

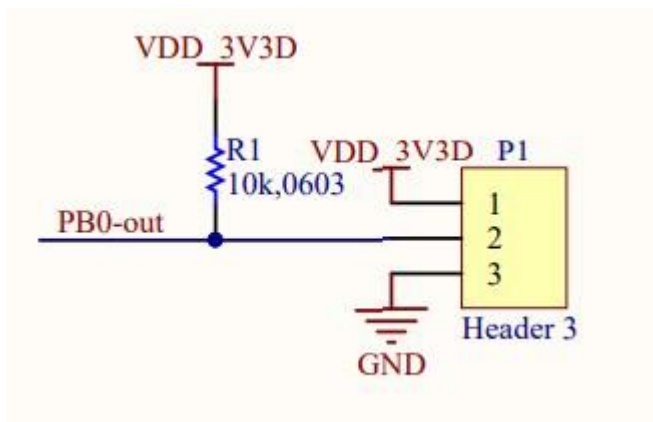
本课学习内容：

- 了解 onewire 是一种芯片级别的通讯
- 实现 18B20 温度读取

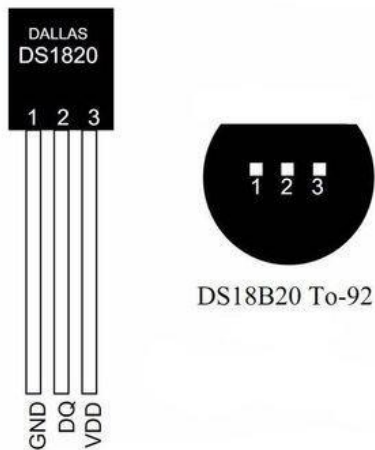
18B20 是常用的数字温度传感器，其输出的是数字信号，具有体积小，硬件开销低，抗干扰能力强，精度高的特点。DS18B20 数字温度传感器接线方便，封装成后可应用于多种场合，如管道式，螺纹式，磁铁吸附式，不锈钢封装式，型号多种多样，有 LTM8877，LTM8874 等等。

主要根据应用场合的不同而改变其外观。封装后的 DS18B20 可用于电缆沟测温，高炉水循环测温，锅炉测温，机房测温，农业大棚测温，洁净室测温，弹药库测温等各种非极限温度场合。耐磨耐碰，体积小，使用方便，封装形式多样，适用于各种狭小空间设备数字测温和控制领域。

18B20 通讯方式为 onewire 通讯，加上供电线和 GND 公共端一共 3 根线就能完成测温通讯。以下是 PYBminiDB 底板电路图。



这是 18B20 底座示意图，用户在测试时候需要自行插入封装为 T0-92 的 18B20 器件。注意不要插反，容易引起短路和发热情况，严重时烧坏器件。



```
G:\main.py - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
main.py x
4 from machine import Pin
5 display = SSD1306(pinout={'sda': 'PB7', 'scl': 'PB6'}, height=64, external_vcc=False)
6 ow = onewire.OneWire(Pin('PB0'))
7 ds = ds18x20.DS18X20(ow)
8 def funcA(sec):
9     try:
10         display.poweron()
11         display.init_display()
12         display.draw_text(1,1,'18b20Tem')
13         while True:
14             try:
15                 roms = ds.scan()#扫描总线上的设备
16                 ds.convert_temp()#获取采样温度
17                 for rom in roms:
18                     tem1 = ds.read_temp(rom)
19                     print("18B20TEM:", tem1)
20                     display.draw_text(55,1,str(round(tem1,1)))
21             except:
22                 display.draw_text(55,1,'NC')
23                 display.display()
24                 time.sleep_ms(30)
25         except Exception as ex:
26             print('Unexpected error: {}'.format(ex))
27 if __name__ == '__main__':
28     _thread.start_new_thread(funcA, (1,))
```

我们解析一下上述程序。

import pyb, _thread, time, onewire, ds18x20 #引用 pyb, 线程, 时间, 一线通讯, 18b20
from pyb import Pin#引用引脚

```
from ssd1306 import SSD1306#引用 SS1306

from machine import Pin#引用引脚

display = SSD1306(pinout={'sda': 'PB7', 'scl': 'PB6'}, height=64, external_vcc=False)#OLED 初始化

ow=onewire.OneWire(Pin('PB0'))#初始化 onewire

ds=ds18x20.DS18X20(ow)#初始化 18B20

def funcA(sec):#线程 A

    try:#异常处理

        display.poweron()#OLED 开启

        display.init_display()#OLED 显示

        display.draw_text(1, 1, '18b20Tem')#显示内容

        while True:#死循环

            try:#非异常处理

                roms = ds.scan()#扫描总线上的设备

                ds.convert_temp()#获取采样温度

                for rom in roms:#遍历获得的器件的数据

                    tem1=ds.read_temp(rom)#读取温度

                    print("18B20TEM:", tem1)#向 REPL 发送数据

                    display.draw_text(55, 1, str(round(tem1, 1)))#OLED 显示

            except :#异常处理

                display.draw_text(55, 1, 'NC')#异常处理函数

        display.display()#OLED 显示

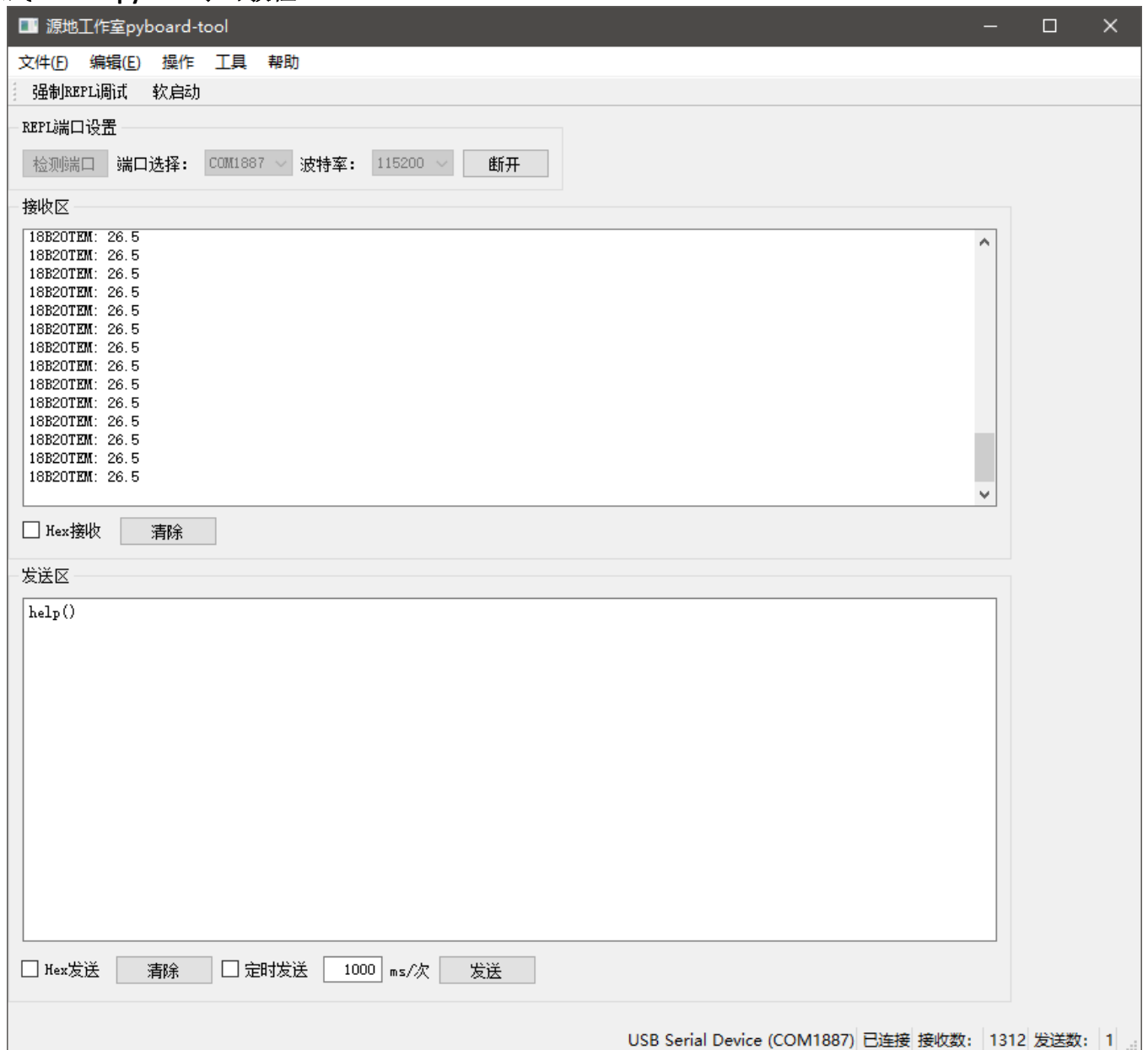
        time.sleep_ms(30)#延时

    except Exception as ex:#异常处理

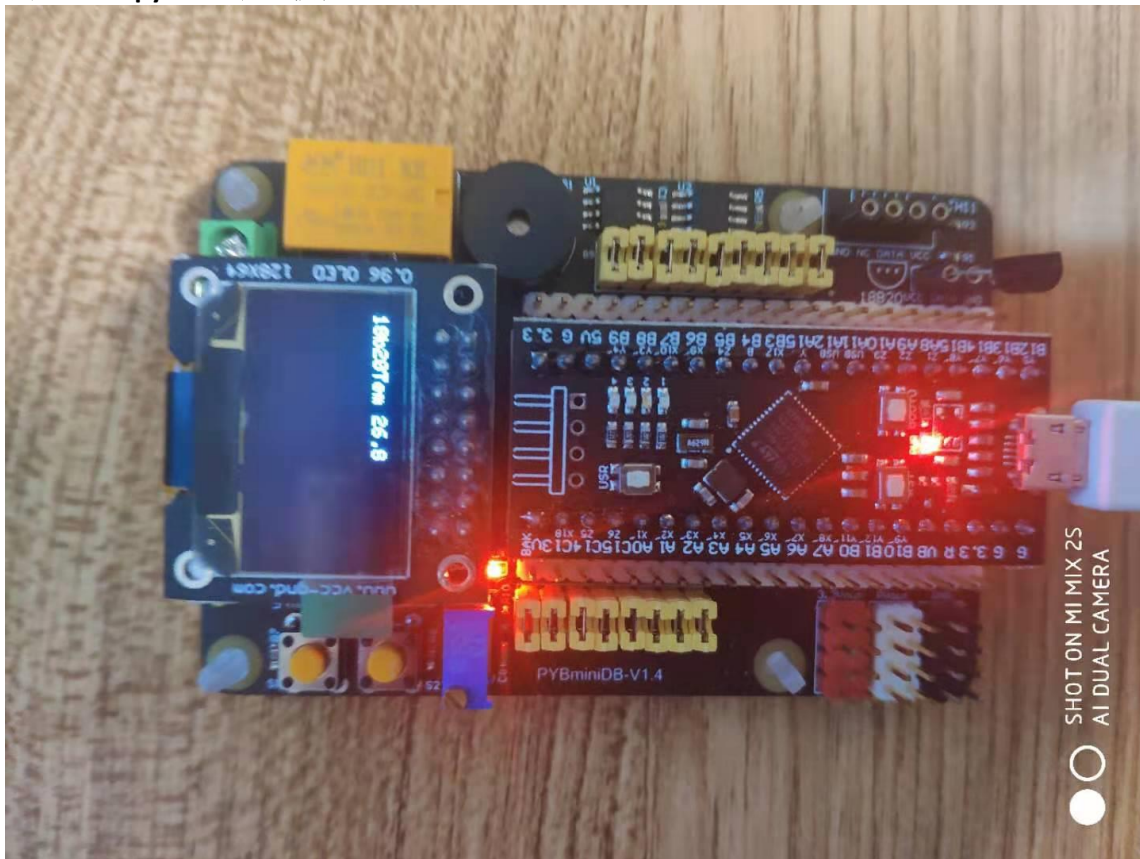
        print('Unexpected error: {0}'.format(ex))#异常处理函数

if __name__ == '__main__':#程序入口

    _thread.start_new_thread(funcA, (1,))#线程 A 开启
```



这是 REPL 打印的数据。



这是硬件显示情况。