

ECXXX-QuecPython

APTU 应用指导

LTE 系列

版本: ECXXX-QuecPython_APTU 应用指导_V1.1

日期: 2021-08-19

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

或联系我司当地办事处，详情请登录：

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

或发送邮件至：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 2019，保留一切权利。

Copyright © Quectel Wireless Solutions Co., Ltd. 2019.

文档历史

修订记录

版本	日期	作者	变更表述
1.0	2020-07-27	陈驰	初始版本
1.1	2021-08-19	蒋子杨	增加阿里云功能

目录

文档历史.....	3
目录.....	4
1 摘要.....	5
2 APTU 简介.....	6
2.1. 概念.....	6
2.2. 脚本文件.....	6
2.3. 功能.....	6
3 使用方法.....	7
3.1. 配置文件.....	7
3.1.1. 配置文件内容.....	7
3.1.2. 配置文件缺省值.....	9
3.2. 导入脚本文件.....	10
3.3. 在业务中调用 aptu 模块.....	10
4 事件上报.....	12
4.1. 事件上报的格式.....	12
4.2. 事件结果释义.....	12

1 摘要

本文旨在介绍 QuecPython 的 APTU（Application Packet Transparent Unit，应用报文透传单元）的概念及应用方法。

2 APTU 简介

2.1. 概念

APTU，全称为 Application Packet Transparent Unit，即应用报文透传单元。

APTU 以数据透传的方式广泛应用于家电、工业等需要联网的应用场景，只需在生产时将配置文件导入到文件系统即可。

2.2. 脚本文件

APTU 功能共 3 个脚本文件，分别说明如下：

- OTA.py：执行用户文件升级工作；该文件本身也可以被升级
- aptu.py：执行透传业务主逻辑；
- aptu_config.json：配置文件，主要包含 UART、网络、服务器、设备信息，及 OTA 等相关的配置信息。

2.3. 功能

- 通过配置文件配置所有功能的参数

aptu_config.json 文件配置了功能相关的所有参数。

- 开机时事件上报

开机后，在数传功能准备好或出错时，会上报且仅上报一次事件通知。

- 数据透传

APTU 以串口作为数据传输链路，来自设备端的串口数据以 TCP 协议/阿里云 MQTT 直接透传到服务器；

来自 TCP/阿里云 MQTT 服务器的数据，将直接通过串口透传到设备端。

- 脚本升级

开机并且网络连通后，会自动检测是否需要升级脚本，当有升级计划时，则会自动升级。

- 自动重连

当从服务器掉线后，模块会自动重连服务器。

设备端接收到数传功能准备好的通知后，即可开始数传业务。

3 使用方法

3.1. 配置文件

3.1.1. 配置文件内容

用户需根据自己的项目需求修改配置文件的参数。

配置文件的内容如代码清单 1 所示，为了方便理解，以下在 json 文本中添加了#打头的不应该存在的注释内容

代码清单 1 aptu_config.json 文件内容

```
{
  "UART":{ # 串口相关配置
    "No":2, # 串口号
    "baudRate":9600, # 波特率
    "startBitsLen":1, # 起始位
    "dataBitsLen":8, # 数据位
    "parity":"None", # 奇偶校验，无："None"，奇校验："Odd"，偶校验："Even"
    "stopBitsLen":1, # 停止位
    "flowCtrl":"disable", # 硬件流控，使能："enable"，禁止："disable"
    "significantBit":"least" # LSB
  },

  "Network":{ # 网络相关配置
    "timeWaitForOK":30 # 等待自动拨号完成的超时时间，单位 s
  },

  "Server":[ # 服务器相关配置
    {
      # TCP 配置
      "ipType":"IPv4", # 连接的 IP 类型，"IPv4"、"IPv6"
      "protocol":"TCP", # 连接的传输层类型，"TCP"、"UDP"
      "domain":"www.baidu.com", # 域名
      "port":80, # 端口
      "keepAlive":{ # 心跳配置
        "useThisItem":"off", # 关闭心跳，打开："on"，关闭："off"
        "parameters":{ # 心跳配置参数
          "keepIdle":60, # 心跳间隔时间，单位 s
          "keepInterval":5, # 每次心跳时，心跳包的重发时间间隔，单位 s
          "keepCount":3 # 每次心跳时，心跳包的重发次数
        }
      }
    }
  ],
}
```

```

{ # 阿里云配置
    "protocol": "ALIYUN", # 链接类型, "ALIYUN"
    "deviceInfo": { # 产品信息和设备信息配置
        "productSecret": null, # 可选参数, 产品密钥, 一机一密认证方案时为 null,
        # 一型一密时传入真实的产品密钥
        "productKey": "", # 产品标识
        "deviceName": "", # 设备名称
        "deviceSecret": "" # 可选参数, 设备密钥, 一机一密时需要传入, 一型一密时传
        # 入 null
    },
    "topic": { # 订阅和发布主题
        "publish": "/1/2/3", # 发布的 mqtt 主题
        "subscribe": "/1/2/3" # 订阅的 mqtt 主题
    },
    "parameters": { # MQTT 参数管理
        "clean_session": false, # 默认为 False, 持久客户端设置
        "keepAlive": 300, # 通信允许最长时间段, 默认为 300 (秒), 范围 60-1200
        "reConnect": true # 是否使用内部重连, 默认为 true
    },
    "qos": { # 订阅与发布 qos 设置
        "publish": 0, # 发布 qos 设置默认为 0, 范围 (0,1)
        "subscribe": 0 # 订阅 qos 设置, 默认为 0, 范围 (0,1)
    }
},
"Transaction": { # Server 选择
    "serverProtocol": "ALIYUN" # 填入需要使用 server protocol, 如果为空, 则默认选择
    # server 的第一个配置
},

"DeviceInfo": { # 移远云平台上注册的设备信息
    "moduleType": "EC600S-CX", # 模块类型, 建议"模块型号-项目名称"的命名方式
    "UID": "305", # 字符串类型, 唯一标识符
    "PK": "2bb2a48bd30b6a525a30bc64d8b3d8e0" # 密钥
},

"OTA": { # OTA 升级配置
    "autoUpgrade": true # 布尔型, 自动升级: true, 非自动升级: false
}
}

```

【注意】

- 冒号左边的 key, 严格区分大小写。

- 冒号右边的 value，除了 DeviceInfo 内部的字段以外，其余字符串类型的值不区分大小写。
- 由于 QuecPython 暂时不支持 TCP 的心跳选项配置，Server.keepAlive.useThisItem 的值请设置为 off。
- 布尔型的值，均为所有字母小写的 true 和 false。

3.1.2. 配置文件缺省值

配置文件中，"UART"的"No"和"baudRate"、"Server"的"domain"和"port"是必选项。其他非必须项都可以删除，但是代码中会对其做缺省处理，现将缺省行为整理如表 1 所示：

表 1 配置文件缺省行为

UART	No		必选	
	baudRate		必选	
	startBitsLen(实际未用)		缺省值 1	
	dataBitsLen		缺省值 8	
	parity		缺省值"None"	
	stopBitsLen		缺省值 1	
	flowCtrl		缺省值"disable"	
	significantBit(实际未用)		缺省值"least"	
Network	timeWaitForOK		缺省值 30	
Server	ipType		缺省值"IPv4"	
	protocol		缺省值"TCP"	
	domain		protocol 为"TCP"时必选	
	port		protocol 为"TCP"时必选	
	keepAlive	useThisItem		缺省值"on"
		parameters	keepIdle	protocol 为"TCP"时必选
			keepInterval	protocol 为"TCP"时必选
			keepCount	protocol 为"TCP"时必选
	deviceInfo	productSecret		一型一密时必选
		productKey		protocol 为"ALIYUN"时必选
		deviceName		protocol 为"ALIYUN"时必选
		deviceSecret		一机一密时必选
	topic	publish		protocol 为"ALIYUN"时必选
		subscribe		protocol 为"ALIYUN"时必选
	parameters	clean_session		缺省值 false
		keepAlive		缺省值 300
		reConnect		缺省值 true
	qos	publish		缺省值 0
		subscribe		缺省值 0
Transaction	serverProtocol		缺省值为 server 第一个配置	
DeviceInfo	moduleType		必选	
	UID		必选	
	PK		必选	
OTA	autoUpgrade		缺省值 true	

上表中：

- 任何字段的字体为黑色时，该字段及其下包含的内容，可从配置文件中删除，使用其缺省值。
- 当黑色字体的字段出现在配置文件中时，其紧邻的下一级的红色字体的字段是必选项。
- 第一列字体为红色的字段，必须出现在配置文件中。
- "Server"对象中不含包"keepAlive"字段时，表示关闭心跳功能。
- "DeviceInfo"和"OTA"对象同时出现，才能开启 OTA 功能。

3.2. 导入脚本文件

参考以下链接目录下的 index.html 文件的 1.3 节《将 Python 脚本文件导入 quecPython 文件系统》：

[https://gitee.com/quecpython/HeliosSDK/tree/master/docs/html/Helios_SDK_指南\(3\)_高级](https://gitee.com/quecpython/HeliosSDK/tree/master/docs/html/Helios_SDK_指南(3)_高级)

根据该方法，将本章 2.2 节提及的三个文件全部导入到模块中。

导入成功后，看到 QPYcom 工具文件传输页面的右侧栏中增加了下图红框中的三个文件。

选中 aptu.py 后，点击图 1 中的执行按钮，即可测试脚本功能。

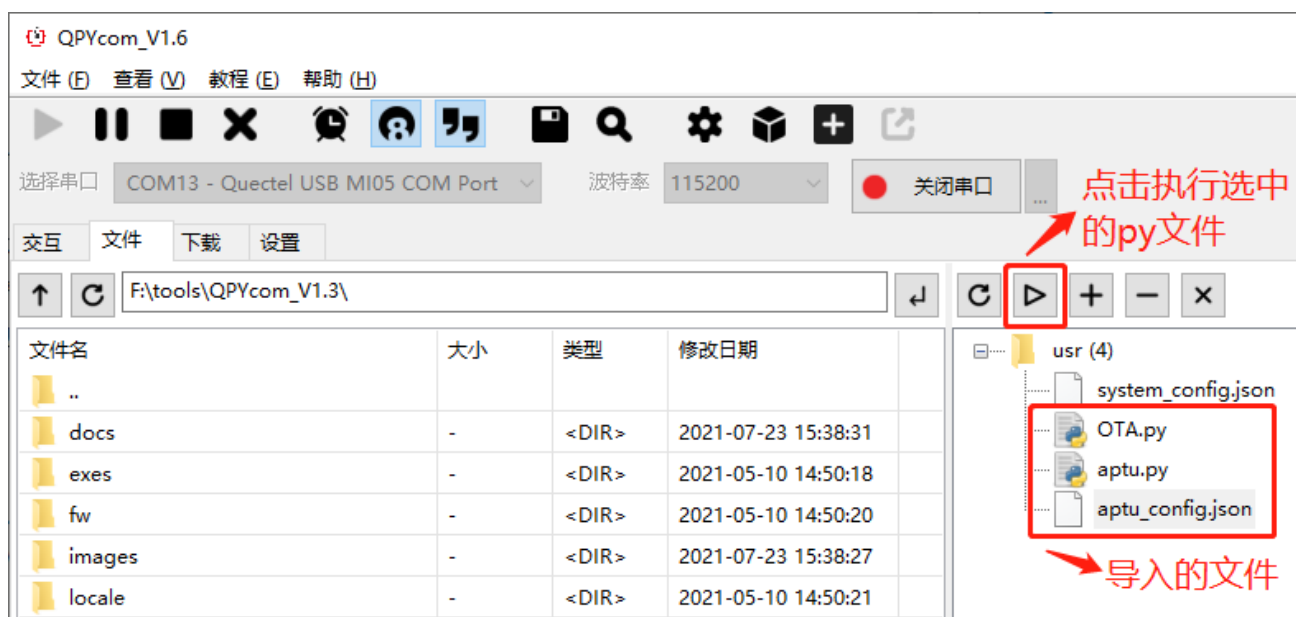


图 1 脚本文件导入方法

3.3. 在业务中调用 aptu 模块

在业务脚本中添加代码清单 2 所示的 3 行代码，即可开启功能：

代码清单 2 启动 APTU 功能

```
from usr import aptu
aptu_obj = aptu.aptu_cls()
aptu_obj.start()
```

aptu_cls 类的对象初始化函数包含两个默认参数，原型如代码清单 3 所示：

代码清单 3 aptu_cls 类的对象初始化函数

```
@classmethod
def __init__(cls, projectName = "APTU", projectVersion = "V1.0.0"):
    try:
        cls.PROJECT_NAME = projectName
        cls.PROJECT_VERSION = projectVersion
        cls.config = cls.__read_config()
        cls.__uart_init()
        cls.__data_call_check()
        cls.__ota_check()
    except Exception as e:
        cls.__exception_handler(e.args[0])
        raise
```

可以看出，缺省的参数分别是项目名称和版本号，业务上根据自己的需要进行填充，默认值分别为"APTU"和"V1.0.0"。

4 事件上报

虽然 APTU 完全工作与透传模式，但是开机后功能是否正常，是有必要通知到设备端的。
上文已经提及，在数传功能准备好或出错后，会上报且仅上报一次事件通知。

4.1. 事件上报的格式

事件上报以 json 文本的格式组织，示例如代码清单 4 所示：

代码清单 4 事件上报的 json 格式

```
{
  "result": {
    "code": 0,
    "desc": "OK"
  },
  "data": {
    "SN": "D1Q21E2130204660P",
    "IMEI": "861681053233719"
  }
}
```

上述文本的 json 对象包含两个子对象"result"和"data"。

- "result"对象包含两个字段"code"和"desc"，用于报告 APTU 是否准备好数传功能。
- "data"对象包含两个字段"SN"和"IMEI"，只有当"result"中的"code"值为 0 时，才会包含"data"对象。

以上各字段的数据类型说明如表 2 所示：

表 2 数据类型说明

字段	数据类型
code	整数
desc	字符串
SN	字符串
IMEI	字符串

4.2. 事件结果释义

"result"对象即为事件的结果通知。

事件的结果释义如表 3 所示：

表 3 事件结果释义

code	desc	mark
1	OTA plain comes	有新版待更新，模块报出此通知后，会自动重启升级至新版本

0	OK	数传准备好，可以开始传输数据
-1	config error	配置文件错误
-2	net error	网络错误
-3	socket create error	socket 创建失败
-4	socket option set error	socket 选项设置失败
-5	socket connect error	服务器连接失败
-6	DNS error	域名解析失败
-7	UART error	串口操作失败
-8	sys error	系统级别错误
-9	Aliyun create error	阿里云创建失败
-10	Aliyun option error	阿里云选项设置失败
-11	Aliyun connect error	阿里云服务器连接失败