

# QuecPython FAQ&Debug 手册

LTE 系列

版本: Quectel QuecPython\_FAQ&Debug 手册\_V1.0

日期: 2020-11-27

状态: 临时文件

上海移远通信技术股份有限公司始终以为客户提供最及时、最全面的服务为宗旨。如需任何帮助，请随时联系我司上海总部，联系方式如下：

上海移远通信技术股份有限公司

上海市闵行区田林路 1016 号科技绿洲 3 期（B 区）5 号楼 邮编：200233

电话：+86 21 51086236 邮箱：[info@quectel.com](mailto:info@quectel.com)

或联系我司当地办事处，详情请登录：

<http://www.quectel.com/cn/support/sales.htm>

如需技术支持或反馈我司技术文档中的问题，可随时登陆如下网址：

<http://www.quectel.com/cn/support/technical.htm>

或发送邮件至：[support@quectel.com](mailto:support@quectel.com)

## 前言

上海移远通信技术股份有限公司提供该文档内容用以支持其客户的产品设计。客户须按照文档中提供的规范、参数来设计其产品。由于客户操作不当而造成的人身伤害或财产损失，本公司不承担任何责任。在未声明前，上海移远通信技术股份有限公司有权对该文档进行更新。

## 版权申明

本文档版权属于上海移远通信技术股份有限公司，任何人未经我司允许而复制转载该文档将承担法律责任。

版权所有 ©上海移远通信技术股份有限公司 2020，保留一切权利。

**Copyright © Quectel Wireless Solutions Co., Ltd. 2020.**

# 文档历史

## 修订记录

版本	日期	作者	变更表述
1.0	2020-11-24	吴永聪	初始版本
1.0	2020-11-26	叶智强	增加客户常见问题解决方案

## 目录

文档历史.....	1
1 QuecPython SDK 包.....	4
1.1 获取 QuecPython SDK 包.....	4
1.2 QuecPython SDK 包解压.....	4
1.3 准备小熊派开发板.....	5
1.4 驱动安装.....	5
1.4.1 驱动安装时 CDC 串口出不来怎么办.....	6
1.5 固件包下载.....	7
1.5.1 固件包下载失败，板子成砖了怎么办.....	8
2 Python 代码编写及下载调试.....	9
2.1 QuecPython API 类库文档.....	9
2.2 编写 python 源程序代码.....	10
2.3 QPYcom 工具脚本下载调试.....	11
3 QuecPython 模块运行常见 Debug.....	14
3.1 umqtt 使用.....	14
3.1.1 使用 MQTT 连接阿里云或者腾讯云.....	14
3.2 usocket 使用.....	14
3.3 _thread 多线程创建.....	15
3.4 定时器 timer 使用.....	16
3.5 QuecPython 是否支持队列.....	17
4 其他常见问题.....	18
4.1 socket 解析 IP 失败.....	18
4.2 执行脚本文件提示语法错误.....	19
4.3 模块运行 main.py 脚本文件.....	19
4.4 SIM 卡进行插拔后出现网络请求失败.....	19
4.5 MQTT 连接一段时间异常断开.....	19
4.6 QuecPython 内部提供给用户的文件存储空间大小.....	20
4.7 升级最新版本固件包失败了，现在上电不开机了串口也不显示.....	20
4.8 拿到开发板后不知道怎么使用.....	20
4.9 如何下载固件.....	20
4.10 固件包从哪里获取.....	20
4.11 USB 串行设备未识别.....	21
4.12 EC100Y 平台支持多少个串口.....	21
4.13 QuecPython 支持的 API 详细介绍.....	21
4.14 实战项目 demo.....	21
4.15 QPYcom 窗口太大.....	21
4.16 模块是什么平台，跑什么系统.....	21
4.17 QPYcom 烧录等待时间过长，无反应.....	21
4.18 QuecPython 的源码文件安全吗.....	22
4.19 在 QPYcom 操作没有任何反应.....	22
4.20 运行 main.py 程序之后，QPYcom 无法操作 EC100Y.....	22
4.21 开发板送的流量卡可以混用吗.....	22
4.22 编写 Python 代码用什么工具.....	22

---

4.23 板子接出的串口无法通信.....	22
4.24 接上 USB 线灯不亮.....	22

# 1 QuecPython SDK 包

## 1.1 获取 QuecPython SDK 包

访问 QuecPython 官网 <http://qpy.quectel.com/down.html#SDK> 下载最新版本 QuecPython SDK 包。

SDK

QuecPython 的 SDK。

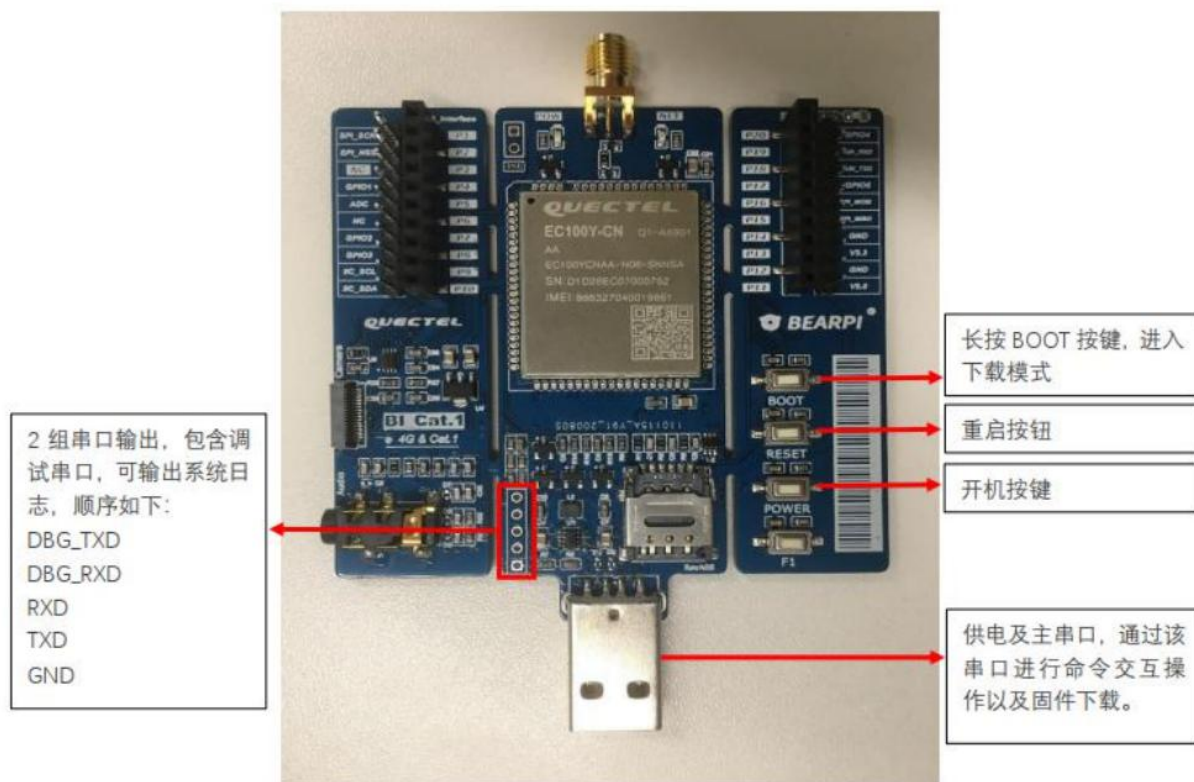
文件名	平台	版本	更新日期	大小	下载地址
QPY_V0006_EC100Y_BIN.zip	EC100Y	V0.6	2020-11-10	173MB	<a href="#">点此下载</a>

## 1.2 QuecPython SDK 包解压

解压下载的 QuecPython SDK 包，解压后的目录结构如图所示：

	boards	板级Feature, 当前支持ASR cat	修改日期: 2020/11/19 11:38
	cloud	云支持, 当前支持阿里/腾讯/QuecCloud	修改日期: 2020/11/19 11:38
	demo	支持的python moudles	修改日期: 2020/11/19 11:38
	document	SDK文档目录, 包括工具使用 等	修改日期: 2020/11/19 11:38
	firmware	固件包, 需要先下载到模块中去	修改日期: 2020/11/19 11:38
	Solution	行业Solution, 包含云喇叭/Tracker等	修改日期: 2020/11/23 14:55
	tools	工具集合、驱动、下载交互工具等	修改日期: 2020/11/19 11:38
	README.md		修改日期: 2020/11/9 18:46 大小: 605 字节

## 1.3 准备小熊派开发板



关于调试串口的使用说明:

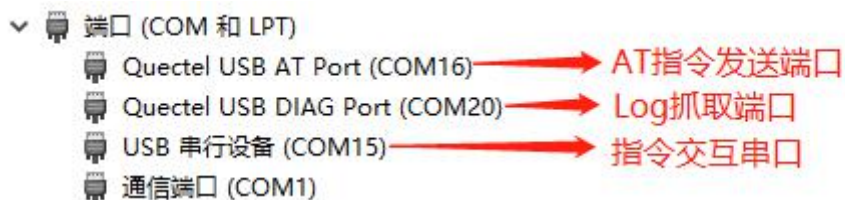
DBG\_TXD、DBG\_RXD 为调试信息输出端口, 用户使用该端口时, 需要将 DBG\_TXD、DBG\_RXD 以及 GND 连接到 USB To TTL 的转换器, 通过转换器连接到电脑串口。

## 1.4 驱动安装

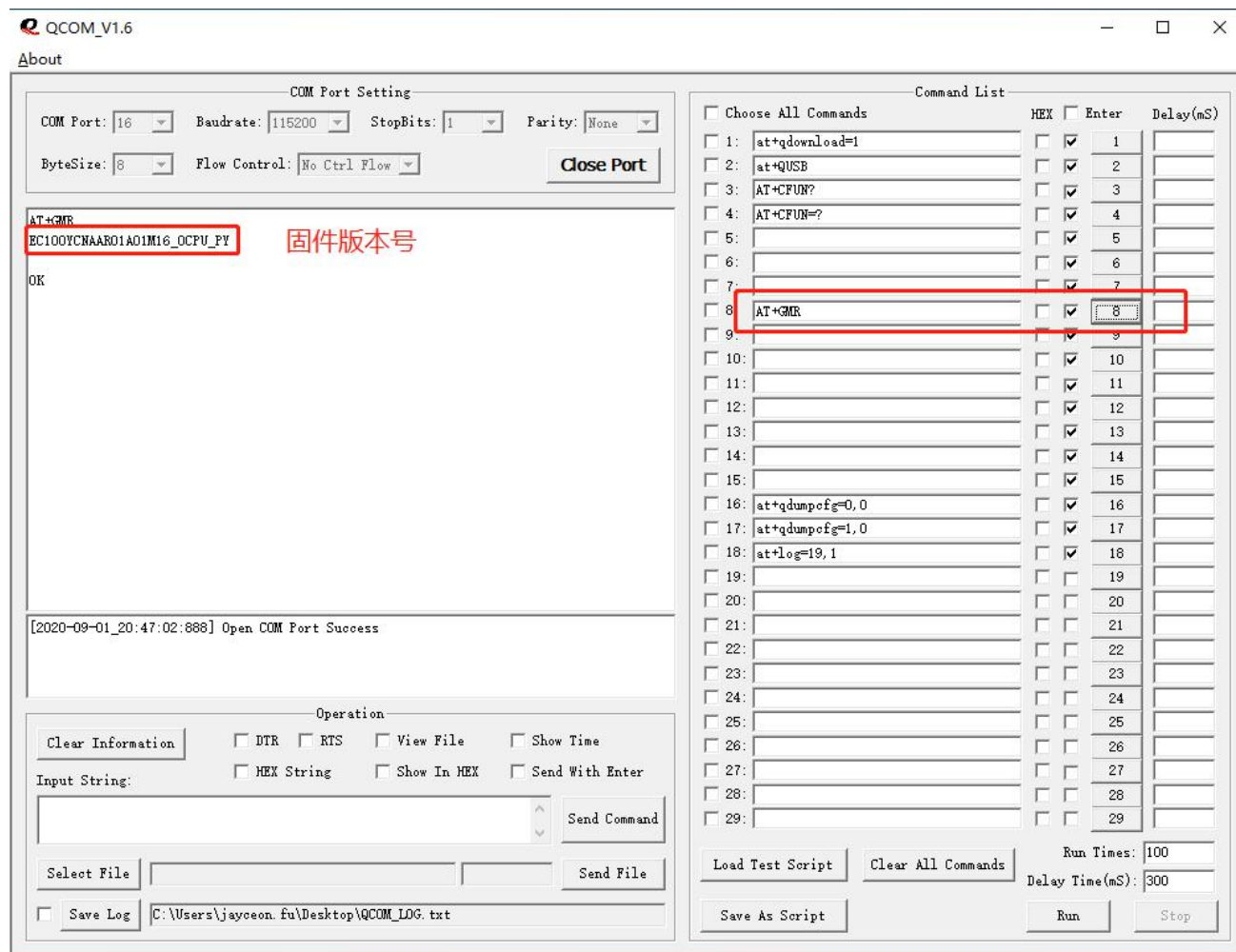
在 SDK 包中【**tools---->USB 驱动 USB Driver**】目录下, 找到如下驱动安装包, 解压后双击运行 setup.exe 直接安装即可。



驱动安装完成后, 插上小熊派开发板, 进入电脑的设备管理器, 点击【端口(COM 和 LPT)】应该会出现如下三个端口 (端口号可能与下图中不同, 但是前面的名称是一样的), 出现以下三个串口, 代表驱动安装成功:



打开 SDK 包中【**tools-->串口工具 Serial Tools**】目录下的 QCOM 软件，通过发送一条 AT 指令，确认驱动是否安装成功，如果能正常发送 AT 指令并回复成功，那么驱动安装成功；注意端口号要选择【**Quectel USB AT Port**】对应的端口号，波特率选择 115200，停止位 1 位，无奇偶校验，8 位数据位，无硬件控制流。



## 1.4.1 驱动安装时 CDC 串口出不来怎么办

- 检查系统环境，在设备管理器中查看串口号，确认串口号没有被其他端口占用，更换插到电脑上的 USB 串口
- 检查 CDC 串口驱动安装是否正确，驱动包的版本是否是官方提供的最新版本
- 参考针对 win7 环境下安装 CDC 串口驱动的安装说明文档
- 如仍未解决，请加 QQ 开发交流群 445121768 联系支持人员提供具体支持



## 1.5 固件包下载

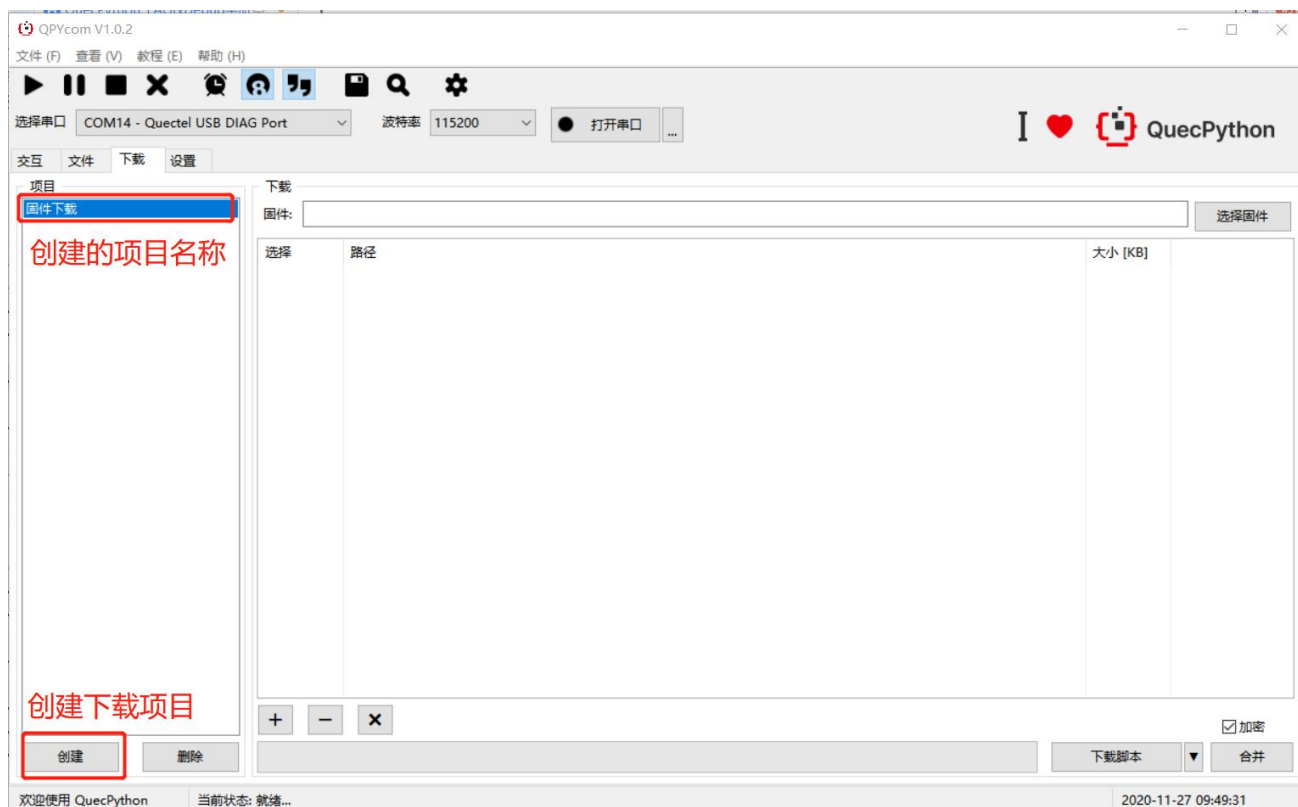
为了对客户 python 应用程序做交互适配，需要先下载 QuecPython SDK 包中提供的固件到开发板中，下载完成后才可以对 python 应用程序做底层适配处理。

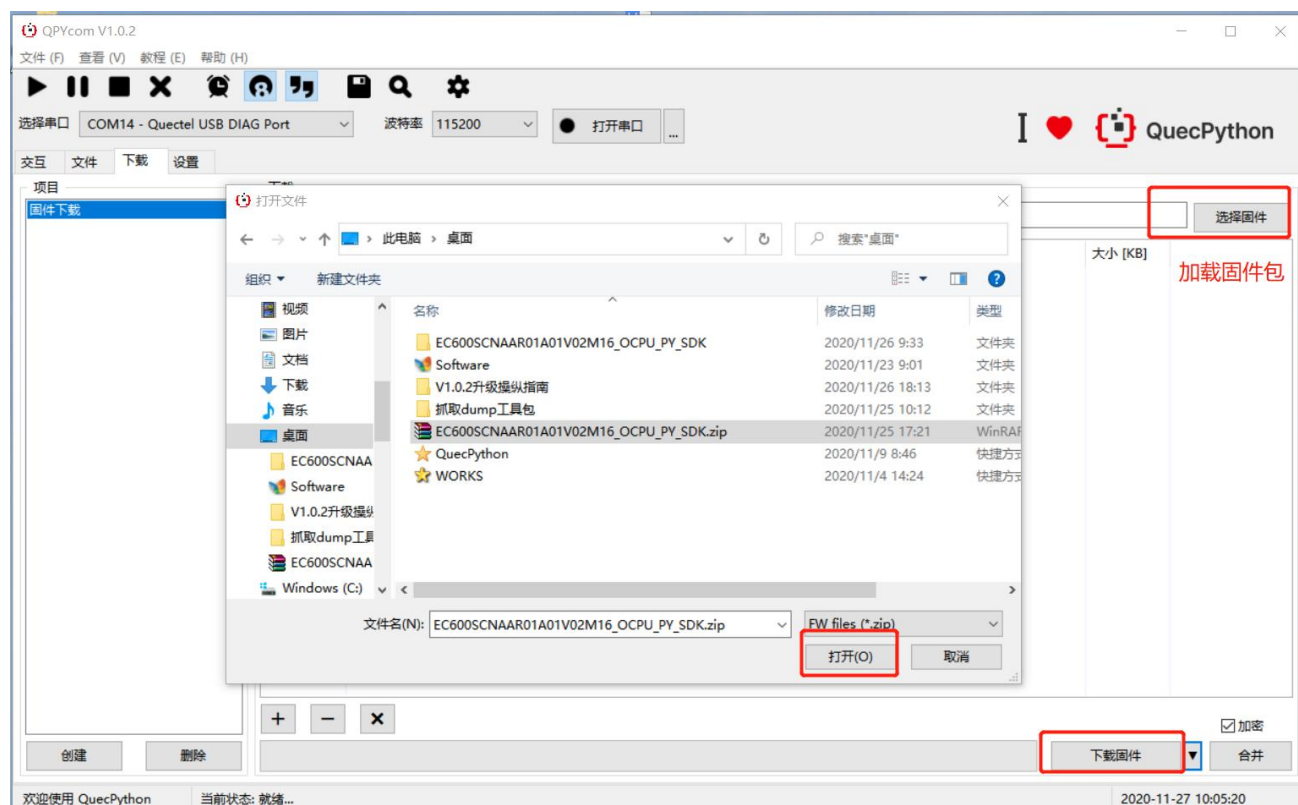
首先确认固件版本，方法是通过 QCOM 软件，发送 AT 指令 ‘AT+GMR’，如下所示：



如果是 python 版的固件（固件号以 PY 结尾），则直接跳过此步骤；如果不是 python 版的固件或者不是最新版本的固件，则需要重新下载固件，下载方式如下：

- (1) 在 SDK 包 【tools-->图形化工具 QPYcom GUI tool】 目录下解压 QPYcom\_Vx.x.x 工具包，双击运行该软件 QPYcom.exe；
- (2) 创建项目名称, 点击【选择固件】选择固件升级包, python 版的固件包在提供的 SDK 包中的 Firmware 目录；
- (3) 左键倒三角形，选择【下载固件】， 点击【下载固件】进行固件下载；





第一步、根据需求，创建用户项目（点击“创建”按钮）；

第二步、选择需要下载到模块的固件包；



第三步、点击 按钮，切换到“下载固件”模式；

第四步、点击“下载固件”按钮；

第五步、进度条到 100%，表示下载完毕。

## 1.5.1 固件包下载失败，板子成砖了怎么办

当使用 QPYcom 工具下载固件失败后，不用担心板子成砖，可以有以下两种方法下载成功。

方法 1:

- 打开电脑任务管理器，找到 QPYcom.exe，强制结束后台任务。
- 按住板子上的 reset 按键，重启板子，看设备管理器的串口列表中是否显示出正常的三个串口（AT、DIAG 以及 CDC 口），如果正常显示出串口则重新打开 QPYcom.exe 工具，参照《Quectel QuecPython\_QPYcom 工具使用说明》重新下载固件；如果上述三个串口没有正常显示出来，则参考方法 2。

方法 2:

- 强制按住开发板上的 BOOT 按钮 3-5 秒，直到设备管理器中串口列表中显示出 Doenload 下载口，说明设备已经进入下载模式。
- 打开 SDK 包 tools 目录下的 QFlash 工具进行固件刷写。QFlash 烧写使用方法参考同目录下的

《Quectel\_QFlash\_用户指导》文档。

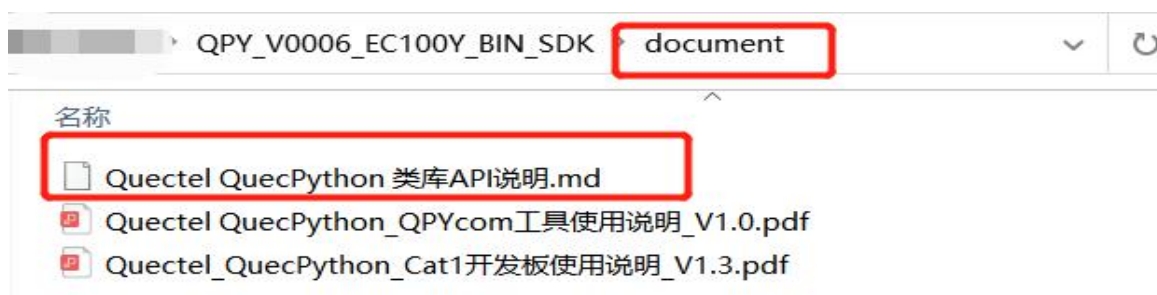
备注：从 QPYcom V1.0.2 版本开始已经解决了 QPYcom 下载失败问题。

## 2 Python 代码编写及下载调试

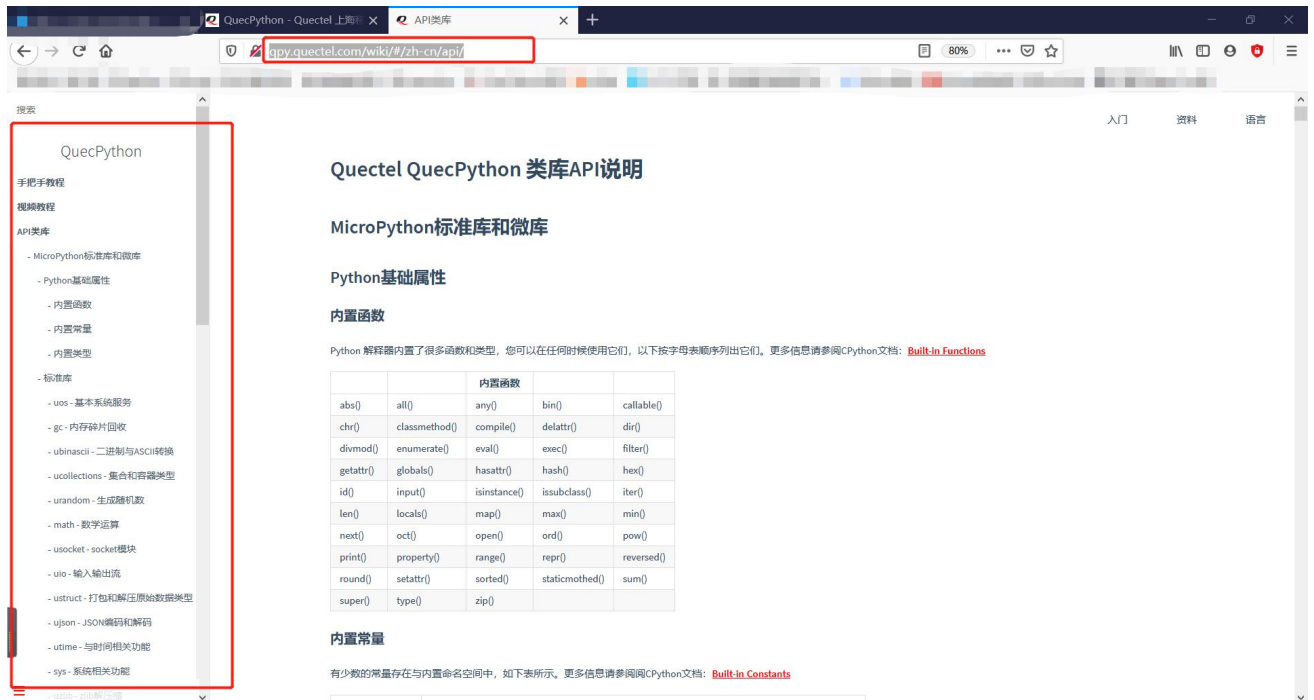
### 2.1 QuecPython API 类库文档

QuecPython API 类库是指已发布的 SDK 固件包中，提供给用户使用的模块接口。用户通过这些接口可以知道目前哪些模块功能是我们已经支持的，哪些还暂不支持。同时可以根据该 API 类库中模块的不同组合开发出你想要的产品及功能。

关于 QuecPython API 类库文档的获取方式，一种是在 SDK 包中 document 文档目录下，



另一种是可以通过官网下载：<http://qpy.quectel.com/wiki/#/zh-cn/api/>



通过 QuecPython API 类库文档我们可以看出目前主要支持的有 Micropython 标准库，QuecPython 类库及第三方库。其中 QuecPython 类库主要包含了数据拨号、基站定位、网络相关、FOTA 升级、音频播放及其他 BSP 驱动接口，第三方库主要包含阿里云、腾讯云、MQTT、HTTP 等类库。

**备注：在 QuecPython 官网上有大量的手把手教学文档和入手视频，这些资料用户可以直接下载，操作使用。**

## 2.2 编写 python 源程序代码

我们在开发产品过程中，编写源程序可以通过 Pycharm IDE 工具，也可以通过 notepad++进行源程序编写。在 SDK 包的 demo 目录下有大量的 demo 示例程序，其中大部分 demo 程序用户可以直接下载到板子上运行。

WORKS > QuecPython > QuecPython-SDK-EC100Y-V0.6 > demo

名称	修改日期	类型	大小
aliyun	2020/10/26 14:56	文件夹	
common	2020/11/6 10:24	文件夹	
dataCall	2020/10/26 14:56	文件夹	
fota	2020/10/26 14:56	文件夹	
gpio	2020/10/26 14:56	文件夹	
http	2020/11/6 10:24	文件夹	
I2C	2020/11/5 10:09	文件夹	
json	2020/10/26 14:56	文件夹	
log	2020/11/6 10:24	文件夹	
mqtt	2020/10/26 14:56	文件夹	
ntp	2020/10/26 14:56	文件夹	
pwm	2020/10/26 14:56	文件夹	
socket	2020/10/26 14:56	文件夹	
TenCentyun	2020/10/26 14:56	文件夹	
thread	2020/10/26 14:56	文件夹	
timer	2020/10/26 14:56	文件夹	
tts	2020/10/26 14:56	文件夹	
uart	2020/10/26 14:56	文件夹	
utime	2020/11/6 10:24	文件夹	
README.md	2020/10/21 17:32	MD 文件	5 KB

以第一个“hello python” 代码为例，我们打开 notepad++编辑器输入以下代码：

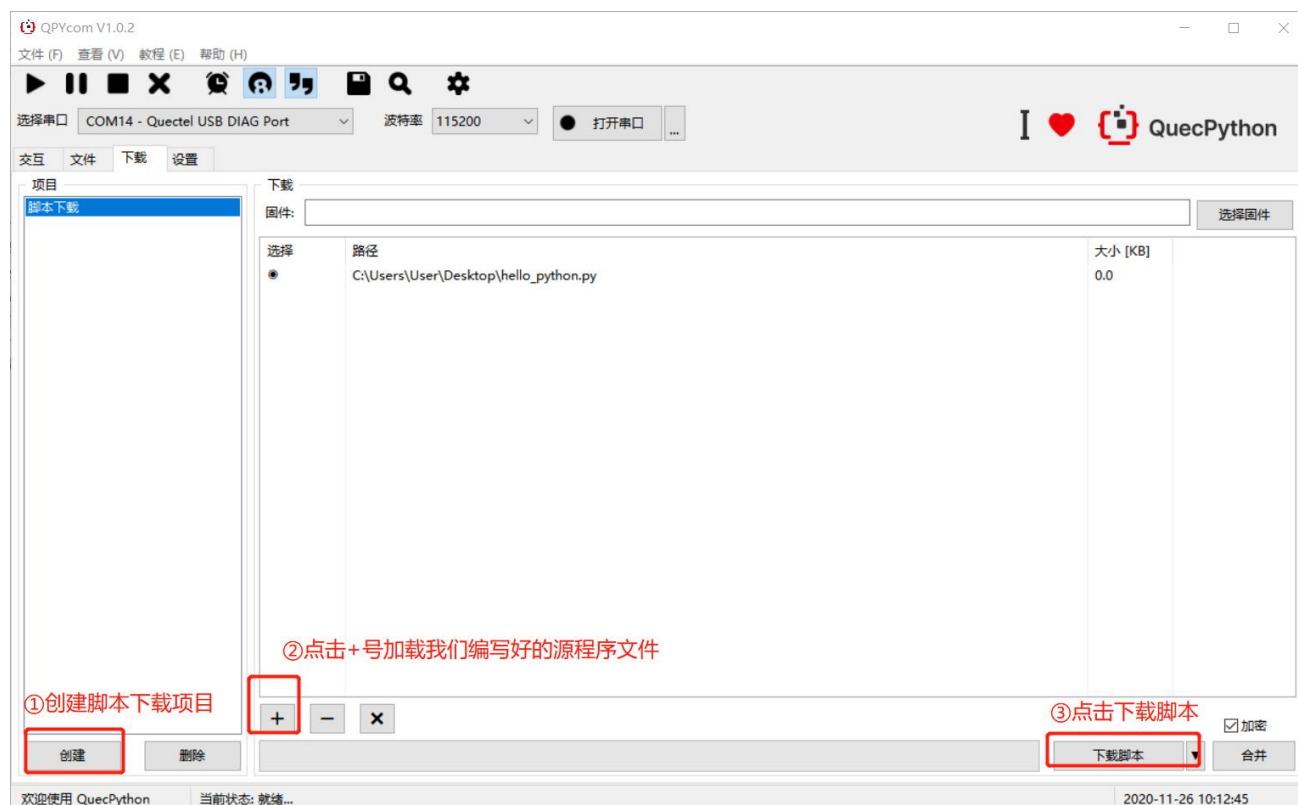
```
hello_python.py
1
2 print("hello python")
```

文件后缀名保存为 hello\_world.py,这样我们的第一个 python 源程序就编写完成了。

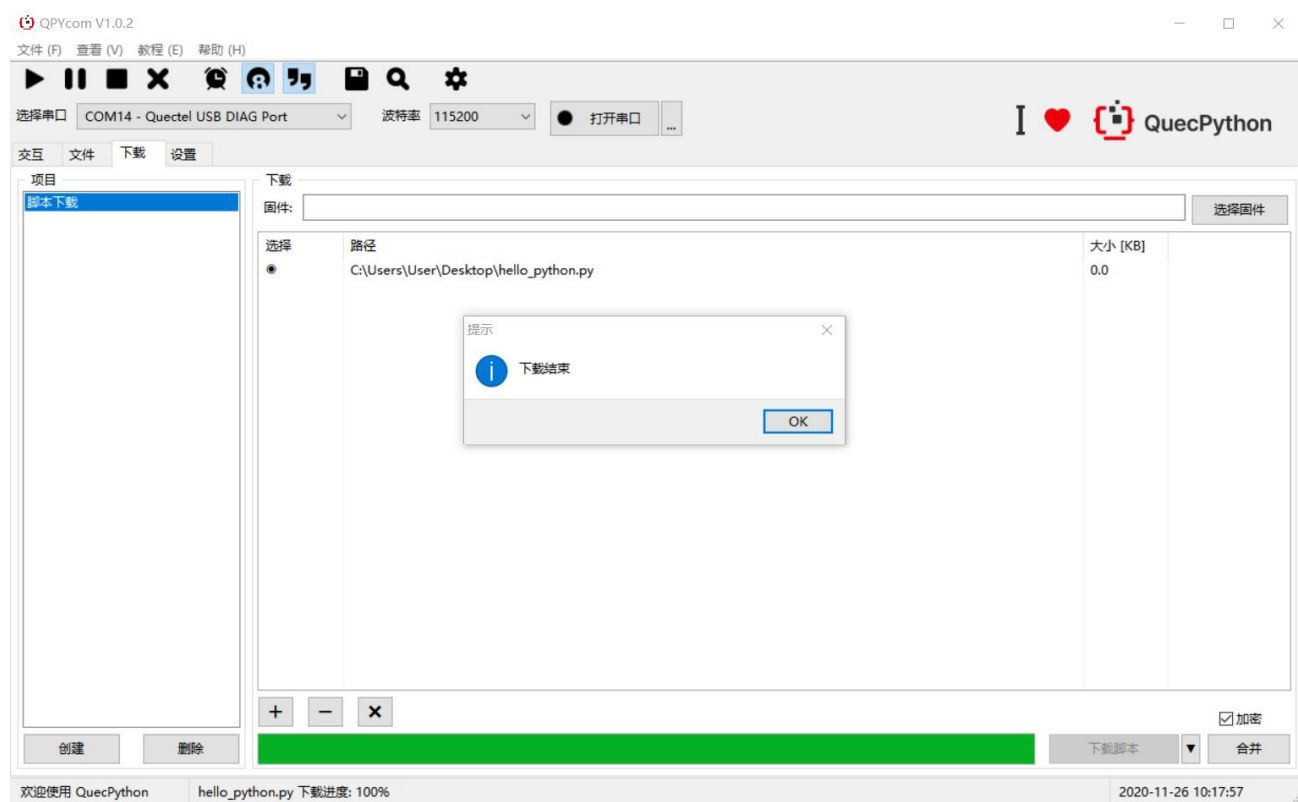
## 2.3 QPYcom 工具脚本下载调试

在 2.2 的步骤中我们编写好了第一个 python 源程序代码,接下来我们使用 QPYcom 工具下载我们的脚本到模块中运行。

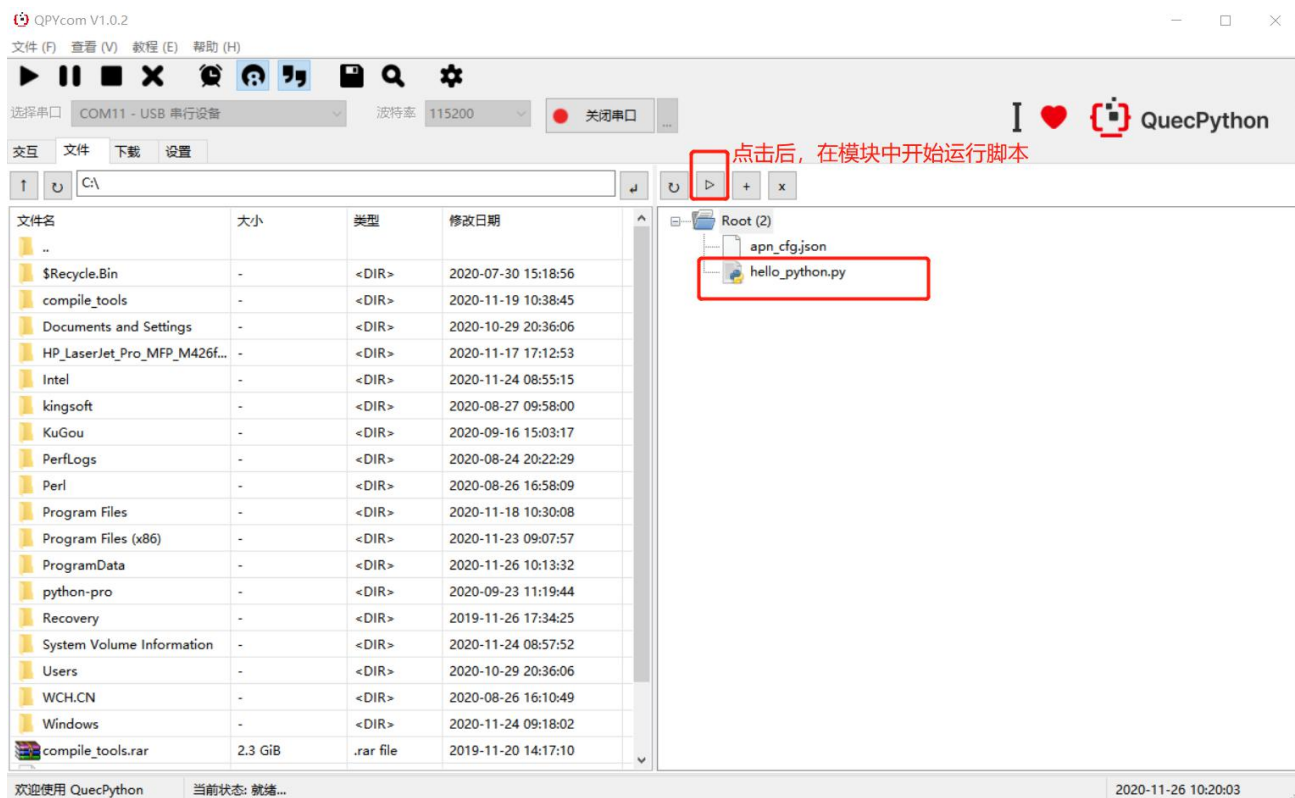
- 首先打开 QPYcom 工具，进入到下载界面：



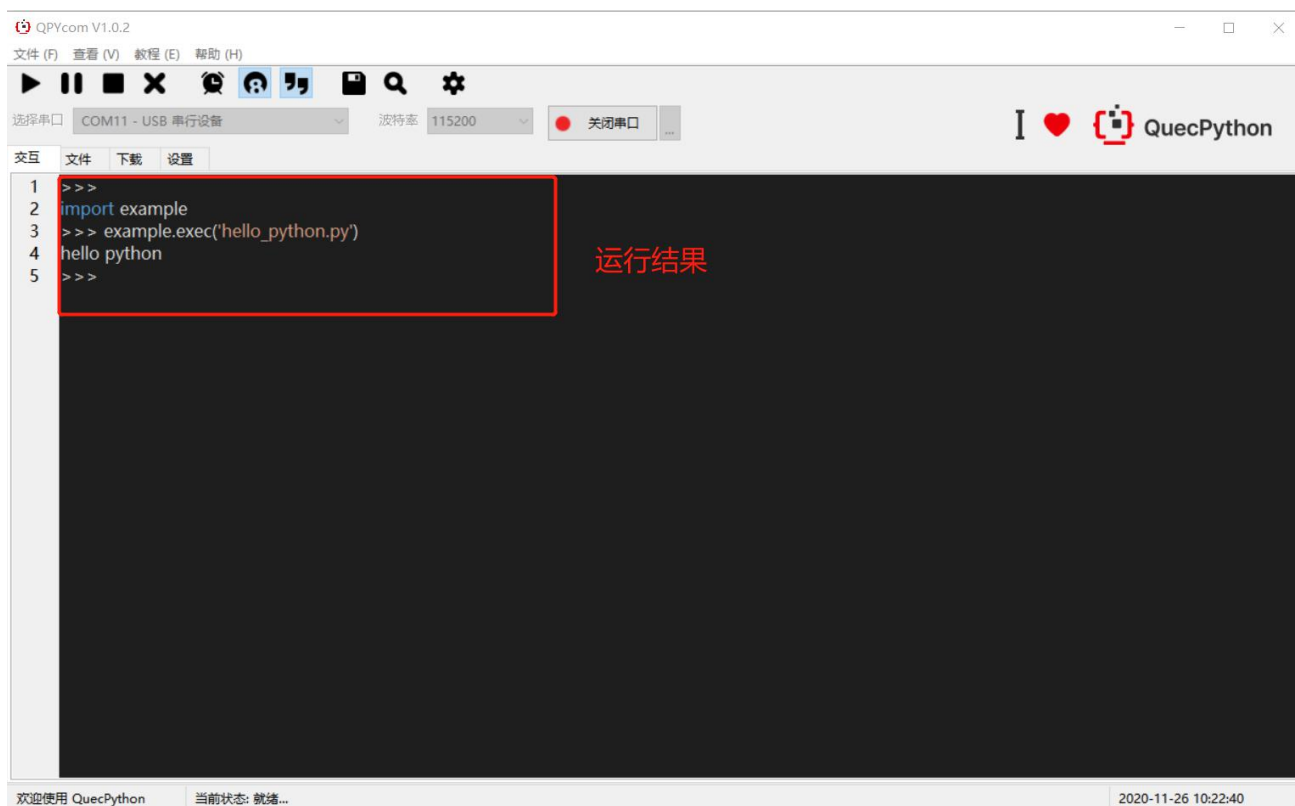
● 下载脚本结束：



● 文件列表查看我们刚才下载进去的脚本：



- 点击运行后, 运行结果如下图所示:





## 3 QuecPython 模块运行常见 Debug

### 3.1 umqtt 使用

Umqtt 模块提供创建 MQTT 客户端的发布订阅功能。该模块可以向服务端发布或者订阅消息，向服务器发送 ping 包,检测保持连通性，与服务器建立或者断开连接，需要注意的是 MQTTClient.check\_msg()和 MQTTClient.wait\_msg()方法中推荐使用 **wait\_msg** 的方法。

#### 3.1.1 使用 MQTT 连接阿里云或者腾讯云

使用 MQTT 连接阿里云或者腾讯云的步骤可以详细参见官网上的手把手教学文档。

文档地址：

<http://qpy.quectel.com/wiki/#/zh-cn/sbs/aliyun/>

<http://qpy.quectel.com/wiki/#/zh-cn/sbs/tencentyun/>

### 3.2 usocket 使用

usocket 模块提供对 BSD 套接字接口的访问，支持对 socket 的地址绑定、监听、连接、数据接收和发送等通信方法。详细可参见：QuecPython API 类库文档 usocket 模块的使用。

代码示例：

```
# 导入 usocket 模块
import usocket
import log

# 设置日志输出级别
log.basicConfig(level=log.INFO)
socket_log = log.getLogger("SOCKET")

# 创建一个 socket 实例
sock = usocket.socket(usocket.AF_INET, usocket.SOCK_STREAM)

# 解析域名
sockaddr = usocket.getaddrinfo('www.tongxinmao.com', 80)[0][-1]

# 建立连接
sock.connect(sockaddr)

# 向服务端发送消息
ret = sock.send('GET /News HTTP/1.1\r\nHost: www.tongxinmao.com\r\nAccept-Encoding: deflate\r\nConnection: keep-alive\r\n\r\n')
socket_log.info('send %d bytes' % ret)

# 接收服务端消息
```



```
data = sock.recv(256)
socket_log.info('recv %s bytes:' % len(data))
socket_log.info(data.decode())

# 关闭连接
sock.close()
```

运行结果示例：

```
1
2 import example
3 >>> example.exec('example_socket_file.py')
4
5 INFO:SOCKET:send 98 bytes
6
7 INFO:SOCKET:recv 256 bytes:
8 INFO:SOCKET:HTTP/1.1 200 OK
9
10 Server: kangle/3.5.8
11
12 Date: Fri, 27 Nov 2020 06:22:35 GMT
13
14 Set-Cookie: PHPSESSID=sqt5mjlu3h0vrhm36t7kdnira3; path=/
15
16 Expires: Thu, 19 Nov 1981 08:52:00 GMT
17
18 Cache-Control: no-store, no-cache, must-revalidate, post-check=0, pre-check=0
19
20 Pra
21 >>>
```

### 3.3 \_thread 多线程创建

QuecPython 支持多线程使用，\_thread 模块提供创建新线程的方法，主要使用：  
**thread.start\_new\_thread(function, args)**的方法创建新的线程。

代码示例：

```
import _thread

def th_func1(id):
    print("this is func1:%d" % id)

def th_func2(id):
    print("this is func2:%d" % id)

_thread.start_new_thread(th_func1, (1,))
_thread.start_new_thread(th_func2, (2,))
```

## 3.4 定时器 timer 使用

EC100YCN 支持定时器 Timer0~Timer3, timer 有 start 和 stop 两个方法, 支持定时器单次执行模式, 也支持周期执行模式。

代码示例:

```
import log
import utime
from machine import Timer

# 设置日志输出级别
log.basicConfig(level=log.INFO)
Timer_Log = log.getLogger("Timer")

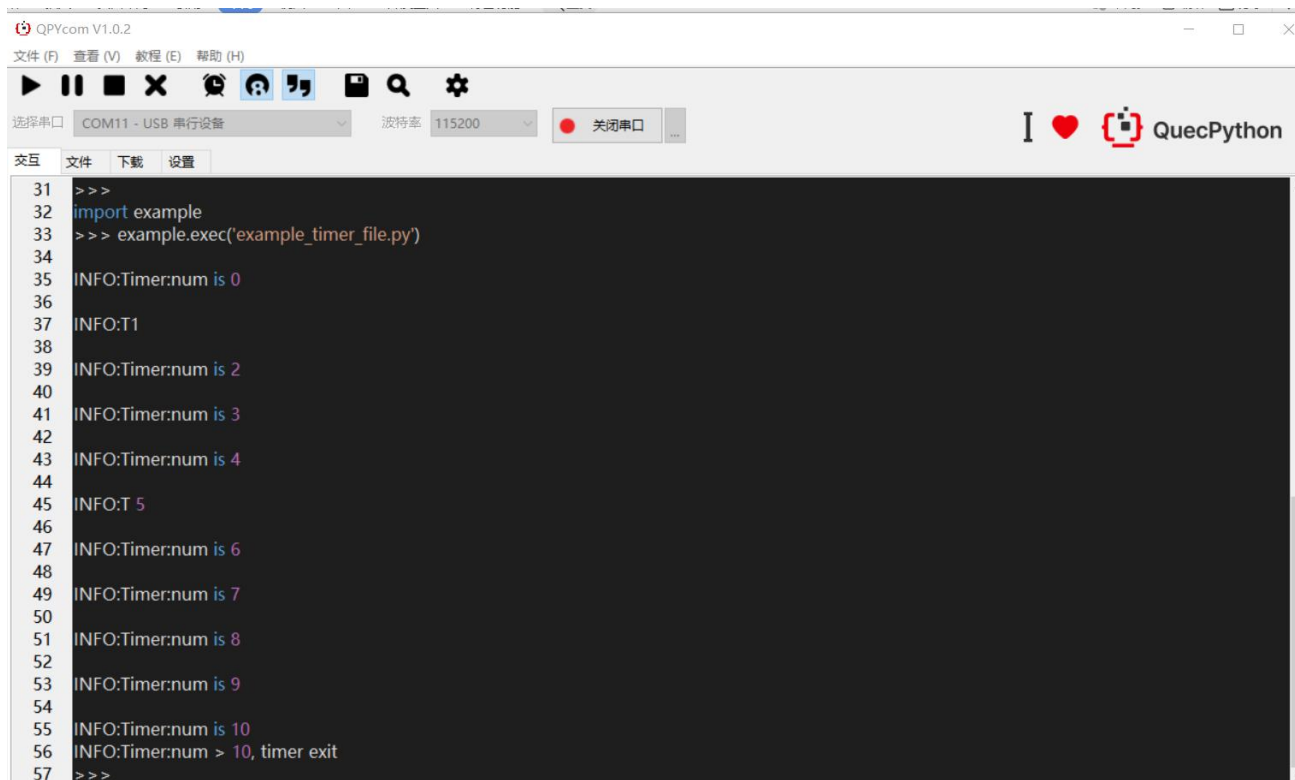
num = 0
state = 1
# 注: EC100YCN 支持定时器 Timer0~Timer3
t = Timer(Timer.Timer1)

# 创建一个执行函数, 并将 timer 实例传入
def timer_test(t):
    global num
    global state
    Timer_Log.info('num is %d' % num)
    num += 1
    if num > 10:
        Timer_Log.info('num > 10, timer exit')
        state = 0
        t.stop() # 结束该定时器实例

t.start(period=1000, mode=t.PERIODIC, callback=timer_test) # 启动定时器

while state:
    pass
```

执行结果如下:



```

31 >>>
32 import example
33 >>> example.exec('example_timer_file.py')
34
35 INFO:Timer:num is 0
36
37 INFO:T1
38
39 INFO:Timer:num is 2
40
41 INFO:Timer:num is 3
42
43 INFO:Timer:num is 4
44
45 INFO:T 5
46
47 INFO:Timer:num is 6
48
49 INFO:Timer:num is 7
50
51 INFO:Timer:num is 8
52
53 INFO:Timer:num is 9
54
55 INFO:Timer:num is 10
56 INFO:Timer:num > 10, timer exit
57 >>>
  
```

### 3.5 QuecPython 是否支持队列

队列是一种先进先出的数据结构，主要操作包括入队,出队。入队的元素加入到对尾，从队头取出出队的元素。在 QuecPython 中我们可以使用 list 操作来模拟队列：

```

class queue:
    def __init__(self):
        self.__alist = []

    def push(self, value):
        self.__alist.insert(0, value)

    def pop(self):
        return self.__alist.pop()

    def size(self):
        return len(self.__alist)

    def clean(self):
        self.__alist.clear()

    def isEmpty(self):
        return self.__alist == []
  
```

```
def showQueue(self):  
    print(self.__alist)
```

运行:

```
if __name__ == '__main__':  
    q = queue()  
    q.push(1)  
    q.push("123")  
    q.push("456")  
    q.push(2)  
    q.showQueue()  
    print(q.pop())  
    print(q.pop())  
    print(q.pop())  
    print(q.pop())  
    q.showQueue()
```

## 4 其他常见问题

### 4.1 socket 解析 IP 失败

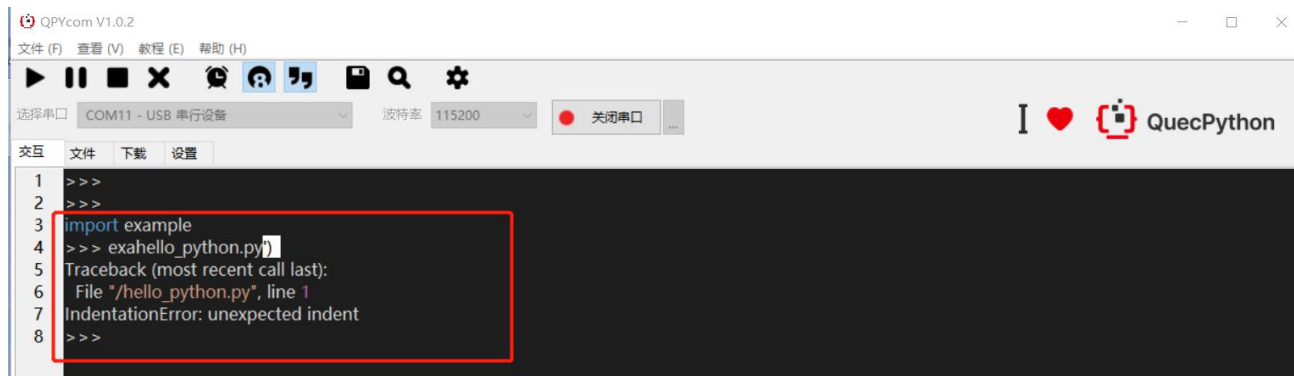
```
>>> import usocket  
>>> socket = usocket.socket(usocket.AF_INET, usocket.SOCK_STREAM)  
>>> sockaddr = usocket.getaddrinfo('www.tongxinmao.com',80)[0][-1]  
Traceback (most recent call last):  
  File "<stdin>", line 1, in <module>  
IndexError: list index out of range  
>>>
```

使用 socket、mqtt 等相关网络连接的 API 时会出现 IP 解析失败导致出现上图异常。

定位问题方向: 检查 SIM 卡是否注网成功以及检查该地址的有效性再次进行尝试。

查询驻网是否成功的 AT 指令, 【**AT+cops?** 查询是否驻网】

## 4.2 执行脚本文件提示语法错误



导致原因：大概率是因为 Python 语法缩进错误（4 个空格缩进）

排查方法：检查代码格式缩进是否规范

推荐使用：PyCharm 或 VsCode 来进行开发，IDE 会提示语法缩进与基本语法上的错误。

## 4.3 模块运行 main.py 脚本文件

现象：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）

导致原因：模块在开机后会自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，建议在测试阶段尽量不要使用 main.py 作为入口文件，可使用 start.py 或其它命名来手动拉起项目，避免重刷固件。

## 4.4 SIM 卡进行插拔后出现网络请求失败

因为目前 EC100Y 暂不支持 SIM 卡热插拔功能，所以如果我们的 SIM 卡在重新插拔后需要手动重启模块才能重新驻网。

## 4.5 MQTT 连接一段时间异常断开

导致原因：MQTT 服务端会有心跳检测机制，一段时间内设备与云端没有通信活动会主动断开连接

尝试解决方向：连接断开是依据配置 mqtt 时的超时值 keepalive，在超出活动时间后会主动断开连接，我们可以根据设置 keepalive 活动时间。使用定时器在活动时间超出前主动向云端发送 ping 包，服务端返回的数据包无需客户处理。

## MQTTClient.ping()

向服务器发送ping包,检测保持连通性

## 4.6 QuecPython 内部提供给用户的文件存储空间大小

平台	用户使用空间	适用固件包版本
EC100Y	5MB	V06 之后固件包版本
EC600S		

## 4.7 升级最新版本固件包失败了，现在上电不开机了串口也不显示

小熊派开发板可以尝试按住 Boot 键 3-5 秒进入强制下载模式，然后使用 QFlash 工具重刷固件包。

## 4.8 拿到开发板后不知道怎么使用

进入【SDK 包----->boards----->小熊派-开发板】目录参考文档《Quectel\_QuecPython\_Cat1 开发板使用说明》一文。

## 4.9 如何下载固件

进入【SDK 包----->document】目录参考文档《Quectel QuecPython\_QPYcom 工具使用说明》一文。

## 4.10 固件包从哪里获取

方式 1：如已经有 SDK 包，可直接在 SDK 包的 firmware 目录下找到；

方式 2：登录 QuecPython 网站下载：<http://qpy.quectel.com/down.html#firmware>

## 4.11 USB 串行设备未识别

win7 系统下需要更改驱动文件，具体参照文档《EC100Y\_windows7 指令交互串口 USB 驱动》

## 4.12 EC100Y 平台支持多少个串口

三个串口

串口 0（Pin21、22）：debug

串口 1（Pin27、28）：SPI\_MISO(Rx)\MOSI(Tx)

串口 2（Pin49、50）：main

## 4.13 QuecPython 支持的 API 详细介绍

QuecPython API 类库文档：<http://qpy.quectel.com/wiki/#/zh-cn/api/>

## 4.14 实战项目 demo

手把手教程链接：<http://qpy.quectel.com/wiki/#/zh-cn/sbs/>

## 4.15 QPYcom 窗口太大

鼠标移至下方拖动边框缩小窗口。

目前在 QPYcom 工具 V1.0.2 版本之后已经做了屏幕分辨率适配，工具会在打开的时候自动适配屏幕分辨率进行窗口大小调整。

## 4.16 模块是什么平台，跑什么系统

模块	内核	系统
EC100Y	ASR3601	ThreadX
EC600S		

## 4.17 QPYcom 烧录等待时间过长，无反应

尝试用 QFlash 工具烧录，必要时拉高 BOOT 引脚或者长按 BOOT 按键 3-5 秒，强制 EC100Y 进入烧录模式。

## 4.18 QuecPython 的源码文件安全吗

QPYcom 下载工具有代码混淆加密功能，确保用户程序不被直接暴露。

## 4.19 在 QPYcom 操作没有任何反应

检查选择的串口是否正确并已打开。

## 4.20 运行 main.py 程序之后，QPYcom 无法操作 EC100Y

开机自动运行 main.py，只能重新烧录固件解决，建议测试时使用其他名称命名 py 文件。

## 4.21 开发板送的流量卡可以混用吗

机卡绑定，第一次开机注网就绑定。

## 4.22 编写 Python 代码用什么工具

推荐使用 pycharm 或 VScode，两款比较流行的 IDE，自带自动补全功能。

## 4.23 板子接出的串口无法通信

要注意接口是 1.8V 还是 3.3V，电平匹配才能正常通信。

## 4.24 接上 USB 线灯不亮

检查模块电压，保证模组的 3.8V 电压稳定，必要时用电池供电。