Course12 --- IR control

Note：

1. When performing infrared remote control, the remote controller should face the infrared receiver on the expansion board.

2. There is a plastic piece on the bottom of the infrared remote controller that needs to be taken down for normal use.

3. The infrared light emitted by the infrared remote controller and the infrared receiver is invisible to the human eye. It can be seen under the camera without filtering infrared light.

Learning goal：

This lesson learns how to use infrared receiver to receive and process signals by Python programming.

Code :

Please use the MU software to open the IR control.py file we provided.

1) Import the library needed for this lesson from micro:bit, display is used for dot matrix display, tinybit controls the car, sleep delay time, music is play music, pin8 is the pin of the infrared receiver.

display.off (): Turn off the micro:bit dot matrix screen. If it is not turned off, it will conflict with the infrared receiver and cause errors.

tinybit.init_IR (pin8): Initialize the infrared receiver. The parameter is the pin of the infrared sensor.

music.play('c')：Buzzer whistle.

Code as shown below:

```
# -*- coding: utf-8-*-# Encoding cookie added by Mu Editor
from microbit import display, pin8, sleep
import tinybit
import music

display.off()
tinybit.init_IR(pin8)
music.play('c')
```

According to the hardware interface reference manual, the infrared receiver connect to P8 of micro:bit.

Category	Function	Number	Drive	The number of Drive pin	The number of connected to the controller	micro:bit
Buzzer	Buzzer	FM	Micro:bit drive directly		FM	P0
Voice sensor	Voice sensor	MIC			MIC	P1
LED light	Water light	LED-RGB			LED-RGB	P12
Tracking sensor	Left tracking	L-DET			L-DET	P13
	Right tracking	R-DET			R-DET	P14
Ultrasonic module	Echo pin	ECHO			ECHO	P15
	Trigger pin	TRIG			TRIG	P16
Infrared receiver	Infrared remote control	RX			RX	P8
I2C interface	I2C interface	SCL			SCL	P19
		SDA			SDA	P20
Motor	Left motor Forward	L-INA	STM8S	PC6/TIM1_CH1	SCL, SDA	P19, P20
	Left motor Reverse	L-INB		PC7/TIM1_CH2		
	Right motor Forward	R-INA		PC3/TIM1_CH3		
	Right motor Reverse	R-INB		PC4/TIM1_CH4		
RGB Searching light	Red	LED-R		PC5/TIM2_CH1		
	Green	LED-G		PD3/TIM2_CH2		
	Blue	LED-B		PD2/TIM2_CH3		

1) Read the data received by the infrared receiver, and only take the upper 8 bits. The key code value of the remote control is shown in the following figure:

value = tinybit.get_IR (pin8): read the value received by the infrared receiver and save it to value;

value = value >> 8: Since the data received from the infrared receiver is 16 digits, and the upper 8 and lower 8 bits are reversed, the higher 8 bits can be used.

```
20  while True:
21      # print(hex(tinybit.get_IR(pin8)))
22      value = tinybit.get_IR(pin8)
23      value = value >> 8
```

2) First,we need to process two special values. By default, you will always receive the value-0x01. If you press and hold a key, you will receive 0xff. The variable a is to solve the problem that you may receive -0x01 when you press and hold the key. The code value of each key.

```
16      # default
17      if value == -0x01:
18          music.pitch(0)
19      # long pressed
20      elif value == 0xff:
21          music.pitch(500)
```

3) Press button, buzzer will whistle.

```
        # off
        if value == 0x00:
            music.play('c')
        # up
        elif value == 0x80:
            music.play('c')
        # light
        elif value == 0x40:
            music.play('c')
        # left
        elif value == 0x20:
            music.play('c')
        # buzzer
        elif value == 0xa0:
            music.play('c')
        # right
        elif value == 0x60:
            music.play('c')
```

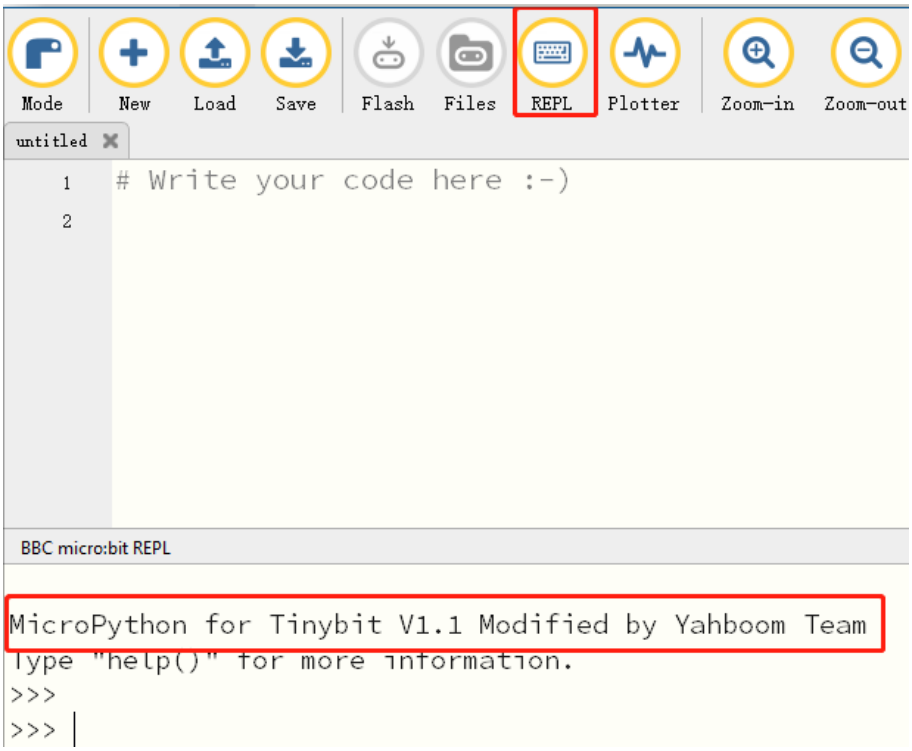
Programming and downloading：

1.You should open the Mu software, and enter the code in the edit window, , as shown below.

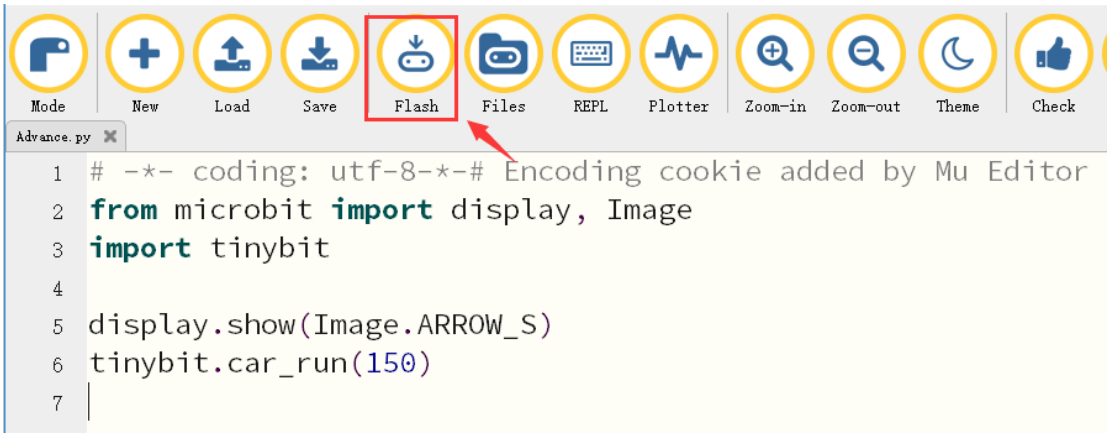
Note! All English and symbols should be entered in English, and the last line must be a space.

2.You can click the **“Check”** button to check if our code has an error. If a line appears with a cursor or an underscore, the program indicating this line is wrong.

3.Click **"REPL"** button,check whether the tinybit library has been downloaded. If not, please refer to the [preparation before class]---> [Python programming]



4.Click the **"Flash"** button to download the program to micro:bit board.



If the program is wrong or the experimental phenomenon is wrong after downloading, please confirm whether you have downloaded the Buildingbit libraryhex file we provided to the micro: bit board.
For the specific method of adding library files, please refer to 【1.Preparation before class】---【Python programming】

Experimental phenomena

After download is complete, we can hear buzzer sound. When we press the button of the infrared remote controller, buzzer will sound, indicating that the reception is successful.