version mainly runs in Java environment, so when installing, you need to install JDK. In this disk, you can directly install jdk-8u71-mac OS x-x64.dmg file, or select the corresponding JDK on

Oracle official website according to your own system.

2. Download the misezzi software compression package

   At present, the official website only supports Baidu online disk to download the Mac version of mischiqi software. Without Baidu online disk, you can also search for the Mac version of mischiqi to download. After downloading, unzip it and run the mixly.jar file to open the mischiqi software on the Mac. If the software fails to run, please select the "general" page in "preferences" - Security and privacy "to allow the software to be used.

3. Install the drive

   If the mixly software is installed, we have written the code. If we want to run on the device, we need to install the serial driver. The serial driver is ch341ser_mac for downloading files. The software can be installed directly. After the installation is successful, "about this machine" - overview page - "system report" - USB ", we see the new device. The basic installation is finished here.

Figure 1-4-1-4

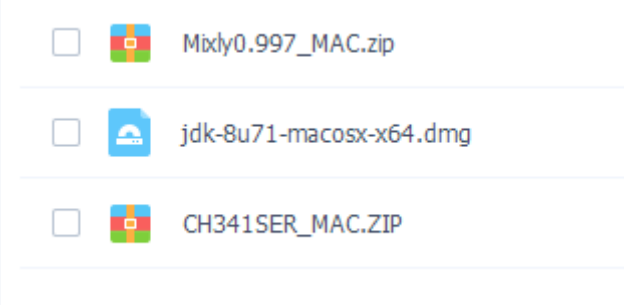


Figure 1-4-1-5

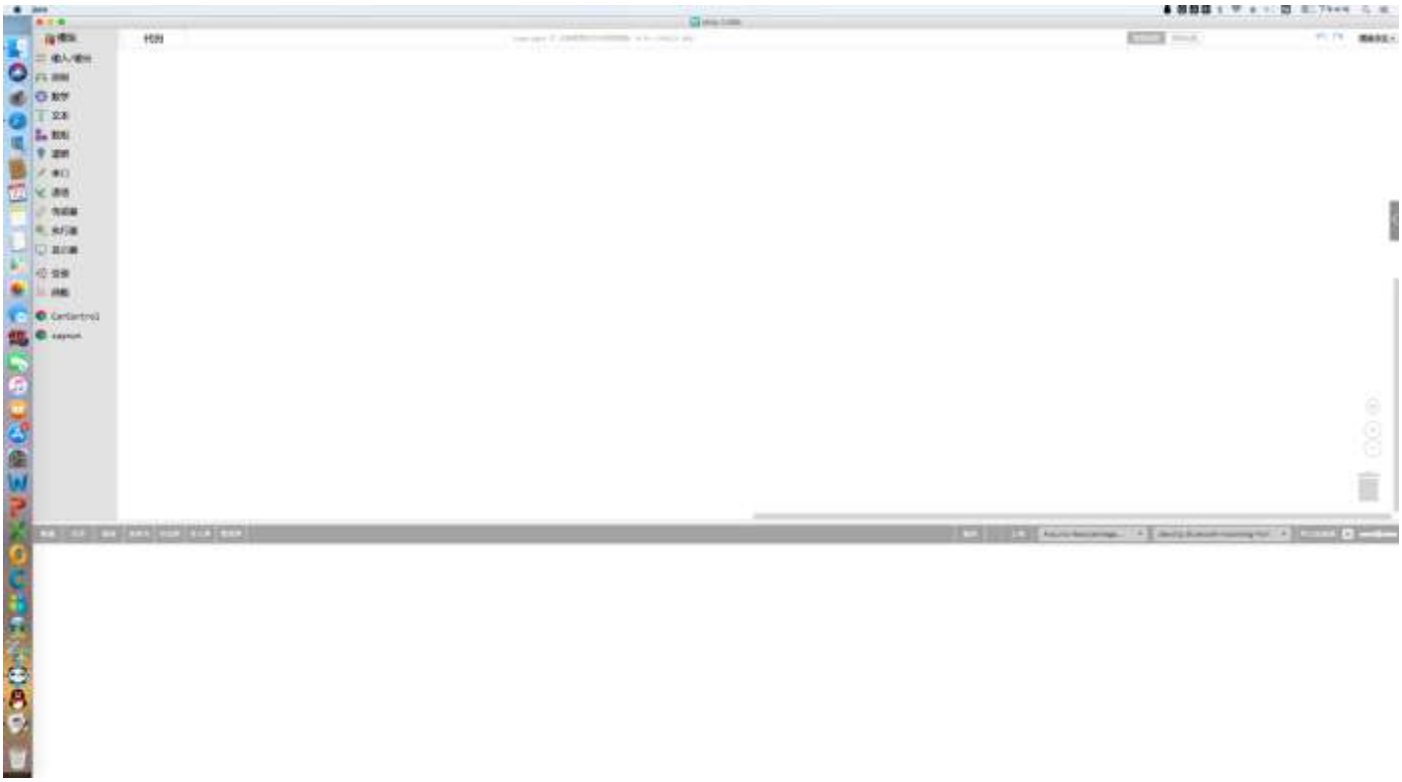


Figure 1-4-1-6

1.4.2 Introduction to compiling environment

The software part of the motordriverboard is programmed in the graphical programming software mixly. With this software, we can write various commands to the expansion board that we want it to execute, so as to control it! The software interface of mixly is shown in figure 1-4-2-1.

Figure 1-4-2-1

Basic module area and library module area: blocks of the same type are divided into the same module and given the same color. Each block represents a control command.

Function area: for the creation and saving of project files, the import, export and management of the library, the serial port connection and upload program, and the area selected by the control board for operation.

Program building area: it is the place to place the building blocks dragged from the basic module area and library module area.

Code area: after dragging out the building blocks, click "code" to see the C language code corresponding to the module you drag.

Language switching area: Chinese simplified, Chinese traditional, English and Spanish can be switched.

Zooming control area: the software operation interface can be zoomed.

Waste bin: used to remove unnecessary building blocks.

1.4.3 Introduction to compiling environment introduction to basic module area and library module area

The basic module area and library module area of mixly are divided into some building blocks brought by mixly software and the library building blocks imported by itself. These building blocks can correspond to C language codes one by one. We can write our own favorite programs by splicing different building blocks. We don't even need to know what codes each building block corresponds to, because when we drag out the building blocks, the program will be written by us. We can see these C language codes by clicking the "code" field. Figure 1-4-3-1 and 1-4-3-2 are module and code parts respectively.

Figure 1-4-3-1

Figure 1-4-3-2

1.4.4 Introduction to building block area

There are many types of building blocks under the script label of mixly, including input / output, control, mathematics, text, array, serial port, communication, etc. if you are interested, you can try to practice it yourself. We will not elaborate here. We mainly understand the four types of building blocks: control, mathematics, serial port and logic.

1) The control building blocks are the building blocks that control the execution process of the program. The main program, such as Figure 1-4-4-1。

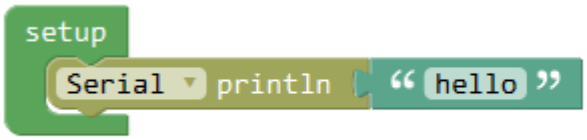


Figure 1-4-4-1

Figure 1-4-4-2

The setup function in the code corresponding to the initialization building blocks. Dragging some building blocks into the initialization building blocks means that the dragged building block programs will run once. If you drag the building blocks outside the initialization building blocks, these programs will enter the loop function and execute these programs circularly. As shown in figure 1-6-2

2) The main function of building blocks of number and logic operation type is to do mathematical operation as the condition of judgment, and compare the size and logic judgment with, or, and not, as shown in figure 1-4-4-3

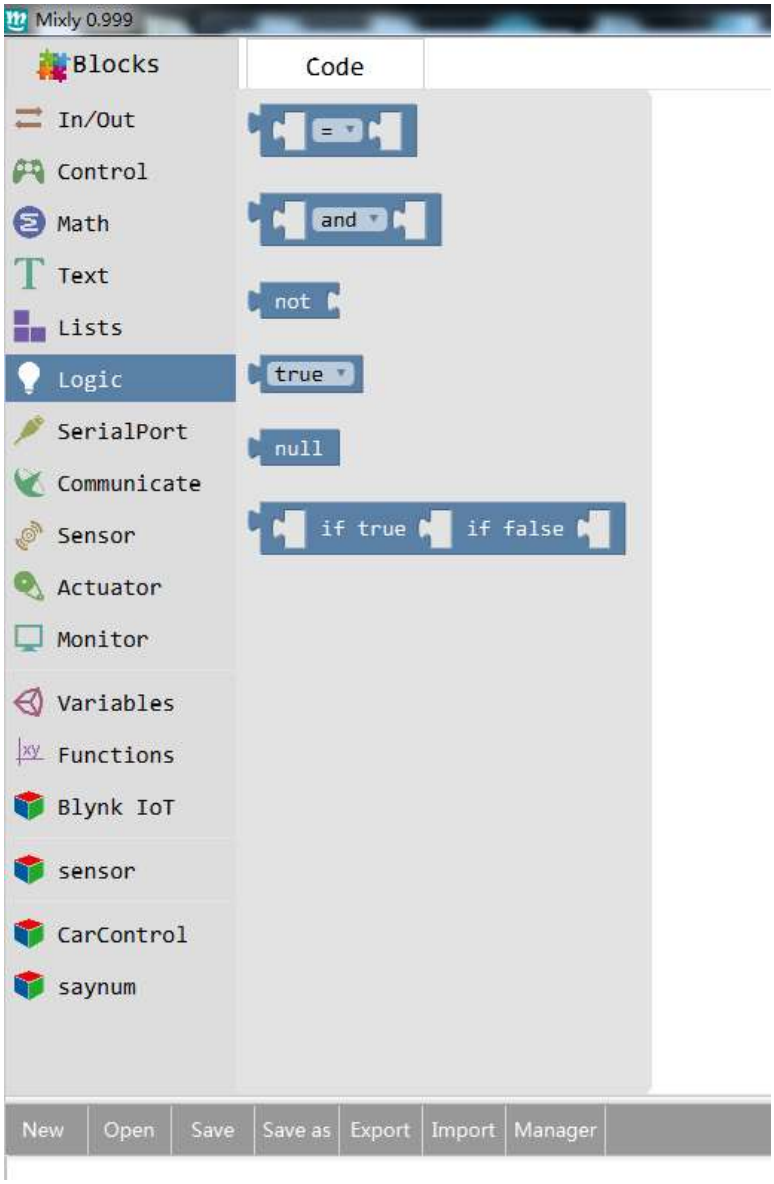


Figure 1-4-4-3

1.4.5 MotorDriverBoard Introduction to building blocks

Motordriverboard is a library that we write for motordriverboard. After importing the library, it will appear in the library module area under the basic module area. Click it and there will appear various small blocks, such as figure 1-4-5-1. Next, we will learn how to write programs for motordriverboard by assembling these modules.

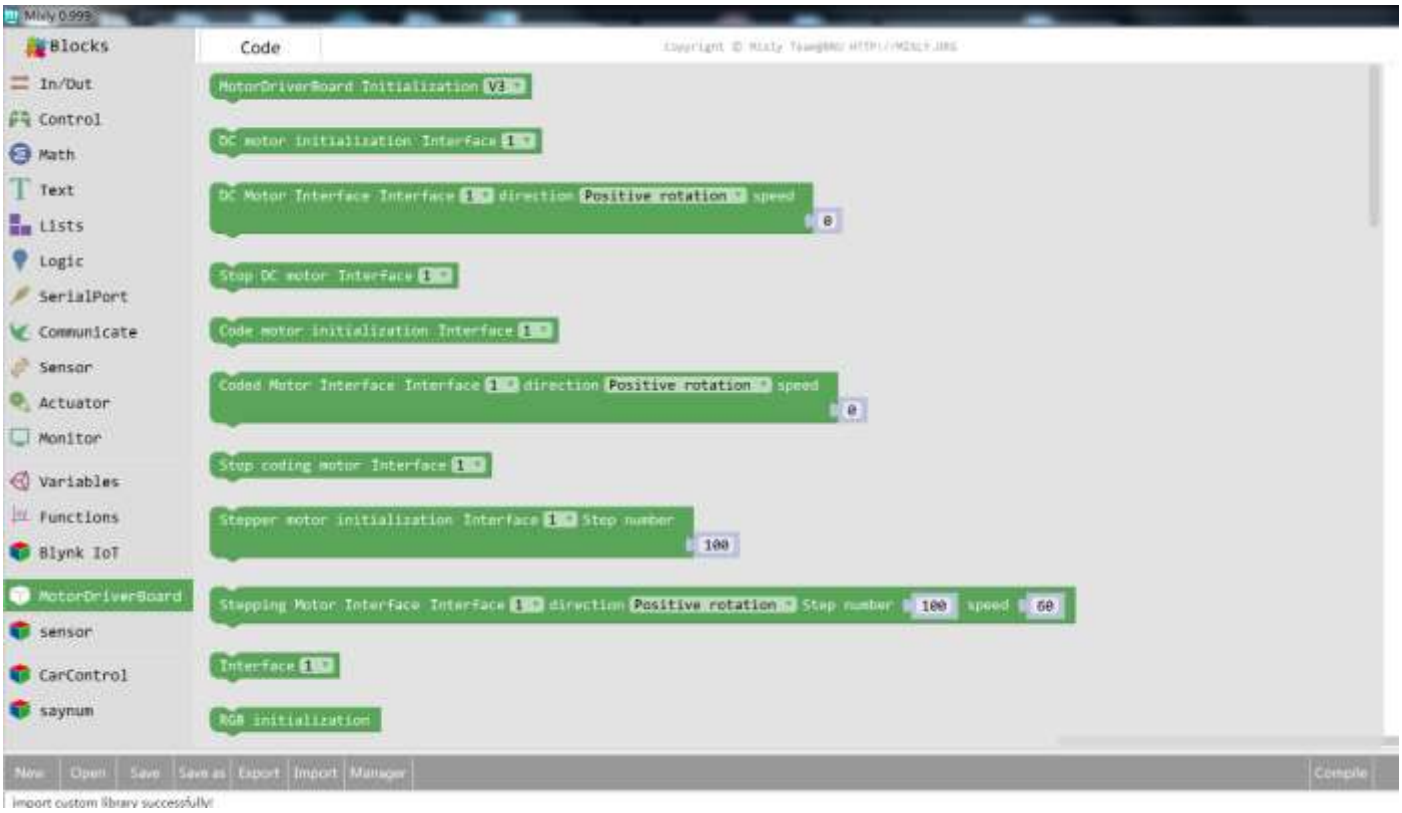


Figure 1-4 -5-1

Chapter two a preliminary understanding of Mixly programming

2.1 Add motordriverboard Library

Before programming, we need to add the MotorDriverBoard library. The specific steps are as follows:

1) Download the motordriverboard library and save it on your own computer
2) Open the mixly software and click "import", as shown in figure 2-1-1;
3) Click the motordriverboard.xml file, and then click "open", as shown in figure 2-1-2;

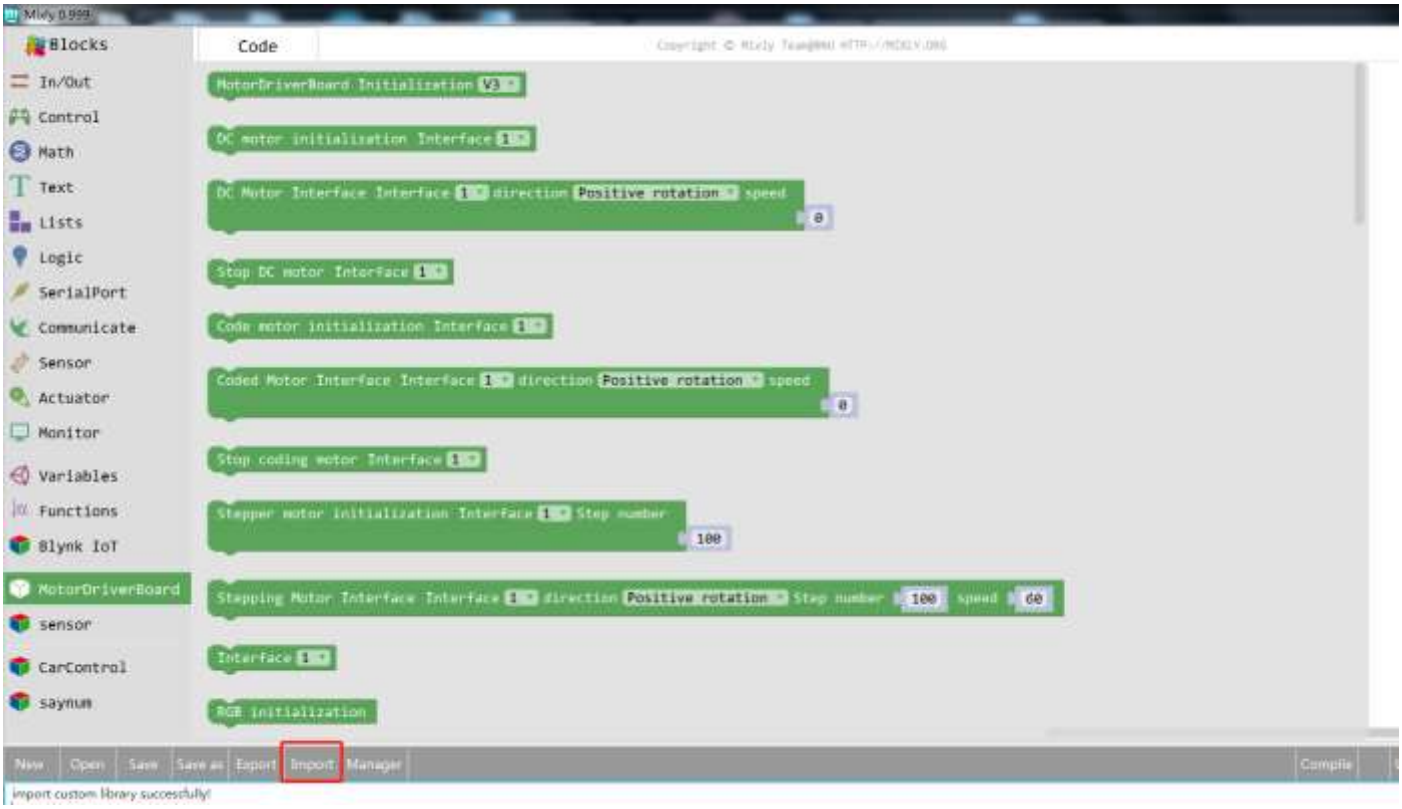


Figure 2-1-1

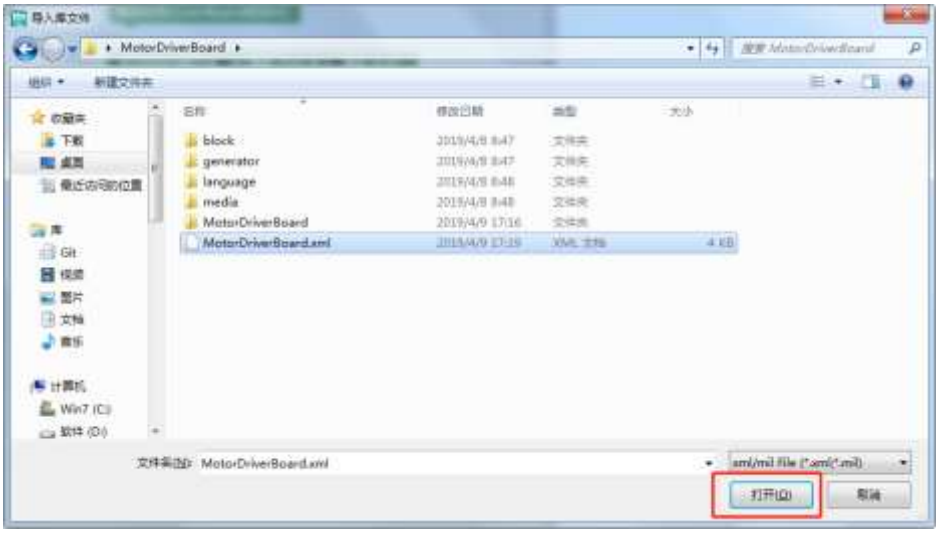


Figure 2-1-2

4) After the import is successful, the prompt area will display: import the custom library successfully! In the library module area, you can see the motordriverboard library module, as shown in figure 2-1-3;

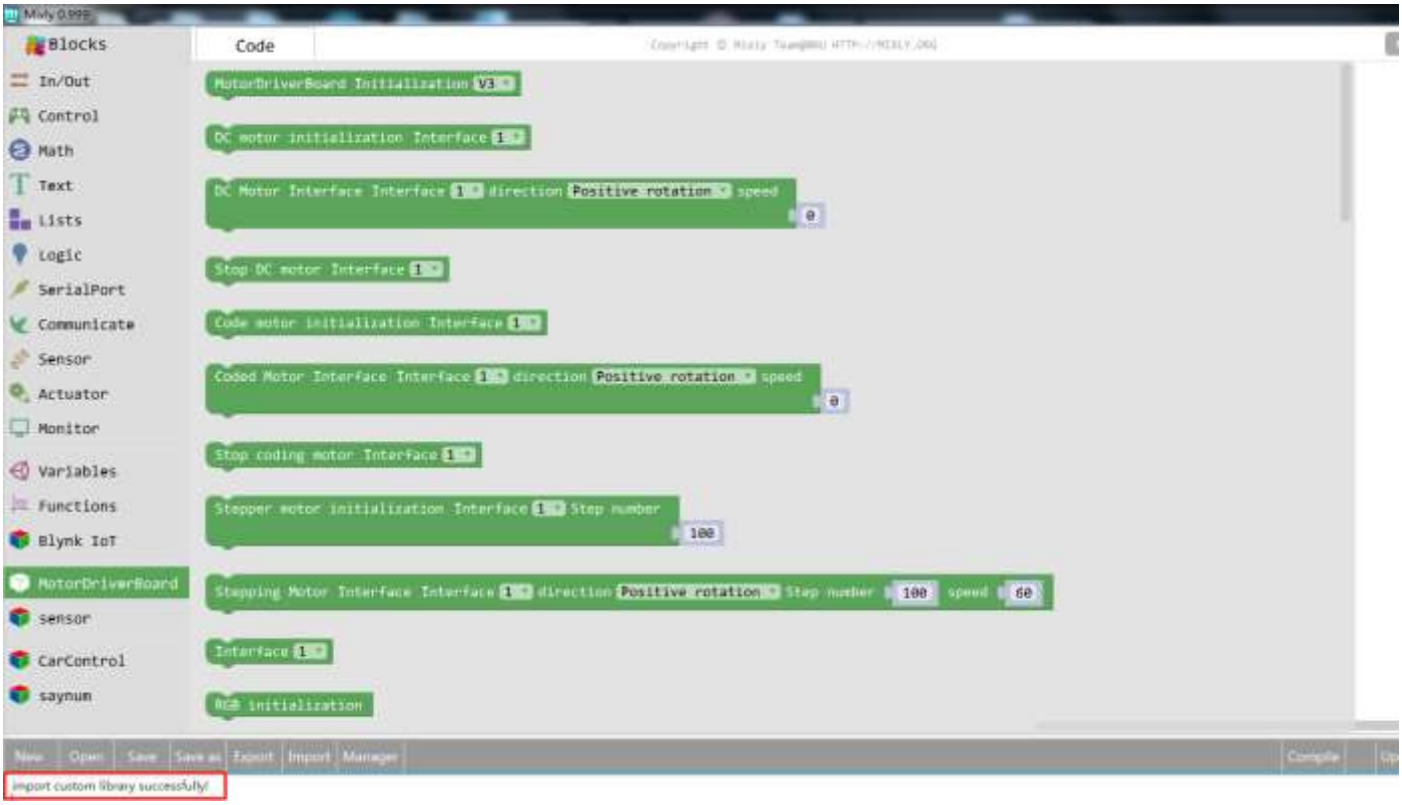


Figure 2-1-3

5) Click "motordriverboard" to display the figure shaped programming block of motordriverboard building block, as shown in figure 2-1-4.

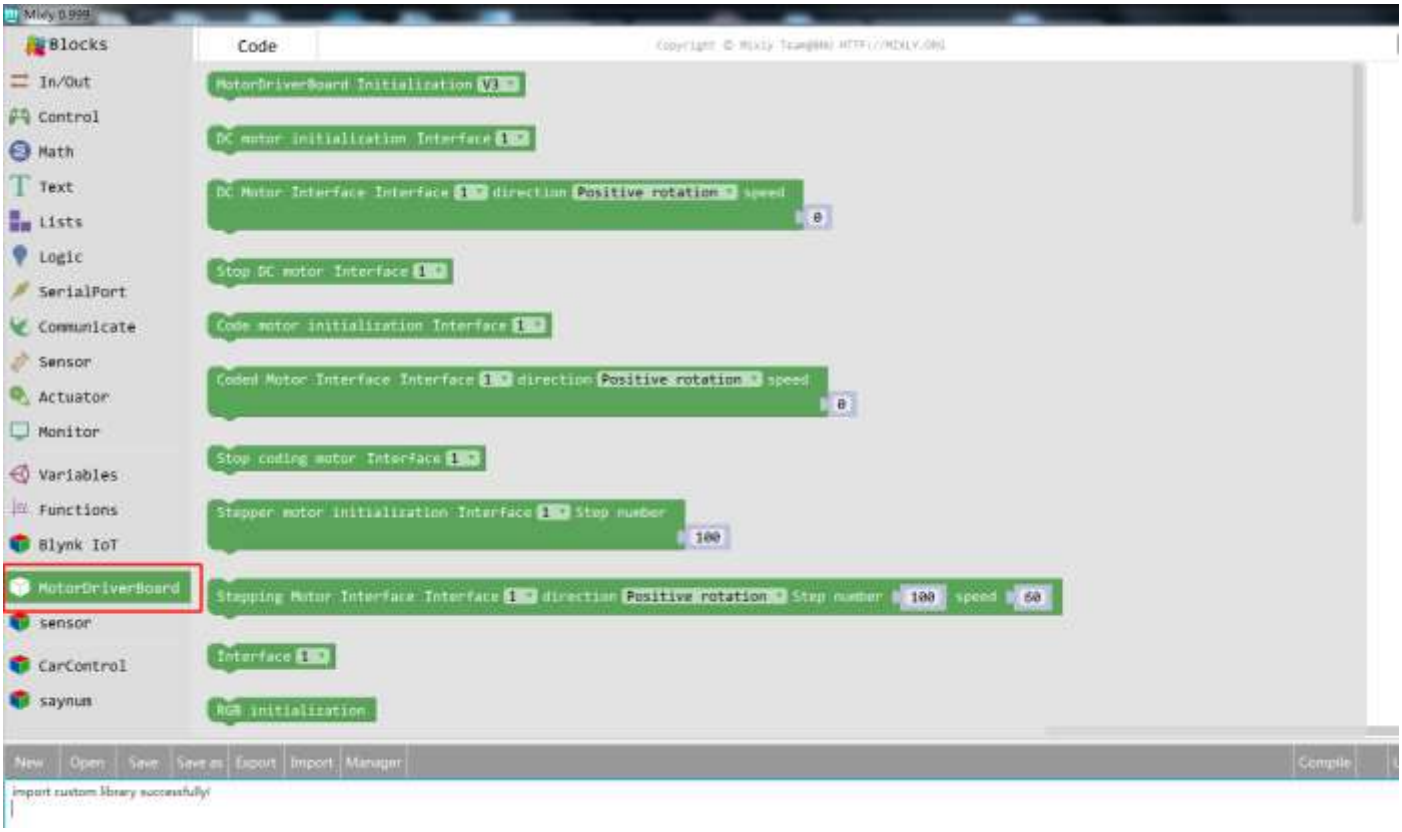


Figure 2-1-4

2.2 Initial programming experience

After adding the motordriverboard library, let's experience the motordriverboard programming! Let's first write a program to let the motordriverboard expansion board print Hello world;

1) First, drag the "motordriverboard initialization" of the building block area to the program architecture area with the mouse. It should be noted that there are three versions of motordriverboard: V3, V4 and V5. We need to select the corresponding version in the "motordriverboard initialization" figure block according to the version used, and then drag "serial baud rate 9600" to the bottom of the "motordriverboard initialization" building block, As shown in figure 2-2-1,

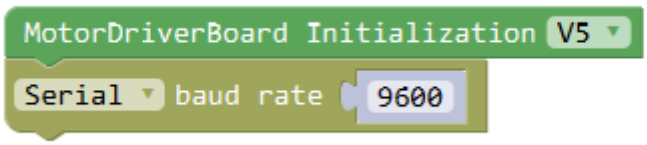


Figure 2-2-1

2) In the serial port module, drag the "serial print (auto wrap)" block to the bottom of "serial baud rate 9600", as shown in figure 2-2-2

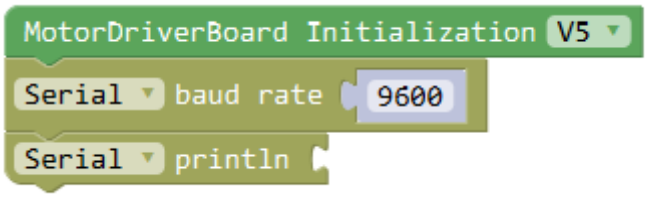


Figure 2-2-2

3) In the text module, drag the building block of "hello" to the right side of serial printing (auto wrap), and input "Hello word" as shown in figure 2-2-3.

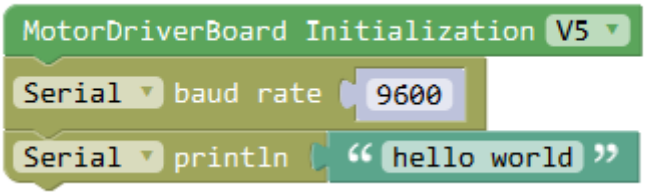


Figure 2-2-3

That's how the motordriverboard extension prints Hello word. After the program is written, we need to transfer the program to the brain (control board) of the motordriverboard expansion board. The expansion board will do the action we want according to the program we wrote. How to transfer the program to the brain (control board) of the motordriverboard expansion board? Only when mixly and the main control board of the expansion board are connected, can we transfer the program written on the

computer to the brain (control board) of the expansion board of the motordriverboard. The following describes the connection method of mixly and the main control board of the expansion board.

2.3 Steps to connect mixly and motordriverboard

1) Use a USB data cable, one end is inserted into the computer, the other end is inserted into the main control board of the expansion board, and connect the main board of the expansion board with the computer;
2) Click the pull-down box on the right of "Upload" to select the model of the main control board, and then click the second pull-down box on the right to select the USB serial port, as shown in figure 2-3-1. After selection, mixly and the motordriverboard are successfully connected.

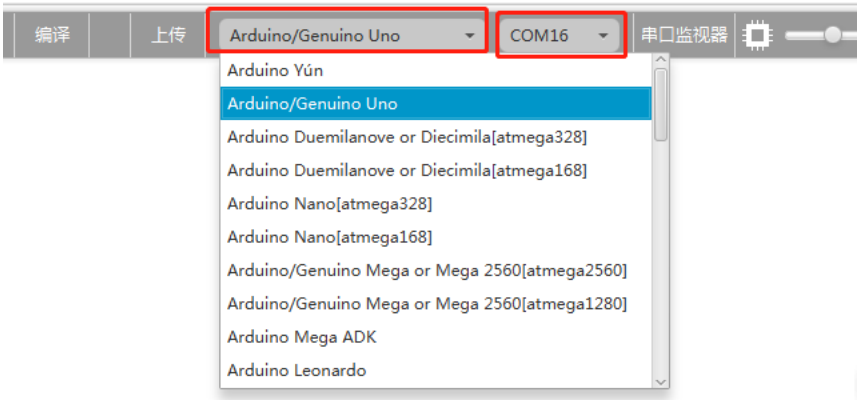


Figure 2-3-1

2.4 Upload the program to the motordriverboard expansion board

When the program is completed and mixly and the motordriverboard expansion board are connected correctly, we can transfer the program we have written to the brain (main control board) of the motordriverboard expansion board. The specific operation steps are as follows:

1) Click upload, and the building block program in the program architecture area will start to upload to the control board. We need to wait for a while, and then prompt "upload completed". After the above steps are completed, there are programs written by us in the brain (main control board) of the motordriverboard expansion board. After the upload is successful, it will prompt that the upload is successful, such as figure. Of course, we can click "generation" at any time Code "to view the program actually uploaded to the control board. These are the C language codes corresponding to the figure block of the building block, such as figure 2-4-1.

Figure 2-4-1

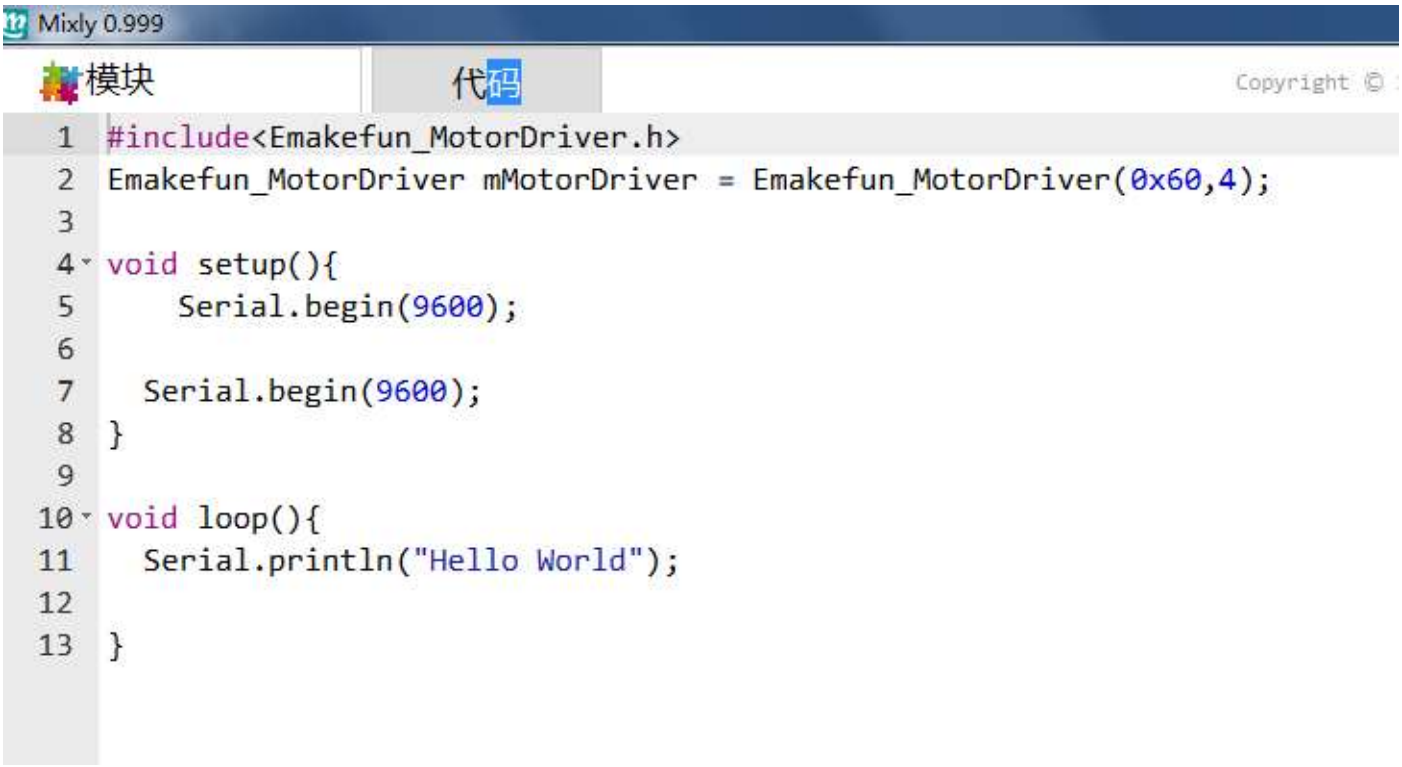


Figure 2-4-1

Turn on the serial monitor. At this time, we will see that hello word is constantly printing on the serial monitor, as shown in figure 2-4-2

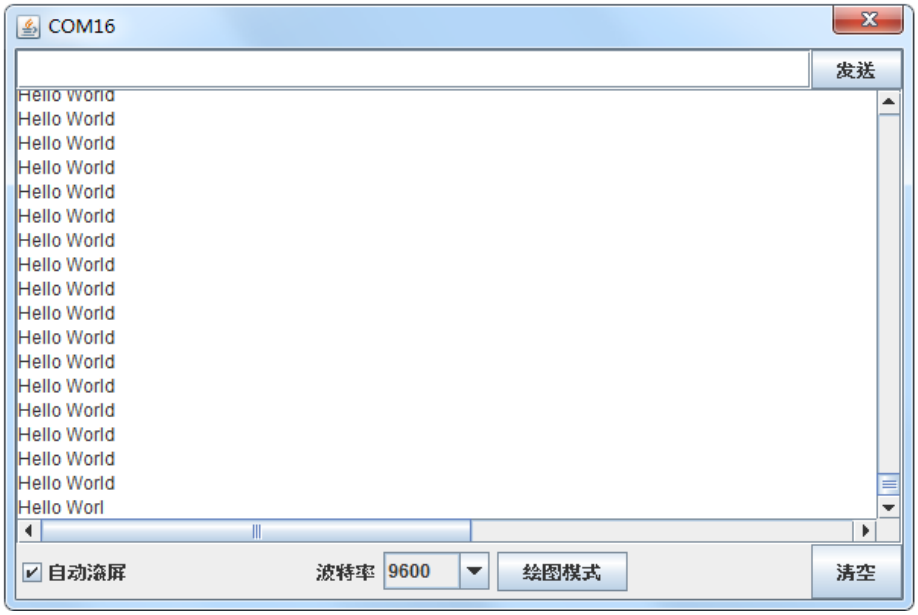


Figure 2-4-2

Chapter 3 full function programming

3.1 DC motor

3.1.1 Principle of DC motor

The reason why a car can move is that it has an engine to power it. The expansion board also has DC motor module to make it move, so what is the motor? In our science textbook, there is an introduction of electromagnetic induction. The motor is driven by electromagnetic induction. It has an iron core wound with copper wire inside and a rotor outside. When the iron core is electrified, there is electromagnetic induction to make the rotor move. This is the motor.

The motordriverboard has four DC motor interfaces, and the wiring diagram of M1 ~ M4 is as follows.

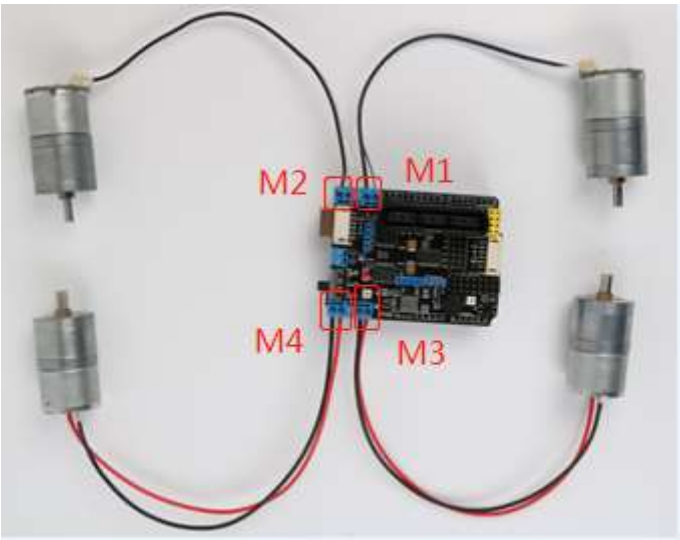


Figure 3-1-1-1

3.1.2 DC motor test

In the expansion board module, we can find three control building blocks of DC motor module, which are initialization motor, setting motor interface, speed and direction, and stopping motor building blocks, as shown in Figure 3-1.

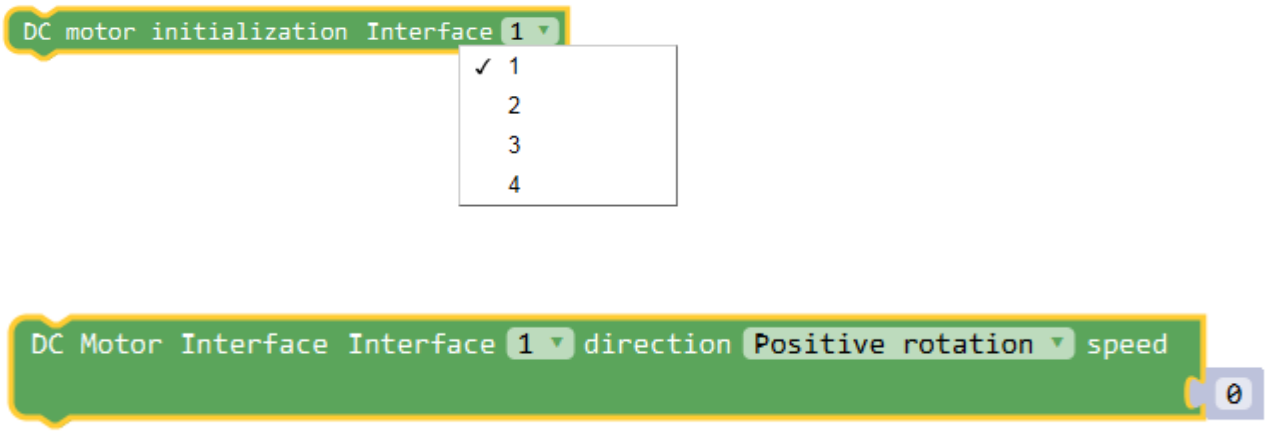


Figure 3-1-2-1

Write the following programs on Mixly, they can respectively control the forward and reverse rotation of the motor. Four DC motors can be connected to the motordriverboard. The M1, M2, m3 and M4 marked on the board correspond to the motor numbers 1, 2, 3 and 4 respectively. The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ DC motor test program

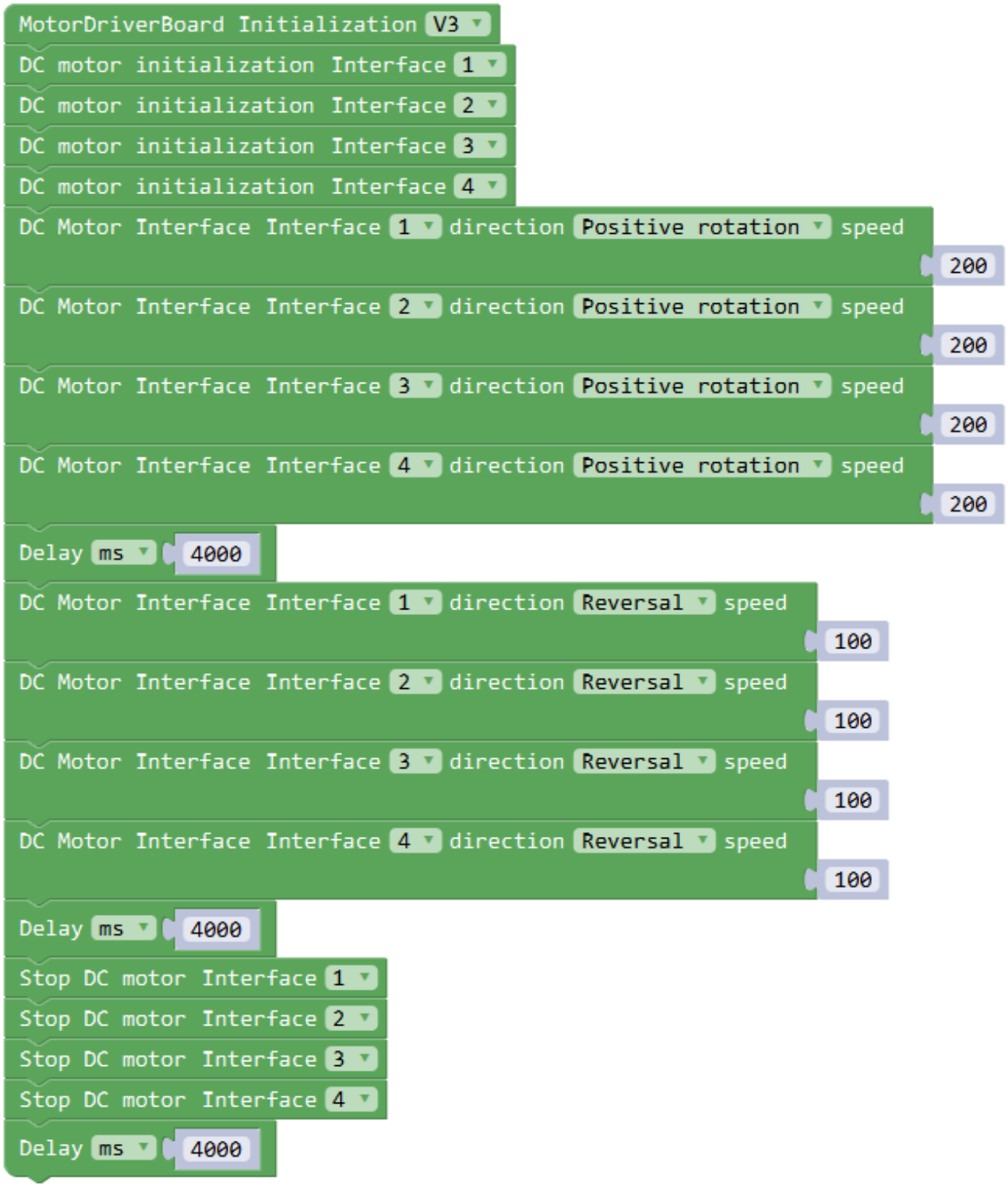

MotorDriverBoard Initialization V3
DC motor initialization Interface 1
DC motor initialization Interface 2
DC motor initialization Interface 3
DC motor initialization Interface 4
DC Motor Interface Interface 1 direction Positive rotation speed 200
DC Motor Interface Interface 2 direction Positive rotation speed 200
DC Motor Interface Interface 3 direction Positive rotation speed 200
DC Motor Interface Interface 4 direction Positive rotation speed 200
Delay ms 4000
DC Motor Interface Interface 1 direction Reversal speed 100
DC Motor Interface Interface 2 direction Reversal speed 100
DC Motor Interface Interface 3 direction Reversal speed 100
DC Motor Interface Interface 4 direction Reversal speed 100
Delay ms 4000
Stop DC motor Interface 1
Stop DC motor Interface 2
Stop DC motor Interface 3
Stop DC motor Interface 4
Delay ms 4000

Figure 3-1-2-2

3.2 Coding motor

3.2.1 Introduction to coding motor

Figure 3-2-1-1

The encoder is a kind of rotary sensor which transforms the angular displacement or angular velocity into a series of electrical digital pulses. We can measure the displacement or velocity information through the encoder.

3.2.2 Code motor program

Write the following program to test whether the forward and reverse rotation of the encoder motor is normal. The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ code motor test program

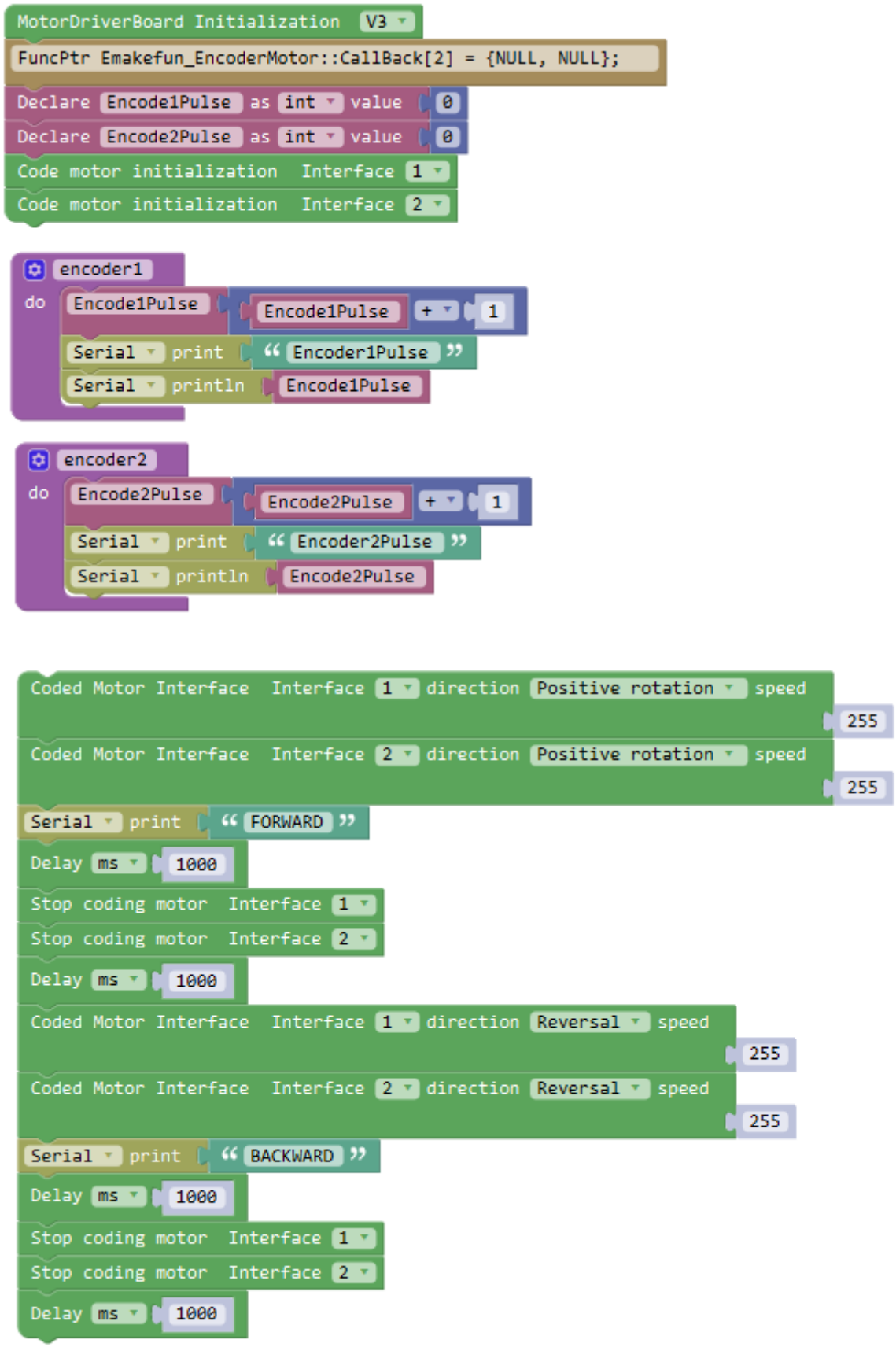

MotorDriverBoard Initialization V3
FuncPtr Emakefun_EncoderMotor::CallBack[2] = {NULL, NULL};
Declare Encode1Pulse as int value 0
Declare Encode2Pulse as int value 0
Code motor initialization Interface 1
Code motor initialization Interface 2
encoder1
do
Encode1Pulse
Encode1Pulse + 1
Serial print “ Encoder1Pulse ”
Serial println Encode1Pulse
encoder2
do
Encode2Pulse
Encode2Pulse + 1
Serial print “ Encoder2Pulse ”
Serial println Encode2Pulse
Coded Motor Interface Interface 1 direction Positive rotation speed
255
Coded Motor Interface Interface 2 direction Positive rotation speed
255
Serial print “ FORWARD ”
Delay ms 1000
Stop coding motor Interface 1
Stop coding motor Interface 2
Delay ms 1000
Coded Motor Interface Interface 1 direction Reversal speed
255
Coded Motor Interface Interface 2 direction Reversal speed
255
Serial print “ BACKWARD ”
Delay ms 1000
Stop coding motor Interface 1
Stop coding motor Interface 2
Delay ms 1000

Figure 3-2-2-1

3.3 Stepper motor

3.3.1 Introduction to stepping motor

Stepping motor is an open-loop control motor which transforms electric pulse signal into angular displacement or linear displacement. It is the main executive component of modern digital program control system and widely used. In the case of no overload, the speed and stop position of the motor only depend on the frequency and number of pulses.

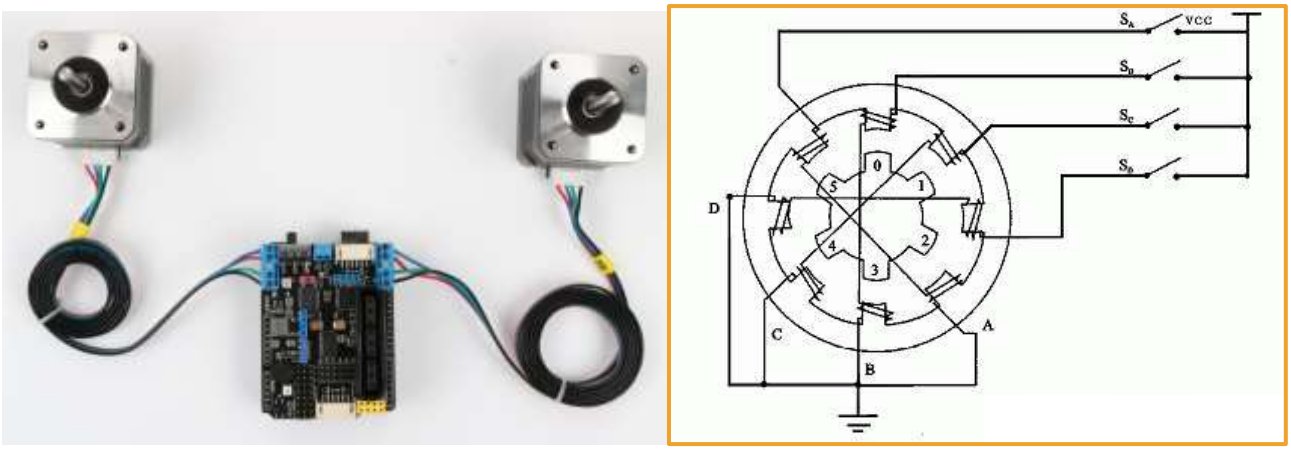


Figure 3-3-1-1

Stepping motor is an open-loop control motor which transforms electric pulse signal into angular displacement or linear displacement. It is the main executive component of modern digital program control system and widely used. In the case of no overload, the speed and stop position of the motor only depend on the frequency and number of pulses. There are many pairs of poles in the stepper motor. If the power on state remains unchanged, the stepper motor will remain in a fixed state. Only by constantly changing the power on state of each pole pair can the stepper motor continue to rotate. Therefore, the stepping motor can not be directly connected to the DC or AC power supply to work, it must use a special driving power supply (stepping motor driver). The controller (pulse signal generator) can control the angular displacement by controlling the number of pulses, so as to achieve the purpose of accurate positioning; at the same time, it can control the speed and acceleration of motor rotation by controlling the pulse frequency, so as to achieve the purpose of speed regulation.

3.3.2 Step motor program

Write the figure shaping program shown in figure below to test the operation of stepping motor. The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ step motor test program

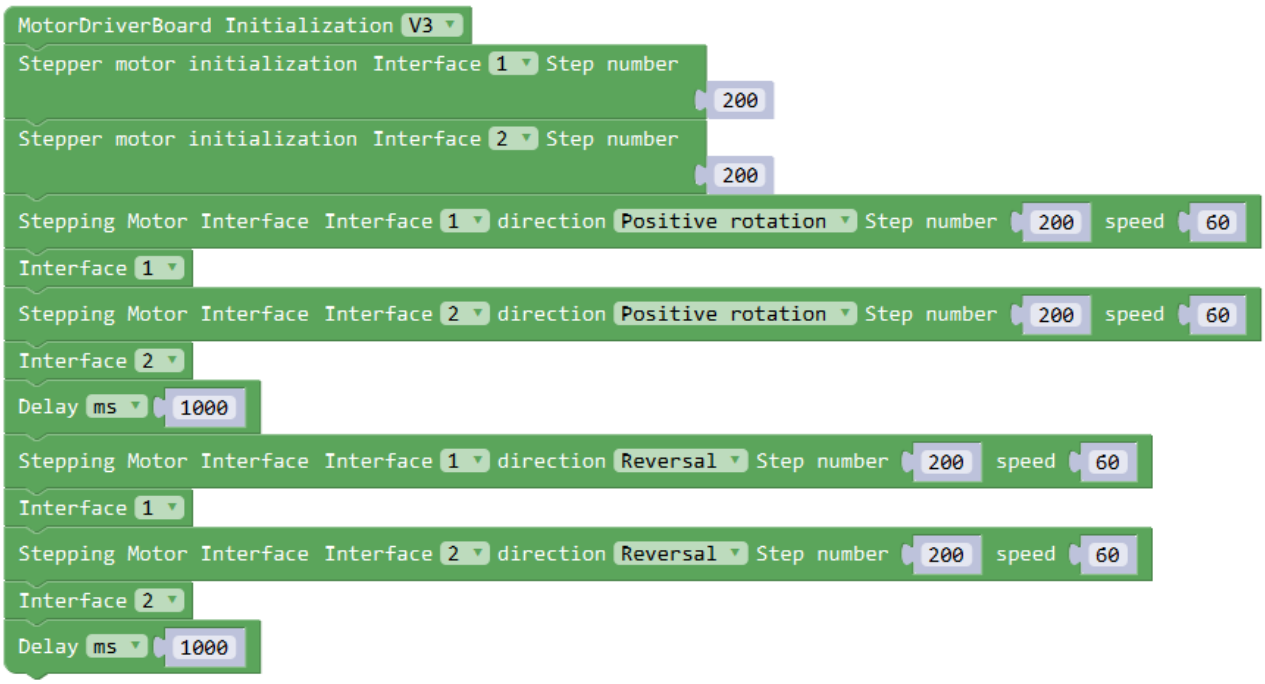


Figure 3-3-2-1

3.4 Servo

3.4.1 Servo Working principle

The steering gear is mainly composed of the following parts: steering wheel, reduction gear set, position feedback potentiometer, DC motor, control circuit, etc., The Bumblebee and Optimus Prime joints of the expansion board we see in the movie all need to be controlled by the steering gear, especially when the expansion board is moving, the mechanical sound of clicking is generated by the rotation of the steering gear on the expansion board.

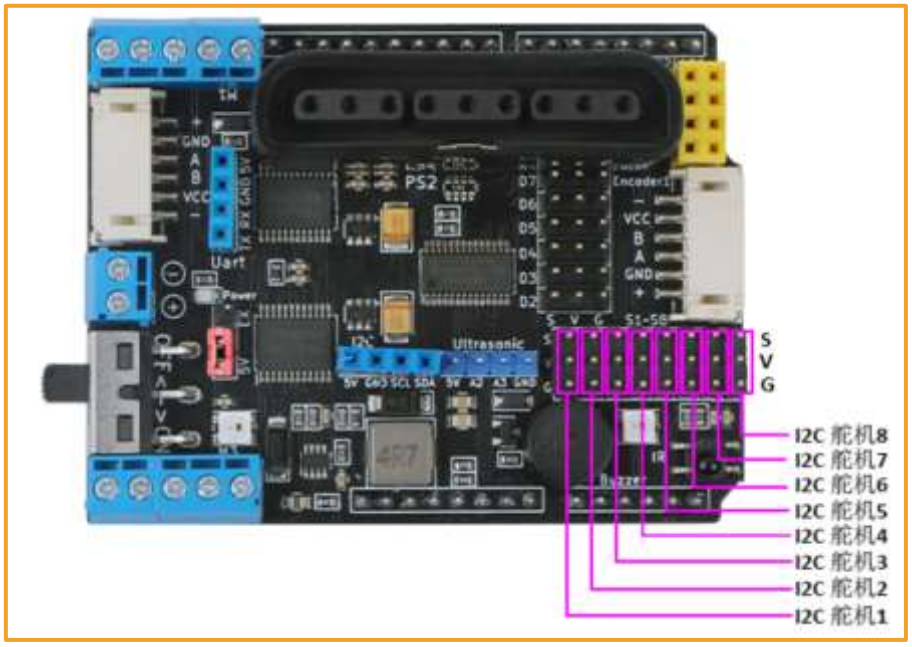
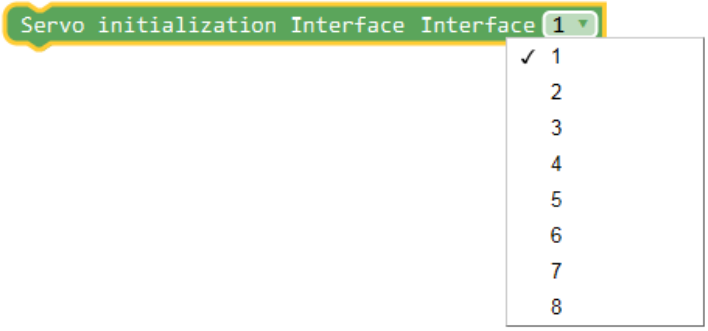


Figure 3-4-1-3

When the control circuit board of the steering gear receives the control signal from the signal line, it controls the rotation of the motor, which drives a series of gear sets, and then transmits them to the output steering wheel after deceleration. Its working process is: control signal → control circuit board → motor rotation → gear set deceleration → steering wheel rotation → position feedback potentiometer → control circuit board feedback.

3.4.2 servo test procedure




MotorDriverBoard Initialization V3
Servo initialization Interface Interface 1
Servo initialization Interface Interface 2
Servo initialization Interface Interface 3
Servo initialization Interface Interface 4
Steering gear interface Interface 1 angle 0
Steering gear interface Interface 2 angle 0
Steering gear interface Interface 3 angle 0
Steering gear interface Interface 4 angle 0
Delay ms 1000
Steering gear interface Interface 1 angle 90
Steering gear interface Interface 2 angle 90
Steering gear interface Interface 3 angle 90
Steering gear interface Interface 4 angle 90
Delay ms 1000
Steering gear interface Interface 1 angle 180
Steering gear interface Interface 2 angle 180
Steering gear interface Interface 3 angle 180
Steering gear interface Interface 4 angle 180
Delay ms 1000

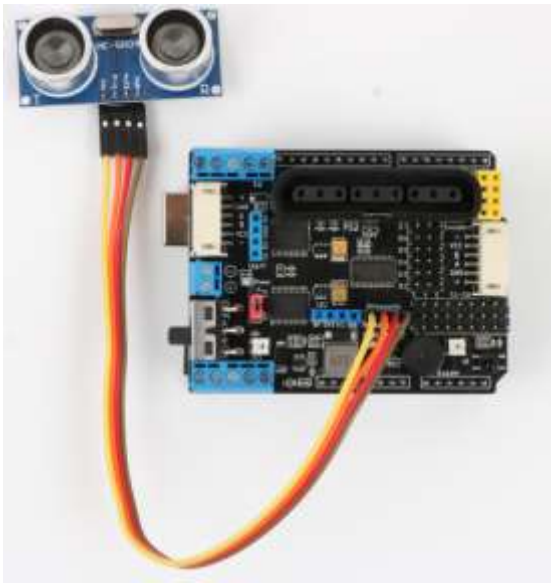
Figure 3-4-2-1

The above is the test program of the steering gear. Connect the steering gear interface 5, upload the program to the motordriverboard, connect the power supply, and you can see that the steering gear turns left and right repeatedly every two seconds. (180 degrees is right, 0 degrees is left.) The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ steering gear test program

3.5 Ultrasonic module

3.5.1 Ultrasonic principle

Ultrasonic sensor is a device which can detect the distance by transmitting ultrasonic wave. Ultrasonic wave is a kind of acoustic wave that can not be heard, and it has the characteristics of returning when touching an object. The ultrasonic sensor has two "eyes", one "eye" emits ultrasonic wave, and the other "eye" receives the ultrasonic wave sent back from an obstacle. When one eye emits ultrasonic wave, it starts timing. When the other eye receives the returned ultrasonic wave, it stops timing. Mathematically, we learned that distance = speed × time, so the ultrasonic can measure distance= The speed of the ultrasonic wave × (timing time ÷ 2); in this way, the distance can be calculated.



3-5-1-1

3.5.2 Usage of ultrasonic module

We found the control block of ultrasonic module in the control module of the expansion board, as shown in figure 3-5-2-1. Using the block, we can detect the distance between the expansion board and the obstacles in front. Note: when using the ultrasonic module, first of all, we need to select the mode control building block, and set the mode as the ultrasonic obstacle avoidance mode. When the obstacle completely blocks the ultrasonic module or faces the ultrasonic module to the distance, the ultrasonic from the

ultrasonic module cannot be received, so the distance between the expansion board and the obstacle cannot be measured. The ultrasonic module installed on the expansion board The detection distance is 5cm ~ 400cm.

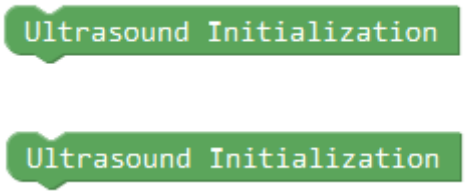


Figure 3-5-2-1

3.5.3 Test program of ultrasonic obstacle avoidance module

We can first write a program to test the ultrasonic obstacle avoidance module, and use serial port printing. When our hands are close to the ultrasonic obstacle avoidance module and far away from the ultrasonic obstacle avoidance module, we can observe the ultrasonic measurement distance printed by the serial port, and more intuitively see the distance measurement process of the ultrasonic obstacle avoidance module. Next, we will write a test program. The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ ultrasonic distance measurement program

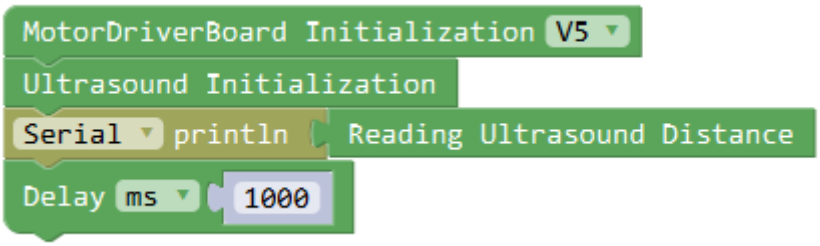


Figure 3-5-3-1

We have written a program like figure 3-5-3-1, and then after the program is uploaded successfully, open the serial port, and then close or away from the ultrasonic module, the serial port will print the corresponding distance.

3.6 RGB

3.6.1 RGB

There are two RGB lights on the motordriverboard. These two modules are called RGB modules. R is the abbreviation of red in English, G is the abbreviation of green in English, and B is the abbreviation of blue in English. Red, green and blue are what we call the three primary colors of color and light. RGB module can make it emit different colors of light through building block control. We can find the control building blocks of two RGB modules in the expansion board module, as shown in figure 3-6-1-1.

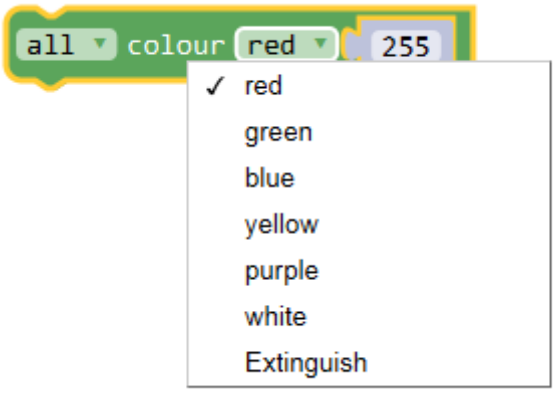


Figure 3-6-1-1

3.6.2 RGB program

We can use RGB module to write a program of light changing and flickering continuously, as shown in figure 3-6-2-1. The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ RGB test program

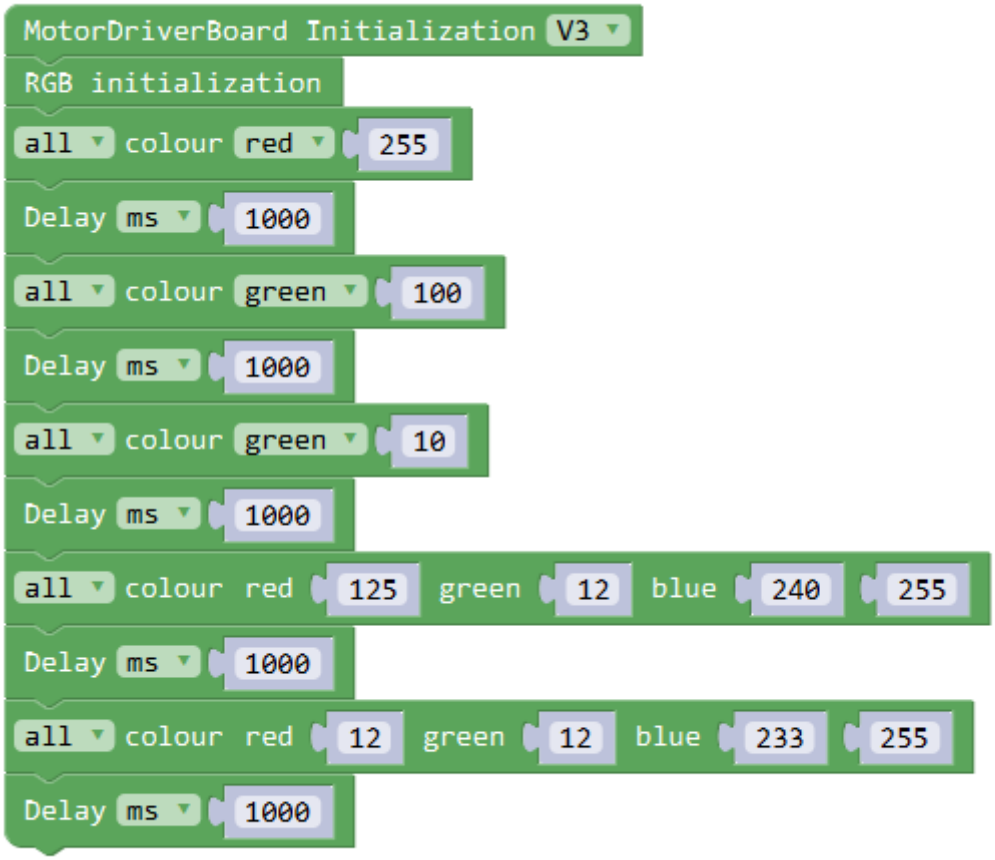


Figure 3-6-2-1

3.7 buzzer

3.7.1 Principle of buzzer

By associating with the common musical instrument playing mode, we can learn that sound is the sound wave produced by the vibration of the right object and the surrounding air, and the different frequency of vibration will make the sound produced different.

3.7.2 buzzer blocks

We can find the control block of buzzer module in the expansion board module, as shown in figure 3-7-2-1.

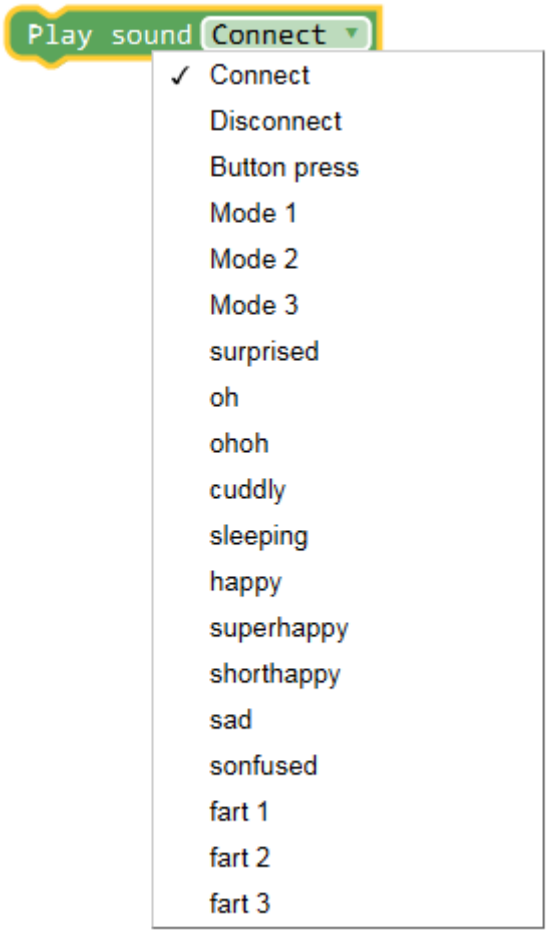


Figure 3-7-2-1

We can use the buzzer module to write a simple sound generating program as shown in figure 3-7-3-1. Of course, we can also select various desired sounds in the drop-down box. The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ buzzer test program

3-7-3-1

3.8 infrared remote control

3.8.1 Principle of infrared remote control

The remote control system is generally composed of a remote control (transmitter) and a receiver. When you press any key on the remote control, the remote control will send a command. After receiving the command from the remote control, the receiver will send the command to the brain of the expansion board. The expansion board will think about what kind of action to do according to the remote control command, and then control its limbs (four wheels) to do the corresponding Figure 4-1 shows the action of remote control and receiver.

Figure 3-8-1-1

To control the expansion board through the remote control, we first need to set the control mode as infrared remote control mode, and set the infrared remote control receiving pin according to the actual connection port, and then define the effect of pressing each key of the remote control, so that when we press the key of the remote control, the expansion board will do the actions defined in our program.

3.8.2 Infrared remote control program

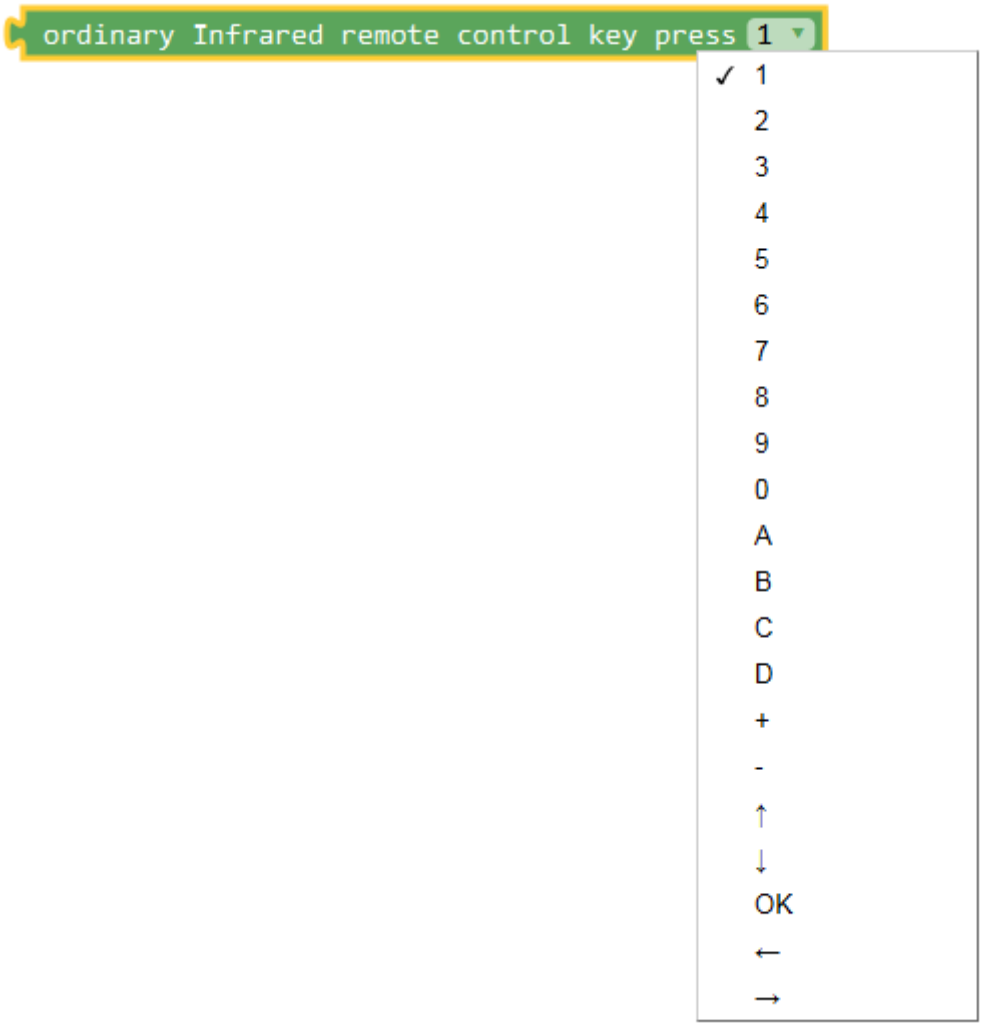


Figure 3-8-2-1

We can first write a program to test the infrared remote control, and use the serial port to print. When the remote control button is pressed, we can observe the characters printed by the serial port, and then we can know which button of the infrared remote control is pressed. Next, we will write a test program. The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ infrared remote control test program

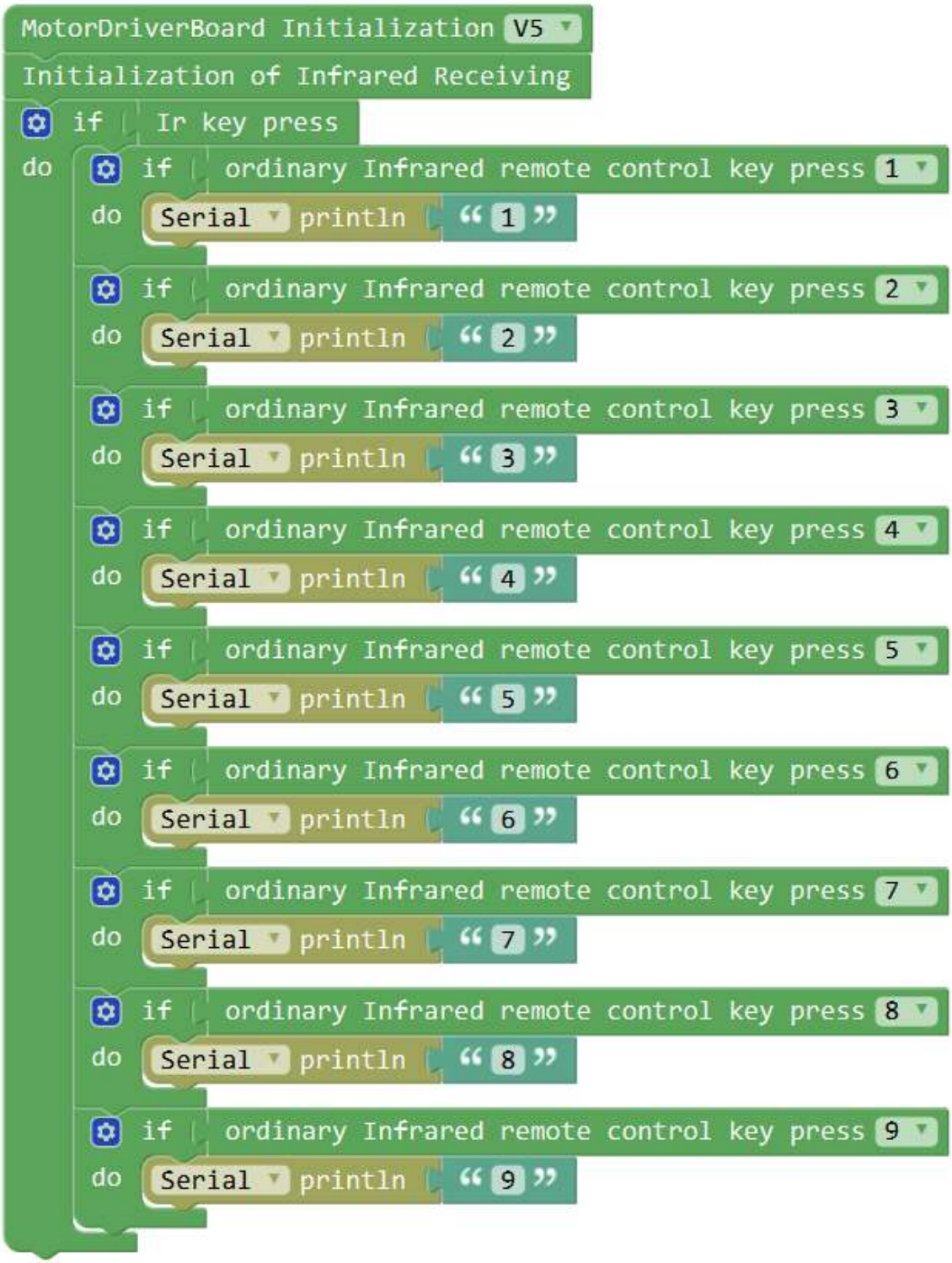


Figure 3-8-2-2

We have written a program like figure 3-8-2-2, and then after the program is uploaded successfully, open the serial port, then hold the remote control to the expansion board and press the key, and the corresponding characters will be printed out through the serial port.

3.9 PS2 remote control handle

3.9.1 PS2 remote control handle

PS2 handle is composed of handle and receiver. The handle needs two sections of No.7 1.5V power supply. Turn the handle switch to on. If the receiver is not searched, the light on the handle will flash continuously. If the receiver is not searched within a certain period of time, the handle will enter the standby mode and the light on the handle will be extinguished. At this time, press the "start" key to wake up the handle.

The working power supply of the receiver is 3-5v, which cannot be reversed or over-voltage, otherwise the receiver will be burnt out.

After normal power on, the handle is automatically paired with the receiver. In the case of no successful pairing, the green light of the receiver flashes, and the light on the handle flashes. After successful pairing, the green light on the receiver is always on, and the light on the handle is always on. Press "mode" (the handle batch is different, and the above mark may be "analog", but it will not affect the use). You can select "red light mode" , green mode.

Figure 3-9-1-1

When the handle is connected with the receiver, we can send key commands with the handle. When the receiver receives these key commands, the brain of the expansion board (main control board) will make its limbs (four wheels, steering gear) do corresponding actions according to the received commands.

3.9.2 PS2 remote control handle program

We can write a program to test the PS2 remote control key and rocker first. The effect of the key test program is that when the PS2 remote control key is pressed, the RGB light is on and the serial port prints the corresponding characters, we will know which key of the PS2 remote control is pressed. When the rocker is pressed and turned, the serial port will continuously print the read rocker value. The program path is: figure shaped programming \ mixly \ Motordrive Rboard sample program \ hardware library test program \ PS2 test program

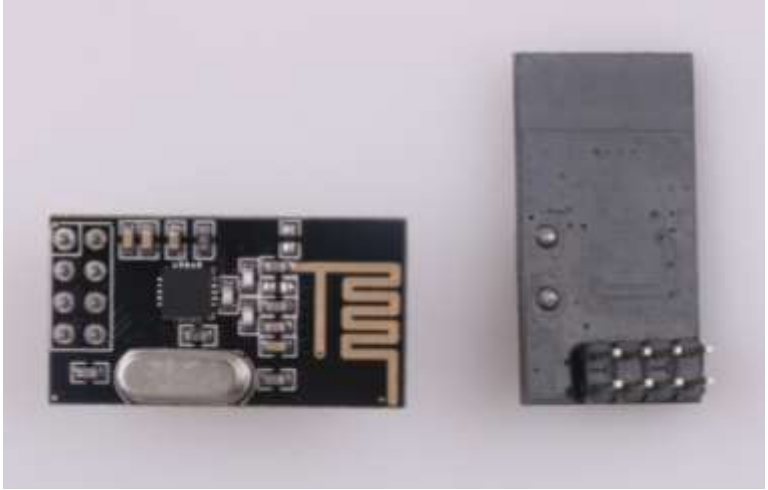

MotorDriverBoard Initialization V3
PS2 Initialization
RGB initialization
if PS2 key pressed
do if PS2 handle press up
do all colour red 255
Serial println " up "
if PS2 handle press down
do all colour green 150
Serial println " down "
if PS2 handle press left
do all colour blue 150
Serial println " left "
if PS2 handle press right
do all colour yellow 100
Serial println " right "
if PS2 handle press △
do all colour purple 50
Serial println " triangle "
if PS2 handle press ○
do all colour white 10
Serial println " round "
if PS2 handle press ×
do all colour purple 255
all colour white 255
Serial println " cross "
if PS2 handle press □
do all colour white 255
all colour purple 255
Serial println " square "
if PS2 handle press Left 3 or PS2 handle press Right 3
do Serial println Getting Rocker Value Left lateral axis
Serial println Getting Rocker Value Left vertical axis
Serial println Getting Rocker Value Right horizontal axis
Serial println Getting Rocker Value Right vertical axis

3.10 Nrf24L01

3.10.1 Nrf24L01 introduce

NRF24L01 + module (as shown in Figure 13-1) is a 2.4G wireless communication module developed by Nordic based on nRF24L01 chip. With FSK modulation, Nordic's enhanced short burst protocol is internally integrated. It can realize point-to-point or 1-to-6 wireless communication. The wireless communication speed can reach up to 2m (BPS). NRF24L01 has four working modes, including transceiver mode, configuration mode, idle mode and shutdown mode.

Insert the nRF24L01 + module into the corresponding interface on the driver board of the motor driver (as shown in figure 3-10-1-1). In order to receive stable nRF24L01 data, it is recommended to connect 10uF capacitors between VCC and gud, such as figure 3-10-1-2. The received data can be sent to each other through two devices for test.



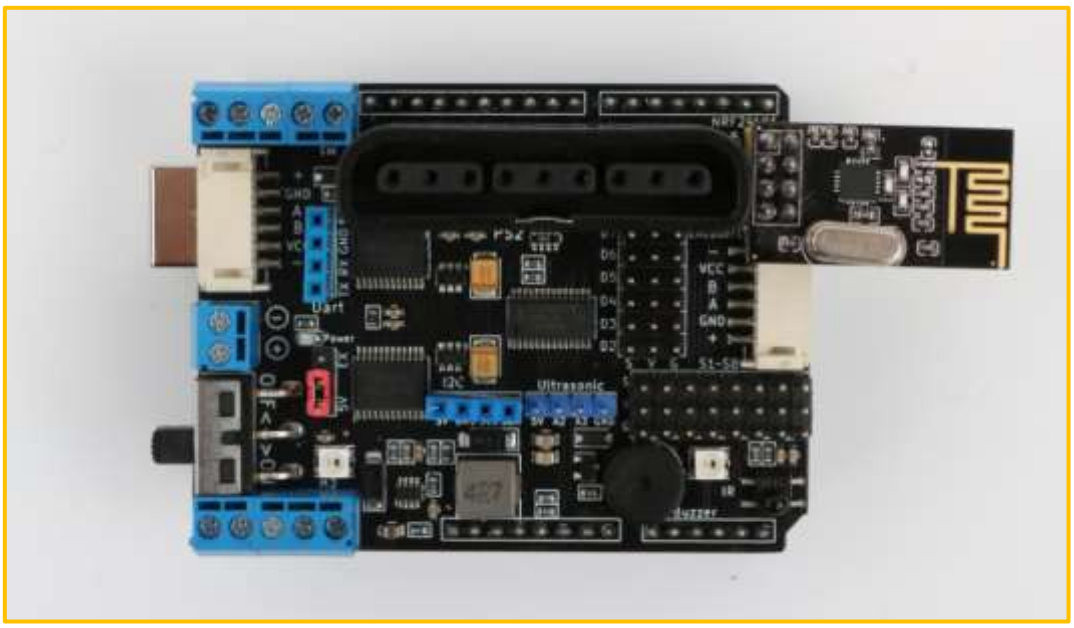
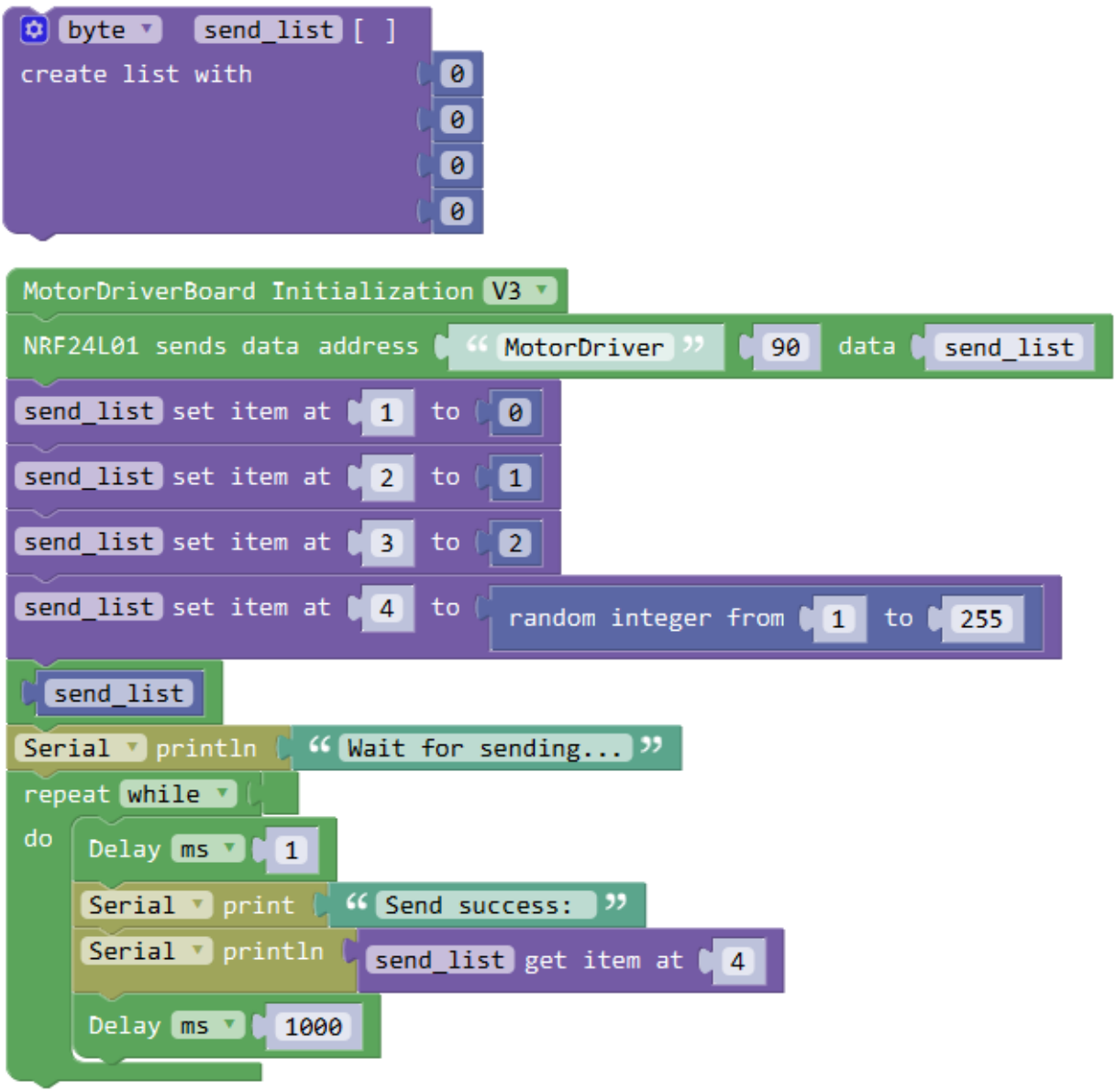


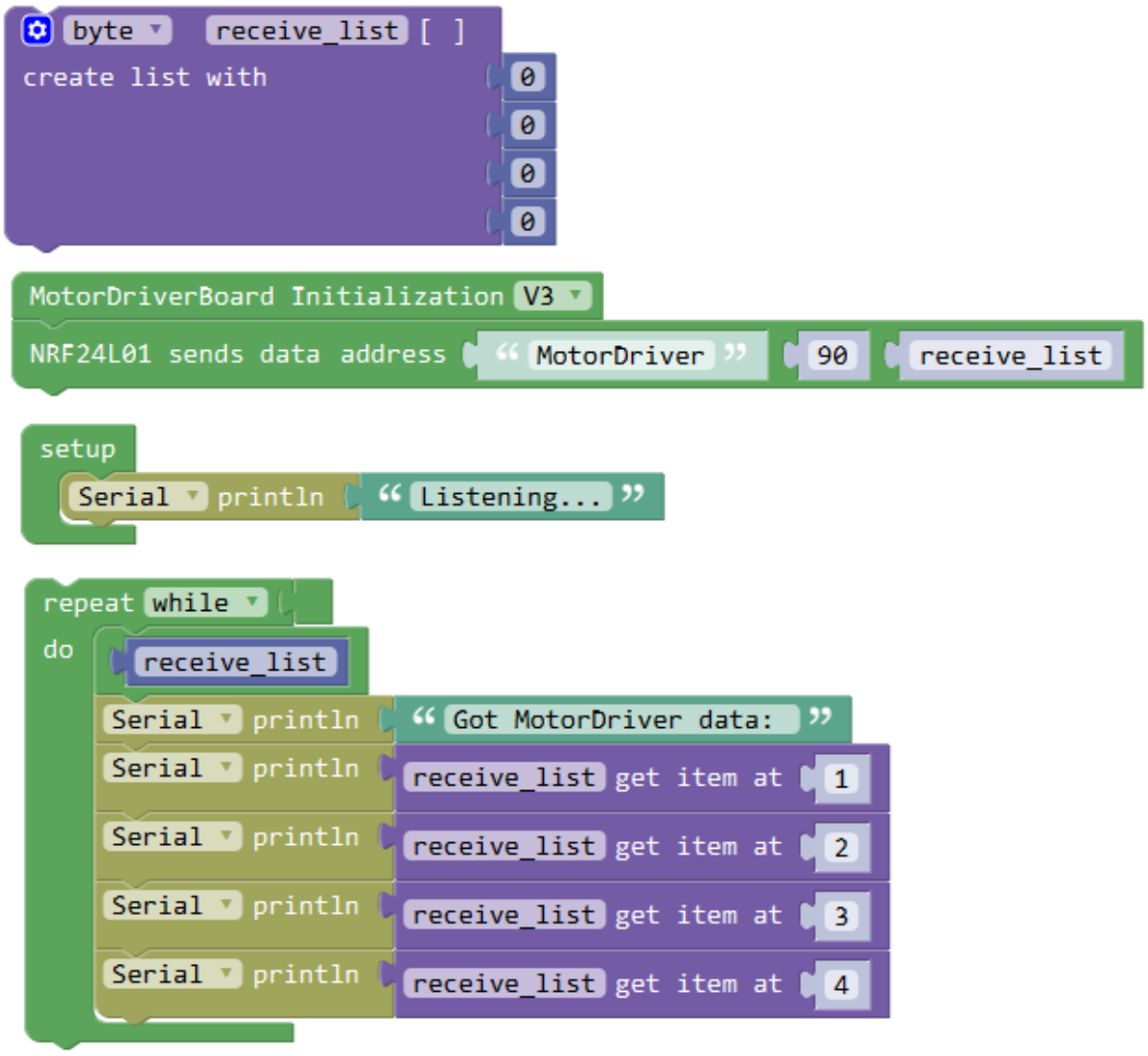
Figure 3-10-1-2

3.10.2 Nrf24L01program

When writing nRF24L01 + test with Mixly, we need to use two motordriverboards and insert two nRF24L01 + modules into the corresponding excuses, and then build two programs with building blocks in misic: one is to send data and the other is to receive data, as shown in the figure below. The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ hardware library test program \ NRF sending and NRF receiving

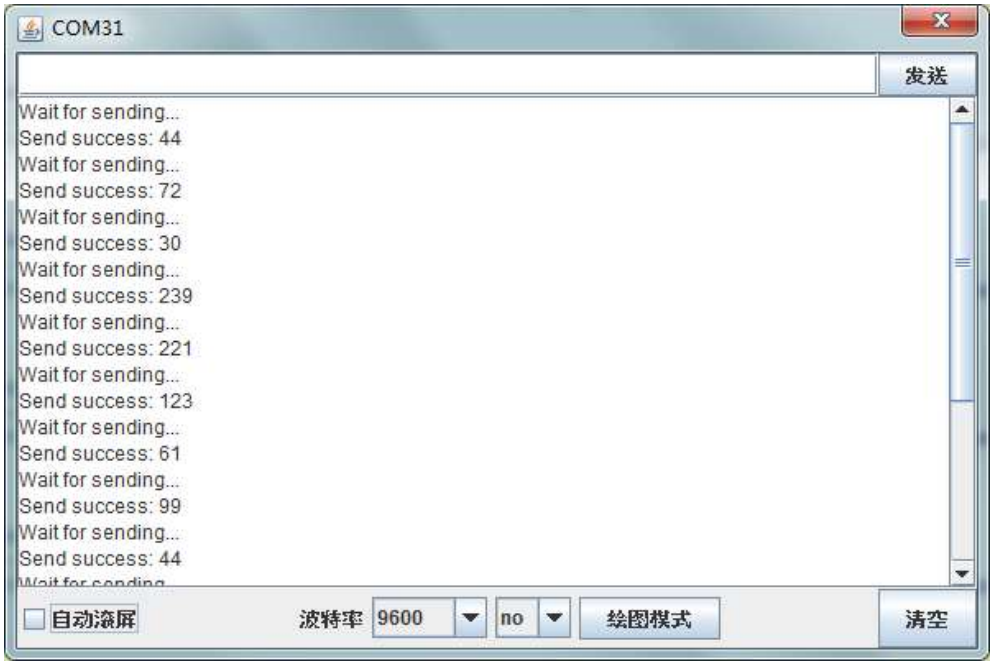


Sending program

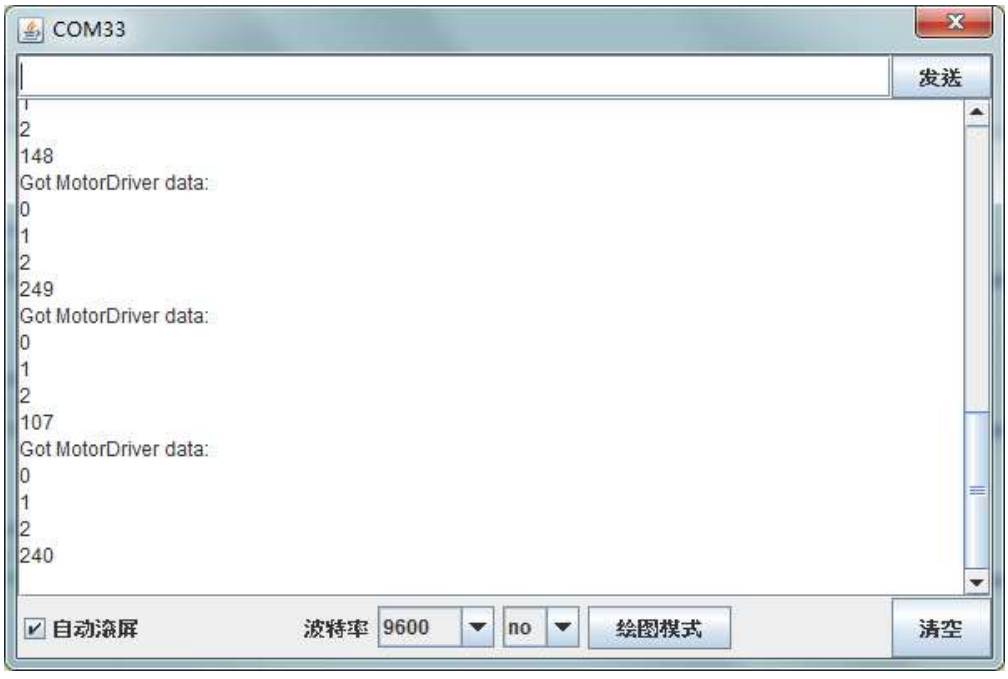


Receiving procedure

Compile with Arduino IDE, open the serial port monitor, and observe that the data is successfully sent from nRF24L01 to another nRF24L01



Sending program



Receiving procedure

Chapter 4 classic applications

In order to make it convenient for you to program your own four-wheel drive car, we have added some figure blocks (such as Figure 4-1) and PS2 control, nrf4l01 +, infrared remote control and Bluetooth control

sample programs in the library of motordriverboard. For those who have hobbies or needs, please refer to the classic applications in this chapter.

Figure 4-1

4.1 PS2 controls 4WD

The figure programming of PS2 has been introduced by turning on the RGB light by pressing the key. Now we extend the function of PS2 to operate a four-wheel drive car driven by the motordriverboard. We have done two example programs. The first one is to operate the four-wheel drive car by pressing the left and right keys of PS2 (figure 4-1-1 for example). The second one is to operate the four-wheel drive car by the rocker of PS2 (see figure 4-1-2). The program path is as follows: figure shaped programming \ mixly \ motordriverboard sample program \ classic application example program \ PS2 key operated 4WD program and PS2 rocker operated 4WD program

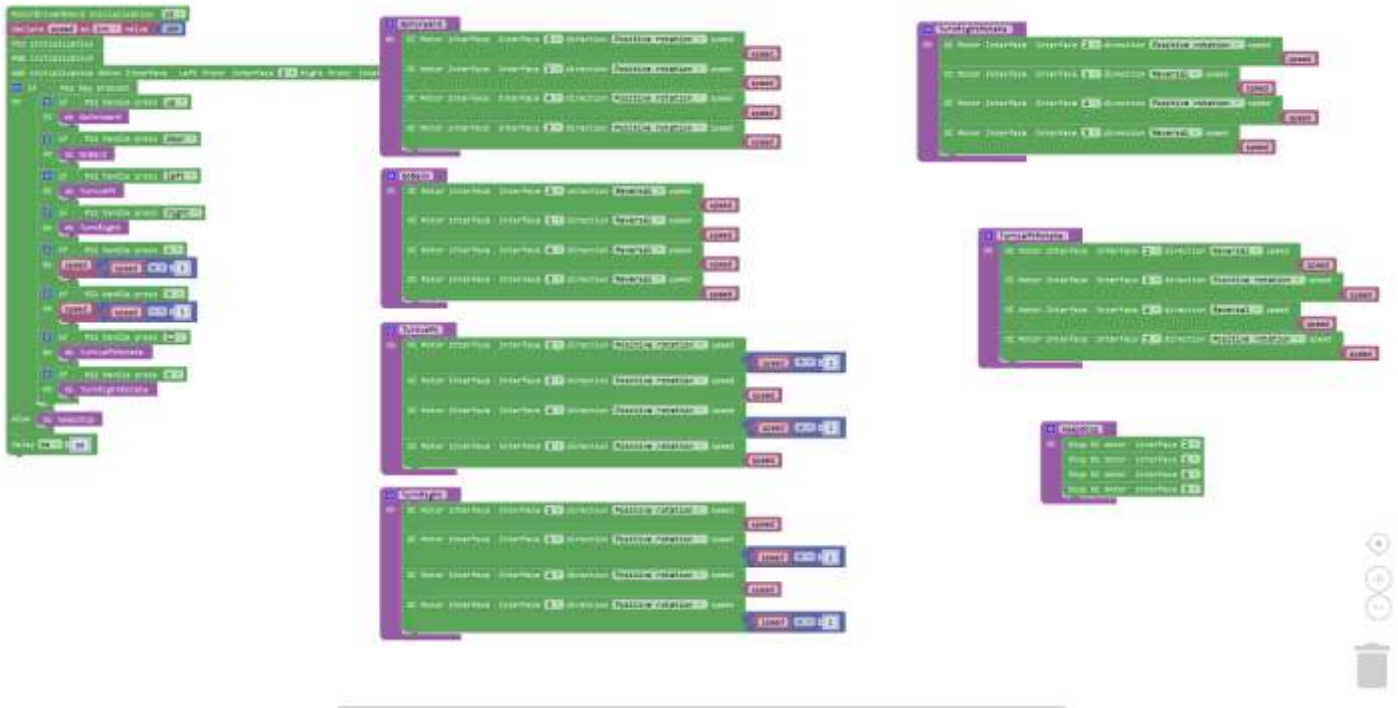


Figure 4-1-1 PS2 button to operate 4WD

The program idea of using PS2 key to control 4WD vehicle is: first, define the functions of forward, backward, left, right, left, right, right, and stop with the most original figure block of DC motor, and then realize these functions respectively by pressing the key, such as pressing the up key to execute the forward function, pressing the key to execute the backward function . in this way, you can write a program to control the 4WD car with the PS2 key.

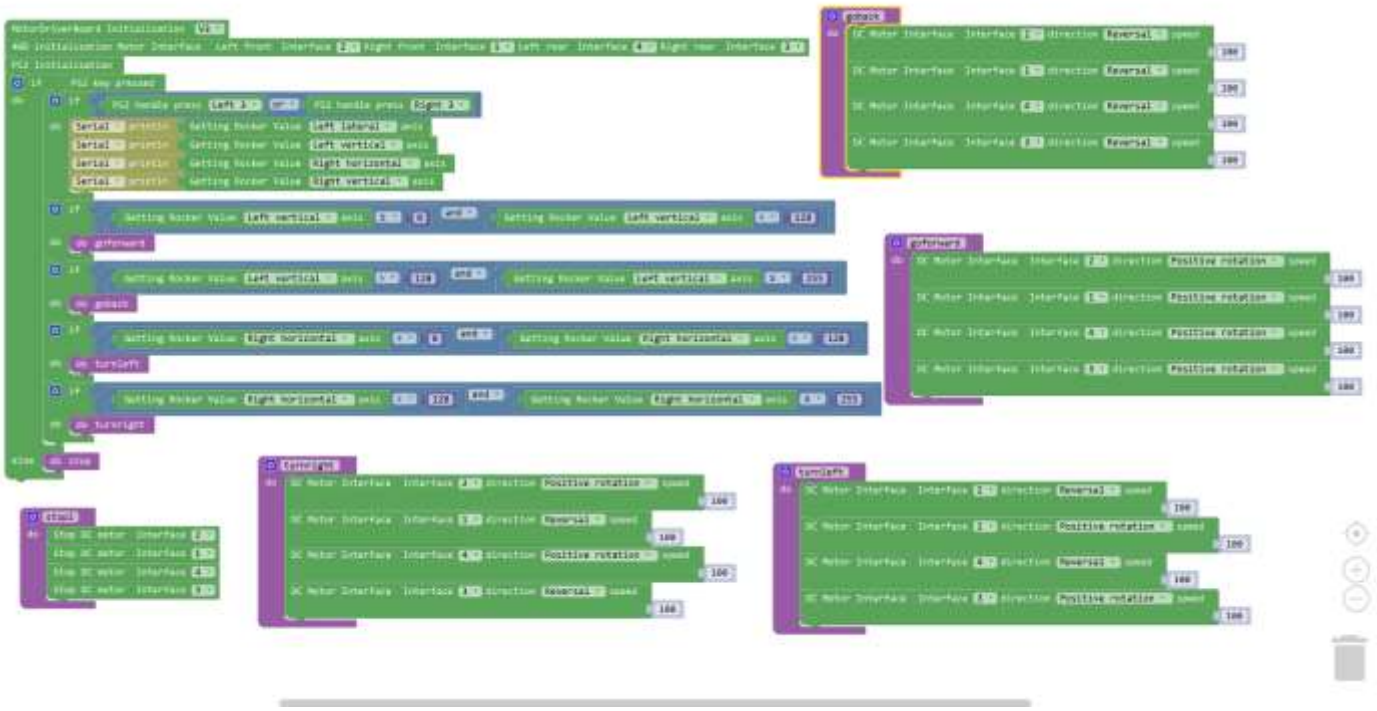


Figure 4-1-2 PS2 joystick control 4WD

The idea of using PS2 rocker control is: the left rocker is pushed up and down to control the forward and backward of the 4WD vehicle, the right rocker is pushed left and right to control the left and right turning of the 4WD vehicle, of course, we can add the functions we want in the example program, such as controlling the movement of the 4WD vehicle while making the RGB light of the vehicle on or making the buzzer ring, etc.

4.2 NRF24L01 controls 4WD

NRF24L01 + needs our wireless handle (such as figure 4-2-1) to operate the motordriverboard 4WD car. It controls the 4WD car by pressing the button of the wireless handle, which is similar to the principle of PS2. The difference is that the four buttons on the right of the wireless handle control the front, back, left, and left remote controls to obtain the angle value, so that the 4WD car turns according to the angle obtained. The program path is as follows: figure shaped programming \ mixly \ motordriverboard sample program \ classic application example program \ NRF remote control 4WD receiver program

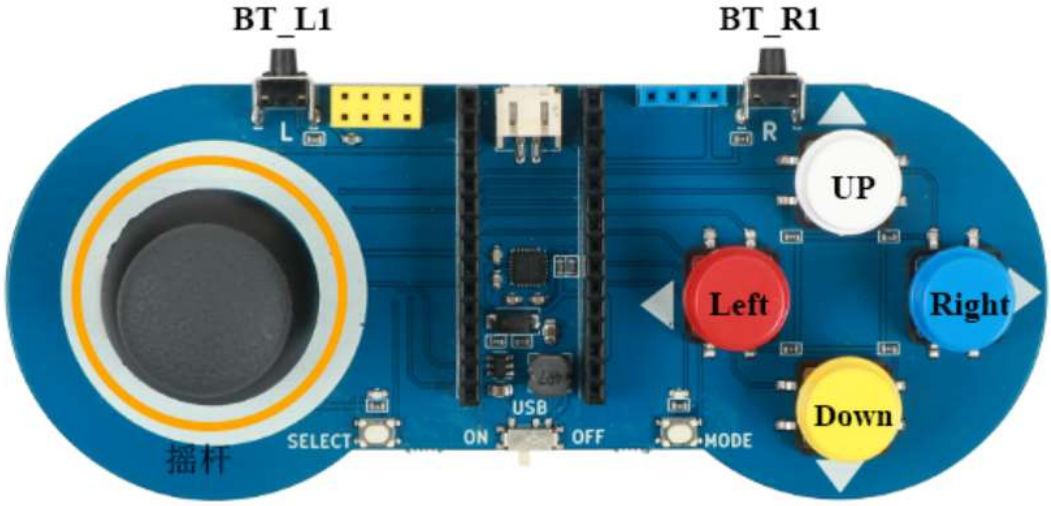
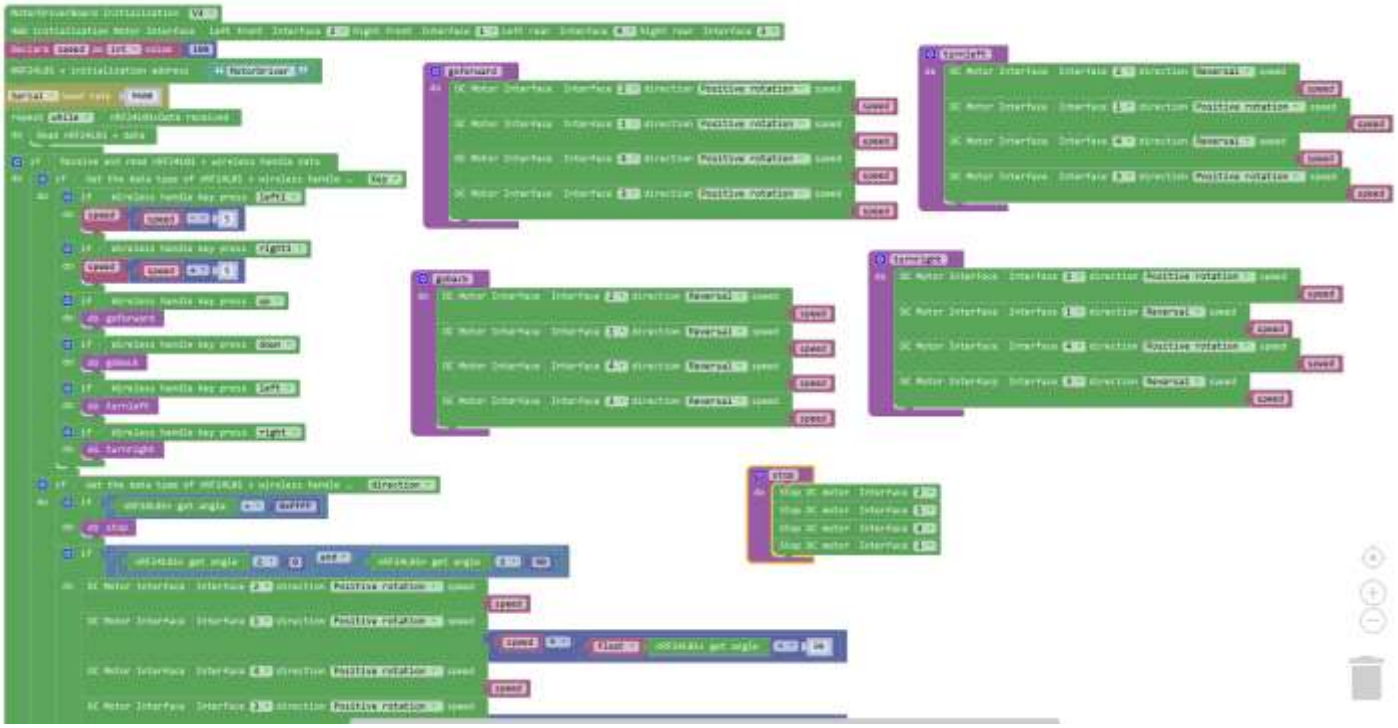


Figure 4-2-1



4.3 The infrared remote control controls the four-wheel drive vehicle

We have written the test program of infrared remote control, now we can make a program of infrared remote control to control the four-wheel drive vehicle (Figure 4-3-1). The program path is as follows: figure shape programming \ mixly \ motordriverboard sample program \ classic application example program \ infrared remote control wipe empty 4WD program

Figure 4-3-1

4.4 Bluetooth control of 4WD

The motordriverboard motor driver board is equipped with a Bluetooth socket. Insert the Bluetooth module into the socket, and you can use the Bluetooth connection of the mobile phone. First, we download the emakefun mobile app, open the app (Figure 4-4-1 for example), and select a four-wheel drive car (Figure 4-4-2 for example) for Bluetooth connection. We need to write a Bluetooth control program for the driver board (Figure 4-4-2 for example) You can enter the app Bluetooth control interface to control the 4WD. The program path is as follows: figure shaped programming \ mixly \ motordriverboard sample program \ classic application example program \ Bluetooth control 4WD program

Figure 4-4-1

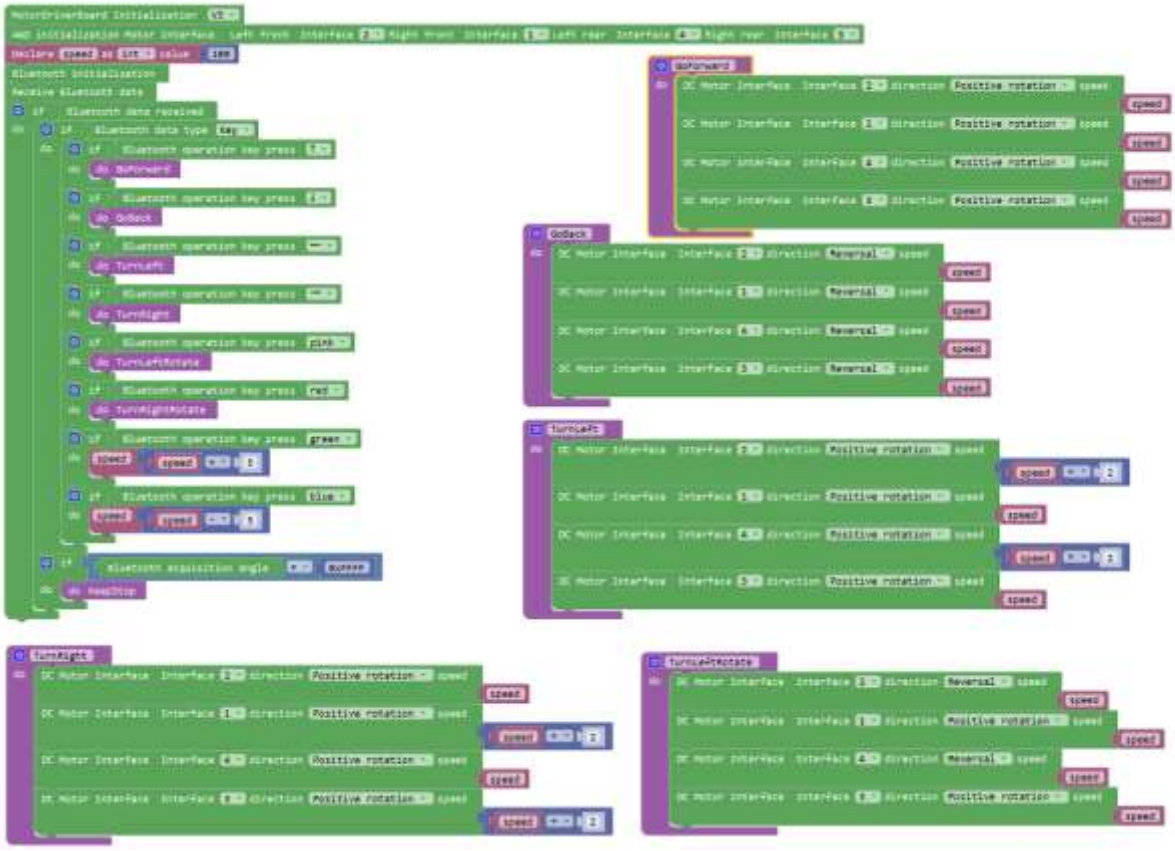


Figure 4-4-2

4.5 PS2 manipulator 4WD

The motordriverboard supports 8-way steering gear. We use 4 steering gear for the robot arm matched with the motordriverboard. The program principle of PS2 controlling the robot arm 4WD car is to add the program of using the rocker to control the steering gear on the program of PS2 controlling the 4WD car, realizing 8 keys to control the driving direction and speed of the 4WD car. The left rocker controls the left and right movement of the robot arm, and the right rocker controls the machinery When the arm moves up

and down, the first side key on the left is to close the pliers of the mechanical arm, and the first side key on the right is to open the pliers. The program path is as follows: figure shaped programming \ mixly \ motordriverboard sample program \ classic application example program \ PS2 control the four-wheel drive program of the mechanical arm