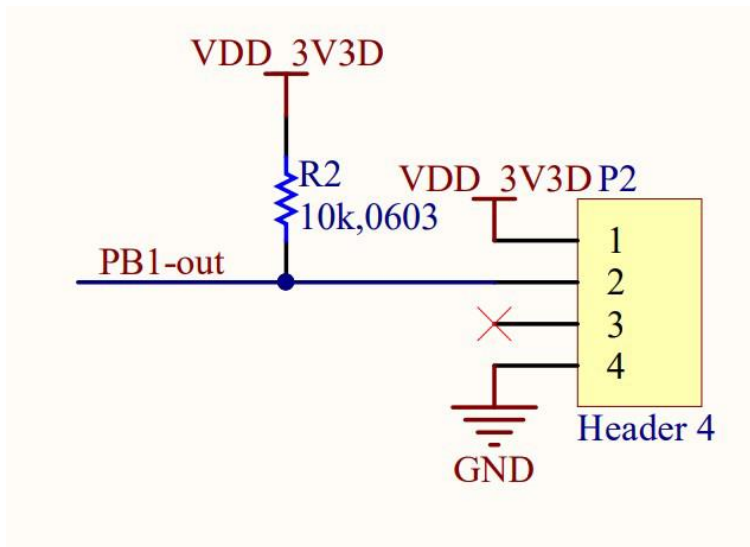


本课学习内容：

- 再次了解 onewire 是一种芯片级别的通讯
- 实现 DHT11 温度与湿度读取显示，向 REPL 打印信息

DHT11 数字温湿度传感器是一款含有已校准数字信号输出的温湿度复合传感器，它应用专用的数字模块采集技术和温湿度传感技术，确保产品具有极高的可靠性和卓越的长期稳定性。传感器包括一个电阻式感湿元件和一个 NTC 测温元件，并与一个高性能 8 位单片机相连接。因此该产品具有品质卓越、超快响应、抗干扰能力强、性价比极高等优点。每个 DHT11 传感器都在极为精确的湿度校验室中进行校准。校准系数以程序的形式存在 OTP 内存中，传感器内部在检测信号的处理过程中要调用这些校准系数。单线制串行接口，使系统集成变得简易快捷。超小的体积、极低的功耗，使其成为该类应用中，在苛刻应用场合的最佳选择。产品为 4 针单排引脚封装，连接方便。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。

DHT11 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，通过原理图我们知道 PB1 连接到 DHT11 的信号引脚。



```
from ssd1306 import SSD1306#引用 SSD1306  
  
import _thread,time,dht#引用线程，时间，dht 库  
  
from machine import Pin#引用 PIN 库
```

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

```
dht = dht.DHT11(Pin("PB1"))#dht11 初始化
```

```
def funcA(sec):#线程 A
```

```
    time.sleep(sec)#延时
```

```
    try:#异常处理
```

```
        display.poweron()#OLED 上电
```

```
        display.init_display()#OLED 初始化
```

```
        display.draw_text(1, 1, 'DHT11Tem')#OLED 显示内容 1
```

```
        display.draw_text(1, 13, 'DHT11Hum')#OLED 显示内容 2
```

```
        while True:#死循环
```

```
            try:#异常处理
```

```
                dht.measure()#DHT 数据获取
```

```
                print("temperature:", dht.temperature())#向 REPL 打印温度
```

```
                print("humidity:", dht.humidity())#向 REPL 打印湿度
```

```
                display.draw_text(55, 1, str(round(dht.temperature(), 1)))#OLED 显示
```

温度

```
                display.draw_text(55, 13, str(round(dht.humidity(), 1)))#OLED 显示湿
```

度

```
            except :#异常处理
```

```
                print("DHT11 lose")#打印就是异常
```

```
                display.display()#OLED 显示
```

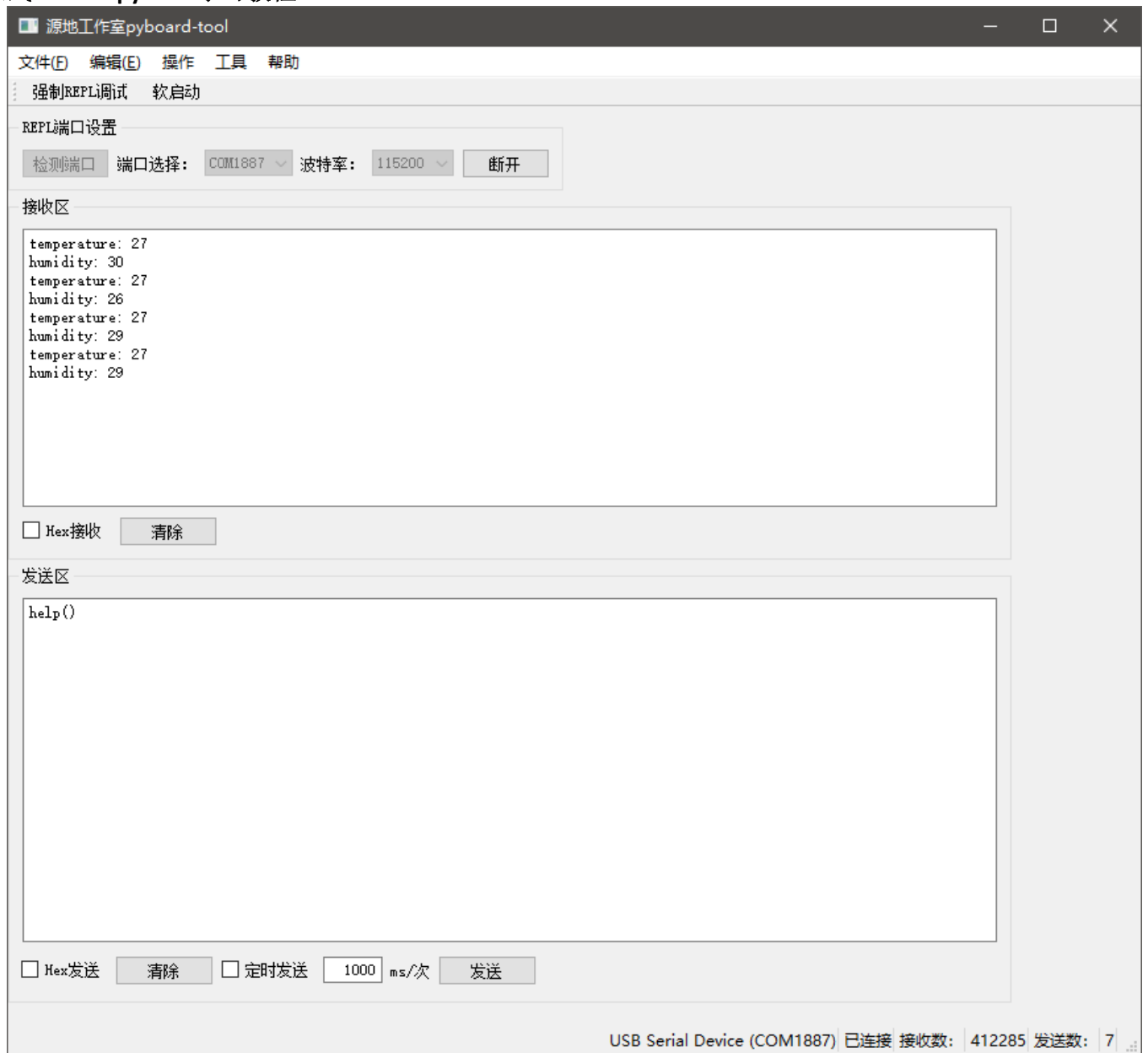
```
                time.sleep_ms(1000)#延时
```

```
        except Exception as ex:#异常处理
```

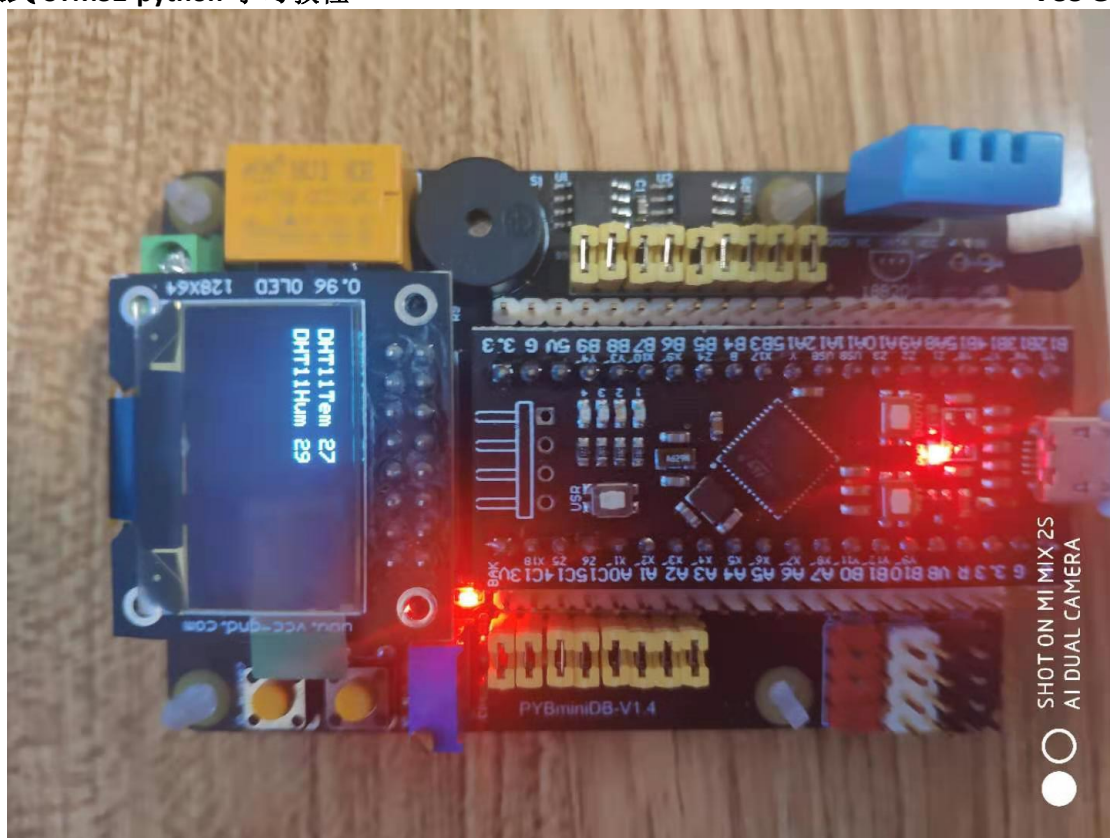
```
            print('Unexpected error: {0}'.format(ex))#异常处理函数
```

```
if __name__ == '__main__':#程序入口
```

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



这是 REPL 显示信息。



这是实物显示效果。