

更改原因:

前期 AT 指令利用主 APIKEY 动态创建设备 ID 和设备 APIKEY。(模块支持保存这三个参数), 设备连接采用设备 ID 和设备 APIKEY (鉴权信息) 进行设备连接, 对应的连接方式 1, 使用的 SDK 函数为

```
/* connect1 (C->S): devid + apikey */  
EdpPacket* PacketConnect1(const char* devid, const char* auth_key)
```

针对产品量产阶段便利性和通用性的考虑, 对前期 OneNET 版本的 AT 指令需要作出调整和改动。

1 设备创建时增加鉴权信息字段"auth_info":{"sn":"xxxxxxxx"} (sn:xxxxxx 为产品生产序列号)

2 设备连接采用项目 ID 和 auth_info(鉴权信息)进行设备连接, 对应连接方式 2, 使用的 SDK 函数为

```
/* connect2 (C->S): userid + auth_info */  
EdpPacket* PacketConnect2(const char* userid, const char* auth_info)
```

3 本地保存的参数更改为主 APIKEY, 项目 ID 及设备生产序列号 SN。

具体修改见文档的红色标注部分。

2.2 错误码定义

错误码说明	错误码
WLAN 错误	-100
网络错误	-101
没有 master key	-102
没有设备序列号	-103
没有设备所属项目的项目 ID	-104
设备不在线	-105
无效的操作	-106
文件发送错误	-107
参数错误	-108
设备连接失败	-109
数据发送失败	-110
设备连接断开失败	-111
设备参数初始化失败	-112

2.3.2 AT+CIOTMKEY

说明: 查询 WIFI 模块的主 APIKEY 是否已经固化存在。

功能：查询主 APIKEY 是否存在	
查询指令格式： AT+CIOTMKEY?<CR>	正确响应： +OK=[mkey]<CR><LF> 错误响应： +ERR=-102<CR><LF>
	参数：无

2.3.5 AT+CIOTID

说明：查询设备 ID

功能：查询设备 ID 是否存在	
查询指令格式： AT+CIOTID?<CR>	正确响应： +OK=[dev_id]<CR><LF> 错误响应： +ERR=-103<CR><LF>
	参数：dev_id 为设备 ID

更改为：

2.3.5 AT+CIOTSN?

说明：查询设备生产序列号

功能：查询设备序列号是否存在	
查询指令格式： AT+CIOTSN?<CR>	正确响应： +OK=[SN]<CR><LF> 错误响应： +ERR=-103<CR><LF>
	参数： SN 为设备序列号

2.3.6 AT+CIOTIDC

说明：删除设备 ID，默认删除对应的设备 APIKEY。

功能：删除设备 ID	
指令格式： AT+CIOTIDC<CR>	正确响应： +OK<CR><LF> 错误响应： +ERR<CR><LF>
	参数：无

更改为

2.3.6 AT+CIOTSN

说明：设置设备序列号，和网络状态无关，本地保存该参数。

功能：设置设备序列号	
指令格式： AT+CIOTSN=[sn]<CR>	正确响应： +OK=[SN]<CR><LF> 错误响应： +ERR<CR><LF>
	参数：SN 为设备序列号

2.3.7 AT+CIOTKEY

说明：查询设备 APIKEY

功能：查询设备 apikey	
查询指令格式： AT+CIOTKEY?<CR>	正确响应： +OK=[apikey]<CR><LF> 正确响应： +ERR=-104<CR><LF>
	参数：apikey 为设备 APIKEY

更改为：

2.3.7 AT+CIOTPID?

说明：查询设备所属项目的项目 ID

功能：查询设备所属项目的项目 ID	
查询指令格式： AT+CIOTPID?<CR>	正确响应： +OK=[PID]<CR><LF> 错误响应： +ERR=-104<CR><LF>
	参数：PID 为设备所属项目的项目 ID

2.3.8 AT+CIOTKEYC

功能：删除设备 APIKEY	
指令格式： AT+CIOTKEYC<CR>	正确响应： +OK<CR><LF> 错误响应：

	+ERR<CR><LF>
	参数：无

更改为：

2.3.8 AT+CIOTPID

说明：设置设备所属项目的项目 ID，和网络状态无关，本地保存该参数。

功能：设置设备所属项目的项目 ID	
指令格式： AT+CIOTPID=[pid]<CR>	正确响应： +OK=[PID]<CR><LF> 错误响应： +ERR<CR><LF>
	参数：PID 为设备所属项目的项目 ID

2.3.1 AT+CIOT

说明：查询 WIFI 模块的主 APIKEY，设备序列号及项目 ID 是否已经固化存在。

功能：查询参数是否存在	
查询指令格式： AT+CIOT?<CR>	正确响应： +OK<CR><LF> 错误响应分三种情况： +ERR=-102<CR><LF> +ERR=-103<CR><LF> +ERR=-104<CR><LF>
	参数：无

内部查询的参数内容改为主 APIKEY，设备序列号及项目 ID

2.3.4 AT+CIOTINIT

说明：AT+CIOTINIT 用来进行设备 ID 和设备 Apikey 参数的创建及有效性验证。

功能：启动参数创建	
指令格式： AT+CIOTINIT<CR>	正确响应： +OK<CR><LF> 错误响应： +ERR=-100<CR><LF> +ERR=-101<CR><LF> +ERR=-102<CR><LF> +ERR=-112<CR><LF>
	参数：无

注：若指令执行返回“+ERR=-100”或者“+ERR=-101”，请确保模块连接可用 AP 的 SSID 和密码配置（配置可使用模块原有的 TCP/IP 工具箱命令或专门的 APP 进行）正确。

更改为:

2.3.4 AT+CIOTINIT

说明: AT+CIOTINIT 用来进行设备初始化处理。

功能: 设备初始化处理	
指令格式: AT+CIOTINIT<CR>	正确响应: +OK<CR><LF> 错误响应: +ERR=-100<CR><LF> +ERR=-101<CR><LF> +ERR=-102<CR><LF> ^① +ERR=-103<CR><LF> ^② +ERR=-104<CR><LF> ^③ +ERR=-112<CR><LF> ^④
	参数: 无

注①: 如果查找本地主 APIKEY 参数不存在, 返回+ERR=-102, 指令终止

注②: 如果查找本地设备序列号 SN 参数不存在, 返回+ERR=-103, 指令终止

注③: 如果查找本地设备所属项目的项目 ID 参数不存在, 返回+ERR=-104, 指令终止

注④:

+ERR=-112 该错误码表示在网络状态正常及设备参数存在的情况下, 在平台查找设备失败, 然后去平台创建设备也失败;

如果网络状态正常及设备参数存在的情况下, 在平台查找设备成功, 则返回 OK, 指令终止;

如果网络状态正常及设备参数存在的情况下, 在平台查找设备失败, 则在平台创建设备, 如创建成功, 则返回 OK, 指令终止;

使用主 APIKEY 及 auth_info 进行设备查找及设备创建示例如下。

提供查询 HTTP 例子如下:

```
GET /devices?auth_info={"sn":"123456789"} HTTP/1.1
```

```
api-key: ez8hHBOA8G4fksX4T9M=Alq8Mlo=
```

```
Host: api.heclouds.com
```

提供创建 HTTP 例子如下:

```
POST /devices HTTP/1.1
```

```
api-key: ez8hHBOA8G4fksX4T9M=Alq8Mlo=
```

```
Host:api.heclouds.com
```

```
Content-Length:193
```

```
{"title":"device123","desc":"http  
device","tags":["http"],"location":{"ele":370000,"lat":17.609997,"lon":177.03403},"private":true,  
"protocol":"EDP","auth_info":{"sn":"123456789"},"interval":60}
```

2.3.10 AT+CIOTSTART

说明：指令功能是建立设备连接，启动数据发送。

功能：建立设备连接，启动数据发送	
指令格式： AT+CIOTSTART=[encrypt]<CR>	正确响应： +OK<CR><LF> 错误响应： +ERR=-108<CR><LF> +ERR=-109<CR><LF>
	参数：encrypt ◇ 0，不使用加密 ◇ 1，使用加密（暂不支持）

更改为：

指令内部执行方式更改，由原来的设备 ID+设备 APIKEY 建立 EDP 连接更改为项目 ID+鉴权信息来建立 EDP 连接。

/* connect1 (C->S): devid + apikey */

EdpPacket* PacketConnect1(const char* devid, const char* auth_key)

/* connect2 (C->S): userid + auth_info */

EdpPacket* PacketConnect2(const char* userid, const char* auth_info)

3.3.11 AT+CIOTDAT

说明：指令功能是长传小数据（支持一次上传多个）

功能：发送数据	
指令格式： AT+CIOTDAT=[num,kay_1,time_1,value_1... kay_n,time_n,value_n]<CR>	正确响应： +OK<CR><LF> 错误响应： +ERR=-108<CR><LF> +ERR=-110<CR><LF>
	参数： Num: 数据记录数量 key: 数据采样名称 time: 数据采样时间 value: 数据采样值
说明： 时间格式：如 2016-01-01 00:00:00 年月日用'-'分隔 时分秒用':'分隔	

如没有采样时间，则时间项不填，格式为 AT+CIOTSTART=[kay,,value]<CR>

3.3.16 AT+CIOTSEND

说明：发送小数据，支持一次上传多个：

功能：发送采样数据	
指令格式： AT+CIOTDAT=[encrypt,num,kay_1,time_1,value_1...kay_n,time_n,value_n] <CR>	正确响应： +OK<CR><LF> 错误响应： +ERR=-108<CR><LF> +ERR=-109<CR><LF> +ERR=-110<CR><LF> +ERR=-111<CR><LF>
	参数： encrypt: 是否加密 num: 数据记录数 ✧ 0, 不加密 ✧ 1, 加密 key: 数据采样名称 time: 数据采样时间 value: 数据采样值
说明： 时间格式：如 2016-01-01 00:00:00 年月日用'-'分隔 时分秒用':'分隔 如没有采样时间，则时间项不填，格式为 AT+CIOTSEND=[encrypt,kay,,value]<CR>	