

3.7 Battery power display

I. Learning objectives

1. Learn the power detection and OLED display of the Raspberry Pi Pico 2/Pico mainboard and the car expansion board.
2. Understand the use of power detection.

II. Hardware use

This course uses the Pico 2/Pico mainboard and the car expansion board's power detection and OLED display. **Note that before running this section of the routine, the OLED display must be plugged in, otherwise the program will report an error.**



The basic principle of the hardware of power detection is to adjust the battery voltage to a suitable value through a resistor and input it to the main control. The main control detects the voltage through the ADC to determine the power.

3. Program Analysis

Code path: Code -> 1. Basic course -> 7. Battery power display.py

```
from machine import Pin, I2C, ADC
from pico_car import SSD1306_I2C
import time

#initialization oled
i2c=I2C(1, scl=Pin(15),sda=Pin(14), freq=100000)
oled = SSD1306_I2C(128, 32, i2c)
#initialization ADC
Quantity_of_electricity = machine.ADC(28)
```

```
while True:
    #Display power on OLED
    #Under 20000, there is no power at all
    oled.text('Battery:', 0, 0)
    oled.text(str(Quantity_of_electricity.read_u16()), 65, 0)
    oled.show()
    oled.fill(0)
    time.sleep(0.1)
```

from pico_car import SSD1306_I2C

Use SSD1306_I2C from pico_car.

import time

The "time" library. This library handles everything to do with time, from measuring it to inserting delays into your program. The unit is seconds.

from machine import Pin, I2C, ADC

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

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

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

oled = SSD1306_I2C(128, 32, i2c)

Initialize the size of OLED to 128*32, and pass the previously set IIC parameters into it.

Quantity_of_electricity = machine.ADC(28)

Initialize ADC port 28.

oled.text(str(Quantity_of_electricity.read_u16()), 65, 0)

Set the OLED to display the battery power value at 65,0. We use Quantity_of_electricity.read_u16() to read the ADC value, convert the value into a string through the str() function, and then display it on the OLED.

oled.show()

Display the set OLED content.

oled.fill(0)

Clear the set content and prepare for the next display.

Fourth, Experimental Phenomenon

After the program is downloaded, we can see that the OLED displays 'Battery: Battery Power'. This power will change with the voltage. After testing, when the value is less than 20000, the battery is in a low power state, and less than 25000, it is in a medium power state.

