

4.5 Infrared Remote Control Display

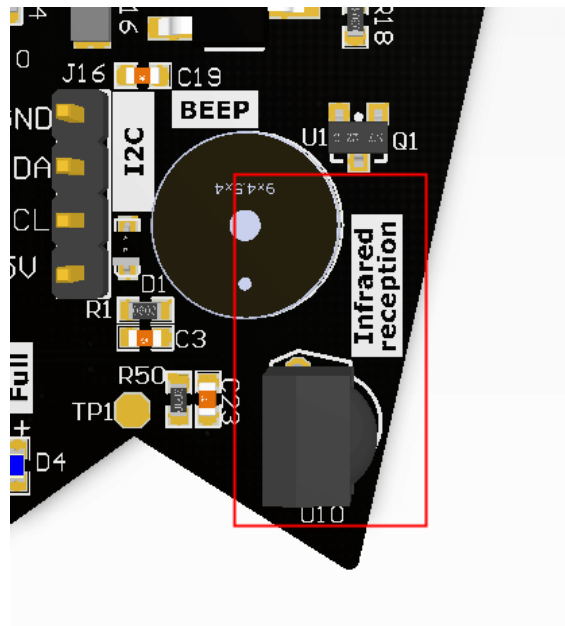
Note: The infrared receiver and remote controller will be affected by light, please run this code in an indoor environment without sunlight to reduce the interference of sunlight on the infrared sensor.

1. Learning Objectives

In this course, we will learn the experiment of infrared remote control and reception.

2. About Hardware

This course uses the infrared receiver and OLED on Pico robot. And we also need to use the infrared remote controller.



The spectrum of infrared light is outside the red light, and the wavelength is 0.76-1.5 μm , which is longer than the wavelength of red light. Infrared remote control is a control method that uses infrared to transmit information. Infrared remote control has the advantages of anti-interference, simple circuit, easy encoding and decoding, low power consumption and low cost. Infrared remote control is suitable for the control of almost all home appliances. The infrared receiving head has built-in photoelectric elements, which can receive infrared light of corresponding wavelength, convert it into digital signal, and judge different remote control buttons by reading the signal value.

The code value of the infrared remote control included in the car kit is shown in the table below.

Keys	Shell prints key values
Power	0
Up	1
Light	2
Left	4
Sound	5
Right	6
Turn Left	8
Down	9
Turn Right	10
+	12
0	13
-	14
1	16
2	17
3	18
4	20
5	21
6	22
7	24
8	25
9	26

3. About code

Code path: Code -> 2.Advanced course -> 5. IR control display.py

```

import time
from machine import Pin, I2C
from pico_car import SSD1306_I2C, ir
#initialization ir
Ir = ir()
#initialization oled
i2c=I2C(1, scl=Pin(15),sda=Pin(14), freq=100000)
oled = SSD1306_I2C(128, 32, i2c)

while True:
    #get value
    value = Ir.Getir()
    time.sleep(0.01)
    if value != None:
        print(value)
        #display press
        if value == 0:
            while value == 0:
                value = Ir.Getir()
                oled.text('Press:Power', 0, 0)
                oled.show()
                oled.fill(0)
        elif value == 1:
            while value == 1:
                value = Ir.Getir()
                oled.text('Press:Up', 0, 0)
                oled.show()
                oled.fill(0)
        elif value == 2:
            while value == 2:
                value = Ir.Getir()
                oled.text('Press:Light', 0, 0)
                oled.show()
                oled.fill(0)
        elif value == 4:
            while value == 4:
                value = Ir.Getir()
                oled.text('Press:Left', 0, 0)
                oled.show()
                oled.fill(0)
        elif value == 5:
            while value == 5:
                value = Ir.Getir()
                oled.text('Press:Sound', 0, 0)
                oled.show()
                oled.fill(0)
        elif value == 6:
            while value == 6:
                value = Ir.Getir()
                oled.text('Press:Right', 0, 0)
                oled.show()
                oled.fill(0)
        elif value == 8:
            while value == 8:
                value = Ir.Getir()
                oled.text('Press:Turn Left', 0, 0)
                oled.show()

```

```
oled.fill(0)
elif value == 9:
    while value == 9:
        value = Ir.Getir()
    oled.text('Press:Down', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 10:
    while value == 10:
        value = Ir.Getir()
    oled.text('Press:Turn Right', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 12:
    while value == 12:
        value = Ir.Getir()
    oled.text('Press:+', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 13:
    while value == 13:
        value = Ir.Getir()
    oled.text('Press:0', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 14:
    while value == 14:
        value = Ir.Getir()
    oled.text('Press:-', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 16:
    while value == 16:
        value = Ir.Getir()
    oled.text('Press:1', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 17:
    while value == 17:
        value = Ir.Getir()
    oled.text('Press:2', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 18:
    while value == 18:
        value = Ir.Getir()
    oled.text('Press:3', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 20:
    while value == 20:
        value = Ir.Getir()
    oled.text('Press:4', 0, 0)
    oled.show()
    oled.fill(0)
elif value == 21:
    while value == 21:
        value = Ir.Getir()
```

```

oled.text('Press:5', 0, 0)
oled.show()
oled.fill(0)
elif value == 22:
    while value == 22:
        value = Ir.Getir()
oled.text('Press:6', 0, 0)
oled.show()
oled.fill(0)
elif value == 24:
    while value == 24:
        value = Ir.Getir()
oled.text('Press:7', 0, 0)
oled.show()
oled.fill(0)
elif value == 25:
    while value == 25:
        value = Ir.Getir()
oled.text('Press:8', 0, 0)
oled.show()
oled.fill(0)
elif value == 26:
    while value == 26:
        value = Ir.Getir()
oled.text('Press:9', 0, 0)
oled.show()
oled.fill(0)
value = None

```

We use `while value == key value:` to achieve the effect of pressing and releasing the remote control and then executing.

from pico_car import SSD1306_I2C, ir

Use SSD1306_I2C and ir from pico_car, which is our packaged OLED and IR receiver library.

import time

The "time" library. This library handles everything time related, from measuring it to inserting delays into programs. The unit is seconds.

from machine import Pin, I2C

The machine library contains all the instructions that MicroPython needs to communicate with Pico and other MicroPython-compatible devices, extending the language of physical computing, using the Pin and I2C libraries here.

i2c=I2C(1, scl=Pin(15),sda=Pin(14), freq=100000)

Set the IIC 1 pin to SCL 15, SDA 14, and the frequency to 100000.

oled = SSD1306_I2C (128, 32, i2c)

Initialize the size of the OLED to 128*32, and pass in the IIC parameters set earlier.

Ir = ir ()

Initialize the infrared remote control.

value = Ir.Getir ()

Read the infrared remote control value and assign it to the variable value.

oled.show ()

Display the set OLED content.

oled.fill (0)

Clear the settings and prepare for the next display.

oled.text ('Press: Power', 0, 0)

Display the corresponding key on the OLED, for example, press the power key to display 'Press: Power'.

4. Experimental Phenomenon

After the code is downloaded, when we press the button on IR controller, the corresponding button name will be displayed on the OLED.

At the same time, the print shell will also print corresponding button value.



As shown below.

Keys	Shell prints key values	OLED display
Power	0	Press:Power
Up	1	Press:Up
Light	2	Press:Light
Left	4	Press:Left
Sound	5	Press:Sound
Right	6	Press:Right
Turn Left	8	Press:Turn Left
Down	9	Press:Down
Turn Right	10	Press:Turn Right
+	12	Press:+
0	13	Press:0
-	14	Press:-
1	16	Press:1
2	17	Press:2
3	18	Press:3
4	20	Press:4
5	21	Press:5
6	22	Press:6
7	24	Press:7
8	25	Press:8
9	26	Press:9