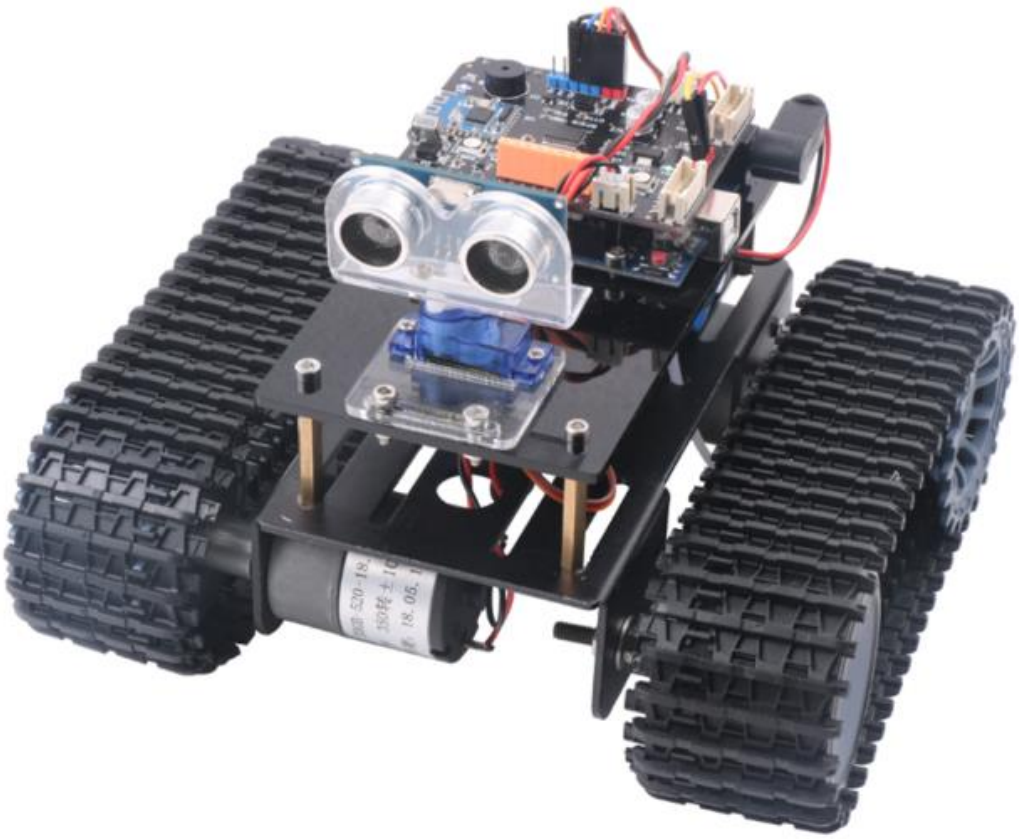
Panther-Tank MagicBlock Graphical Programming Course V.1.0



Revision of edition

date	version number	description	author
2019-8-20	**V.1.0**	**created documents**	**Twisted**

content

Chapter 1 Understanding Panther-Tank and Its Programming Environment 5
1.1 Introduction of life 5
1.2 Summary 5
1.2.1 Panther-Tank introduction 5
1.2.2 Panther-Tank Function list 5
1.3 Panther-Tank Hardware introduction 6
1.3.1 Introduction to analogy 6
1.3.2 Introduction of Main Control Board 6
1.3.3 Introduction to Extension Board 7
1.4 Panther-Tank Introduction to Software Compiling Environment 8
1.4.1 Software installation 8
1.4.2 Introduction to Compiler Environment 9
1.5 Introduction to Label Area 10
1.6 Panther-Tank Introduction of building blocks 12
Chapter 2 Initial Knowledge Programming 16
2.1 Hello world 16
2.1.1 Add Panther-Tank Library 16
2.1.2 First Experience of Programming 20
2.2 MagicBlock and Panther-Tank Connection Steps 22
2.3 Upload program to Panther-Tank robot 23
2.4 RGB LED 28
2.5 Buzzer 29
Chapter Three Robots Move 30
3.1 DC motor 30
3.1.1 Principle of DC Motor 30
3.1.2 DC motor test 30
3.1.3 Programming control and debugging the robot to take the square route. 31
3.1.4 Discover the Rule of Program and Optimize Program 32
3.2 Ultrasound module 33
3.2.1 Principle of Ultrasound 33
3.2.2 Usage of Ultrasound Module 33

3.2.3 Testing of the Ultrasound Obstacle Avoidance Module....34
3.2.4 Writing Robot Program for Tango Dance....34
3.2.5 Optimizing Dancing Robot Program....36
3.3 Ultrasonic obstacle avoidance robot....36
3.3.1 Operating Principle of Steering Engine....36
3.3.2 Steering Engine Testing....37
3.3.3 Programming of Obstacle Avoidance Robot....38
Chapter IV Infrared Telecontrol Robot....40
4.1 Principle of Infrared Remote Control....40
4.2 The Use of Remote Controller....40
4.3 Infrared remote control test....40
5.4 Programming of Remote Control Robot....42
Chapter 5 Bluetooth Remote Control Robot....44
5.1 Bluetooth Control Principle....44
5.2 Connection Method of Bluetooth Module and Anter-Tank Robot APP KeywishBot....44
5.3 Bluetooth module testing....46
5.4 Programming Thought of Bluetooth Remote Control Robot....47
Chapter 6 PS2 Remote Control Robot....47
6.1 Principle of PS2 remote control handle....47
6.2 PS2 remote control test....49
6.3 Programming Thought of PS2 Remote Control Robot....49

Chapter 1 Understanding Panther-Tank and Its Programming Environment

1.1 Introduction of life

Hello, everyone. Welcome to the world of Panther-Tank robots!

What is a robot? Robots don't necessarily have to be long as human beings call them robots. Generally, robots can perform their work automatically. They can be commanded by human beings and run pre-programmed programs. Such things can be called robots.

What are the responsibilities of robots? Robot, originally Robo, means slave, that is, human servant, so the duty of robot is to help human work.

1.2 Summary

1.2.1 Panther-Tank introduction

Through the study of this tutorial, we will experience the interactivity of the physical world and software. Panther-Tank consists of three components: Panther-Tank smart car, Arduino UNO master board and MagicBlock graphical programming software. MagicBlock is a graphical programming learning software for STEAM education, which is developed based on Scratch 2.0. It not only enables users to create interesting stories, games, animations and other works, but also supports Arduino hardware programming. We can use this software to control the Arduino UNO motherboard on Panther-Tank, so that Panther-Tank can issue various commands and control it. In this software, programming is no longer tedious code, using graphical representation and drag-and-drop interaction to complete the core logic of programming. The whole programming is like building an interesting building block toy. Through Magic Block's stage area or Panther-Tank's behavior, we can intuitively see the actual effect of various script designs.

1.2.2 Panther-Tank Function list

1) Obstacle avoidance function of ultrasound
2) Ultrasound + Infrared Barrier Avoidance Function
3) Automatic Tracking Function
4) Infrared remote control function
5) Bluetooth remote control function
6) PS2 remote control function
7) Bluetooth Remote Mode Switching Function
8) Ligh Finding Function

1.3 Panther-Tank Hardware introduction

1.3.1 Introduction to analogy

The hardware part of the Panther-Tank robot we use is a Panther-Tank robot, which consists of a body model built by acrylic board, various sensors, motor wheels, control panel and power supply battery, as shown in Figure 1-3-1-1.

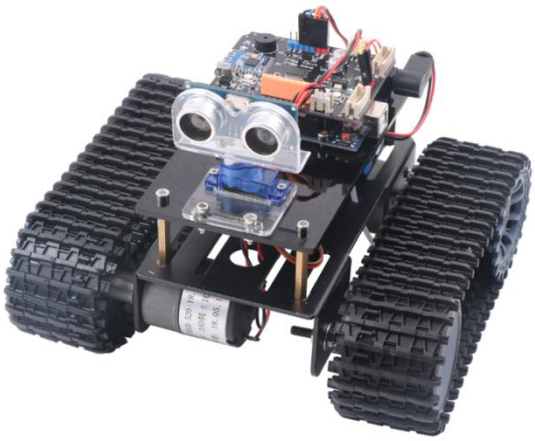


Figure 1-3-1-1

In this robot, every part of it is similar to the structure of our human body and plays a vital role. It is similar to human beings, as shown in Table 1-3-1-2.

Robot	**Human**	**Function**
Control panel	Brain	Accept instructions, process information, and give instructions to control limbs and organs
Motors and wheels	limb	Responsible for limb movements
Battery	heart	Provide motivation
sensor	Facial features	Acceptance of perceived information
electronic circuit	nerve	Connect each part of the limb to form a whole.

Table 1-3-1-2

1.3.2 Introduction of Main Control Board

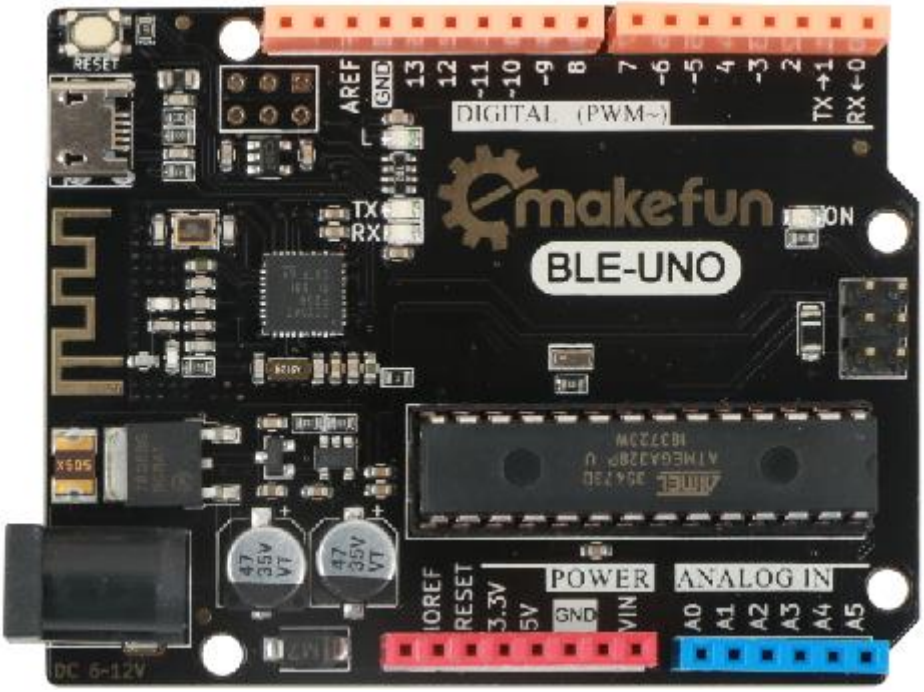


Figure 1-3-2-1

1.3.3 Introduction to Extension Board

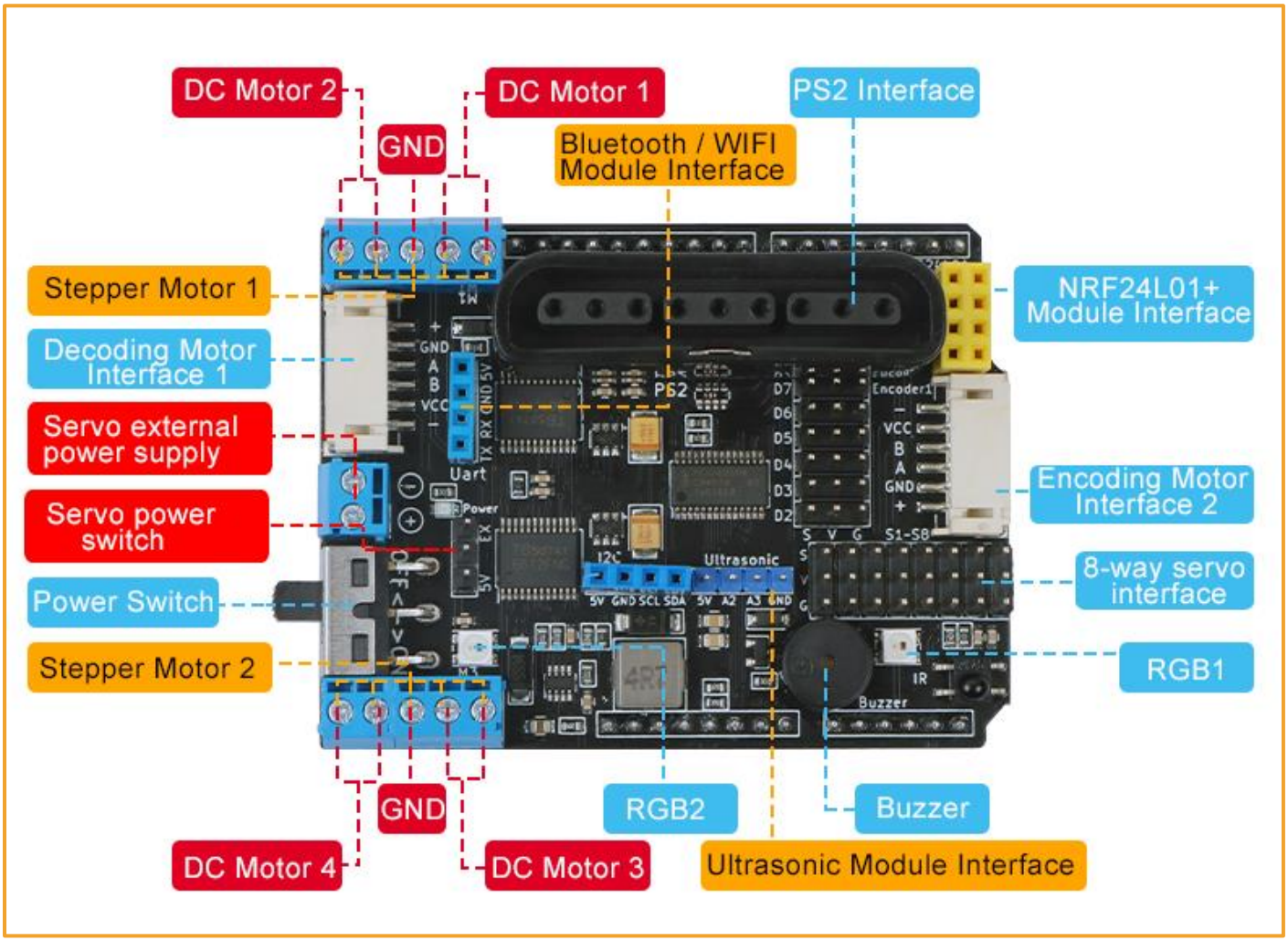


Figure 1-3-3-1

1.4 Panther-Tank Introduction to Software Compiling Environment

1.4.1 Software installation

The installation package of MagicBlock, a graphical programming software based on scratch 2.0, has been placed in the Panther-Tank product data. It can be installed directly.

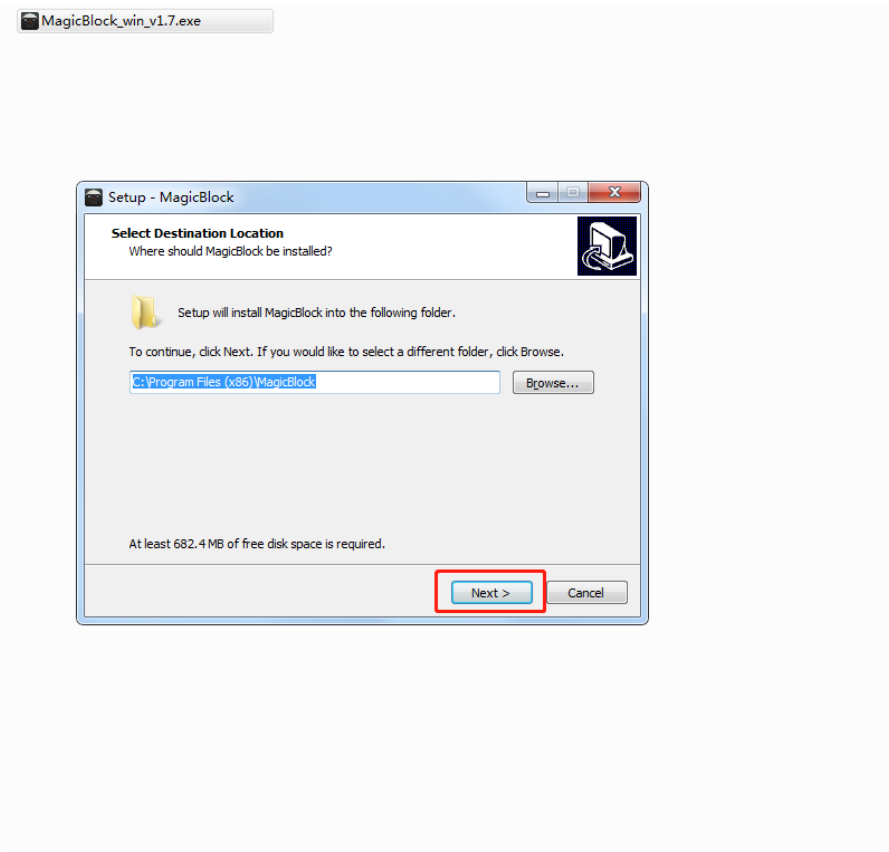


Figure 1-4-1-1

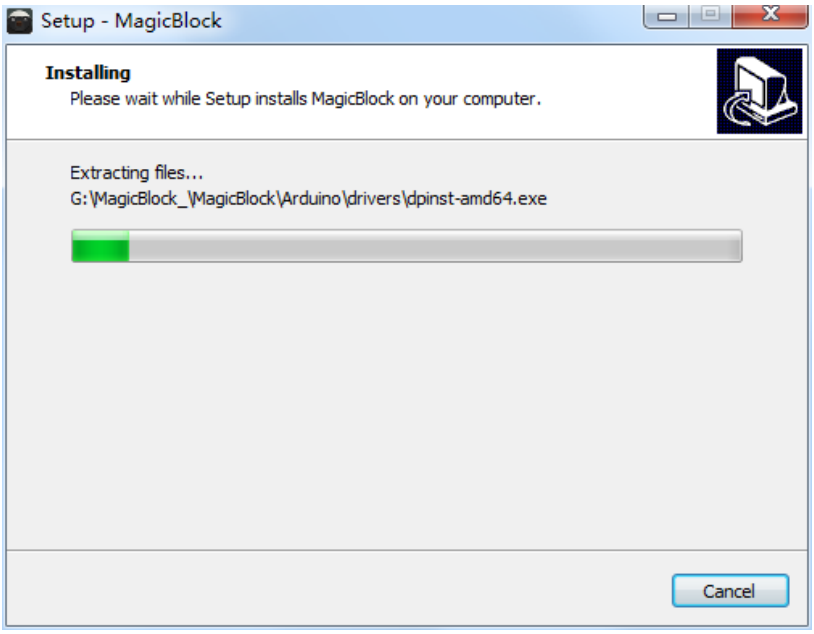


Figure 1-4-1-2

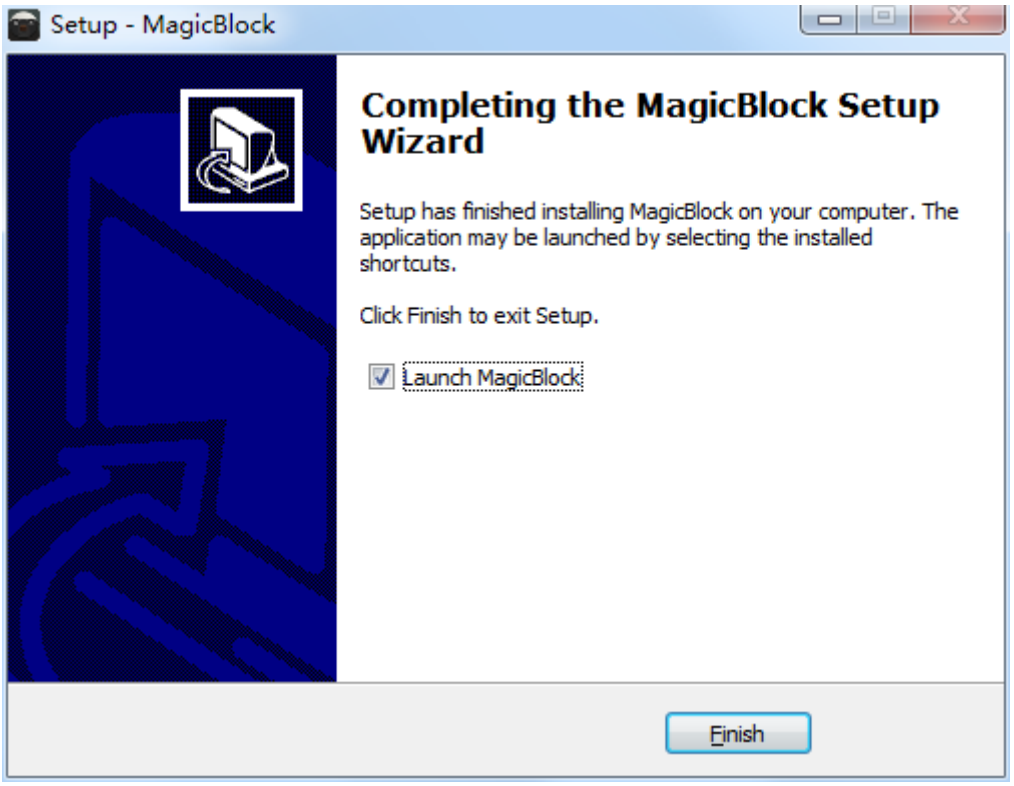


Figure 1-4-1-3

1.4.2 Introduction to Compiler Environment

We know that in order for a robot to move, besides the hardware cooperation, the more important thing is to write a program for it. We already know the hardware part of the robot, and then we will know its

software part. Panther-Tank's software is programmed on MagicBlock, a graphical programming software. With this software, we can control the robot by writing all kinds of commands we want it to execute. MagicBlock's software interface is shown in Figure 1-4-2-1.

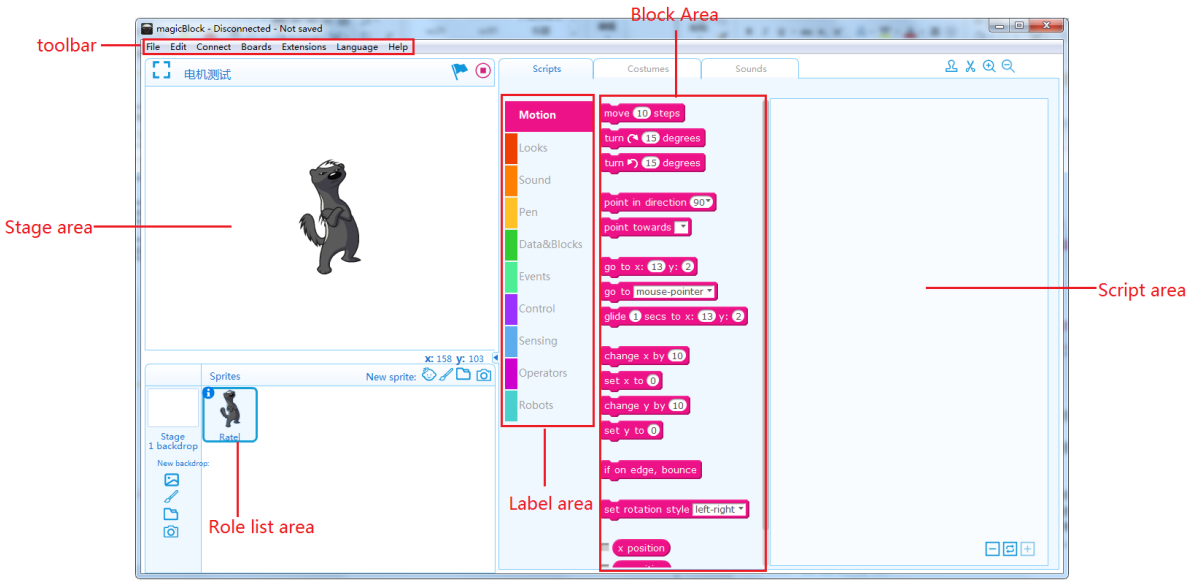


Figure 1-4-2-1

- **Toolbar：** For project files, software interface mode, serial connection and upload program, control board selection, software using language, software update operation area.
- **Stage area：** The role and role in the work, the interaction area between the role and the user, is the place to show the running effect of the program.
- **Roles list area：** All the role prototypes display areas, where you can see the name of the role, rotation direction, location, and so on.
- **Label area：** Contains script tab, styling tab, voice tab, which can operate on script, styling and voice of characters.
- **Block area：** The same type of building blocks are divided into the same module and given the same color. Each building block represents a control instruction.
- **Script area：** Programming area, the building blocks stacked in the script area can be programmed.

1.5 Introduction to Label Area

MagicBlock's label area is mainly divided into scripts (as shown in Figure 1-5-1), shapes (as shown in Figure 1-5-2) and sounds (as shown in Figure 1-5-3). The scripts are mainly control blocks to perform actions; under the model label, you can draw some graphics to add to the program; under the sound label, you can record some sounds by yourself or put them into the program. The main purpose of Panther-Tank is to control blocks under script labels.

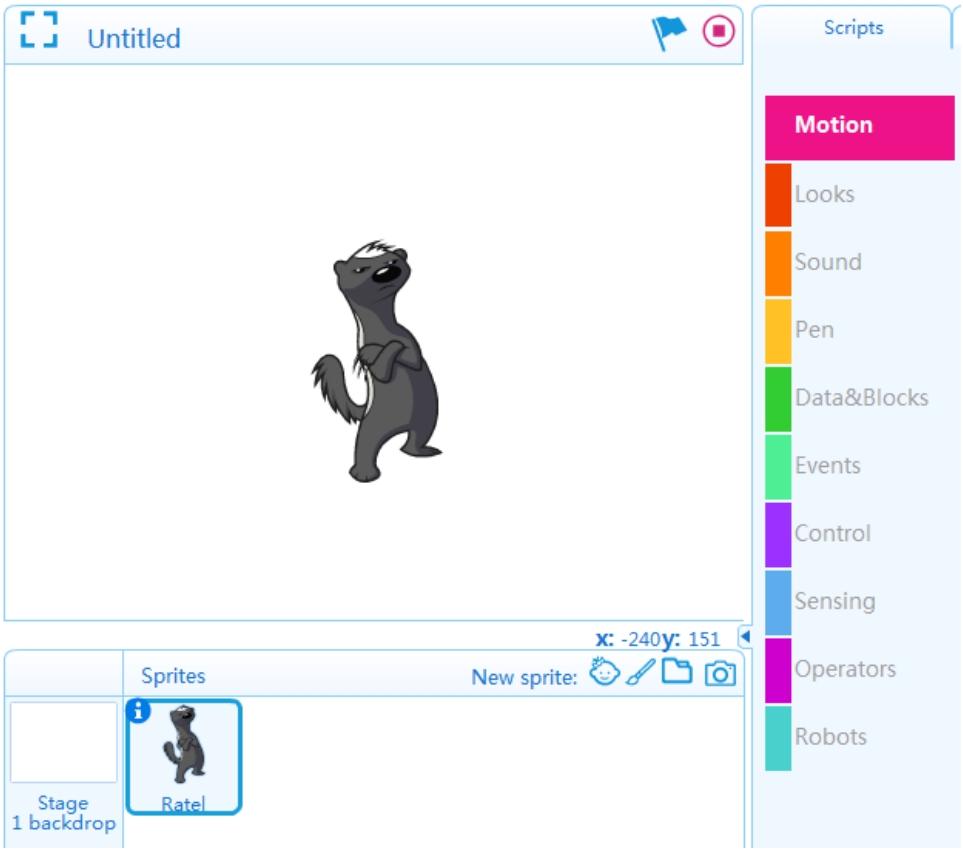

Untitled
Scripts
Motion
Looks
Sound
Pen
Data&Blocks
Events
Control
Sensing
Operators
Robots
x: -240 y: 151
Sprites
New sprite:
Stage
1 backdrop
Ratel

Figure 1-5-1

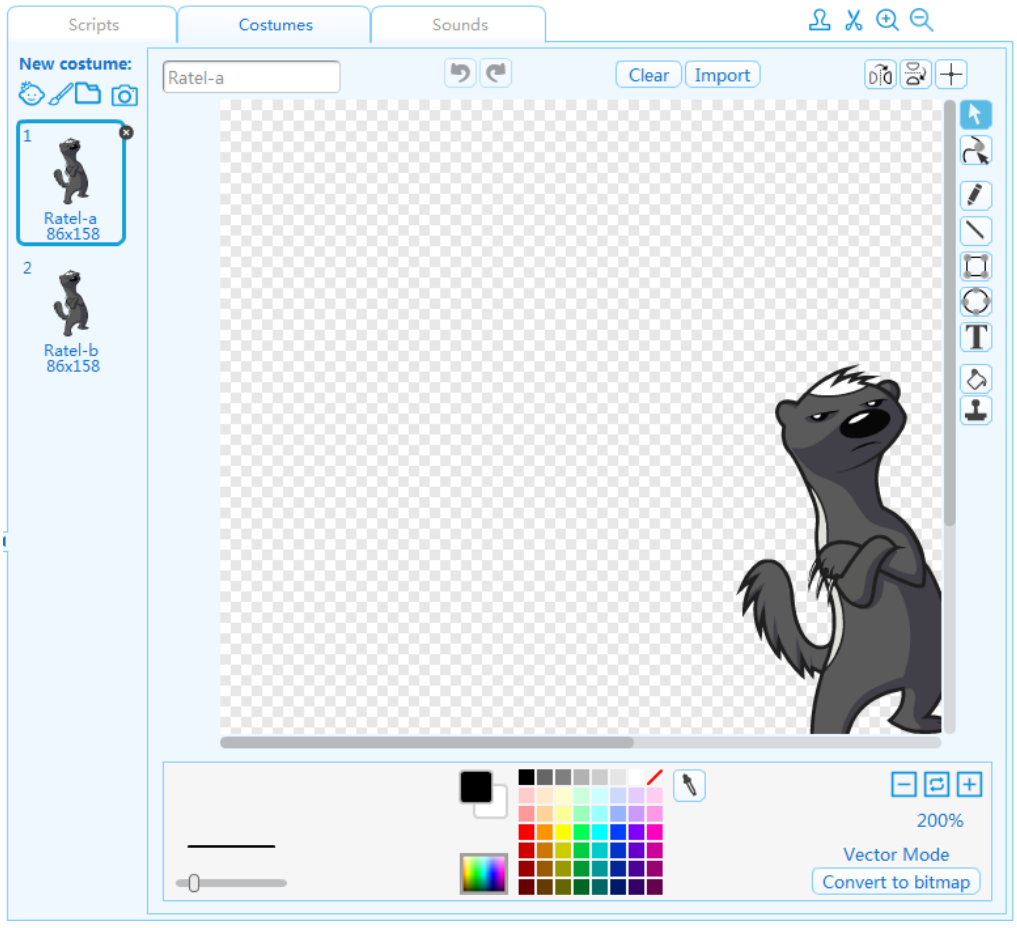

Scripts
Costumes
Sounds
New costume:
Ratel-a
Clear
Import
1
Ratel-a
86x158
2
Ratel-b
86x158
200%
Vector Mode
Convert to bitmap

Figure 1-5-2

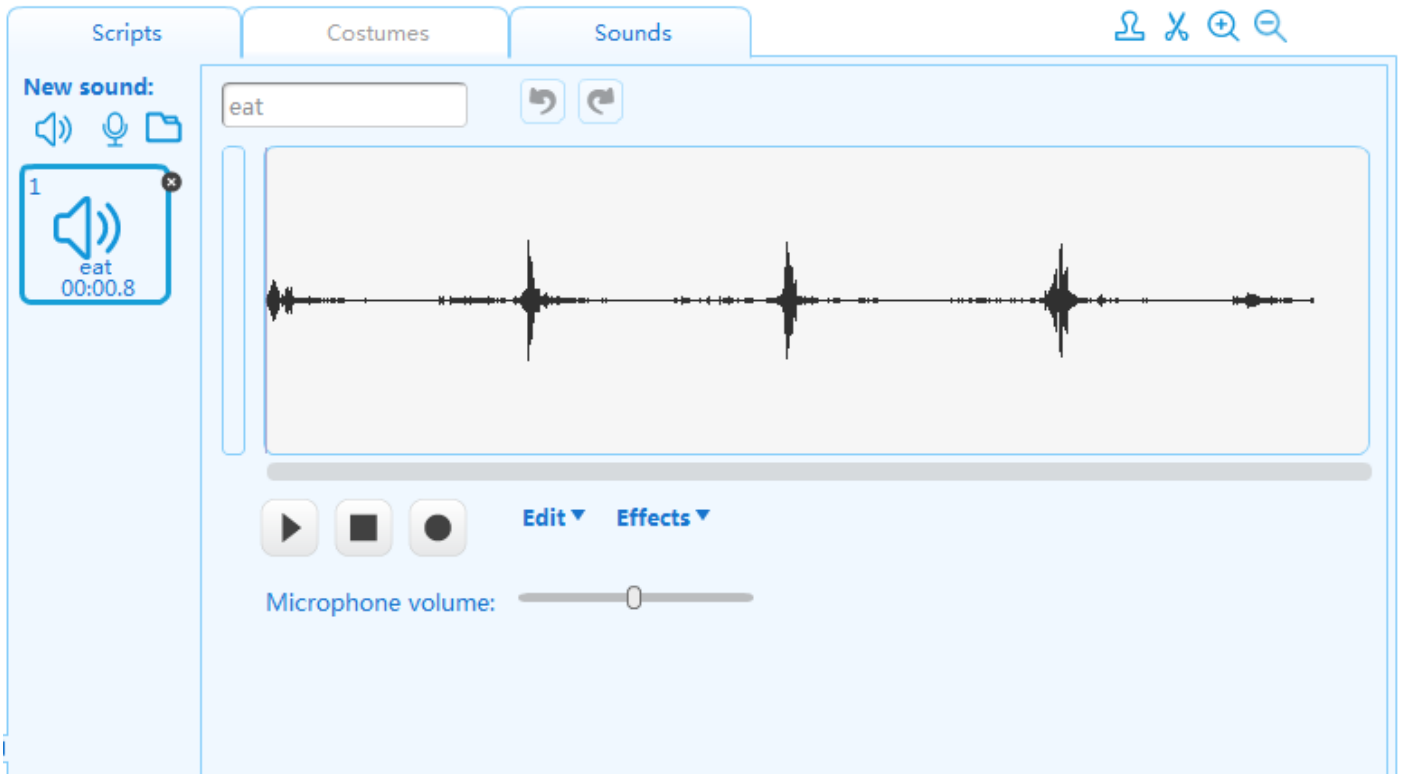


Figure 1-5-3

1.6 Panther-Tank Introduction of building blocks

MagicBlock's script label has 10 types of building blocks, including action, appearance, sound, brush, event, detection, which are some effects and data. Starterbox will not be used. If you are interested, you can try it out by yourself. Here we do not go into any more details. We mainly understand the three types of building blocks: control, digital and logical operations, and robotic module.

1) The control blocks are all the building blocks that control the execution process of the program (Fig. 1-6-1), the main program.

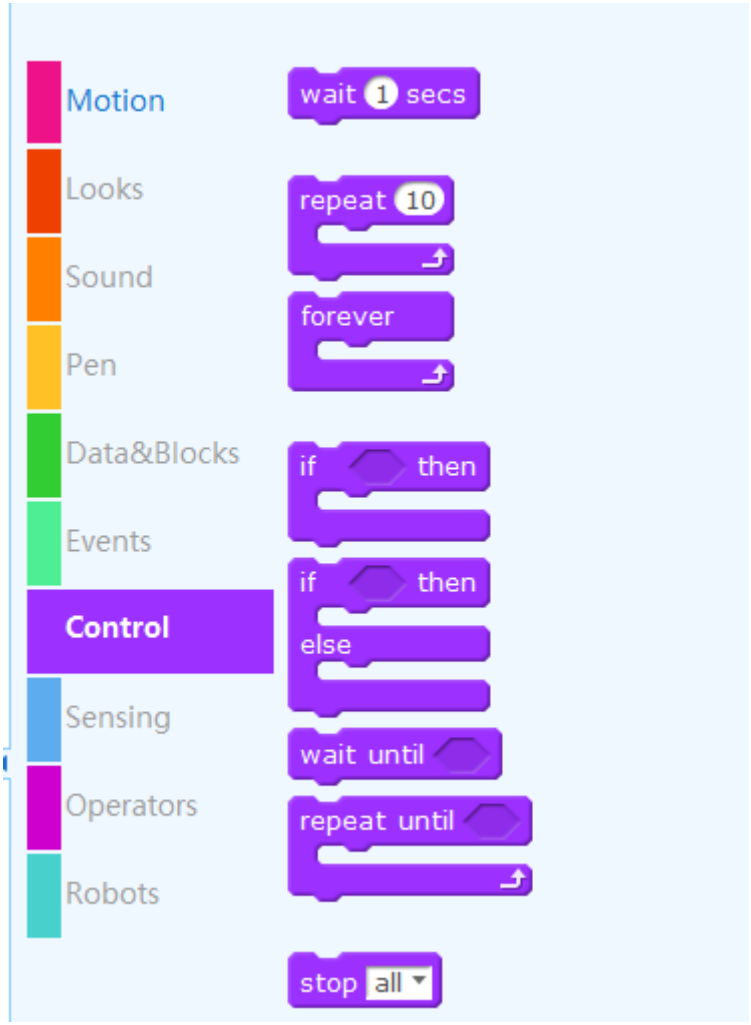
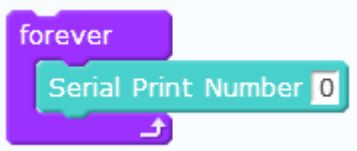


Figure 1-6-1

For example， 

This building block means to repeat the procedure in the building block all the time. Another example 

That is, the robot has been printing the number 0 in the serial port.

2) The main function of building blocks of numeric and logical operation types is to do mathematical operation as a condition of judgment, and to compare the size and logical judgment with, or with, or without, as shown in Figure 1-6-2.

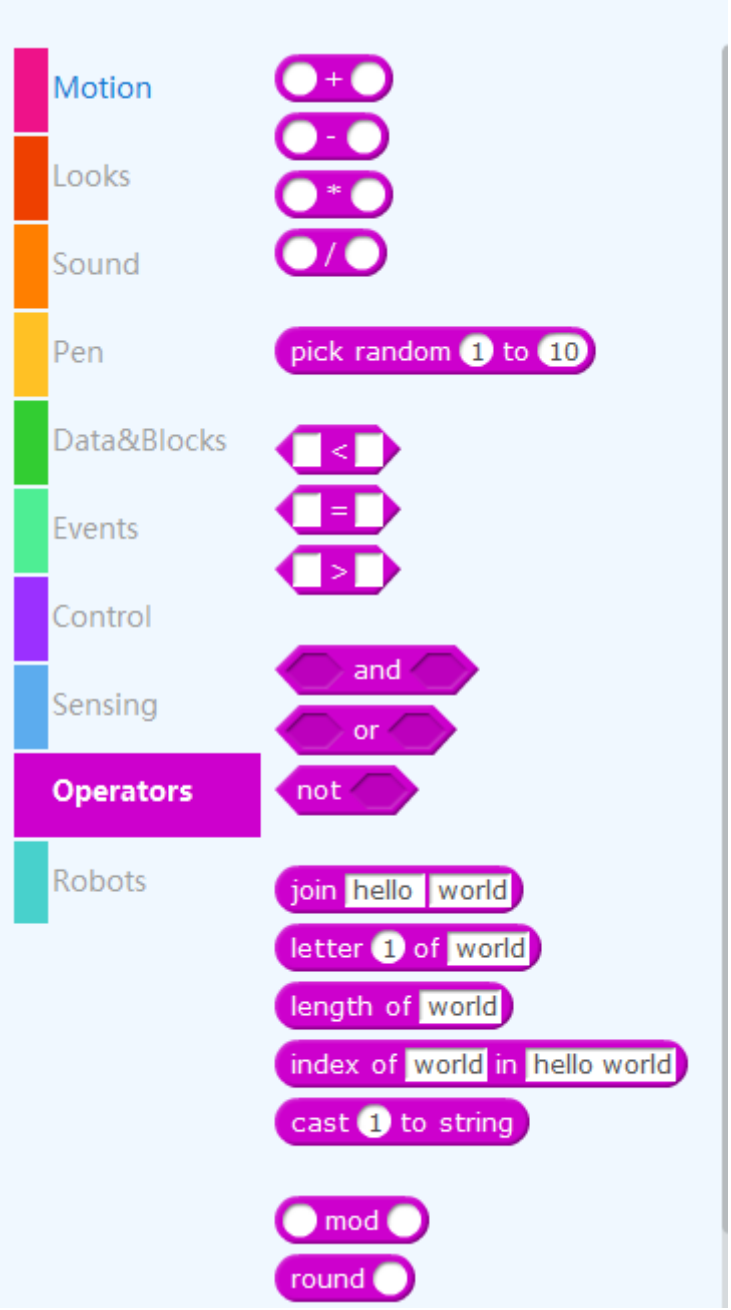
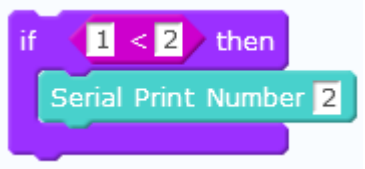
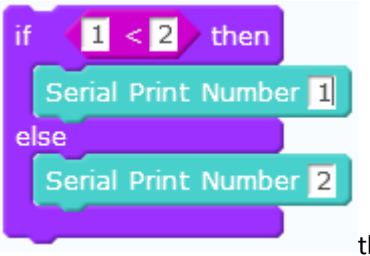


Figure 1-6-2

For example，  are the operations of judging size, when they are used together with control building blocks, digital and logical operations and robotic module building blocks.， You can write such a program.



， It means to judge  Is this condition valid? If the condition holds,



then the serial port printing number 2 is executed. Expanding, this program means

Is this condition valid? If the condition is valid, the serial port print number 1 is executed, and if the condition is not valid, the serial port print number 2 is executed.

3) The main function of the robot module is to control the robot to perform corresponding actions, such as forward, backward, steering and so on, as shown in Fig. 1-6-3 and Fig. 1-6-4.

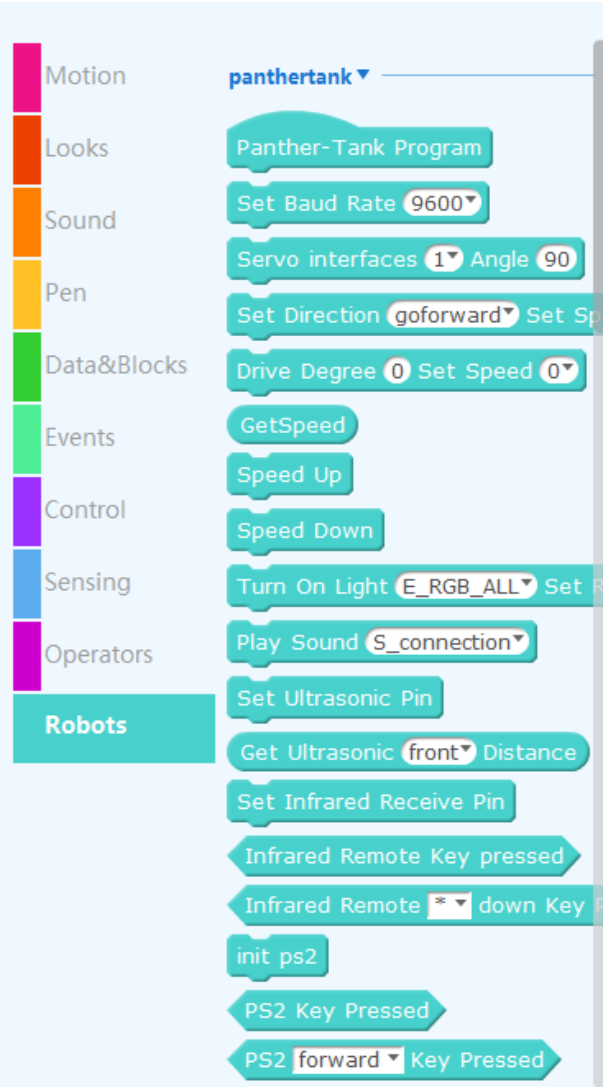


Figure 1-6-3

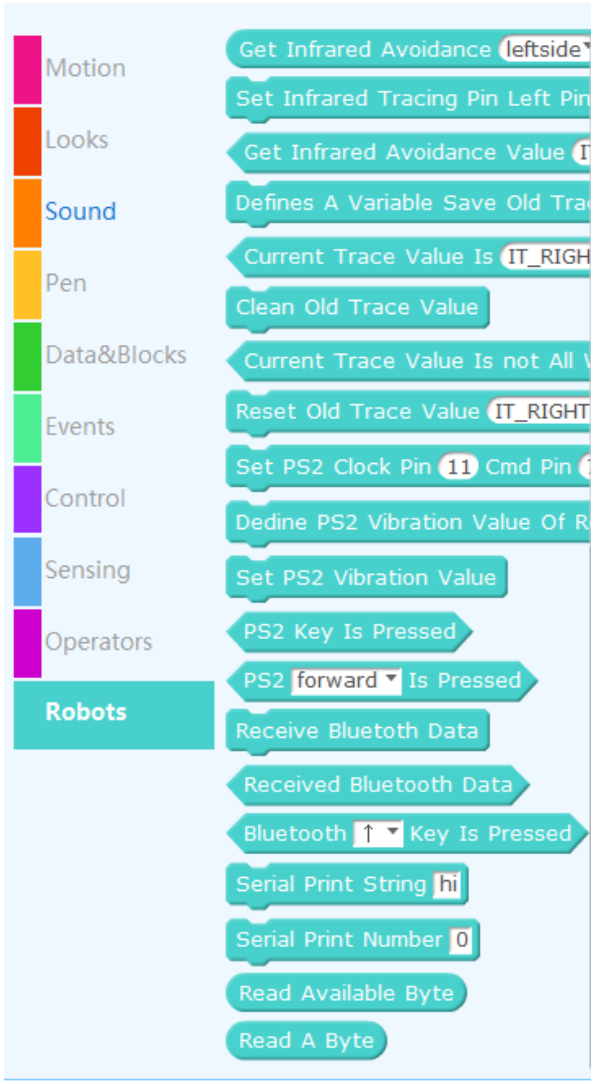


Figure 1-6-4

Chapter 2 Initial Knowledge Programming

2.1 Hello world

If we want the Panther-Tank robot to move, we need to store instructions (programs) in its brain (control board) beforehand for the Panther-Tank robot. How to compile instructions for the Panther-Tank robot? Now let's take you through the experience of writing a Panther-Tank robot print Hello World program.

2.1.1 Add Panther-Tank Library

Before programming, we need to add the Panther-Tank library. The steps are as follows:

1) Download the Panther-Tank library and save it on your computer. File name: Panther-Tank.zip
2) Open MagicBlock software and click on "Extension Manager", as shown in Figure 2-1-1-1
3) Add extensions in the lower right corner of the extension manager center, as shown in Figure 2-1-1-2;

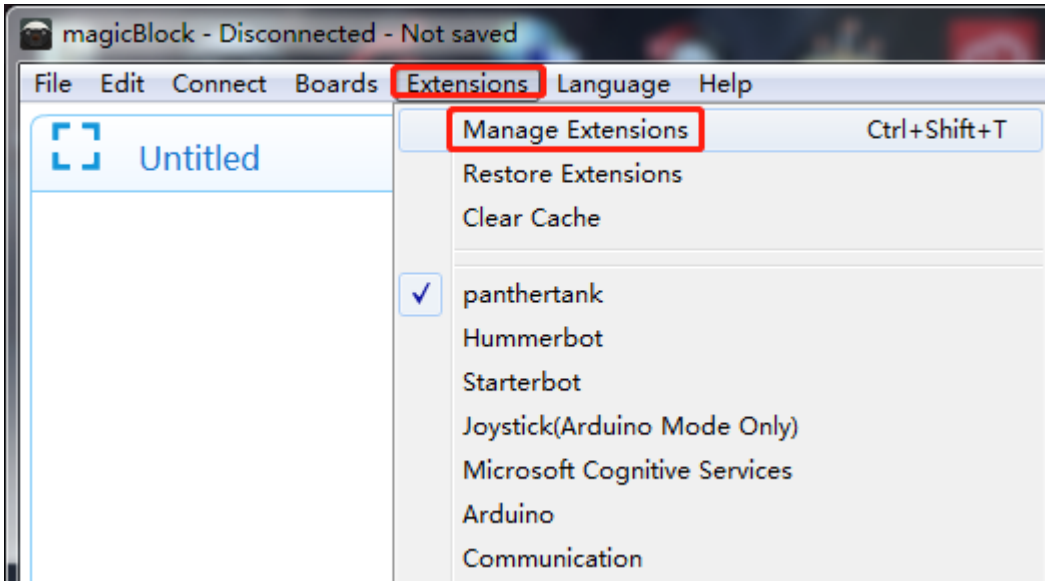


Figure 2-1-1-1

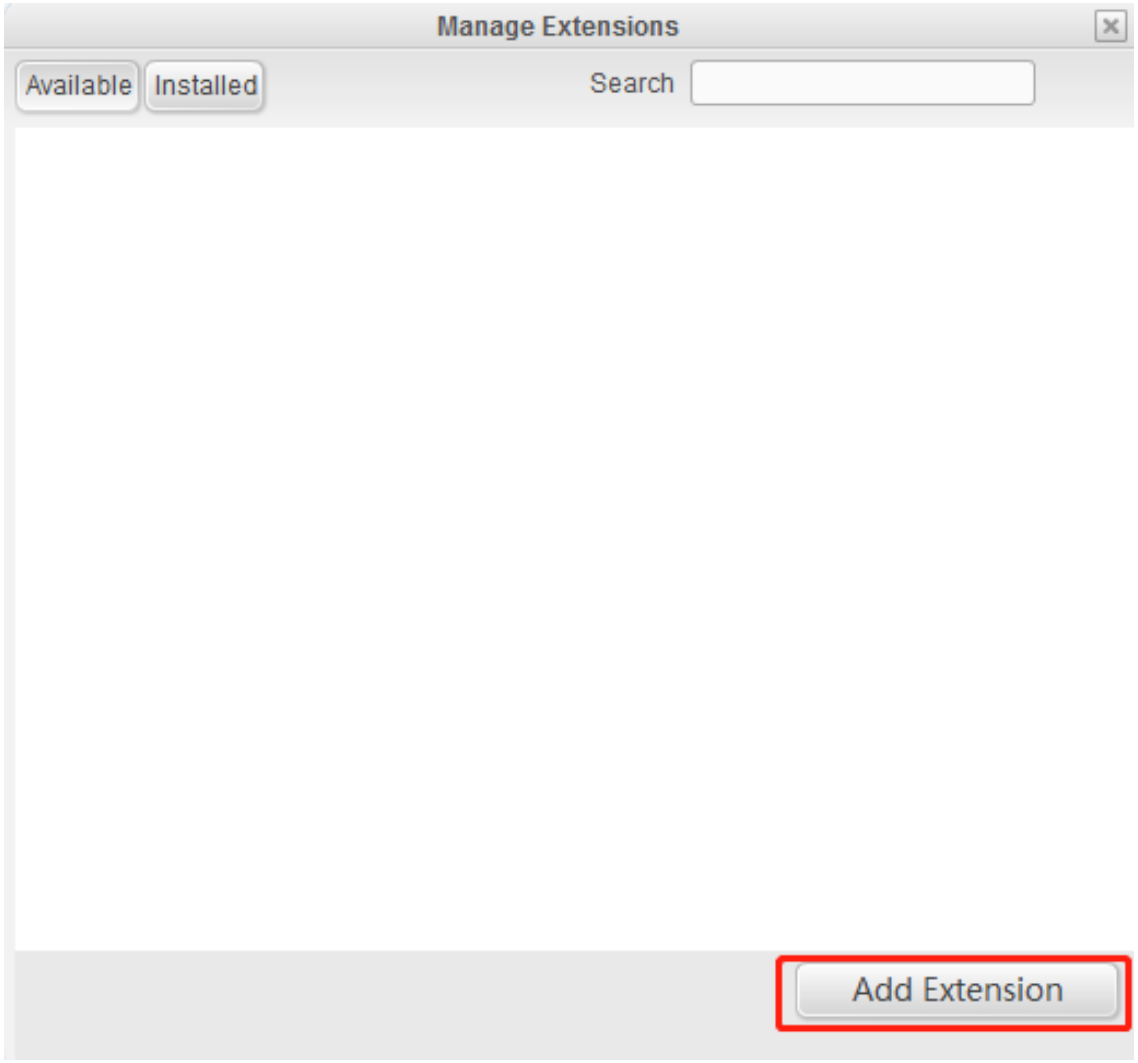


Figure 2-1-1-2

4) Select "Zip. file" for the file type, then select "Panther-Tank. zip" and click "Open", as shown in Figure 2-1-1-3.

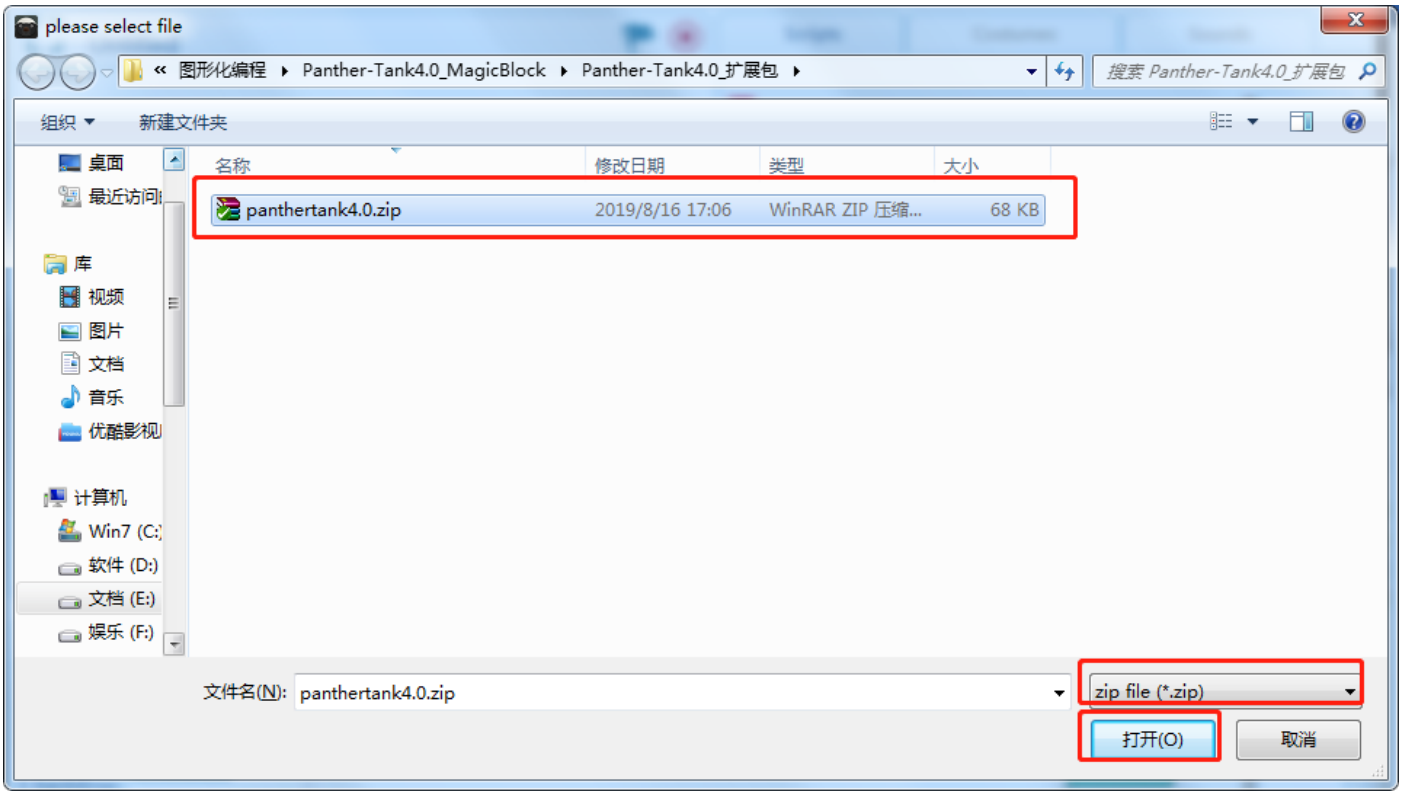


Figure 2-1-1-3

5) In "Extension Manager", click "Installed", and you will see that the Panther-Tank library has been successfully added, as shown in Figure 2-1-1-4;

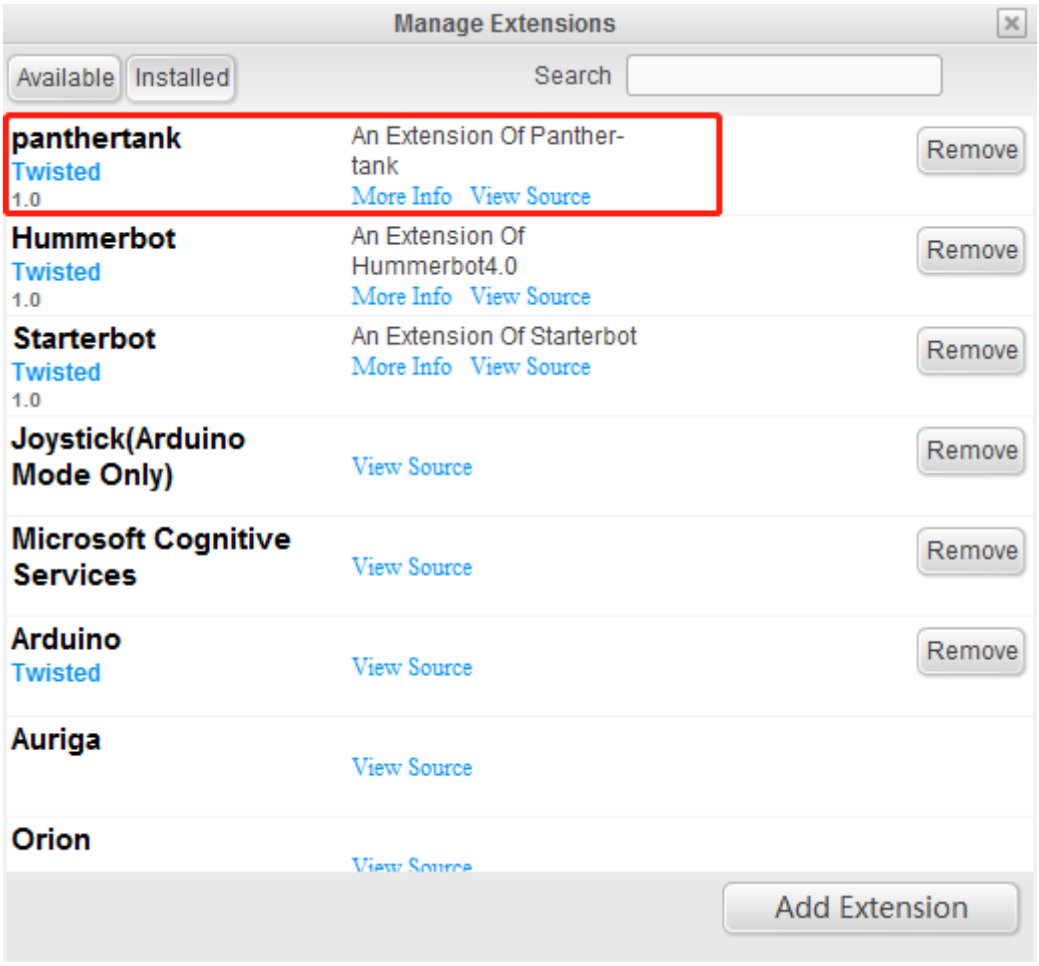


Figure 2-1-1-4

6) Click on "Extension", select "Panther-Tank", and then click "Script -> Robot Module", the Panther-Tank Block Graphics Programming Block will be displayed in the building block area, as shown in Figure 2-1-5.

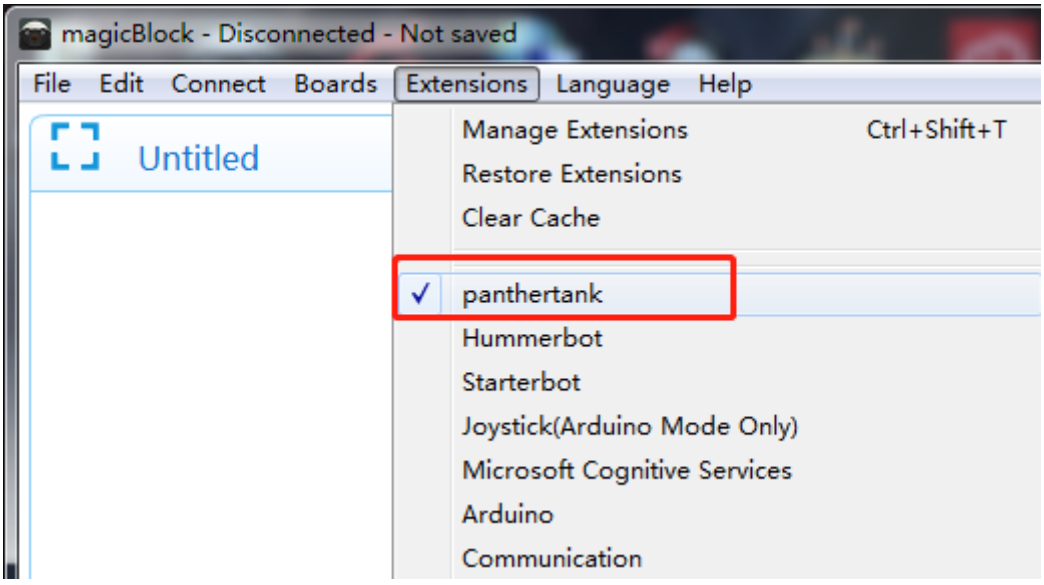


Figure 2-1-1-5

2.1.2 First Experience of Programming

After learning about Magic Block's graphics blocks, let's experience Panther-Tank programming. Let's first write a program for Panther-Tank robots to print Hello world.

First, drag the Panther-Tank main program in the building block area to the script area with the mouse, and then drag the "Setting Serial Port Baud Rate 9600" to the bottom of the "Panther-Tank Main Program" building block, as shown in Figure 2-1-2-1.

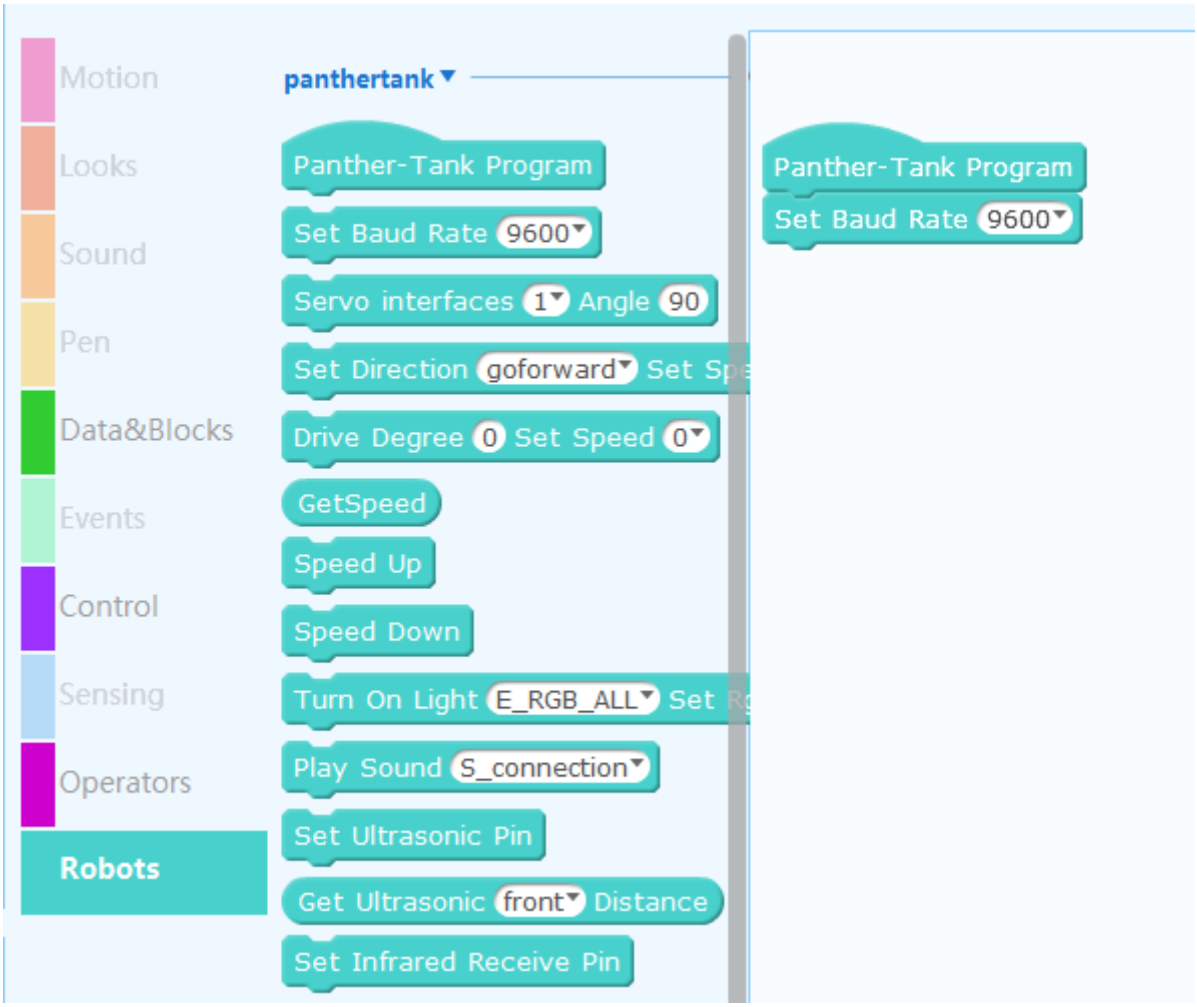


Figure 2-1-2-1

1) Click on the "Control" tab in the label area and drag the "Repeated Execution" block under the Panther-Tank main program in the script area.

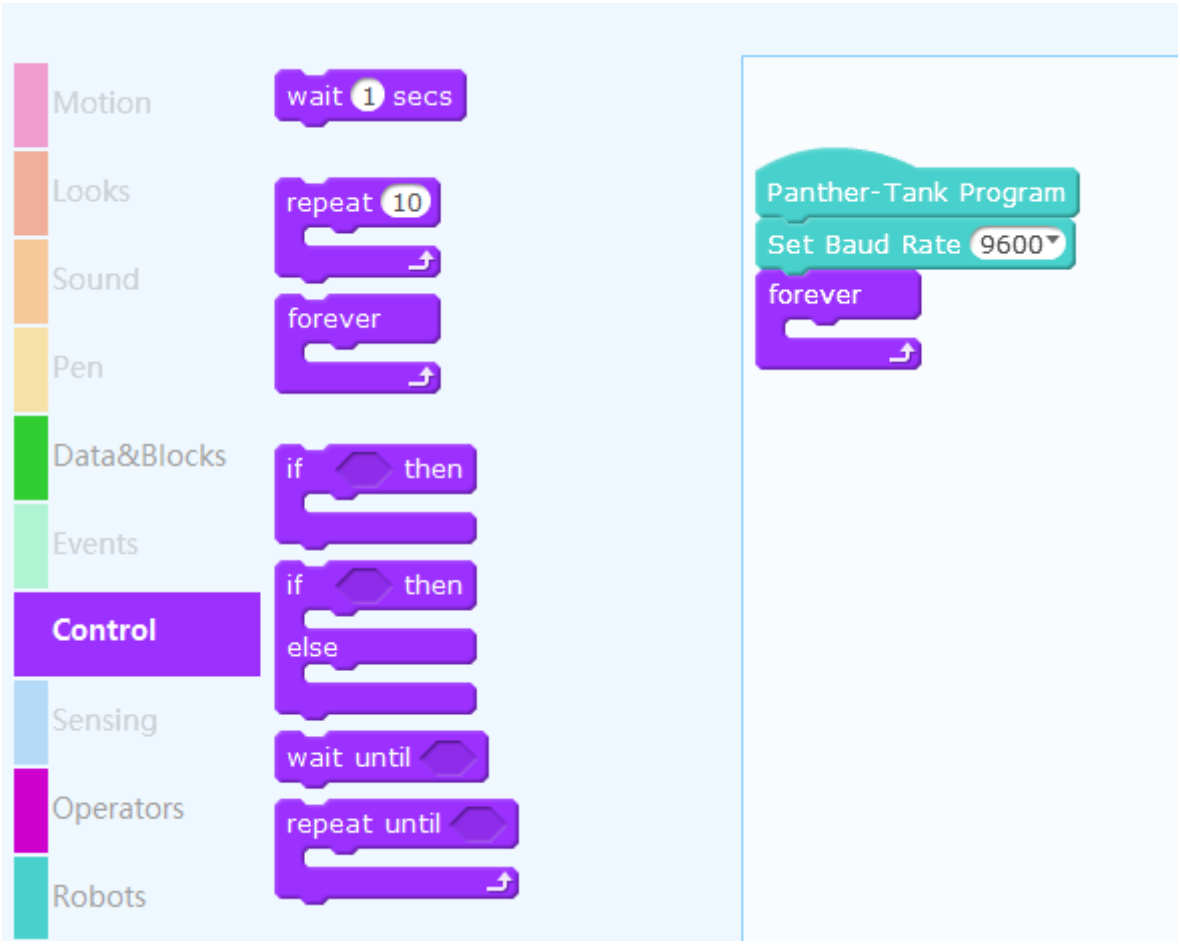


Figure 2-1-2-2

2) Drag the "serial port print string" building block into the repeated execution building block and enter "Hello world" as shown in Figure 2-1-2-3.

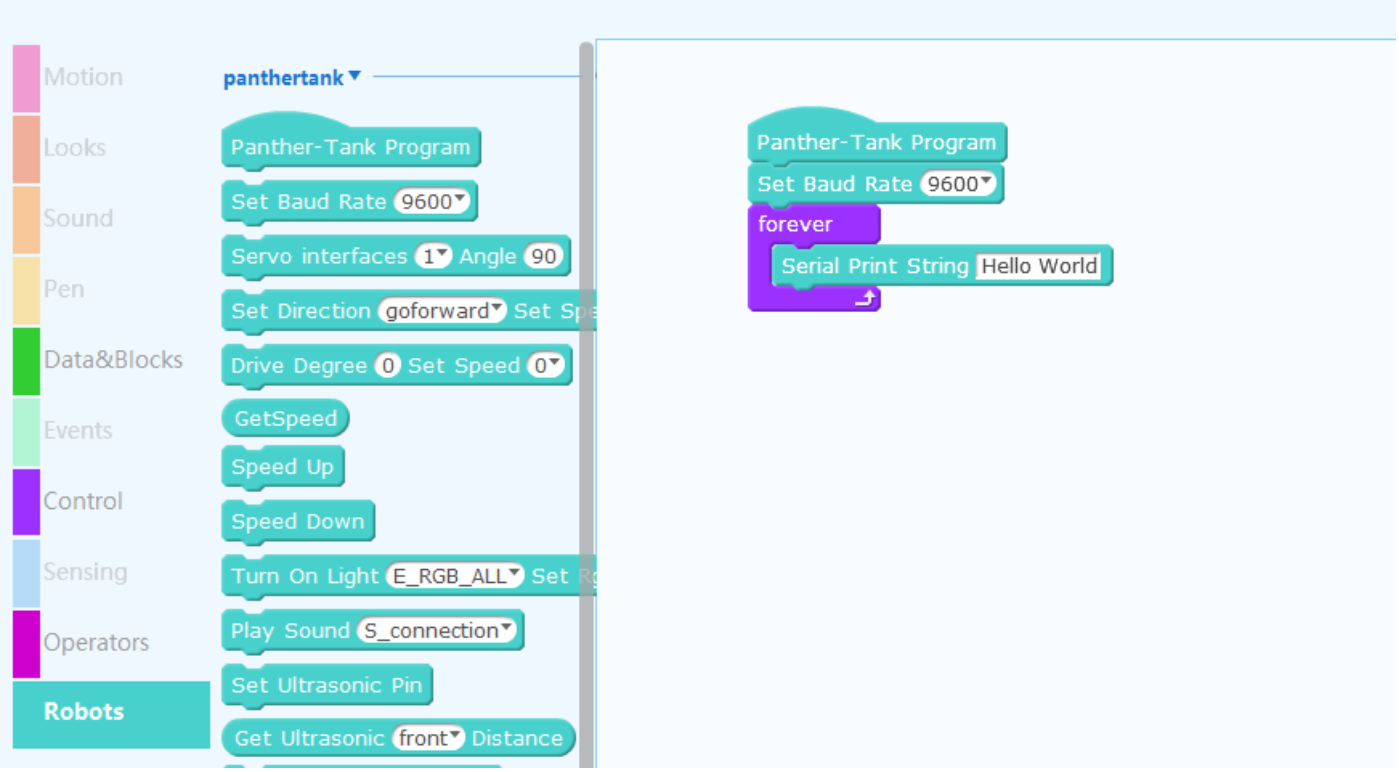


Figure 2-1-2-3

This is the Panther-Tank Robot Printing Hello World program. After the program is written, we need to transfer the program to the brain of the Panther-Tank robot (control motherboard) to do the desired action according to the program we wrote. How to transfer the program to the brain of the Panther-Tank robot (control motherboard)? Only when MagicBlock is connected with the robot motherboard can we transfer the written program from the computer to the brain of Panther-Tank robot (control motherboard). Here is the connection method between MagicBlock and the robot motherboard.

2.2 MagicBlock and Panther-Tank Connection Steps

1) Using a USB data line, one end is inserted into the computer, and the other end is inserted into the robot master board, which connects the robot master board with the computer.
2) Install the driver of Panther-Tank Master Control Board, click "Connect to Install Arduino Driver" as shown in Figure 2-2-1, and click "Install" as shown in Figure 2-2-1.

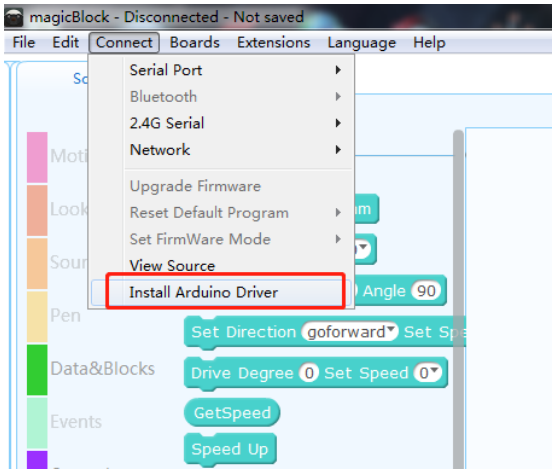


Figure 2-2-1

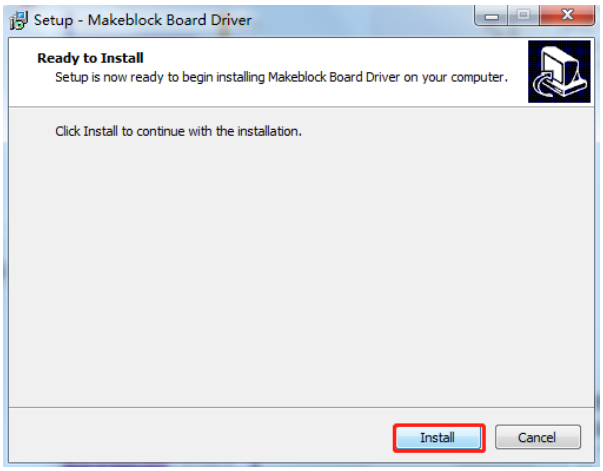


Figure 2-2-2

3) Click on "Connect to COM4 (different computers have different numbers of COM ports)", as shown in Figure 2-2-3. After the correct connection, there will be a "Serial Port Connected" prompt at the top of the software. At this time, MagicBlock and Panther-Tank are successfully connected, as shown in Figure 2-2-4.

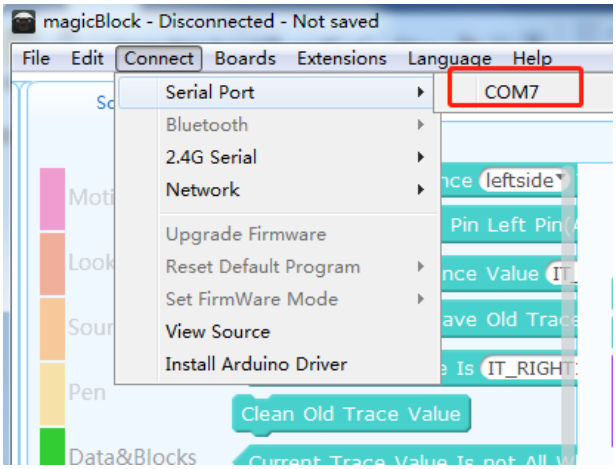


Figure 2-2-3

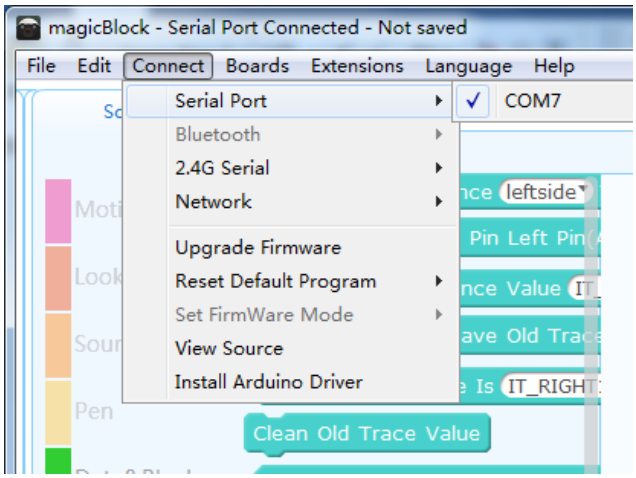


Figure 2-2-4

2.3 Upload program to Panther-Tank robot

When the program is finished and MagicBlock and Panther-Tank robot are connected correctly, we can transmit the program to the brain of Panther-Tank robot (main control board). The specific steps are as follows:

1) Select the type of control board for program transmission, select "Arduino Uno" as shown in Figure 2-3-1, and select "Edit Arduino Mode" as shown in Figure 2-3-2.

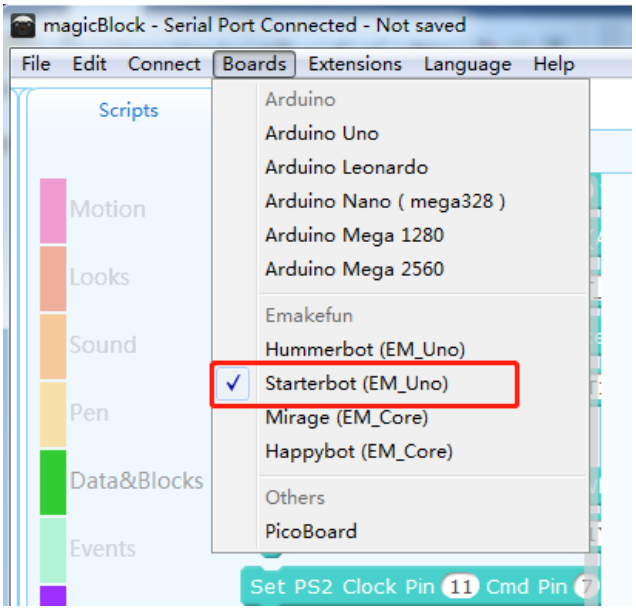


Figure 2-3-1

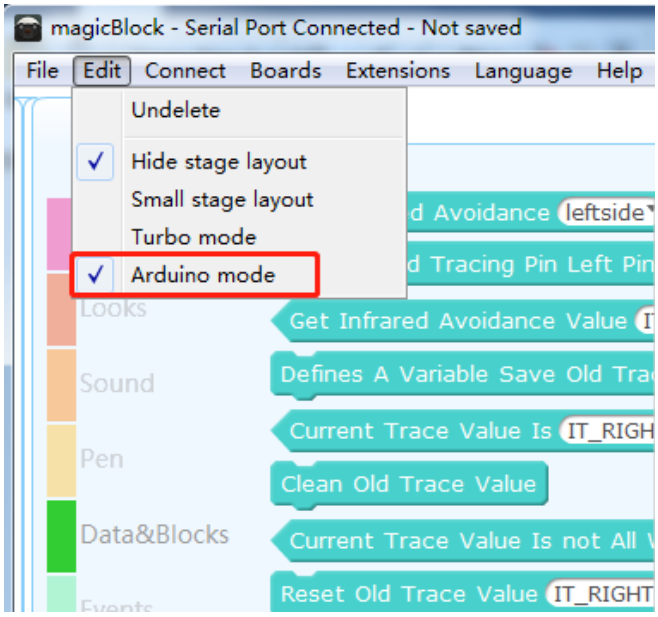


Figure 2-3-2

2) Click on any building blocks in the script area with the mouse. On the right side of the building blocks area, the corresponding building blocks program code will appear. Click "Upload to Arduino", start to generate the building blocks offline code and upload it to the Panther-Tank Robot. The screen will display a prompt window "Upload in", as shown in Figure 2-3-3; after the upload is completed, the prompt "Upload completed" will appear as shown in Figure 2-3-4.

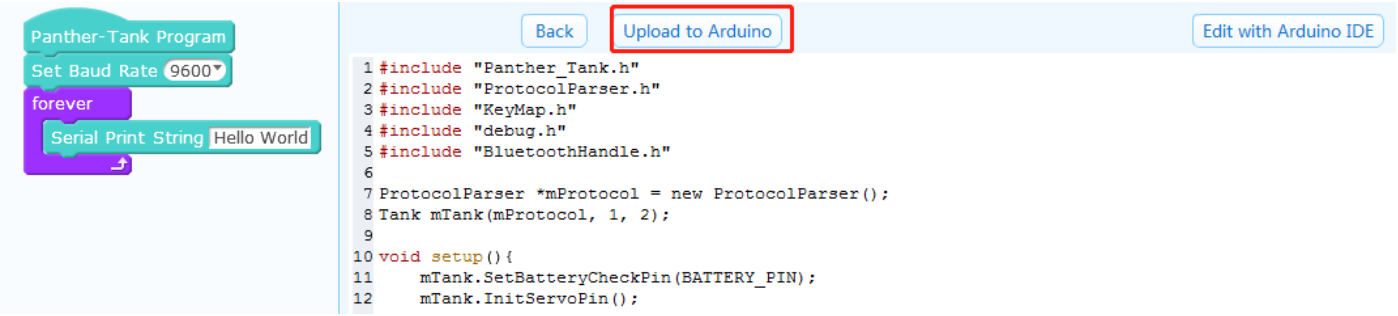


Figure 2-15

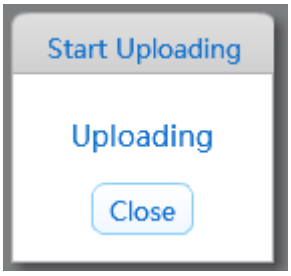


Figure 2-3-3

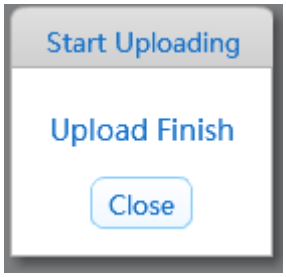


Figure 2-3-4

After completing the above steps, the Panther-Tank robot's brain (master board) already has the program we wrote, so how can we see the Hello Word printed by the robot? At this

point, we will use a serial monitor to see, click the "Edit with Arduino IDE" in the upper right corner, as shown in Figure 2-3-5.

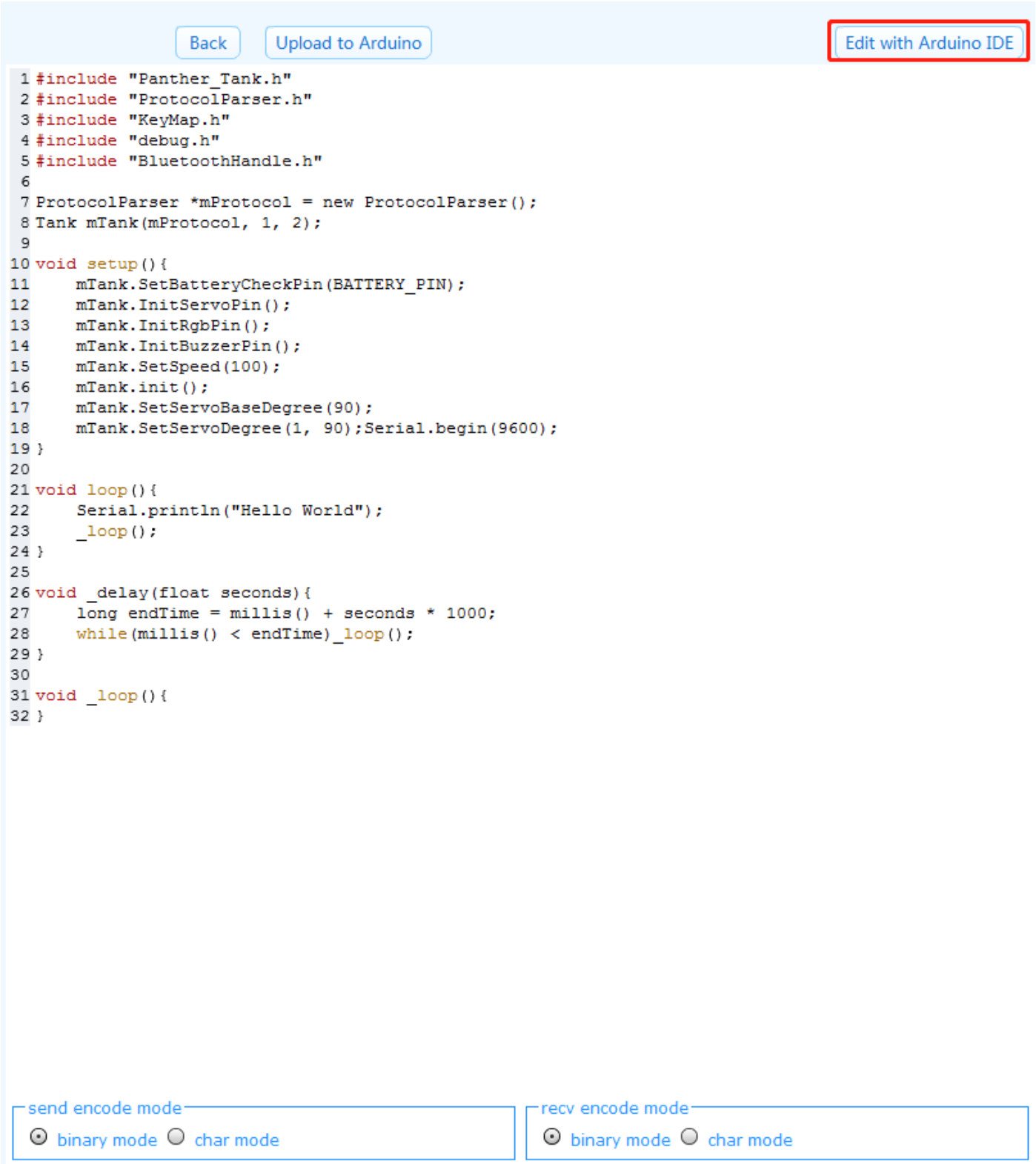


Figure 2-3-5

After opening the Auduino software, click on the serial monitor in the upper right corner, as shown in Figure 2-3-6.

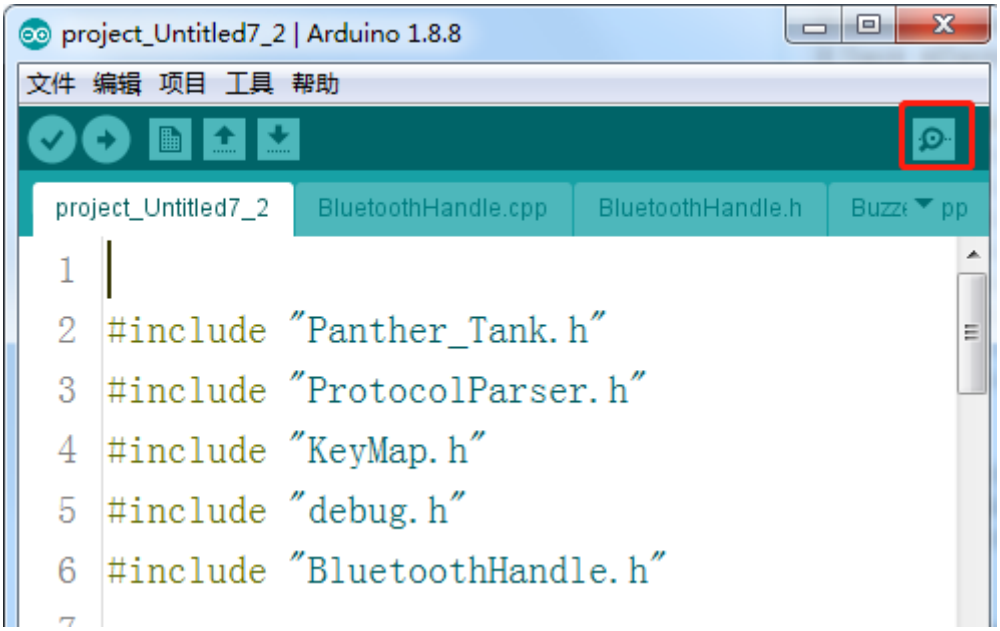


Figure 2-3-6

Turn on the serial monitor and we will see Hello Word printing continuously on the serial monitor, as shown in Figure 2-3-7.

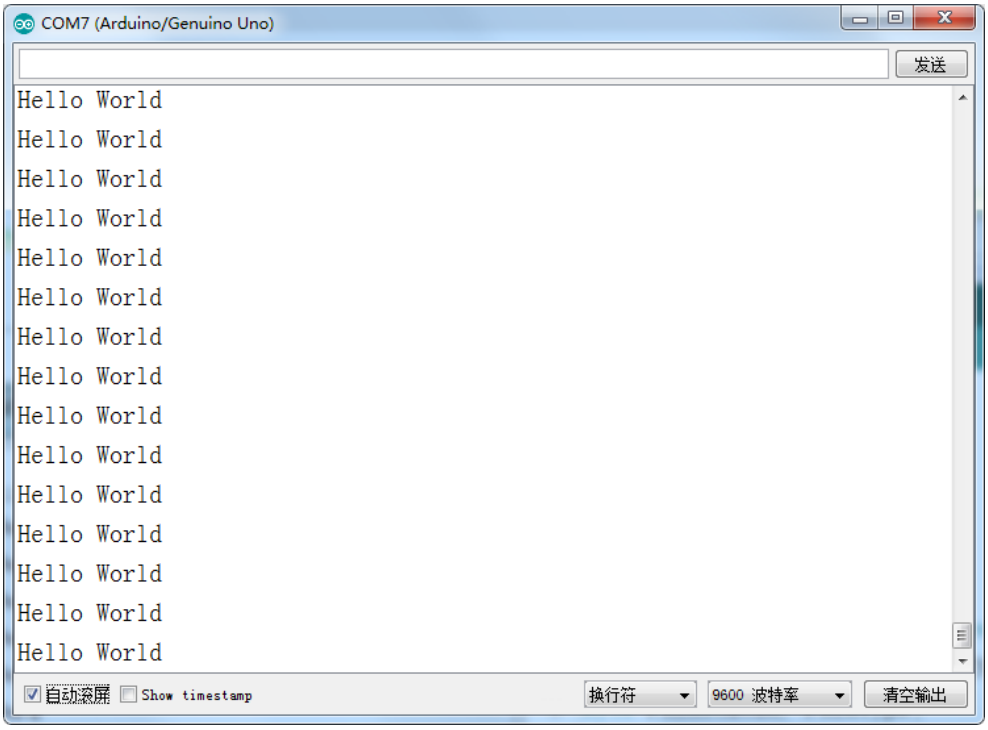


Figure 2-3-7

When using serial port monitor in Arduino, we need to disconnect the serial port connection between MagicBlock and Panther-Tank first. Otherwise, we will report a busy serial port error when opening the serial port monitor, as shown in

Figure 2-3-8. In this case, we need to click MagicBlock's connection-serial port-port number to cancel the connection, and then open the serial port monitor with Arduino.（ Fig. 2-3-9) There will be no errors in Fig. 2-3-8.

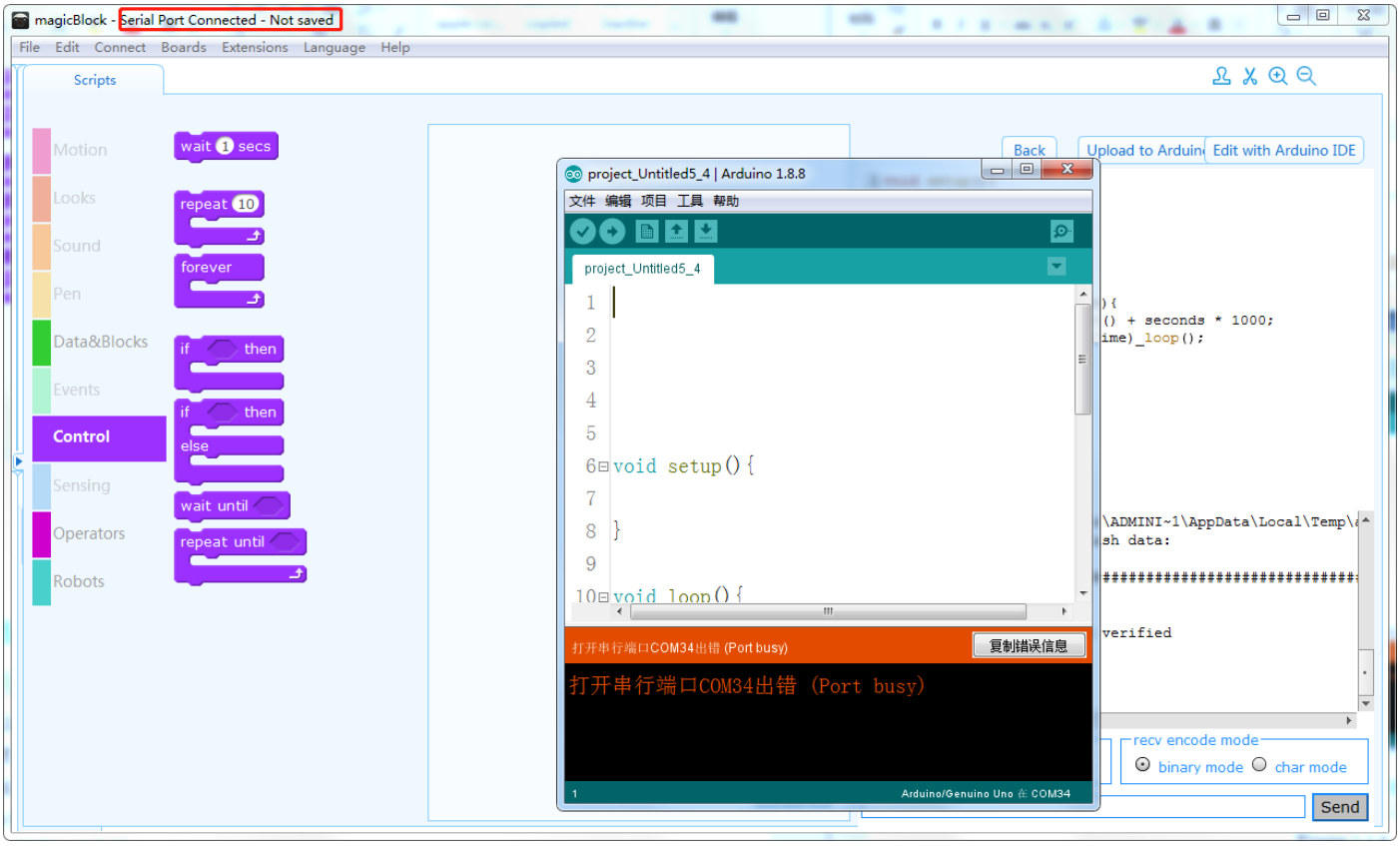


Figure 2-3-8

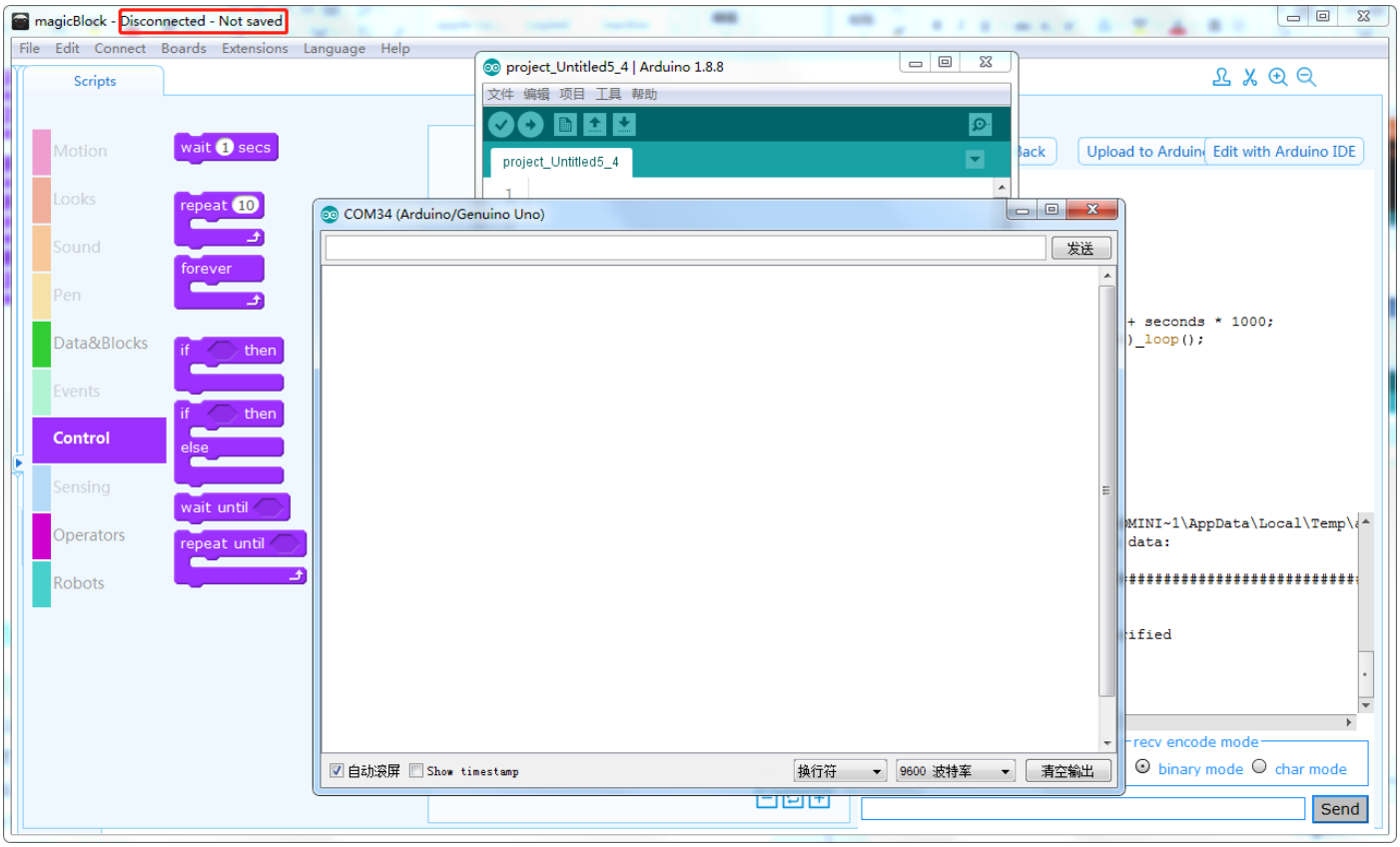


Figure 2-3-9

2.4 RGB LED

We usually see some colorful lights in the evening. The colorful lights are very gorgeous. There are two colorful lights on Panther-tank. These two LED lights can emit colorful light. Then why can a light emit colorful light? This is related to the three primary colors of light we learn from physics. The three primary colors of light are red. Green, blue, these three colors can be mixed together to present different colors, which is the principle of Panther-tank RGB LED lamp. The program building blocks in Panther-tank are shown in Figure 2-21 below.

Figure 2-21

After recognizing the building blocks of LED lights, let's write a simple program to make the two lights light up.

Figure 2-22

By running the program shown in Figure 2-22 above, we will find that when we turn on the Panther-tank power supply, the two lights will turn on, and the color can be set by ourselves. That lamp is not bright enough, we want to let the lamp flash how to write the program, let's write a flashing LED program.

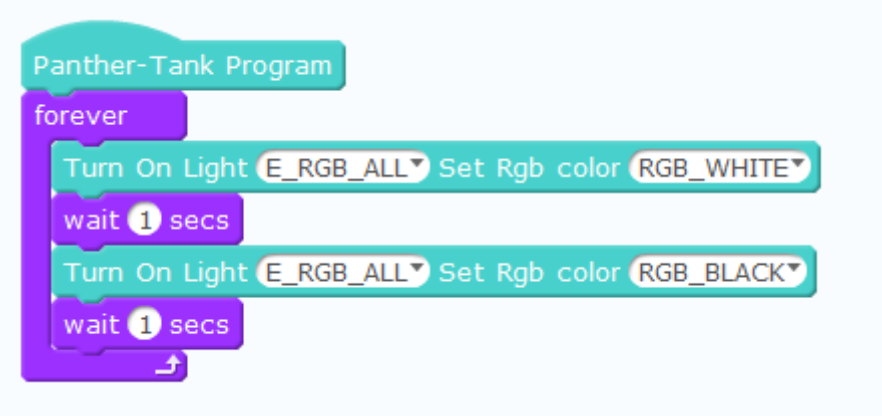


Figure 2-23

When we turn on the power of the robot, we can see that two LED lights will have a flash effect.

2.5 Buzzer

In the last section, we wrote a program to let the robot shine, and saw the robot lights flashing, very beautiful, so apart from shining, can we let the robot sing? Of course, there's a buzzer on the Panther-tank robot, through which the robot can sing. Buzzer is the current through the electromagnetic coil, so that the magnetic field generated by the electromagnetic coil to vibrate the vibration film to make sound. Now to write a program for the robot to sing.

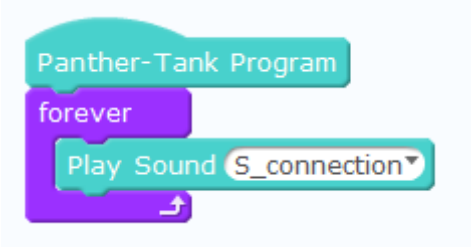


Figure 2-24

When the above program is written, we upload the program, turn on the power of the robot, and you will hear the robot emit a "drop-drop-drop" sound. Of course, we can also add building blocks to play songs and let robots sing.

Chapter Three Robots Move

3.1 DC motor

3.1.1 Principle of DC Motor

The reason why cars are active is that they have engines to power them. Robots also have DC motor modules that allow them to move, so what is a motor? In our science textbook, there is an introduction of electromagnetic induction. The motor is rotated by electromagnetic induction. It has an iron core winding 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 Panther-tank robot has two DC motors. We can use the DC motor to control the building blocks so that the DC motor can rotate, thus driving the Panther-tank robot to move.

3.1.2 DC motor test

In the robot module, we can find three control blocks of DC motor module, which are motor pin building blocks, direction setting, speed building blocks, turning angle setting and speed building blocks, as shown in Figure 3-1.

Figure 3-1

- Setting up the motor pin building blocks: Used to set the pins connecting the four motors of the control robot to the main control board. The default pins are selected in this tutorial.
- Set the direction and speed, speed building blocks: Set the direction of the robot movement and the speed in that direction;
- Set the turning angle and speed: Set the turning angle and speed when the robot moves.

In MagicBlock, the following four programs are written. They can control the motor forward and reverse respectively, and then let the robot move forward, backward, left and right directions. They run the program separately and carefully observe the direction of two DC motors when the robot moves in each direction. The programming of the four directions is shown in Fig. 3-2.

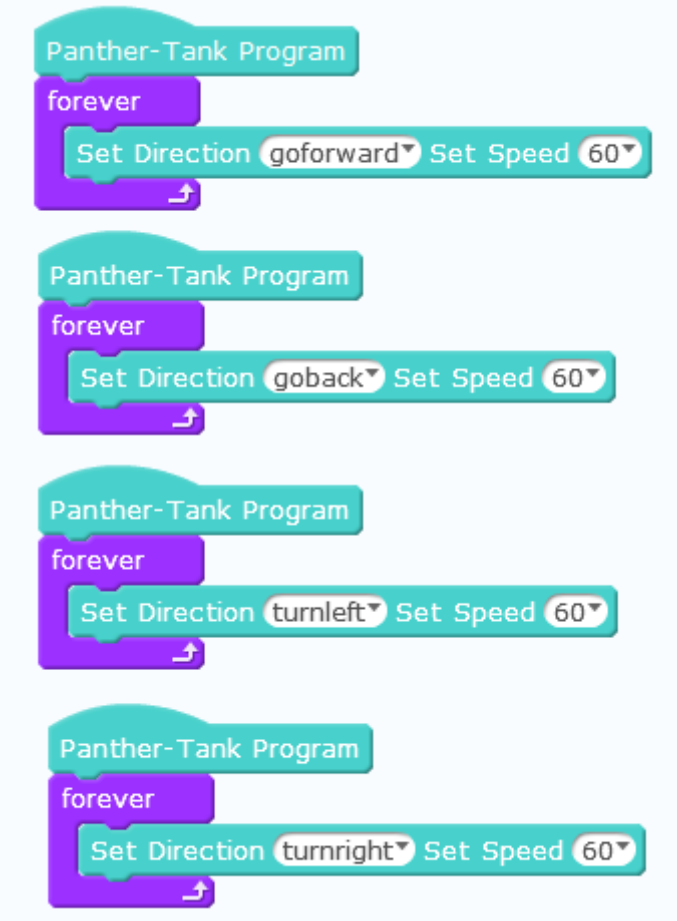


Figure 3-2

Through the above practical operation, we should have understood the programming method of letting the robot move in a single direction by controlling the motor. Now let's explore the programming method of letting the robot walk in a square.

3.1.3 Programming control and debugging the robot to take the square route.

The way the robot moves in a square is to move forward, turn left, turn left, turn forward, turn left (or turn backwards). The path of the robot is shown in Figure 3-3, and the reference program is shown in Figure 3-4.

Figure 3-3 Figure 3-4

Note:

1) The delay blocks under the "forward" building blocks are to control the forward distance of the robot, while the delay blocks under the left turn are to control the turning angle of the robot. When writing the program, the delay time should be modified according to the actual walking condition of the robot.
2) At the end of the program, the motor must stop turning, otherwise the robot will remain in its original state of motion and turn left all the time.

3.1.4 Discover the Rule of Program and Optimize Program

By observing the program of Figure 3-4, we can find that the path of the robot walking square is actually to control the wheel to repeat four times forward and turn left 90 degrees. The optimization program is shown in Figure 3-5.

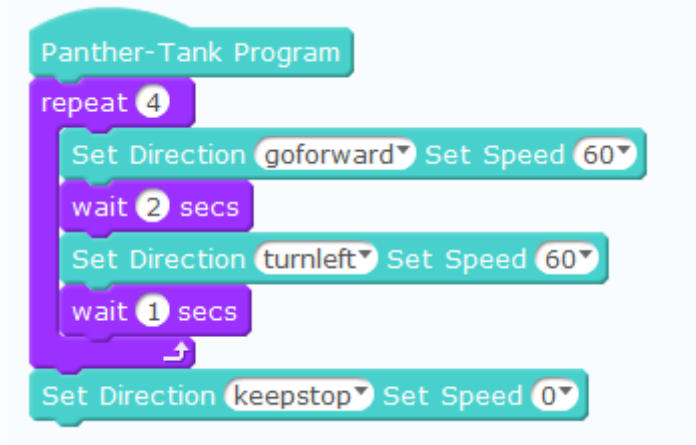


Figure 3-5

3.2 Ultrasound module

3.2.1 Principle of Ultrasound

Ultrasound sensor is a device that detects distance by transmitting ultrasound. Ultrasound is an inaudible sound wave, which has the characteristics of returning when it touches an object. Ultrasound sensors have two "eyes", one of which emits ultrasound, and the other "eyes" receive the ultrasound emitted from obstacles. When one eye emits ultrasound, it begins to time, and when the other eye receives the returned ultrasound, it stops the time. Mathematically, we have learned to go through the road. Range = speed * time, then the distance measured by ultrasound = the speed * of ultrasound (timing time 2); thus distance can be calculated.

3.2.2 Usage of Ultrasound Module

We find the control building blocks of the ultrasonic module in the control module of the robot. As shown in Figure 3-6, the distance between the robot and the obstacle ahead can be detected by using the building blocks. Note: In order to use the ultrasonic module, we first need to choose the mode to control the building

blocks and set the mode to the ultrasonic obstacle avoidance mode. When the obstacles completely block the ultrasonic module or face the ultrasonic module far away, the ultrasonic wave emitted by the ultrasonic module can not be received, so the robot and obstacle can not be detected. The detection distance of the ultrasonic module installed on the robot is 5CM ~ 400CM.

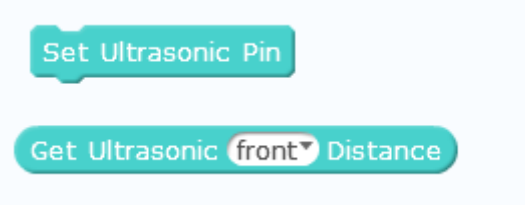


Figure 3-6

3.2.3 Testing of the Ultrasound Obstacle Avoidance Module

We can first write a program to test the ultrasonic obstacle avoidance module, using serial port printing, when we hand close to the ultrasonic obstacle avoidance module and far away from the ultrasonic obstacle avoidance module, we observe the distance measured by the serial port printing, we can more intuitively see the process of measuring the distance of the ultrasonic obstacle avoidance module. Let's write a test program first.

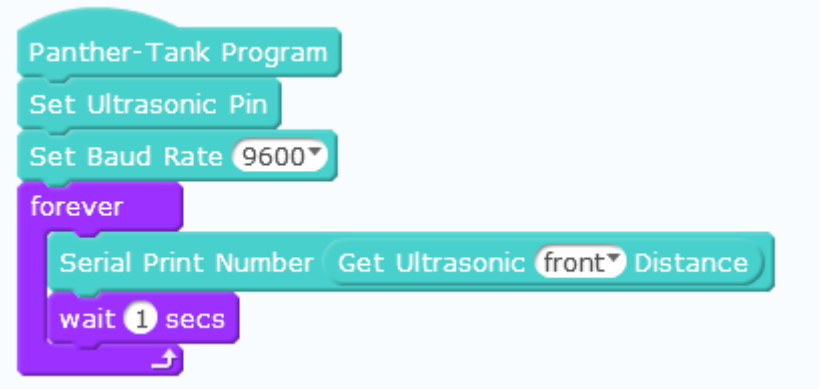


Figure 3-7

We write the program as shown in Figure 3-7, then upload the program successfully, open the serial port, and then close or away from the ultrasonic module, the serial port will print the corresponding distance.

3.2.4 Writing Robot Program for Tango Dance

Tango dance is a kind of double dance. The dance step is to let the robot dance with our palms when one person is close. If the hand is close to it and the hand moves away from it, the distance between the hand and the robot should be measured by the ultrasonic module to determine whether the robot moves forward or backward. The larger the measuring distance of ultrasound means that the hand is far away from the robot, and after a certain distance, the robot moves forward; the smaller the measuring distance of ultrasound

means that the hand is close to the robot, and after a certain distance, the robot retreats; let's set the median value to 20CM first, then let's start programming.

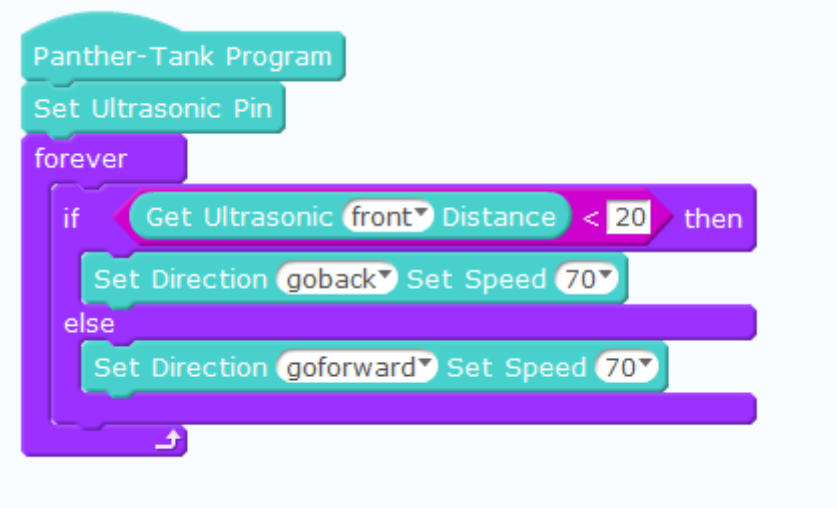


Figure 3-8

Run the program shown in Figure 3-8 above. When the hand is far away, the robot moves forward; when the hand is near, the robot retreats. But when the hand is still, the robot will move back and forth repeatedly, because we did not set a stop motion interval for the robot. How to stop the robot in an interval?

The Value Measured by the Ultrasound Module	Motion State of Robot
More than 20	Forward
Between 12 and 20	Stop
Less than 12	Back off

表 3-9

As shown in Table 3-9 above, a new judgment statement is needed for this situation between 12 and 20. As shown in Figure 3-10, when the distance measured by ultrasound is greater than 12 CM and less than 20 CM, the robot stops.

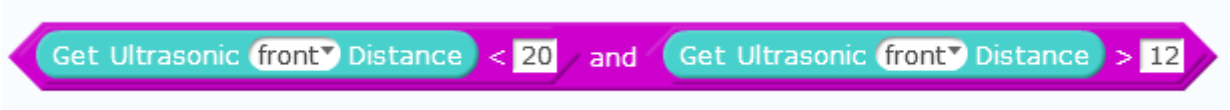


Figure 3-10

3.2.5 Optimizing Dancing Robot Program

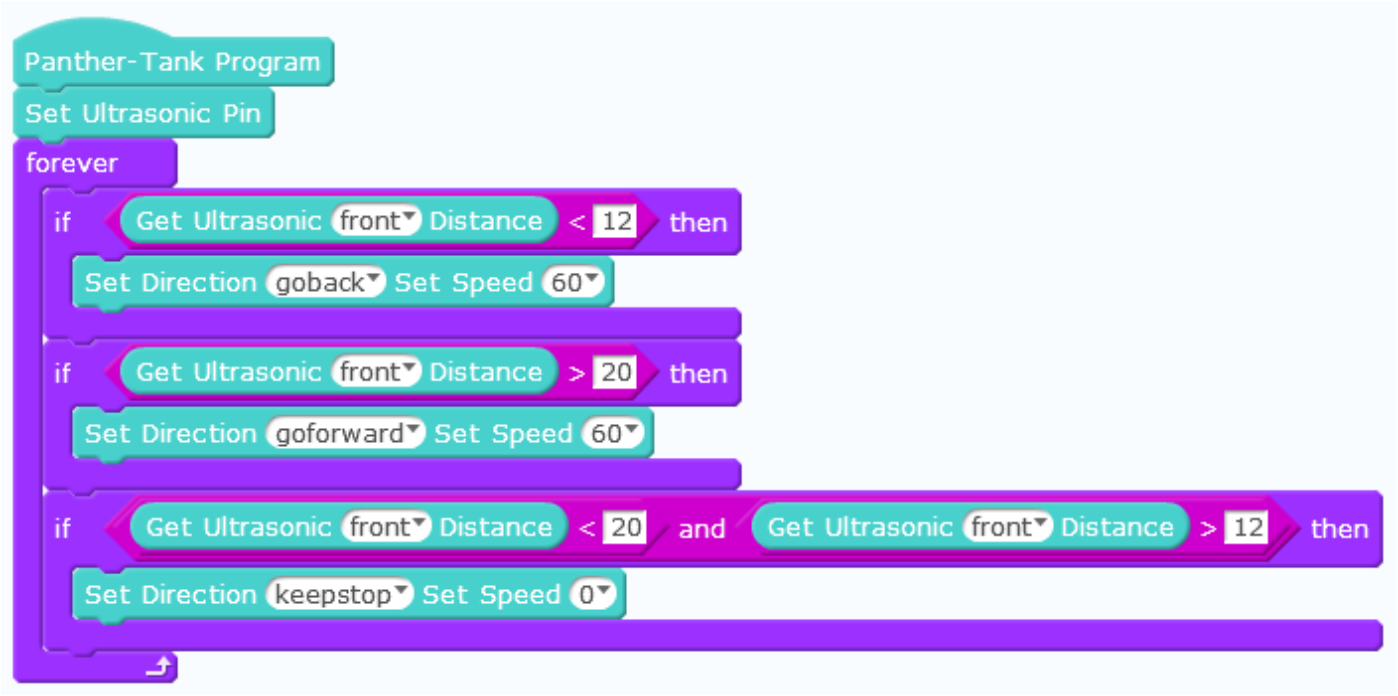


Figure 3-11

3.3 Ultrasonic obstacle avoidance robot

We have learned the roles of robots in moving forward, backward, left, right and ultrasonic modules. Now let's explore how robots can use ultrasound to avoid obstacles in motion. The so-called obstacle avoidance of robots is to let robots move when there are obstacles in front of them. If the robot wants to turn left or right and judge the distance between left and right directions, then the ultrasonic module needs to rotate left and right to explore the way. At this time, the steering gear is used to realize the left and right turn of the ultrasonic module. How does the steering gear work? Next we will introduce the use of the steering gear.

3.3.1 Operating Principle of Steering Engine

The steering gear is mainly composed of the following parts: steering wheel, deceleration gear set, position feedback potentiometer, DC motor, control circuit and so on, as shown in Figure 3-11. The Bumblebee and Optimus Prime joints we see in the movies need to be controlled by the steering gear. Especially when the robot is walking, it makes the mechanical sound of clicking and clicking, which is generated by the steering gear rotating on the robot. Figure 3-12 is the most commonly used SG90 steering gear physical diagram at this stage.

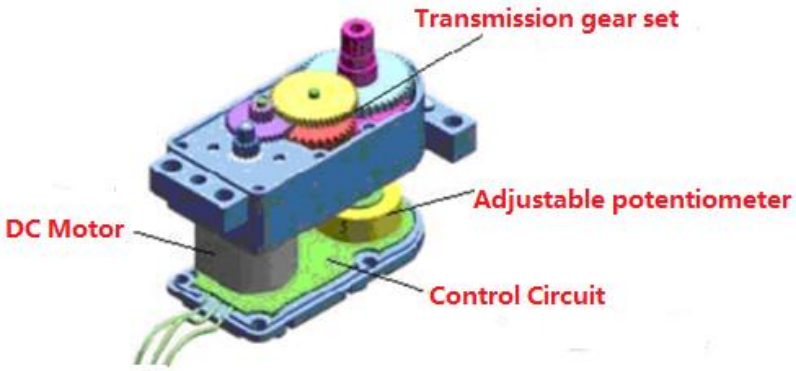


Figure 3-11

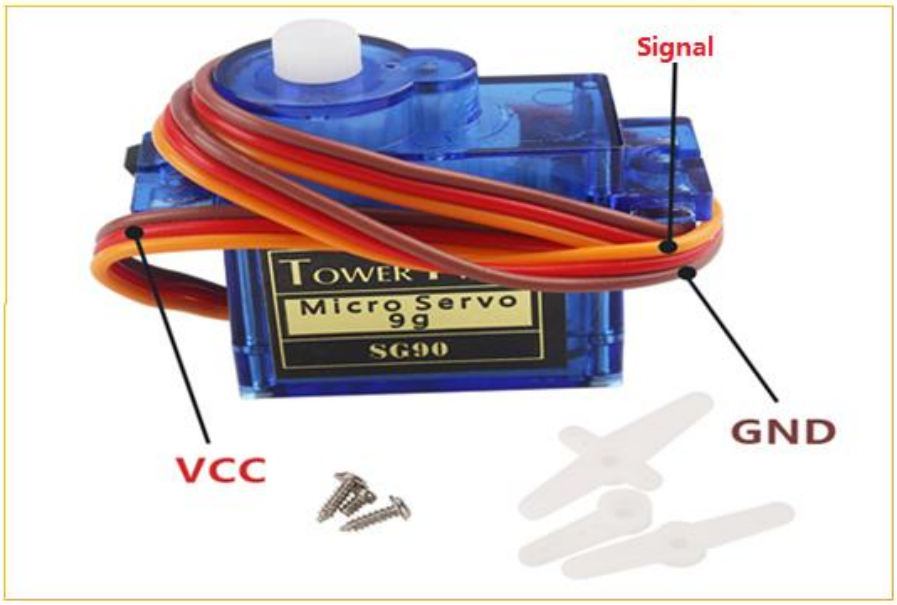


Figure 3.-12

When the control circuit board receives the control signal of the self-confidence signal line, it controls the motor to rotate, and the motor drives a series of gears, then reduces the speed and drives them to the output steering wheel. Its work flow is: control signal control circuit board motor rotation gear set deceleration steering wheel rotation position feedback potentiometer control circuit board feedback.

3.3.2 Steering Engine Testing

The steering gear on the Panther-tank robot is installed in the front, and the ultrasonic module is fixed on the steering gear. When the ultrasonic module is to measure the distance between the robot and the front obstacle, the steering gear is 90 degrees. When the ultrasonic module is to measure the distance between the robot and the left obstacle, the steering gear rotates 90 degrees to the left, and when the ultrasonic module is to measure the distance between the robot and the left obstacle, the steering gear rotates 90 degrees to the left. When measuring the distance between the robot and the right obstacle, the steering gear rotates 90

degrees to the right. Next, we write a program to test the steering gear by using the steering gear to make the ultrasonic wave move forward, then left and then right.

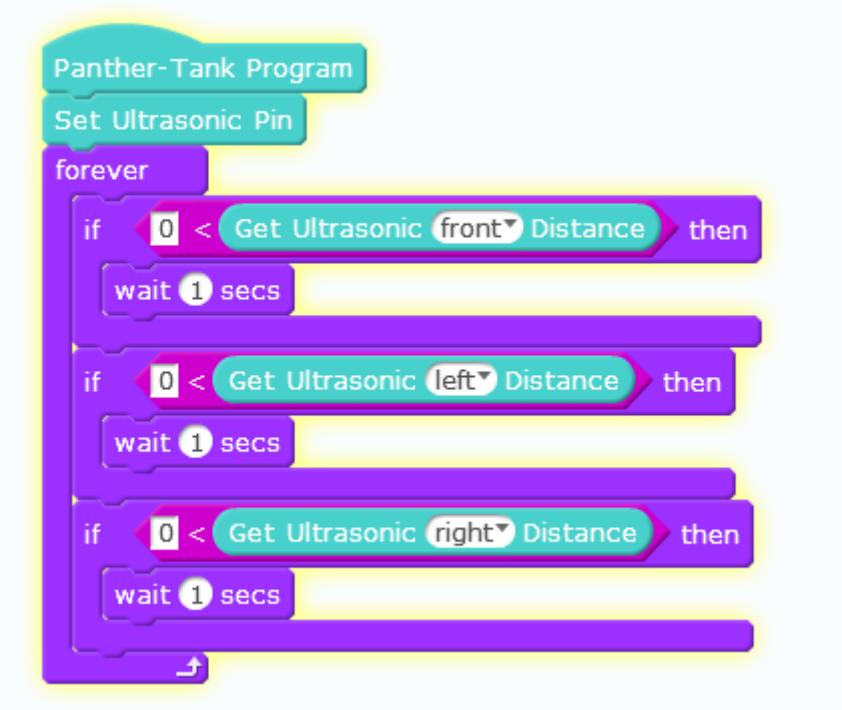


Figure 3-12

Above is the testing procedure of the steering gear. After uploading the program to the robot, turning on the power supply, we can see that the steering gear makes the ultrasonic module turn forward, then left, and then right.

3.3.3 Programming of Obstacle Avoidance Robot

The main work flow of the ultrasonic obstacle avoidance robot is: after power-on, the steering gear makes the ultrasonic module automatically turn to 90 degrees (alignment in front of the robot). The ultrasonic module measures the distance of the obstacle in front. If the value is larger than the set safe distance (12CM), the robot continues to move forward, otherwise stops. At this time, the rudder. The machine rotates the ultrasonic wave 90 degrees to the right. The ultrasonic module measures the distance between the ultrasonic wave and the right obstacle. Then the steering gear rotates the ultrasonic wave 180 degrees to the left. The ultrasonic module measures the distance between the ultrasonic wave and the left obstacle. Then the steering gear makes the ultrasonic module return to 90 degrees. The robot compares the distance measured twice, if left. If the distance measured on both sides is less than the safe distance, the robot will turn around and walk. Following this train of thought, let's start to write the program of obstacle avoidance robot.


Panther-Tank Program
Set Ultrasonic Pin
forever
set UlFrontDistance to Get Ultrasonic front Distance
if UlFrontDistance < 12 then
Set Direction goback Set Speed 80
wait 0.2 secs
if UlFrontDistance < 20 then
Set Direction keepstop Set Speed 0
wait 0.1 secs
set UlRightDistance to Get Ultrasonic right Distance
wait 0.05 secs
set UlLeftDistance to Get Ultrasonic left Distance
wait 0.05 secs
if UlRightDistance > 20 then
Set Direction turnright Set Speed 100
wait 0.4 secs
else
if UlLeftDistance > 20 then
Set Direction turnleft Set Speed 100
wait 0.4 secs
else
if UlRightDistance < 20 and UlLeftDistance < 20 then
Set Direction turnleft Set Speed 100
wait 0.8 secs
else
Set Direction goforward Set Speed 80

Figure 3-13

Chapter IV Infrared Telecontrol Robot

4.1 Principle of Infrared Remote Control

The remote control system is generally composed of a remote controller (transmitter) and a receiver. When you press any key on the remote controller, the remote controller will issue an instruction. When the receiver receives the instruction from the remote controller, it will transmit the instruction to the robot's brain. The robot will think about what kind of action to do according to the remote control instruction. After controlling their limbs (four wheels) to do the corresponding action, the remote controller and receiver are shown in Figure 4-1.

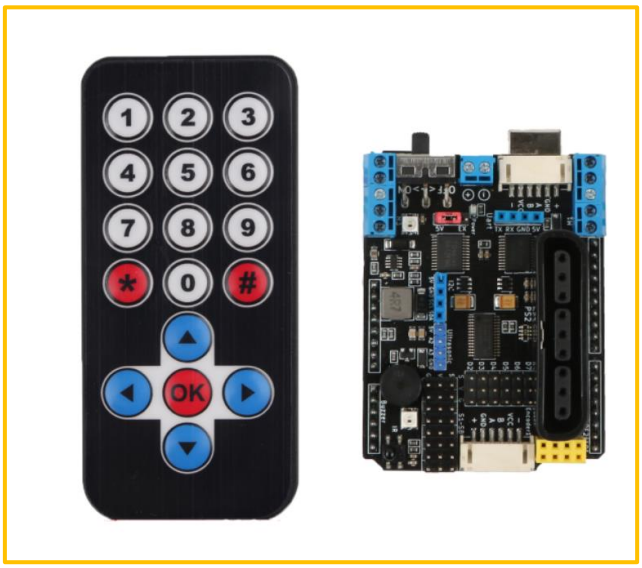


Figure 4-1

4.2 The Use of Remote Controller

To control the robot through the remote control, we first set the operation mode to infrared remote control mode, and set the infrared remote control receiving pin according to the actual connection port, then define the effect of each key press of the remote control, so that when we press the key of the remote control, the robot will do the settings in our program.

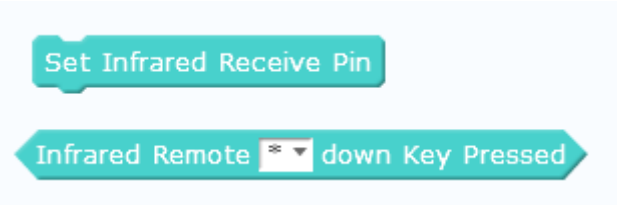


Figure 4-2

4.3 Infrared remote control test

We can first write a program to test infrared remote control, using serial port printing, when the remote control button is pressed, the serial port prints the corresponding key values, we observe the contents of

serial port printing, we can see which key of the infrared remote control is pressed. Now we first write a test program.

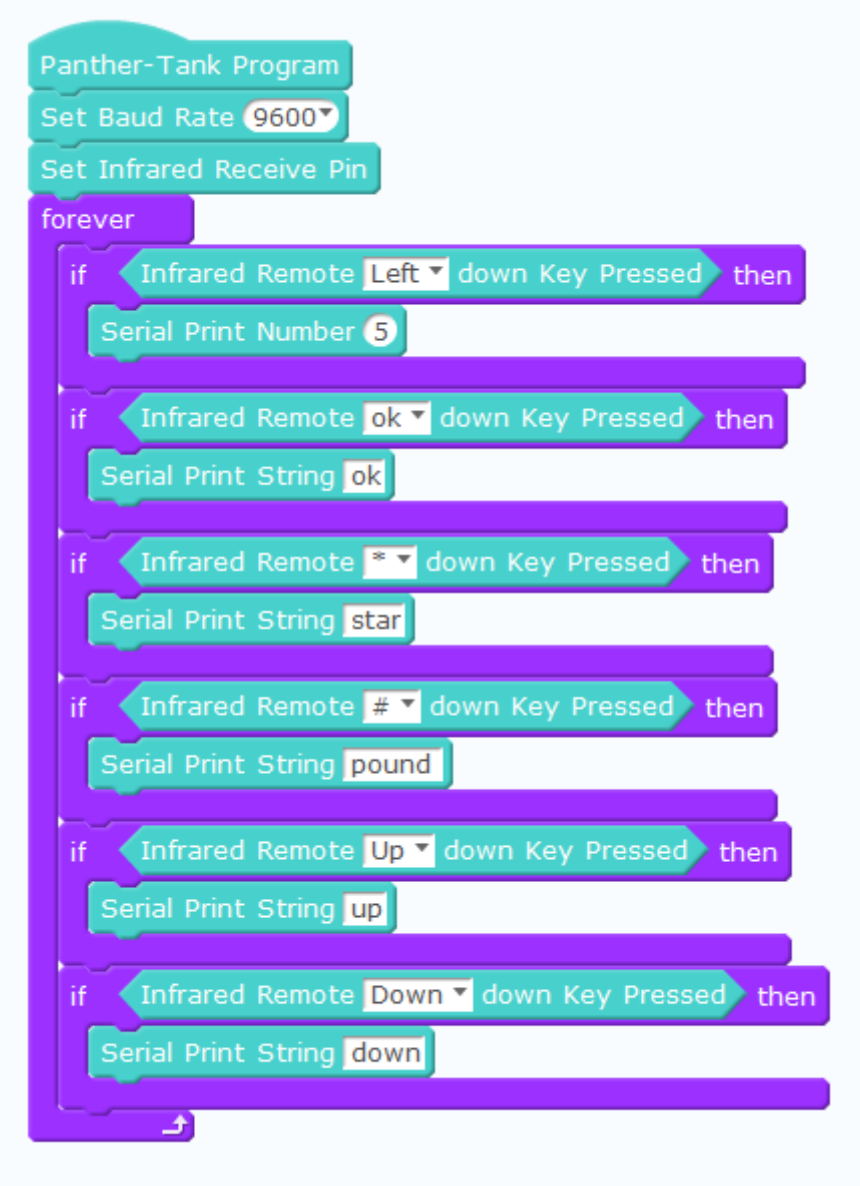


Figure 4-3

We write the program as shown in Figure 4-3, then upload the program successfully, open the serial port, then hold the remote control to press the front and back buttons on the robot, and the serial port will print the corresponding key numbers.

5.4 Programming of Remote Control Robot

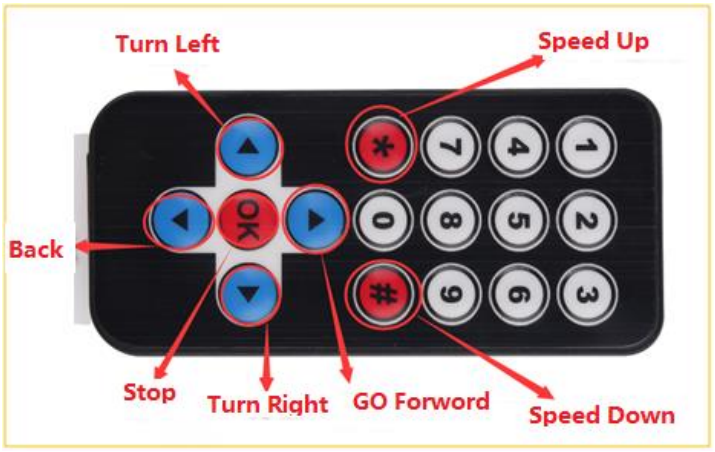


Figure 4-4

We write a remote control key function as defined in Figure 4-4, and set speed regulation 1 (*) for acceleration and speed regulation 2 (#) for deceleration. We need to make a 0.1S delay after the acceleration and deceleration procedure, which is to make the robot accelerate and decelerate a little more smoothly. Without delay, the speed will directly change to the maximum or minimum.

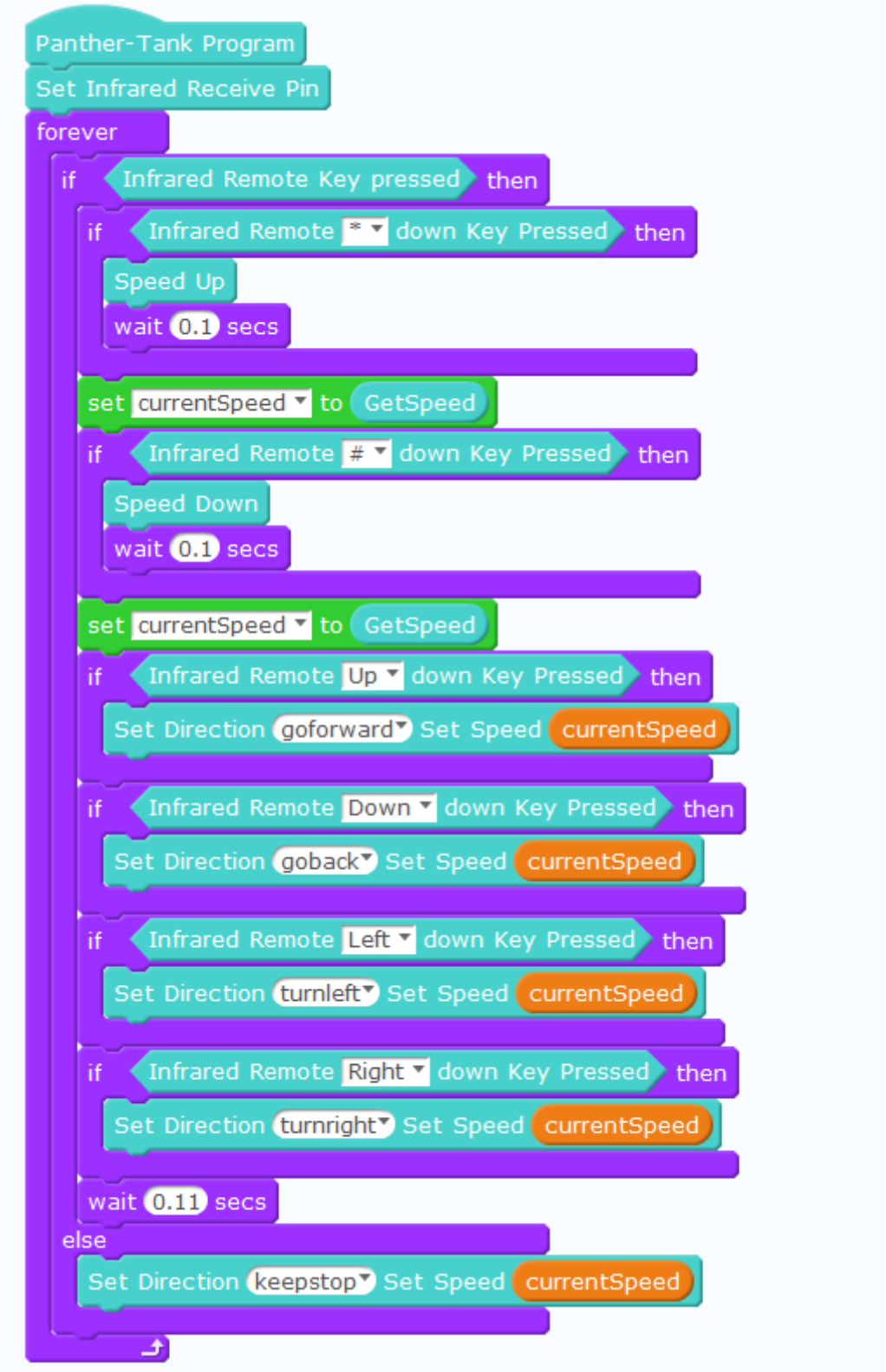


Figure 4-5

Note: When setting the steering speed, the speed value can not be set too small, otherwise the car is not strong enough when turning, there will be stuck in place, so the best setting speed value is more than 40.

Chapter 5 Bluetooth Remote Control Robot

5.1 Bluetooth Control Principle

Bluetooth is a way of long-distance communication. We send the instructions to the Bluetooth module on the robot from the APP end of the mobile phone through the Bluetooth of the mobile phone. Then the Bluetooth module on the robot sends the instructions to the robot brain (main control board), and the robot controls its limbs (four wheels) to do the corresponding actions. At the same time, the robot will send the action being done to the mobile app through the Bluetooth module installed by itself, so that the mobile app can remote control the robot through Bluetooth.

5.2 Connection Method of Bluetooth Module and Anter-Tank Robot APP KeywishBot

The Panther-tank robot is equipped with a Bluetooth module (Fig. 5-3). When the power is turned on, the blue light on the Bluetooth module will flicker. After turning on the APP, select Panther-tank, Bluetooth module and the Panther-tank robot APP KeywishBot (Fig. 5-1) on the mobile phone to connect through Bluetooth. After the connection is successful, the blue light will flicker. Changliang, then select the "remote control interface" (Fig. 5-2) and enter the gravity induction control interface, Fig. 5-4. You can also switch the handle control mode as shown in Figure 5-5.

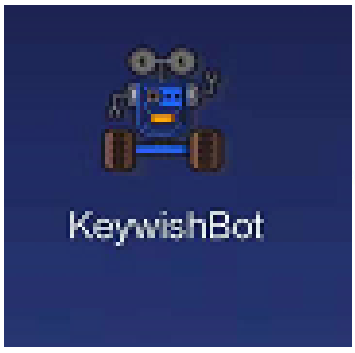


Figure 5-1

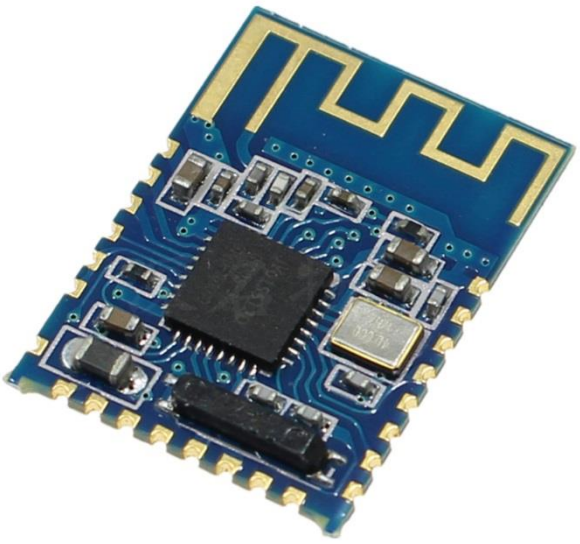


Figure 5-3

Figure 5-2

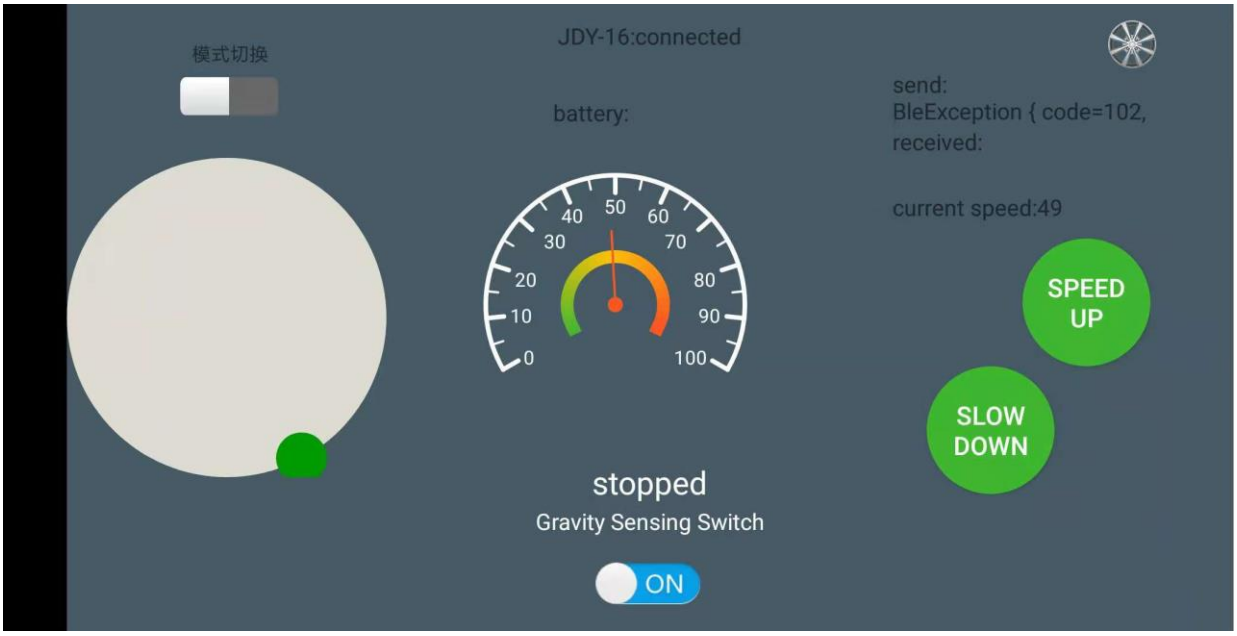


Figure 5-4

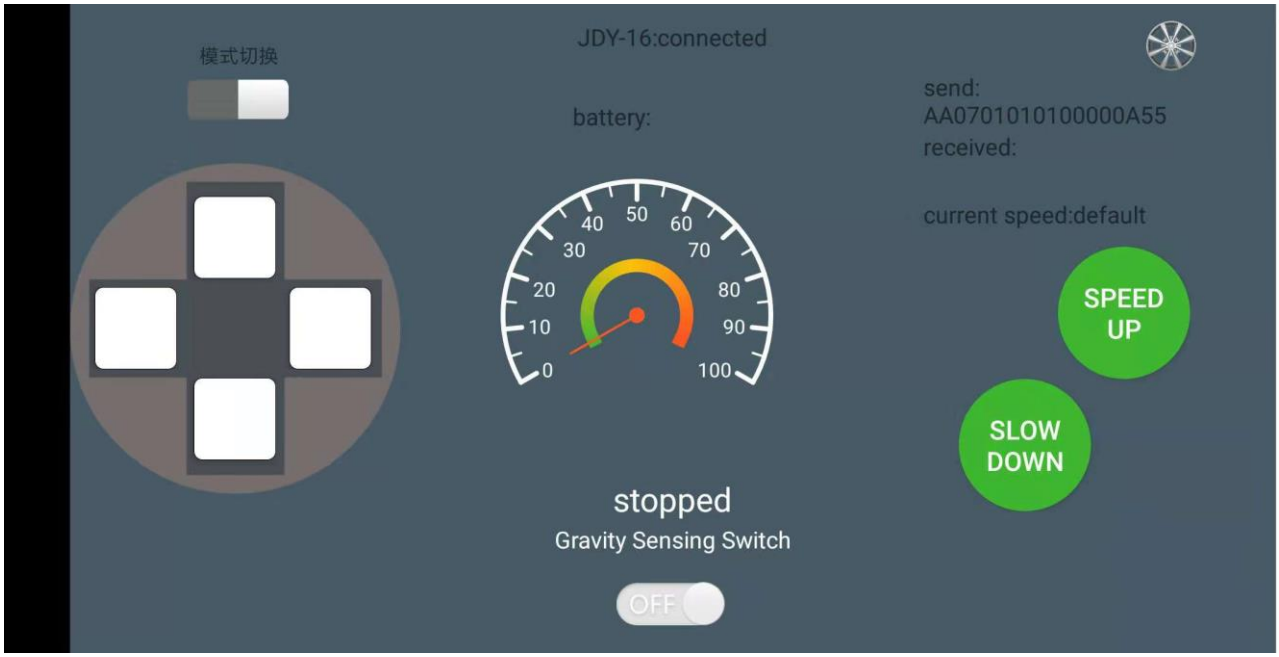


Figure 5-5

5.3 Bluetooth module testing

We can first write a program to test Bluetooth module, using serial port printing. When the Bluetooth module of the robot receives data and the serial port prints the data received by Bluetooth, we can see the effect of Bluetooth data transmission by observing the content sent by Bluetooth of the mobile phone and the content printed by the serial port. Trial procedure.

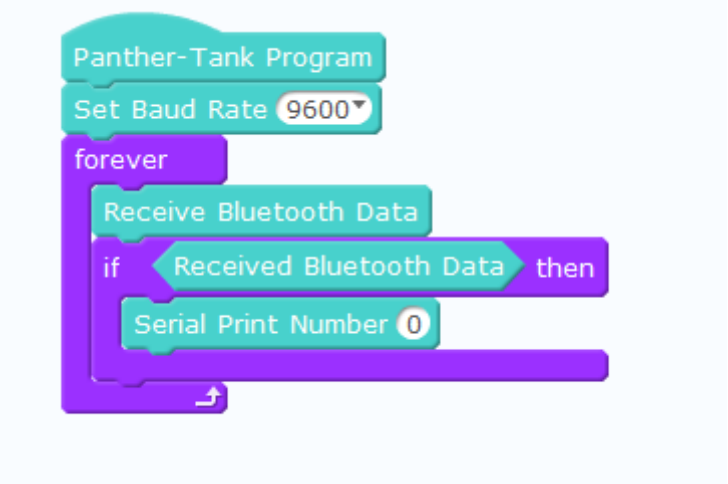


Figure 5-7

We write the program as shown in Fig. 5-7, then upload the program successfully, open the serial port, then connect with the robot Bluetooth with the mobile APP, press the left and right buttons, and the serial port will print the number.

5.4 Programming Thought of Bluetooth Remote Control Robot

The programming idea of Bluetooth remote control robot is to determine whether Bluetooth data is received or not, and then make corresponding actions according to the received data commands. Next, we write the program of Bluetooth robot together.

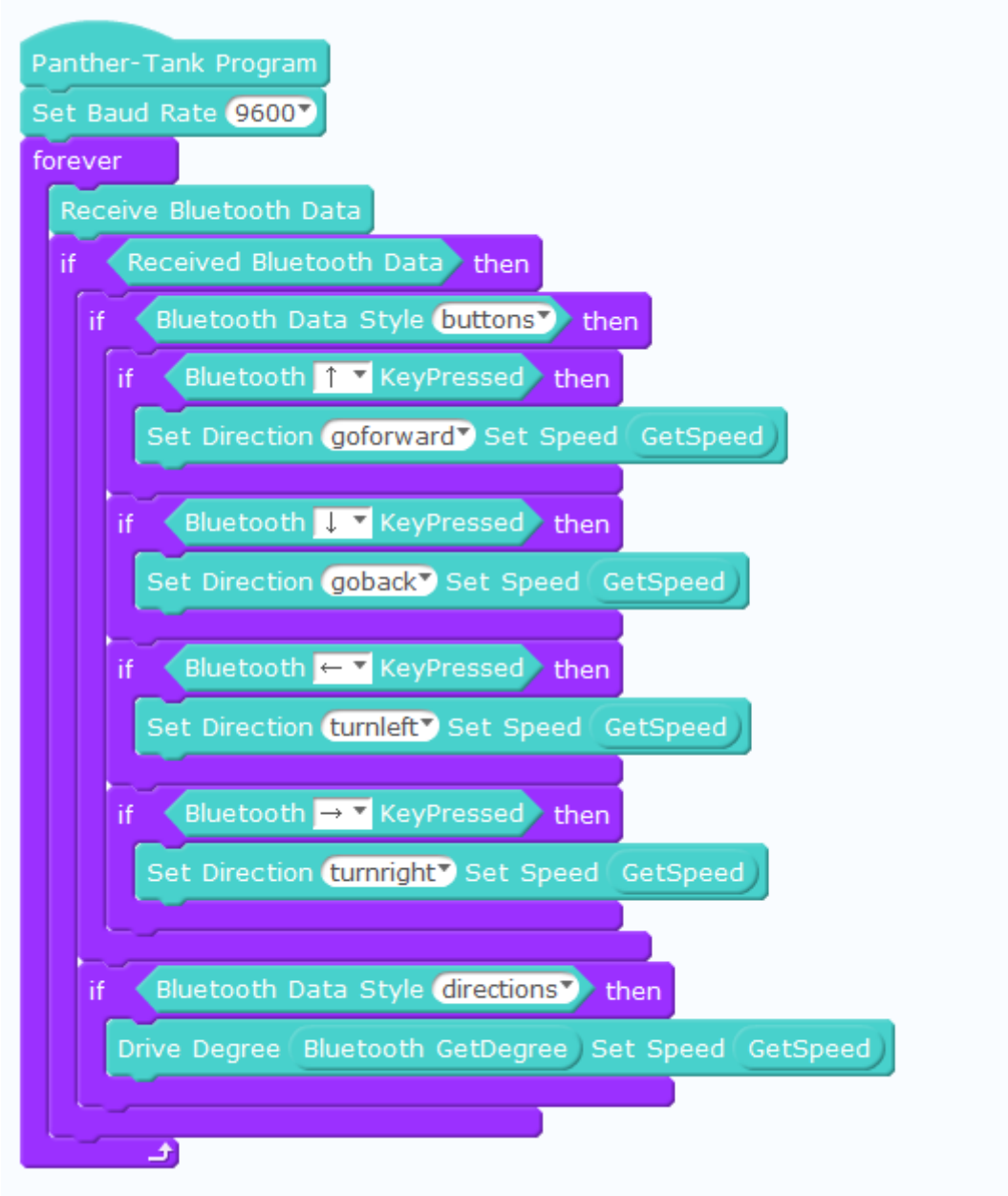


Figure 5-8

Chapter 6 PS2 Remote Control Robot

6.1 Principle of PS2 remote control handle

The PS2 handle consists of a handle (Fig. 6-1) and a receiver (Fig. 6-2). The handle needs two 1.5V batteries to supply power. Put the handle switch on ON. Without searching for the receiver, the light on the

handle flashes continuously. For a certain period of time, before searching for the receiver, the handle will enter standby mode. The lamp on the handle will be extinguished. At this time, press the "START" button to wake up the handle.

After normal power-on, the handle and the receiver are automatically paired and connected. In the unsuccessful state, the green light of the receiver flashes, and the light on the handle also flickers. After matching success, the green light on the receiver is always on, the light on the handle is always on, and the button "MODE" (the batch of the handle is different, the above logo may be "AnalyOG", but not. Will affect the use of), you can choose "red light mode" and "green light mode".

Figure 7-1

Figure 7-2

When the handle is connected to the receiver, we can use the handle to send button commands. When the receiver receives these button commands.

The robot's brain (main control board) will let his limbs (four wheels, steering gear) do the corresponding action according to the command received.

6.2 PS2 remote control test

We can first write a program to test the PS2 remote control, using serial port printing, when the PS2 remote control keys press, serial port printing the corresponding key values, we observe the contents of serial port printing, you can see which keys of the PS2 remote control are pressed, next we first write a test program.

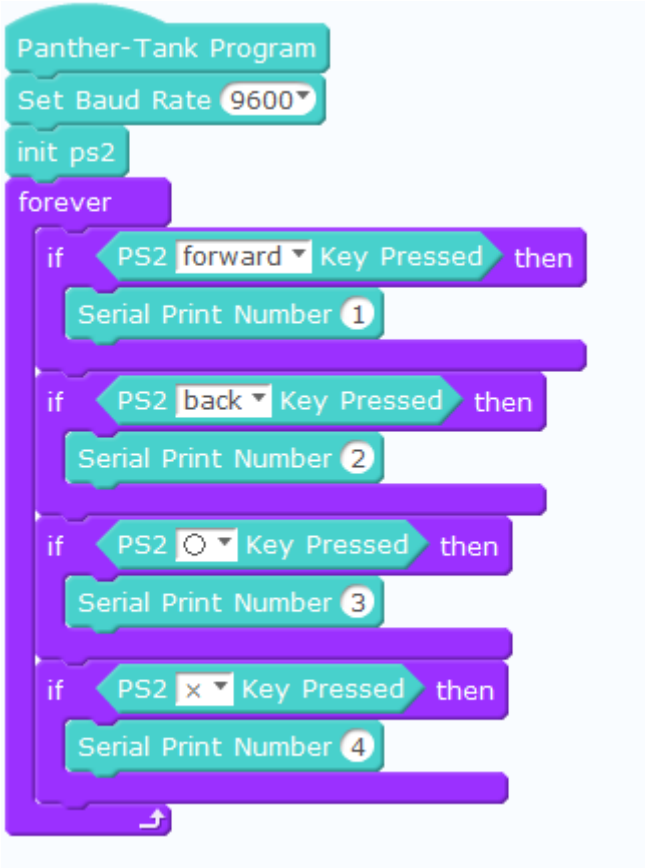


Figure 7-3

Fig. 7-3 bit PS2 test program, after uploading the program to the robot, open the serial port monitor, press the buttons around PS2, and the serial port will print the corresponding number.

6.3 Programming Thought of PS2 Remote Control Robot

The programming idea of PS2 remote control robot is to set the mode as PS2 control mode first, then define the key function on the PS2 handle according to its own needs. Our next programming is programmed according to the key definition shown in Figure 7-3. Then the robot executes the corresponding left, forward, right movement according to the received key value of the handle, or Acceleration, deceleration, steering gear operation, PS2 handle remote control car, all key functions are defined as shown in Figure 7-3:

Figure 7-4

Identify UP: Forward

Identify DOWN: Back

Identify LEFT: Turn left

Identify RIGHT: Turn right

Logo A: Acceleration

Identification B: Left Spin

Logo C: Deceleration

Identification D: Right Spin

Identification 3: Right rocker (Identification 1) control key, that is, the right rocker will work only when R1 is pressed.

Logo 4: The left rocker (logo 2) control key, that is, the left rocker will work only when L1 is pressed.

Joystick left: Adjust left RGB lamp

Joystick Right: Adjust the right RGB lamp

Now let's write the program of PS2 remote control robot.

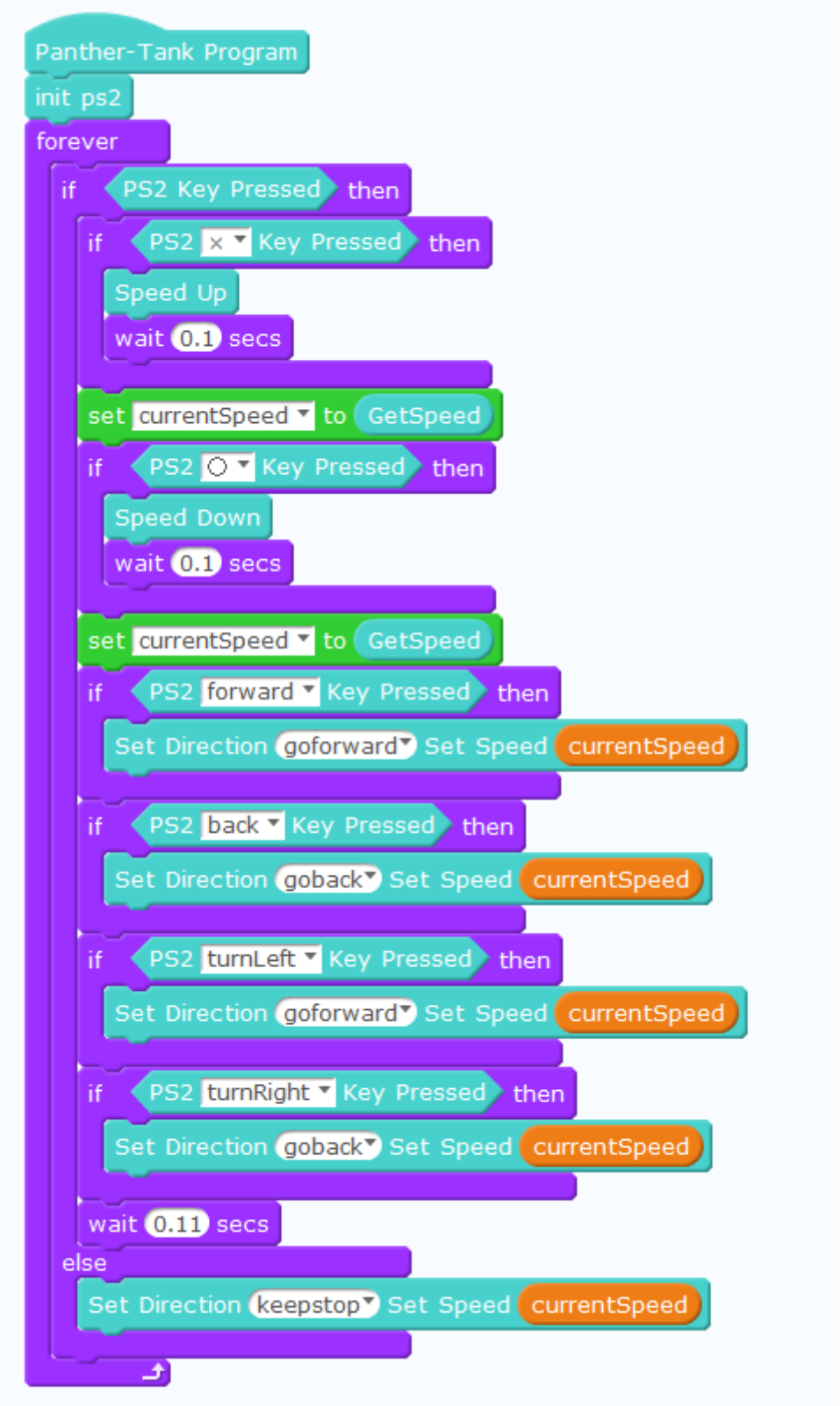


Figure 6-5

Transfer the program of Figure 6-5 to the robot. We turn on the power of the robot and connect the PS2 handle to the receiver. When we press the button on the PS2 handle, the robot will perform the corresponding action.