

QuecPython main.py 文件使用

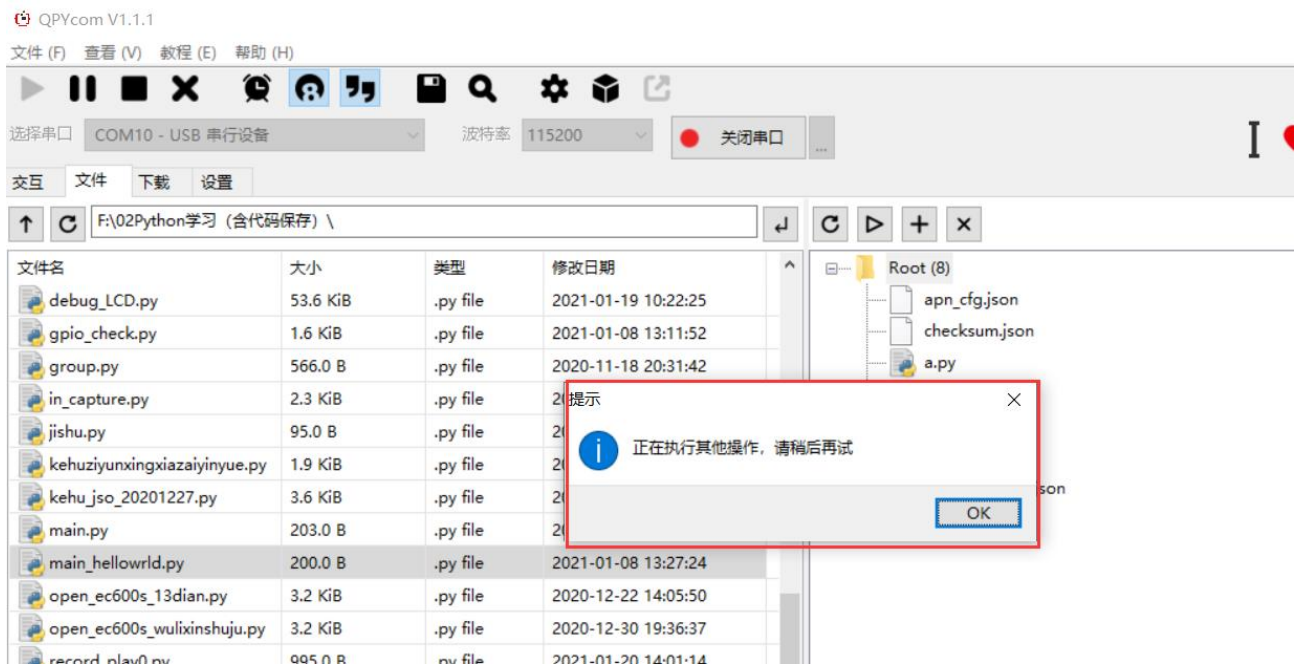
对于 main.py 的几点建议：

- 对于开始的调试，建议不要将程序命名为：**main.py**，建议使用其他名字，例如：**start.py** 等等。如果里面存在死循环，那么在程序运行时无法中断，对于此情况，暂时只能通过重刷固件解决。
- 如果真的想使用 **main.py** 程序，同时又需要使用死循环，记得在死循环里面加一个可中断的条件，例如（如下的例子是在一个线程里面加入了死循环，同时在这个死循环里面加入了中断退出的条件，当一直输出时，可通过 **GPIO2** 对应的按键中断输出）：

```
import log
import _thread
from machine import Pin
import utime
log.basicConfig(level=log.NOTSET)
KEY_log = log.getLogger("KEY")
gpio2 = Pin(Pin.GPIO2, Pin.IN, Pin.PULL_DISABLE, 0)
def in_capture():
    KEY_log.debug("in_capture start!")
    while True:
        KEY_log.info("1111")
        utime.sleep(1)
        if gpio2.read() == 0:
            KEY_log.info("in_capture thread end")
            break
        else:
            pass
if __name__ == "__main__":
    _thread.start_new_thread(in_capture, ())
```

- 对于自运行没有 log 输出，可以参考下面现象 2 的解决方法。

现象 1：上传 **py** 文件且文件名为 **main.py** 到模块后无法任何执行指令（包括上传文件等），类似现象如下截图



导致原因：模块在开机后会主动寻找运行文件名为 `main.py` 的脚本文件，如果 `main.py` 中存在 `while`, `for(,)` 这种循环语句，会导致程序阻塞，串口被占用，无法进行其他操作
 解决方案：目前版本只能通过重刷固件解决。

现象 2: 手动运行 `main.py` 程序, 可以看到通过 QPYcom 连接“USB 串行设备”看到 `print` 输出和 LOG 打印信息, 但是自运行 `main.py` 程序后, 在 QPYcom 的交互界面, 什么都看不到。

解决方法：可以加入串口打印。

(1) 软件代码可参考：

```
from machine import UART # 导入 UART 模块
uart = UART(UART.UART2, 115200, 8, 0, 1, 0) #配置成 UART2 输出（硬件的连接）
count = 50
while count:
    uart.write('main_py_UART_msg:{}\r\n'.format(count))
    count -= 1
```

说明：

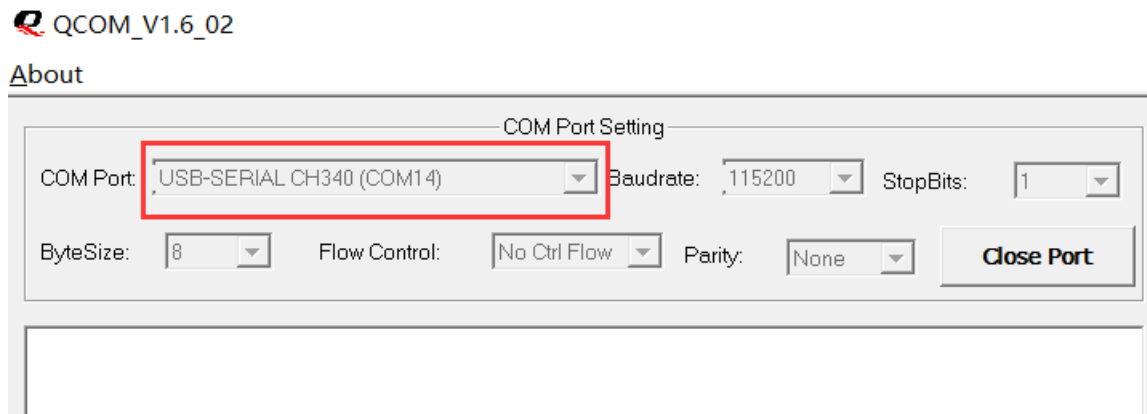
- 串口打印是需要导入 `UART` 模块，按照 `UART` 的 API 库来书写输出代码

(2) 硬件连接可参考：

对于代码中提到的硬件连接（`UART2`），是使用 `type-c` 给模块供电，`UART` 与 `TTL` 转 `USB` 模块的连接如下表，`TTL` 转 `USB` 模块直接插 `PC` 上：

模块 UART_pin 脚	TTL 转 USB 模块
RX1	Tx
TX1	Rx
GND	GND

使用 QCOM 连接“TTL 转 USB 模块”对应的串口，如下图：



鉴于上述的软硬件配置后，在自运行 main.py 程序时，就会在 QCOM 看到对应的打印，如下图：

