

1. 文档介绍

日期	更新内容	备注
2024.09.29	v3.2版本更新如下： 1.新增AT指令 1.1 AT+WIOTAREGNEWCB 1.2 AT+WIOTASENDORDER 1.3 AT+WIOTABL05 1.4 AT+WIOTARESEND 1.5 AT+WIOTAFN 1.6 AT+WIOTARSBYFN 1.7 AT+WIOTAQC 2.兼容性说明 2.1 AT+GNSSPOSQUERY新增参数type，当type为0时查询三维坐标信息，当type为1时查询经纬度 2.2 由于设置类接口增加了返回值和参数校验，如果AT输入参数错误就会返回失败	
2024.11.09	v3.3版本更新如下： 1.新增AT指令 1.1 AT+WIOTAPAGINGTXPTF 1.2 AT+WIOTABNACK	

1.1 文档范围

本手册详细介绍了UC8088 WIOTA AP模块提供的AT指令集。

1.2 命令语法

1.2.1 命令格式

本手册中所有命令行必须以“AT”或“at”作为开头，以回车作为结尾。响应通常紧随命令之后，且通常以“<回车><换行><响应内容><回车><换行>”的形式出现。在命令介绍时，“<回车><换行>”通常被省略了。

1.2.2 命令类型

通常命令可以有如下表所示的四种类型中的一种或多种形式。

类型	格式	说明
查询命令	AT+<cmd>?	用于返回参数的当前值
设置命令	AT+<cmd>=<...>	用于设置用户自定义的参数值
执行命令	AT+<cmd>	用于读取只读参数或不需要额外参数的情况

1.2.3 参数类型

命令参数虽然多种多样，但是都可以简单地归结为整数类型和字符串类型（包括不带双引号的字符串和带双引号的字符串）这两种基本的类型，如下表所示。

类型	示例
整数类型	123
字符串类型	abc
	"hellow ,world"

1.2.4 注意事项

- AT串口输入时不支持回删键(backspace)功能。
- AT指令不区分大小写。

2. 基础 AT指令详细说明

2.1 AT

AT测试指令。

Command	Possible response(s)
AT	OK
	ERROR

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT	OK	AT指令功能可正常使用
	ERROR	AT指令功能无法正常使用

2.2 AT+RST 重启

系统重启，会同时重启ap8288和ap8088。

Command	Possible response(s)
+RST	OK
	ERROR

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+RST	+Select modem,enter follow char: a. Only ymodem down file b. OTA update c. Flash rtthread d. Flash all e. Ymodem down bin, flash rtthread f. Ymodem down bin, flash uboot and rtthread g. Ymodem down ota page, ota update +SYSTEM:START	指令执行成功
	ERROR	指令执行失败

2.3 ATE 回显

AT指令回显功能。

Command	Possible response(s)
ATE<value>	OK
	ERROR

- <value>：默认AT回显关闭。
0：关闭回显。
1：打开回显。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
ATE0	OK	关闭回显成功
	ERROR	指令执行失败
ATE1	OK	开启回显成功
	ERROR	指令执行失败

2.4 AT&L 查询AT指令列表

查询所有支持的AT指令列表。

Command	Possible response(s)
AT&L	OK
	ERROR

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT&L	Commands list : AT&L AT+RST OK	输出所有的AT指令
	ERROR	指令执行失败

2.5 AT+UART AT串口配置

AT默认使用UART0，故该命令可配置UART0的属性，一般不建议更改。

Command	Possible response(s)
+UART=<baud_rate>,<data_bits>,<stop_bits>,<parity>,<flow_control>	OK
	ERROR
+UART?	+UART=<baud_rate>,<data_bits>,<stop_bits>,<parity>,<flow_control>
	ERROR

- <baudrate>：波特率，默认UART0的波特率配置为115200，常见波特率最大支持460800，经测试还支持部分不常见波特率：1024000、2048000、4096000。
- <databits>：有效数据长度。
- <stopbits>：停止位。
- <parity>：奇偶检验。
- <flow_control>：流控，默认值为1，不支持配置。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+UART=115200,8,0,0,1	OK	设置UART0属性成功
	ERROR	指令执行失败
AT+UART?	+UART=115200,8,0,0,1 OK	输出UART0的配置信息
	ERROR	指令执行失败

2.6 AT+REFLASH 基站侧基带恢复

恢复成UC8088中保存的镜像，重刷一次UC8288（基带）的FLASH。通过烧写器烧写UC8288（基带）后，需要执行一次该指令，确保UC8088与UC8288版本一致。

Command	Possible response(s)
+REFLASH	reflash 8288 ready reboot +Select modem,enter follow char: a. Only ymodem down file b. OTA update c. Flash rtthread d. Flash all e. Ymodem down bin, flash rtthread f. Ymodem down bin, flash uboot and rtthread g. Ymodem down ota page, ota update +SYSTEM:START
	reflash 8288 ERROR

3. WITOA AT命令详细说明

3.1 AT+WIOTAINIT 初始化

初始化WIoTa协议栈。

Command	Possible response(s)
+WIOTAINIT	OK
	ERROR

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAINIT	OK	WIoTa协议栈初始化成功
	ERROR	指令执行失败

3.2 AT+WIOAFREQ 设置WIoTa工作频点

设置频点，终端和AP需要设置相同频点才能同步，v2.3版本后该参数支持热配置。

Command	Possible response(s)
+WIOTAFREQ=<freq_idx>	OK
	ERROR
+WIOTAFREQ?	+WIOTAFREQ:<freq_idx>
	ERROR

- <freq_idx>: 频点idx，范围0~200

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAFREQ=145	OK	设置频点为145
	ERROR	指令执行失败
AT+WIOTAFREQ?	+WIOTAFREQ:145	查询到配置的频点为145
	ERROR	指令执行失败

3.3 AT+WIOTAACTIVETIME 设置连接态保持时间

设置AP连接态保持时间，必须和终端保持一致，否则会导致无法正常进行上下行业务。

Command	Possible response(s)
+WIOTAACTIVETIME=<active_time>	OK
	ERROR
+WIOTAACTIVETIME ?	+WIOTAACTIVETIME:<active_time>
	ERROR

- <active_time>: 该值不可设置太大，否则可能导致其他终端无法接入，范围0~0xffffffff，单位：秒
该值有默认值在128配置下为2秒，256配置下为3秒。512配置下为4秒，1024配置下为8秒。非特殊需求不建议更改连接态保持时间。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAACTIVETIME=5	OK	设置连接态保持时间为5秒成功
	ERROR	指令执行失败
AT+WIOTAACTIVETIME?	+WIOTAACTIVETIME:5	查询到配置的连接态保持时间为5秒
	ERROR	指令执行失败

3.4 AT+WIOTACONFIG 系统配置

设置系统配置，注意v2.4版本后调整了参数位置，并新增了两个参数设置。

Command	Possible response(s)
+WIOTACONFIG=<ap_max_pow>,<id_len>,<symbol_len>,<dlul_ratio>,<bt_value>,<group_num>,<specturm_idx>,<old_subsys_v>,<bitscb>,<system_id (v2.5版本之后无该参数)>,<subsystem_id>	OK
	ERROR
+WIOTACONFIG?	+WIOTACONFIG:<ap_max_pow>,<id_len>,<symbol_len>,<dlul_ratio>,<bt_value>,<group_num>,<specturm_idx>,<old_subsys_v>,<bitscb>,<system_id (v2.5版本之后无该参数)>,<subsystem_id>
	ERROR

- <id_len>: user id长度, 取值0,1,2,3代表2,4,6,8字节。默认四字节, IOTE该变量需要与AP保持一致, 现在只支持设置为1, 即四字节。
- <symbol_len>: 帧配置, 取值0,1,2,3代表128,256,512,1024。
- <dlul_ratio>: 帧配置, 该值代表一帧里面上下行的比例, 取值0,1代表1:1和1:2。
- <bt_value>: 该值和调制信号的滤波器带宽对应, BT越大, 信号带宽越大, 取值0,1代表BT配置为1.2和BT配置为0.3, bt_value为0时, 代表使用的是低阶mcs组, 即低码率传输组。bt_value为1时, 代表使用的是高mcs组, 即高码率传输组。
- <group_num>: 帧配置, 取值0,1,2,3代表一帧里包含1,2,4,8个上行group数量。
- <ap_max_pow>: ap射频最大功率, 默认22dbm, 由于该值可能为负数, 但at暂不支持负数解析, tag0.09版本之后, 实际需要设置的功率加20则为输入值, 例如想要设置功率-10dbm, 则 at+wiotapow=0,10, 想要设置功率20dbm, 则 at+wiotapow=0,40。
- <specturm_idx>: 频谱序列号, 默认为3, 即470-510M。
- <old_subsys_v>: 匹配老版本iote (v2.3及之前的版本) 标志位, 默认值为0, 表示不匹配老版本, 如果需要匹配老版本, 将该值设为1。
- <bitscb>: 比特加扰标志位, 默认值为1, 表示开启比特加扰, 为0表示关闭比特加扰。
- <system_id>: 系统id, 每个id是0-0xFFFFFFFF, 16进制格式输入, 不需要0x。此值为预留值, 必须设置, 但是不起作用。v2.5版本之后无该参数
- <subsystem_id>: 子系统id, 每个id是0-0xFFFFFFFF, 16进制格式输入, 不需要0x。(子系统的识别码, 终端IOTE如果要连接该子系统 (AP), 需要将config配置里的子系统ID参数配置成该ID), v2.3版本后该参数支持热配置

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTACONFIG=47,1,1,0,1,0,3,0,1,21456981	OK	配置系统信息成功
	ERROR	指令执行失败
AT+WIOTACONFIG?	+WIOTACONFIG:47,1,1,0,1,0,3,0,1,21456981	输出当前系统配置信息
	ERROR	指令执行失败

3.5 AT+WIOTARUN 启动/关闭WIoTa协议栈

启动WIoTa系统。
启动WIoTa后，收到数据会主动上报串口，数据最长为310字节。
格式是：+WIOTARECV:<user_id>,<type>,<data_len>:<data>。

Command	Possible response(s)
+WIOTARUN=<state>	+WIOTARUN:OK OK
	ERROR

- <state>:
0：退出协议栈。
1：启动协议栈。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTARUN=1	+WIOTARUN:OK OKOK	WIoTa协议栈启动成功
	ERROR	指令执行失败
AT+WIOTARUN=0	OK	WIoTa协议栈关闭成功
	ERROR	指令执行失败

3.6 +WIOTARECV WIoTa数据上报

注册接收数据回调函数后，会将接受到的上行数据主动上报（不需要单独的AT命令设置，系统直接上报接收到的数据到串口），v1.3版本后当type为0时会上报接入提示，然后上报数据。

Command	Possible response(s)
无	+WIOTARECV:<user_id>,<type>,<data_len>: <recv_data>

- <user_id>：上报数据的终端id。
- <type>：
 - 0：接入短消息。
 - 1：连接态短消息。
 - 2：子帧模式数据。
- <data_len>：上报数据的长度。
- <recv_data>：上报数据的内容。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
无	+WIOTARECV:0x12345678,0,13: Hello WIoTa	接收到id为0x12345678的终端上报的13个字节的接入短消息“Hello WIoTa”

3.7 AT+WIOTABLACKLIST 添加或移除黑名单

添加一个user_id到黑名单或从黑名单中移除一个user_id。

Command	Possible response(s)
+WIOTABLACKLIST=<user_id>,<mode>	OK
	ERROR
+WIOTABLACKLIST?	+WIOTABLACKLIST:<blacklist_num> +WIOTABLACKLIST:<user_id> ... OK
	ERROR

- <userid>: 要添加或移除的user_id (0~0xffffffff)
- <mode>:
 - 0：添加黑名单。
 - 1：移除黑名单。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTABLACKLIST=12345678,0	OK	添加id为0x12345678的终端到黑名单成功
	ERROR	指令执行失败
AT+WIOTABLACKLIST=12345678,1	OK	从黑名单中移除id为0x12345678的终端成功
	ERROR	指令执行失败
AT+WIOTABLACKLIST?	+WIOTABLACKLIST:1 +WIOTABLACKLIST:0x12345678 OK	黑名单中共有1个成员，id为0x12345678
	ERROR	指令执行失败

3.8 AT+WIOTAIOTEINFO 查询iote信息（该AT指令在v2.4及之后的版本将不再支持）

查询当前时刻下在线的iote信息和离线的iote信息，iote_status为0时表示离线，为1时表示在线，con_num表示在线的iote数量，discon_num表示离线的iote数量。

Command	Possible response(s)
+WIOTAIOTEINFO?	+WIOTAIOTEINFO:<con_num>,<discon_num> +WIOTAIOTEINFO:<user_id>,<iote_status>,<group_idx>/<subframe_idx> ... OK
	ERROR

- <userid>: 终端id。
- <iote_status>: 终端状态。
 - 0: 离线状态。
 - 1: 在线状态。
- <con_num>: 在线的终端个数。
- <discon_num>: 离线的终端个数。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAIOTEINFO?	+WIOTAIOTEINFO:1,0 +WIOTAIOTEINFO:0x12345678,1,0/7 OK	目前共有1个在线终端，0个离线终端，id为0x12345678的终端在线，位置为0/7
	ERROR	指令执行失败

3.9 AT+WIOTABC 发送广播数据

发送广播数据，执行该命令后，需在10秒内在串口工具的发送区输入长度为len的数据，并点击发送，不然会获取字符失败。

Command	Possible response(s)
+WIOTABC=<data_id>,<len>,<mode>,<timeout>,<is_block>	OK +WIOTABC:<result>,<data_id>
	ERROR

- <data_id>:非阻塞发送数据时使用，一般为数据地址，相当于某段数据的标记，当发送结束后，该参数会在发送结束后返回，可以让应用层感知哪段数据发送成功了或失败了，如果不需要可填任意值，阻塞发送也可填数据地址或任意值。
- <len>: 要发送的广播数据长度，**若使用的串口工具打开了“加回车换行”，输入数据的长度应扣除回车换行字符的长度（2个字节），若没有打开则不需要扣除**，范围0~1024byte。
- <mode>:
0: 普通广播数据。
1: OTA广播数据。
- <timeout>:发送数据超时时间，范围0~0xffffffff。
- <is_block>:阻塞发送标识，0: 非阻塞发送，1: 阻塞发送。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTABC=123456,11,0,10000,0\r\nAP ready!	OK +WIOTABC:0,123456	非阻塞发送普通广播数据 “AP ready!”成功
	ERROR	指令执行失败
AT+WIOTABC=123456,11,1,10000,1\r\nOTA test data	+WIOTABC:0,123456 OK	阻塞发送OTA广播数据 “OTA test data”成功
	ERROR	指令执行失败

3.10 AT+WIOTAMCID 设置或删除组播ID

用于设置或删除组播ID，发送组播前必须先设置组播ID，否则无法发送组播。可多次设置，但最多只能设8个不同的组播ID。

Command	Possible response(s)
+WIOTAMCID=<type>,<id0>,<id1>,<id2>,<id3>	OK
	ERROR

- <type>:操作类型，0表示添加组播ID，1表示删除主播ID。
- <id0~id3>: 要设置的组播ID，十六进制不带0x，为0时表示无效id，且不能在中间留0。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAMCID=0,ce98c377,8e0f6cd9,8aff8783,0	OK	添加组播ID： ce98c377,8e0f6cd9,8aff8783 成功
	ERROR	指令执行失败

3.11 AT+WIOTAMC 发送组播数据

AP发送组播数据，执行该命令后，需在10秒内在串口工具的发送区输入长度为len的数据，并点击发送，不然会获取字符失败。

Command	Possible response(s)
+WIOTAMC=<data_id><len>,<mc_id>,<timeout>,<is_block>	OK
	ERROR

- <data_id>:非阻塞发送数据时使用，一般为数据地址，相当于某段数据的标记，当发送结束后，该参数会在发送结束后返回，可以让应用层感知哪段数据发送成功了或失败了，如果不需要可填任意值，阻塞发送也可填数据地址或任意值。
- <len>: 要发送的组播数据长度，**若使用的串口工具打开了“加回车换行”，输入数据的长度应扣除回车换行字符的长度（2个字节），若没有打开则不需要扣除**，范围0~1024byte。
- <mc_id>: 组播id，范围0~0xffffffff。
- <timeout>: 发送数据超时时间，范围0~0xffffffff。
- <is_block>:阻塞发送标识，0：非阻塞发送，1：阻塞发送。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAMC=123456,14,12345678,10000,1\r\nHello WloTa!	+WIOTAMC:0,0x12345678,0x123456 OK	阻塞发送数据“Hello WloTa!”到组播id为0x12345678的一组终端成功
	ERROR	指令执行失败

3.12 AT+WIOTASEND 发送单播数据

AP发送下行单播数据，执行该命令后，需在10秒内在串口工具的发送区输入长度为len的数据，并点击发送，否则会获取字符失败。

Command	Possible response(s)
+WIOTASEND=<data_id>,<len>,<user_id>,<timeout>,<is_block>	OK
	ERROR

- <data_id>:非阻塞发送数据时使用，一般为数据地址，相当于某段数据的标记，当发送结束后，该参数会在发送结束后返回，可以让应用层感知哪段数据发送成功了或失败了，如果不需要可填任意值，阻塞发送也可填数据地址或任意值。
- <len>: 要发送的单播数据长度，**若使用的串口工具打开了“加回车换行”，输入数据的长度应扣除回车换行字符的长度（2个字节），若没有打开则不需要扣除**，范围0~310byte。
- <user_id>: 终端id，范围0~0xffffffff。
- <timeout>: 发送数据超时时间，范围0~0xffffffff。
- <is_block>:阻塞发送标识，0：非阻塞发送，1：阻塞发送。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTASEND=123456,14,12345678,10000,0\r\nHello WIoTa!	OK +WIOTASEND:0,0x12345678,0x123456	非阻塞发送数据“Hello WIoTa!”到id为0x12345678的终端成功
	ERROR	指令执行失败

3.13 AT+WIOTARESEDN 设置单播重发次数

v3.2新增，设置单播重发次数，指单个协议包重发次数。

Command	Possible response(s)
+WIOTARESEDN=<times>	OK
	ERROR
+WIOTARESEDN?	+WIOTARESEDN:<times> OK
	ERROR

- <times>：范围0~3，表示不重发、重发1~3次。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTARESEDN=0	OK	不重发
	ERROR	指令执行失败
AT+WIOTARESEDN=1	OK	重发一次
	ERROR	指令执行失败
AT+WIOTARESEDN?	+WIOTARESEDN:1 OK	查询到重发次数为1
	ERROR	指令执行失败

3.14 AT+WIOTASENDORDER 发送顺序业务数据（v3.2版本新增）

AP发送顺序业务数据，执行该命令后，需在10秒内在串口工具的发送区输入长度为len的数据，并点击发送，否则会获取字符失败。

Command	Possible response(s)
+WIOTASENDORDER=<data_id>,<len>,<user_id>,<timeout>,<order_business>,<is_block>	OK
	ERROR

- <data_id>:非阻塞发送数据时使用，一般为数据地址，相当于某段数据的标记，当发送结束后，该参数会在发送结束后返回，可以让应用层感知哪段数据发送成功了或失败了，如果不需要可填任意值，阻塞发送也可填数据地址或任意值。

- <len>: 要发送的单播数据长度，若使用的串口工具打开了“加回车换行”，输入数据的长度应扣除回车换行字
符的长度（2个字节），若没有打开则不需要扣除，范围0~310byte。
- <user_id>: 终端id，范围0~0xffffffff。
- <timeout>: 发送数据超时时间，范围0~0xffffffff。
- <order_business>: 顺序业务标识，为0时和AT+WIOTASEND一样，为1时表示顺序业务，指同一子帧的多个
终端顺序执行。
- <is_block>:阻塞发送标识，0：非阻塞发送，1：阻塞发送。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTASENDORDER=123456,14,12345678,10000,1,0\r\nHello WloTa!	OK +WIOTASEND:0,0x12345678,0x123456	非阻塞发送数据“Hello WloTa!”到id为0x12345678的终端成功
	ERROR	指令执行失败

3.15 AT+WIOTASCANFREQ 扫频

AP端扫频命令，可扫一组频点和全扫，返回扫频结果，执行该命令后需要在窗口工具的发送区输入长度为dataLen（dataLen只能大于或等于输入的字符串长度，不能小于否则会获取字符串失败），个数为freqNum的字符串，并点击发送。

Command	Possible response(s)
+WIOTASCANFREQ=<timeout>,<data_len>,<freq_num>,<scan_type>,<is_gwmode>	+WIOTASCANFREQ:<freq_idx>,<rssi>,<snr>,<is_synced>
	...
	OK
	ERROR

- <timeout>: 扫频的总超时时间，必须要设置，范围0~0xffffffff。
- <data_len>: 发送字符串的总长度+
1) 执行at命令AT+WIOTASCANFREQ=10000,11,5
2) 当出现>时十秒钟内在串口工具的发送区内输入字符串1,2,3,4,5
3) 点击发送。
4) 等待扫频结果返回，结果会通过串口打印出来。
- <freq_num>: 频点个数，范围0~200，该参数为0时为全扫。
- <scan_type>: 扫频类型，0是正常扫频，1是快速扫频（只扫rssi）
- <is_gwmode>: 是否为网关模式扫频，0为正常模式，1为网关模式，区别在于输出格式不同，其他完全一样。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIoTASCFREQ=60000,70,20,0,0 \\n5,15,25,35,45,55,65,75,85,95,105,115,125,135,145,155,165,175,185,195	+WIoTASCFREQ:5,-58,0,1 ... OK	扫描20个频点成功，并输出个频点的rssi、snr和is_synced信息
	ERROR	指令执行失败
AT+WIoTASCFREQ=180000,0,0,0	+WIoTASCFREQ:0,-68,12,0 ... OK	全扫，扫描201个频点成功，并输出个频点的rssi、snr和is_synced信息
	ERROR	指令执行失败

3.16 AT+WIoTATEMP 读取ap8288芯片实时温度

读取ap8288芯片内部实时温度，无任务是两帧内完成结果上报，有任务会直接返回读取失败。

Command	Possible response(s)
+WIoTATEMP?	+WIoTATEMP:<temp> OK
	ERROR

- <temp>: 读取到的UC8288温度，单位：摄氏度。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIoTATEMP?	+WIoTATEMP:47 OK	读取到基带芯片温度为47摄氏度
	ERROR	指令执行失败

3.17 AT+WIoTAPOW 设置ap8288射频功率

设置ap8288射频功率，设置范围为：0 - 29dbm，v2.3版本后该参数支持热配置。

Command	Possible response(s)
+WIoTAPOW=<power>	OK
	ERROR
+WIoTAPOW?	+WIoTAPOW=<power> OK
	ERROR

- <power>: 最大功率（0-29）dbm，v0.09版本之后，实际需要设置的功率加20则为输入值，例如想要设置功率-10dbm，则 at+wiotapow=10，想要设置功率20dbm，则 at+wiotapow=40。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOAPOWER=47	OK	设置射频功率为27（47-20）dbm成功
	ERROR	指令执行失败或设置射频功率失败
AT+WIOAPOWER?	+WIOAPOWER:27 OK	查询到射频功率为27dbm
	ERROR	指令执行失败

3.18 AT+WIOAVERSION 查询当前软件版本

查询当前WIoTa库的版本号、git 信息、编译生成库的时间，如果未开启协议栈只能查到ap8088的信息，ap8288的信息需要启动WIoTa协议栈（[3.5 AT+WIOARUN 启动WIoTa协议栈](#)）后方可查询到。

Command	Possible response(s)
+WIOAVERSION?	+WIOAVERSION:<version_8088>,<version_8288> +GITINFO:<gitInfo_8088>,<gitInfo_8288> +TIME:<make_time_8288>,<make_time_8288> +CCEVERSION:<cce_version> OK
	ERROR

- version_8088：当前UC8088的版本号。
- version_8288：当前UC8288的版本号。
- gitInfo_8088：当前UC8088的git信息。
- gitInfo_8288：当前UC8288的git信息。
- make_time_8288：当前UC8088的创建时间。
- make_time_8288：当前UC8088的创建时间。
- cce_version：当前CCE的版本号。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOAVERSION?	+WIOAVERSION:v0.10_ap8088, v0.10_ap8288 +GITINFO:Fri Apr 15 14:16:23 2022,Fri Apr 15 10:00:12 2022 +TIME:Apr 18 2022 09:34:56, Apr 15 2022 14:37:36 +CCEVERSION:b7a380 OK	输出查询到的版本信息
	ERROR	指令执行失败

3.19 AT+WIOAHOPPING 设置跳频

设置跳频频点或模式。

Command	Possible response(s)
+WIOAHOPPING=<type>,<value>, <value1>	OK
	ERROR

- <type>:
0: 设置跳频频点。
1: 设置跳频模式。
- <value>: type为0时表示跳频频点（0-200）；type为1时表示在原频点工作的帧数（0-255）。
- <value1>: type为0时无意义设为0；type为1时表示在跳频频点工作的帧数（0-255）。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOHAHOPPING=0,147,0	OK	设置跳频频点为147
	ERROR	指令执行失败
AT+WIOHAHOPPING=1,10,20	OK	表示在原频点工作10帧后在跳频频点工作20帧，如此循环
	ERROR	指令执行失败

3.20 AT+WIOTAIOTENUM 设置连接态终端数量

设置同时处于连接态终端的最大数量。

Command	Possible response(s)
+WIOTAIOTENUM=<max_num>	OK
	ERROR

- <max_num>: 最大数量，dlul_radio为0时默认72个，为1时默认144个。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAIOTENUM=144	OK	设置连接态终端的最大数量为144个成功
	ERROR	指令执行失败

3.21 AT+WIoTABCMCS 设置和查询广播传输速率

设置广播（普通广播、OTA和组播）的MCS。

Command	Possible response(s)
+WIoTABCMCS=<bc_mcs>	OK
	ERROR
+WIoTABCMCS?	+WIoTABCMCS:<bc_mcs> OK
	ERROR

- <bc_mcs>: mcs等级，在不同symbol length和不同MCS下，对应每帧传输的应用数据量（byte）会有差别，NA表示不支持，见下表：

（备注：下表中为单播数据包的数据量，如果是普通广播包，下表每项减2，如果是OTA包，下表每项减1）

symbol length	mcs0	mcs1	mcs2	mcs3	mcs4	mcs5	mcs6	mcs7
128	6	8	51	65	79	NA	NA	NA
256	6	14	21	51	107	156	191	NA
512	6	14	30	41	72	135	254	296
1024	6	14	30	62	107	219	450	618

Note1：由于协议限制，广播和单波在不同symbol_length下支持的最大MCS不同，但设置超过最大MCS时，默认设置为最大MCS，见下表：

symbol length	广播最大MCS	单波最大MCS
128	4	4
256	6	6
512	6	7
1024	5	7

Note2：当OTA的MCS为高阶MCS且一直发送时，此时发送上行会失败，在此种场景下要发上行，请采用低阶MCS发送OTA。128配置MCS大于等于MCS2为高阶小于MCS2为低阶；256配置MCS大于等于MCS3为高阶小于MCS3为低阶；512和1024配置MCS大于等于MCS4为高阶小于MCS4为低阶。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTABCMCS=3	OK	设置广播的mcs为3成功
	ERROR	指令执行失败
AT+WIOTABCMCS?	+WIOTABCMCS:3 OK	设置广播的mcs为3成功
	ERROR	指令执行失败

3.22 AT+WIOTARATE设置和查询传输模式和速率

三种模式：

第一种基本模式，是基本速率设置。

在第一种模式的基础上，在系统配置中dlul_ratio为1:2时，才能打开第二种模式，打开该模式能够提高该帧结构情况下两倍速率，默认第二种模式开启状态。

在第一种模式的基础上，打开第三种模式，能够提升（8*(1 << group_number)）倍单终端的速率，但是会影响网络中其他终端的上行，建议在大数据量快速传输需求时使用。

第四种为减少基带CRC为1字节，提高上层单帧数据量3字节。

备注：group_number为系统配置中的参数。

Command	Possible response(s)
+WIOTARATE=<rate_mode>, <rate_value>	OK
	ERROR
+WIOTARATEQUERY=<rate_mode>	+WIOTARATE QUERY:<rate_value> OK
	ERROR

- <rate_mode>:
 0: 当rate_mode为UC_RATE_NORMAL时, rate_value为 [uc mcs level e](#)。单波的最大MCS请参考3.20中的Note1。
 1: 当rate_mode为UC_RATE_MID时, rate_value为0或1, 表示关闭或打开, 必须和终端的状态保持一致。
 2: 当rate_mode为UC_RATE_HIGH时, rate_value为0, 表示关闭, rate_value为其他值, 表示当实际发送数据量 (byte) 大于等于该值时才会真正开启该模式, 常用建议设置rate_value为100, 可单独开启, 建议最好和终端状态保持一致。
 3: 当rate_mode为UC_RATE_CRC_TYPE时, rate_value为0或1, 默认为0, 表示四字节CRC校验, 为1表示1字节CRC校验, 也就是说开启该模式后, 上层单帧可多携带3字节数据

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTARATE=1,1	OK	当dlul_ratio为1:2时, 开启模式二
	ERROR	指令执行失败
AT+WIOTARATE=1,0	OK	关闭模式二
	ERROR	指令执行失败
AT+WIOTARATE=2,0	OK	关闭模式三
	ERROR	指令执行失败
AT+WIOTARATE=2,100	OK	当数据长度大于等于100byte时, 开启模式三
	ERROR	指令执行失败
AT+WIOTARATEQUERY=0	+WIOTARATEQUERY:3 OK	查询到当前mcs为3
	ERROR	指令执行失败

3.23 AT+WIOTALOG 设置WIoTa LOG

设置WIoTa AP log模式，可根据情况开关log和切换log串口。

Command	Possible response(s)
+WIOTALOG=<mode>	OK
	ERROR

- <mode>:
 0: 关闭UART log。
 1: 开启UART log。
 2: UART log使用UART0, 如果从UART1切换到UART0, 会把UART0的波特率改为460800, 此时AT的波特率也是用该值。
 3: UART log使用UART1, 如果从UART0切换到UART1, 会把UART0的波特率恢复为115200。
 4: 关SPI log。
 5: 开SPI log。
- 注意：默认状态下, UART log使用UART1, 波特率460800, AT使用UART0, 波特率115200, 在UART log的串口切换后 (AT会自动换到另外一个串口上), 需要特别注意串口工具使用的波特率是否对应, 如果AT的波特率不对时, 发送AT CMD会直接导致AT挂住！

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTALOG=0	OK	关闭串口log
	ERROR	指令执行失败
AT+WIOTALOG=1	OK	开启串口log
	ERROR	指令执行失败
AT+WIOTALOG=2	OK	串口log使用UART0
	ERROR	指令执行失败
AT+WIOTALOG=3	OK	串口log使用UART1
	ERROR	指令执行失败
AT+WIOTALOG=4	OK	关闭SPI log
	ERROR	指令执行失败
AT+WIOTALOG=5	OK	开启SPI log
	ERROR	指令执行失败

3.24 AT+WIOTACRC设置WIoTa CRC校验

设置WIoTa AP CRC校验长度限制和关闭crc，默认长度为大于等于100时加CRC校验。

Command	Possible response(s)
+WIOTACRC=<crc_limit>	OK
	ERROR
+WIOTACRC?	+WIOTACRC:<crc_limit>
	OK
	ERROR

- <crc_limit>:
0: 关闭CRC校验。
大于0: CRC校验的数据长度。 如， crc_limit为100，则大于等于100字节的数据会在末尾自动加CRC16的校验。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTACRC=0	OK	关闭CRC16校验
	ERROR	指令执行失败
AT+WIOTACRC=100	OK	当发送数据大于等于100byte时，开启CRC16校验
	ERROR	指令执行失败
AT+WIOTACRC?	+WIOTACRC:100 OK	查询到crc_limit的值为100
	ERROR	指令执行失败

3.25 AT+WIOTAPOSQUERY 根据ID查询在帧结构的位置信息

根据USERID查询连续最多200个终端的在帧结构里的信息，位置信息跟subsystem_id有关，即使是同一个id，subsystem_id不同，位置信息可能不同。

Command	Possible response(s)
+WIOTAPOSQUERY=<start_addr>,<addr_cnt>	+POS:<group_idx>,<burst_idx>,<slot_idx> OK
	ERROR

- <start_addr>：终端的user_id。
- <addr_cnt>：查询个数，至少为1个。
- <group_idx>：组号。
- <burst_idx>：子帧号。
- <slot_idx>：时隙号。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAPOSQUERY=ba4cb037,1	+POS:0,6,6 OK	id 0xba4cb037在帧结构的位置为0,6,6
	ERROR	指令执行失败

3.26 AT+WIOTASTATE查询ap上下行状态（该AT指令在v2.4及之后的版本将不再支持）

查询AP端上下行状态，可查询单个终端的单个状态，单个终端的所有状态，全部终端的所有状态。

Command	Possible response(s)
+WIOTASTATE=<get_or_reset>,<user_id>,<state_type>	+WIOTASTATE:<user_id>,<ul_rcv_len>,<ul_rcv_suc>,<dl_send_len>,<dl_send_suc>,<dl_send_fail>
	+WIOTASTATE:<user_id>,<state>
	OK
	ERROR

- <get_or_reset>：
 - 0：查询状态。
 - 1：重置状态。
 - <user_id>：终端的user_id。
 - <state_type>：查询的状态类型。
 - 1：查询上行接受成功的数据总长度。
 - 2：查询上行成功接受的次数。
 - 3：查询下行成功的总长度。
 - 4：查询下行成功的次数。
 - 5：查询下行失败的次数。
- 注：上行失败的AP端不做统计。
- 如： id不帶0x。
- 1) AT+WIOTASTATE=0,0,0为查询所有终端的所有状态。
 - 2) AT+WIOTASTATE=0,12345678,0为查询0x12345678的所有状态。
 - 3) AT+WIOTASTATE=0,12345678,1为查询0x12345678的上行接受成功的数据总长度。
 - 4) AT+WIOTASTATE=1,0,0为重置所有终端的所有状态。
 - 5) AT+WIOTASTATE=1,12345678,0为重置0x12345678的所有状态。
 - 6) AT+WIOTASTATE=1,12345678,1为重置0x12345678的上行接受成功的数据总长度。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTASTATE=0,0,0	+WIOTASTATE:0x12345678, 100,1,100,1,0 OK	查询到id为0x12345678的终端，成功收到长度为100byte的上行数据，成功接收到1次完整的上行数据，成功发送长度为100byte的下行数据，成功发送1次完整的下行数据，下行发送失败的次数为0
	ERROR	指令执行失败
AT+WIOTASTATE=0,12345678,0	OK	查询到id为0x12345678的终端，成功收到长度为100byte的上行数据，成功接收到1次完整的上行数据，成功发送长度为100byte的下行数据，成功发送1次完整的下行数据，下行发送失败的次数为0
	ERROR	指令执行失败
AT+WIOTASTATE=0,12345678,1	+WIOTASTATE:0x12345678,100 OK	查询到id为0x12345678的终端，成功接收到长度为100byte的上行数据
	ERROR	指令执行失败
AT+WIOTASTATE=1,0,0	OK	重置所有终端的所有状态成功
	ERROR	指令执行失败
AT+WIOTASTATE=1,12345678,0	OK	重置id为0x12345678的终端的所有状态成功
	ERROR	指令执行失败
AT+WIOTASTATE=1,12345678,1	OK	重置id为0x12345678的终端的上行接受数据长度成功
	ERROR	指令执行失败

3.27 AT+WIOTAREADMEM查询某片地址的内容（非程序异常不建议使用）

查询AP或基带某个地址或寄存器的内容。

Command	Possible response(s)
+WIOTAREADMEM=<type>,<addr>,<len>	+WIOTAREADMEM:<value> OK
	ERROR

- <type>:
0: 查询ap8088某片内存地址内容。
1: 查询ap8288某片内存地址内容。
- <addr>: 要查询的地址，如3b0014，注意地址不带0x。
- <len>: 要查询的内存长度。
- <value>: 从输入地址查询到的前四个字节的内容。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAREADMEM=0,2fe000,4	+WIOTAREADMEM:0xffffffff OK	查询UC8088内存地址为0x2fe000的四个字节内容为0xffffffff
	ERROR	指令执行失败
AT+WIOTAREADMEM=1,3b1014,4	+WIOTAREADMEM:0x2b47e49c OK	查询UC8288内存地址为0x3b1014的四个字节内容为0x2b47e49c
	ERROR	指令执行失败

3.28 AT+WIOTAFBALIGN 开启帧边界校准功能

v2.4版本之后，可通过该指令开启帧边界同步功能，默认关闭。

Command	Possible response(s)
+WIOTAFBALIGN=<state>	OK
	ERROR

- <state>:
0：关闭帧边界同步校准功能。
1：开启帧边界同步校准功能。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAFBALIGN=1	OK	开启帧边界同步校准功能
	ERROR	指令执行失败

3.29 AT+WIOTATSMODE 设置授时类型

v2.4版本之后，可通过该指令设置开启或关闭授时类型。

Command	Possible response(s)
+WIOTATSMODE=<type>,<state>	OK
	ERROR

- <type>：0表示授时类型为GPS，1表示授时类型为1588协议，2为同步助手。
- <state>：0表示关闭该类型授时，1表示开启该类型授时。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTATSMODE=0,1	OK	开启GPS授时功能
	ERROR	指令执行失败

3.30 AT+WIOTATIME 同步1588协议时间到AP（该指令只能有1588授时类型使用）

v2.4版本之后，可通过该指令同步带1588协议的master的时钟源到AP模组。

Command	Possible response(s)
+WIOTATIME=<sec>,<usec>	OK
	ERROR

- <sec>：1588授时时间戳整秒部分。
- <usec>：1588授时时间戳整微秒部分。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTATIME=235,568	OK	1588授时时间戳为235s.568微秒
	ERROR	指令执行失败

3.31 AT+WIOTATSRUN 开始授时或停止授时

v2.4版本之后，可通过该指令开启或关闭授时功能。

Command	Possible response(s)
+WIOTATSRUN=<state>	OK
	ERROR

- <state>：0表示关闭授时，1表示开启授时。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTATSRUN=1	OK	开始授时
	ERROR	指令执行失败

3.32 AT+WIOTATSCYCLE 设置授时周期

v3.1版本新增指令，用于设置授时周期。

Command	Possible response(s)
+WIOTATSCYCLE=<cycle_min>	OK
	ERROR

- <cycle_min>：默认15分钟，范围5~60分钟。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTATSCYCLE=20	OK	设置授时周期为20分钟
	ERROR	指令执行失败

3.33 AT+WIOTATSSTATE 查询授时状态

v2.4版本之后，开启授时时，可通过该指令查询授时状态。

Command	Possible response(s)
+WIOTATSSSTATE?	+WIOTATSSSTATE=<state>
	ERROR

- <state>：授时状态。
 - 0：授时线程创建，未开启gps或1588授时的状态。
 - 1：授时开始的状态。
 - 2：一次授时成功的状态。
 - 3：授时结果偏差过大无法完成对齐校验的状态。
 - 4：初次开机经过授时完成帧头计算成功后的状态，在该状态时需要立即将协议栈run起来。
 - 5：非初次开机，每隔固定时间进行帧头对齐校准成功的状态。
 - 6：一次授时停止的状态。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTATSSSTATE?	+WIOTATSSSTATE:4 OK	已完成初始帧头计算
	ERROR	指令执行失败

3.34 AT+GNSSPOSQUERY 查询GPS授时时的位置信息（当授时类型为GPS时有效）

v2.4版本之后，开启GPS授时时，可通过该指令查询授时时的三维坐标位置信息。

v3.2版本之后还可查询经纬度位置信息

Command	Possible response(s)
+GNSSPOSQUERY=<type>	+GNSSPOSQUERY:<pos_x>,<pos_y>,<pos_z>或 +GNSSPOSQUERY:<longitude>,<latitude>,<altitude>
	ERROR

- <type>：为0时查询三维位置信息，为1时查询经纬度信息。
- <pos_x>：位置坐标x。
- <pos_y>：位置坐标y。
- <pos_z>：位置坐标z。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+GNSSPOSQUERY=0	+GNSSPOSQUERY:2356,1255,45886 OK	授时位置信息成功
	ERROR	指令执行失败

3.35 AT+GNSSRELOCATION 设置GPS重新定位（当授时类型为GPS时有效）

v2.4版本之后，开启GPS授时时，可通过该指令重新让GPS定位，一般来说AP位置固定，只会在开机第一次进行定位操作，后续校准流程会从flash中直接读取位置信息，减少定位时间，但是如果AP位置发生变动时或需要重新定位值，可通过该指令设置让GPS下次启动重新定位已获取更加准确的授时时间。

Command	Possible response(s)
+GNSSRELOCATION=<state>	OK
	ERROR

- <state>: 0表示不重新定位, 1表示重新定位。

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+GNSSRELOCATION=1	OK	开启GPS重新定位成功
	ERROR	指令执行失败

3.36 AT+WIOTAPAGINGTX 设置连续信号唤醒配置

v2.4版本之后, 该指令可设置空中唤醒终端设置的寻呼配置。

Command	Possible response(s)
+WIOTAPAGINGTX=<freq>,<sepc_idx>,<band>,<symbol>,<awaken_id>,<send_time>	OK
	ERROR

- <freq>: 需要唤醒设备的频点配置。
- <sepc_idx>: 需要唤醒设备的频谱配置。
- <band>: 需要唤醒设备的带宽配置。
- <symbol>: 需要唤醒设备的symbol length配置。
- <awaken_id>: 需要唤醒的ID。
- <send_time>: 最小值为接收端检测周期。

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTAPAGINGTX=145,3,1,2,1235648,20	OK	设置寻呼配置成功
	ERROR	指令执行失败

3.37 AT+WIOTASENDPT 开始发送连续信号唤醒

v2.4版本之后, 当设置好寻呼配置后, 该指令控制基带开始唤醒终端, 该唤醒信号无法维持同步信号, 已经连接的终端会因为AP发送唤醒信号而失步。

Command	Possible response(s)
+WIOTASENDPT	OK
	ERROR

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTASENDPT	OK	发送唤醒寻呼成功
	ERROR	指令执行失败

3.38 AT+WIOTASENDPTF 开始发送连续信号唤醒 (新)

v3.3版本新增，与3.37的区别为，使用该命令发送唤醒信号的同时能维持同步信号，已经连接上该AP的终端不会因为AP发送唤醒信号而失步。

Command	Possible response(s)
+WIOTASENDPTF	OK
	ERROR

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTASENDPTF	OK	发送唤醒寻呼成功
	ERROR	指令执行失败

3.39 AT+WIOTAPAGINGRX AP进入低功耗配置

v3.1版本新增指令，配置paging rx配置，可以使AP进入低功耗，但需要新硬件的支持。

Command	Possible response(s)
+WIOTAPAGINGRX=<freq>,<spec_i>,<band>,<symbol>,<aw_id>,<d_period>,<nlen>,<ut>,<thres>,<ex_flag>,<ex_period>,<period_mult>,<aw_id_ano>	OK
	ERROR
+WIOTAPAGINGRX?	+WIOTAPAGINGRX =<freq>,<spec_i>,<band>,<symbol>,<aw_id>,<d_period>,<nlen>,<ut>,<thres>,<ex_flag>,<ex_period>,<period_mult>,<aw_id_ano>
	ERROR

- <freq>：频点。
- <spec_i>：频段。
- <band>：带宽。暂时只支持200K，即band为1。
- <symbol>：symbol length。
- <aw_id>：唤醒ID，根据symbol length不同，最大值不同，当symbol length为[0,1,2,3]时，唤醒ID最大值限制分别为[41,82,168,339]（可等于，最小值为0）。
- <d_period>：接收端检测周期（单位ms，最大值44000），每隔该时间，基带会自动单独起来检测一次信号，如果检测到信号，则唤醒整个系统，如果没有则继续sleep，该时间越长，整体功耗越低，相应的发送端想要唤醒接收端时则需要发送更长的时间。
- <nlen>：检测头配置，1,2,3,4，默认值4。
- <ut>：检测头配置，1,2,3，默认值2。
- <thres>：threshold检测门限，3~15，默认值10。增大该值，漏检率增大，虚警率减小。（虚警率即对噪声的敏感程度，漏检率即对唤醒信号的敏感程度）
- <ex_flag>：物理层检测到唤醒信号后，自动继续休眠的功能flag配置，设为1则开启该功能。
- <ex_period>：物理层检测到唤醒信号后，自动继续休眠的时长配置，单位ms，如果extra_period 小于等于（detect_period + 10）ms，则继续休眠 detect_period 时长，否则继续休眠 extra_period 时长。
- <period_mult>：第二个唤醒id的检测周期只能是第一个唤醒id的检测周期的倍数，该参数即为倍数，当倍数为0时，表示不检测第二个唤醒id，当倍数为1时，周期与第一个唤醒id相同，以此类推，注意，换算之后的周期，仍然有44秒的限制。
- <aw_id_ano>：第二个唤醒id，范围与第一个一样，不建议两个awaken id相同，当period_multiple不为0时才有效。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAPAGINGRX=160,3,1,1,30,1000,4,2,10,0,3000,2,25	OK	设置paging rx配置成功
	ERROR	指令执行失败

3.40 AT+WIOTASENDPR AP进入低功耗模式

v3.1版本新增指令，AP进入低功耗。

Command	Possible response(s)
+WIOTASENDPR=<is_need_32k_div>,<timeout_max>	OK
	ERROR

- <is_need_32k_div>：是否开启32k分频，0不开启，1开启。
- <timeout_max>：进入paging rx最大超时时间，超过超时时间未被唤醒则会自动醒来。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTASENDPR=0,10000	OK	AP进入paging rx低功耗模式
	ERROR	指令执行失败

3.41 AT+WIOTA AWAKEN 查询AP唤醒原因

v3.1版本新增指令，获取AP唤醒原因。

Command	Possible response(s)
+WIOTA AWAKEN?	+WIOTA AWAKEN=<aw_cause>,<is_cs>,<last_wakeup_cause>,<det_times>,<last_wakeup_idx>
	ERROR

- <is_need_32k_div>：是否开启32k分频，0不开启，1开启。
- <timeout_max>：进入paging rx最大超时时间，超过超时时间未被唤醒则会自动醒来。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTA AWAKEN?	+WIOTA AWAKEN=0,0,1,10,0	查询到唤醒原因
	ERROR	指令执行失败

3.42 AT+WIOTA HW 查询当前AP模组是否支持低功耗

v3.1版本新增指令，查询当前模组是否为新硬件，只有新硬件才支持低功耗。

Command	Possible response(s)
+WIOTA HW?	+WIOTA AWAKEN=<is_new_hw>
	ERROR

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAHW?	+WIOTAAWAKEN=0	当前为旧硬件，不支持 paging rx
	ERROR	指令执行失败

3.43 AT+WIOTAPAGINGMODE 设置是否开启唤醒ID扩展功能

v3.1版本新增指令，用于开启或关闭拓展ID功能。

Command	Possible response(s)
+WIOTAPAGINGMODE=<rx_mode>,<tx_mode>	OK
	ERROR

- <rx_mode>：0不开启paging rx ID扩展，1开启。
- <tx_mode>：0不开启paging tx ID扩展，1开启。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAPAGINGMODE=1,1	OK	paging rx和paging tx均开启ID扩展功能
	ERROR	指令执行失败

3.44 AT+WIOTASP 周期信号唤醒

v2.7版本新加功能，发送周期唤醒信号唤醒进入sync paging低功耗模式的终端。

Command	Possible response(s)
+WIOTASP=<user_id>,<fn_index>,<period>,<send_round>,<continue_fn>,<is_block>	OK +WIOTASP:<result>,<user_id>
	ERROR

- <user_id>：同步paging的终端id。
- <fn_index>：终端睡眠时的帧号，由callback传出，paging时传入。
- <period>：检测周期，与终端检测paging信号周期一致。
- <send_round>：发送唤醒信号的轮数，一般为1，表示发送一轮。
- <continue_fn>：单轮发送唤醒信号的帧数，一般为1，表示只在周期点发送，如果为2，则表示在周期点再前后各发送一帧唤醒信号，为3则表示前后各发两帧，以此类推，当超过detection_period的一半时，将变为每帧都发送唤醒信号
- <is_block>：为0表示非阻塞调用，为1表示阻塞调用
- <result>：为0表示寻呼结束。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTASP=12345678,200,10,1,1	OK +WIOTASP:0,0x12345678	非阻塞方式对id为0x12345678的终端，发送周期唤醒信号一次
	ERROR	指令执行失败

3.45 AT+WIOTASPNUM 查询某位置同步paging的终端数量

v2.7版本新加功能，用于查询帧结构某一位置上，正在进行同步paging的终端个数。

Command	Possible response(s)
+WIOTASPNUM=<group_idx>,<subframe_idx>	+WIOTASPNUM:<num> OK
	ERROR

- <group_idx>：帧结构组号。
- <subframe_idx>：帧结构子帧号。
- <num>：该位置同步paging的终端个数。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTASPNUM=0,7	+WIOTASPNUM:1 OK	查询到位置0,7上有1个iote正在进行同步paging
	ERROR	指令执行失败

3.46 AT+WIOTAAPSTATE 查询WIoTaAP的运行状态

v2.4版本之后，该指令可查询WIoTa AP的整体运行状态。

Command	Possible response(s)
+WIOTAAPSTATE?	+WIOTAAPSTATE:<state> OK
	ERROR

- <state>：0表示运行异常，1表示运行正常。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAAPSTATE?	+WIOTAAPSTATE:1 OK	AP运行正常
	ERROR	指令执行失败

3.47 AT+WIOTABCFNCYCLE 查询或设置广播帧发送周期

v2.8版本之后，该指令可设置或查询广播帧发送周期，一般不建议设置，广播帧固定在子帧6周期发送，用于同步帧号给终端。

Command	Possible response(s)
+WIOTABCFNCYCLE=<bc_fn_cycle>	OK
	ERROR
+WIOTABCFNCYCLE?	+WIOTABCFNCYCLE:<bc_fn_cycle> OK
	ERROR

- <bc_fn_cycle>：默认值11，范围0~11，0表示关闭广播帧发送。1~11表示间隔1~11帧发送一次广播帧。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTABCFNCYCLE=10	OK	设置广播帧发送周期为10帧发送一次成功
	ERROR	指令执行失败
AT+WIOTABCFNCYCLE?	+WIOTABCFNCYCLE:10 OK	查询到广播帧发送周期为10帧
	ERROR	指令执行失败

3.48 AT+WIOTABCROUND 查询或设置广播发送轮数

v2.8版本之后，该指令可设置或查询广播发送轮数，一般不建议设置，当信号好时可减少发送轮数，信号差时增加发送轮数。

Command	Possible response(s)
+WIOTABCROUND=<bc_send_round>	OK
	ERROR
+WIOTABCROUND?	+WIOTABCROUND:<bc_send_round> OK
	ERROR

- <bc_send_round>：广播发送轮数，默认为3，不建议更改。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTABCROUND=2	OK	设置广播发送轮数为2轮成功
	ERROR	指令执行失败
AT+WIOTABCROUND?	+WIOTABCROUND:2 OK	查询到广播发送轮数为2轮
	ERROR	指令执行失败

3.49 AT+WIOTAFRAMELEN 查询当前配置的帧长

v2.8版本之后，该指令可查询当期配置的帧长，单位微妙。

Command	Possible response(s)
+WIOTAFRAMELEN?	+WIOTAFRAMELEN:<frame_len> OK
	ERROR

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAFRAMELEN?	+WIOTAFRAMELEN:145968 OK	查询当前配置帧长为145968 微妙
	ERROR	指令执行失败

3.50 AT+WIOTASINGLETONE AP发送单音

v2.8版本之后，该指令控制AP发送单音。

Command	Possible response(s)
+WIOTASINGLETONE=<is_open>	OK
	ERROR

- <is_open>: 1: 打开单音发送, 0: 关闭单音发送。
- 实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTASINGLETONE=1	OK	设置AP发送单音成功
	ERROR	指令执行失败

3.51 AT+WIOTAMODULEID 获取AP模组ID

v2.9版本之后，该指令用户获取AP模组的ID。

Command	Possible response(s)
+WIOTAMODULEID?	+WIOTAMODULEID :<module_id> OK
	ERROR

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTAMODULEID?	+WIOTAMODULEID:861380000000081636 OK	查询到AP模组ID为 861380000000081636
	ERROR	指令执行失败

3.52 AT+WIOTAFRAMENUM 查询AP当前帧号

v2.9版本之后，该指令用户查询AP当前帧号。

Command	Possible response(s)
+WIOTAFRAMENUM?	+WIOTAFRAMELEN:<frame_num> OK
	ERROR

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTAFRAMENUM?	+WIOTAFRAMENUM:12563 OK	查询当前配置帧号为12563
	ERROR	指令执行失败

3.53 AT+WIOTAIOTELEAV 终端主动离开连接态

v2.9版本之后，该指令控制终端主动离开连接态。

Command	Possible response(s)
+WIOTAIOTELEAV=<user_id>	OK
	ERROR

- <user_id>：要离开连接态的终端ID。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAIOTELEAV=123456	OK	id为123456的终端离开连接态
	ERROR	指令执行失败

3.54 AT+WIOTASUBFCFG 设置子帧模式配置

v3.0版本新增指令，目前只用于语音数据传输配置。

Command	Possible response(s)
+WIOTASUBFCFG=<block_size>,<send_round>	OK
	ERROR

- <block_size>：语音数据的单帧大小。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTASUBFCFG=12,1	OK	设置语音单帧数据量为12字节，发送轮数为1轮
	ERROR	指令执行失败
+WIOTASUBFCFG?	+WIOTASUBFCFG: <block_size>,<send_round> OK	查询到子帧模式配置
	ERROR	指令执行失败

3.55 AT+WIOTAULSUBF 进入上行子帧数据接收模式

v3.0版本新增指令，使某终端进入上行子帧接收模式，目前只用于接收上行运营数据。

Command	Possible response(s)
+WIOTAULSUBF=<is_open>,<user_id>,<rach_delay>	OK
	ERROR

- <is_open>: 0表示退出该模式, 1表示进入该模式, 默认为0。
- <user_id>: 要进入该模式的终端ID。
- <rach_delay>: 接入delay, 会在接入消息上报, 需要用户保存并下发, 一般为3。

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTAULSUBF=1,12345,3	OK	终端12345进入上行子帧数据接收模式, 并设置接收的delay为3
	ERROR	指令执行失败

3.56 AT+WIOTADLSUBF 添加下行子帧数据

v3.0版本新增指令, 发送下行子帧数据。执行该命令后, 需在10秒内在串口工具的发送区输入长度为len的数据, 并点击发送, 不然会获取字符失败。

Command	Possible response(s)
+WIOTADLSUBF=<data_len>,<fn>	OK
	ERROR

- <data_len>: 下行子帧数据长度, 范围1~72字节。
- <fn>: 上行子帧数据上报时的序号, 0~255。

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTADLSUBF=12,23\r\n012345678901	OK	添加下行子帧数据成功
	ERROR	指令执行失败

3.57 AT+WIOTASUBFTEST 子帧模式测试

v3.0版本新增指令, 用于开启、关闭子帧模式测试, 清除或查询测试结果。

Command	Possible response(s)
+WIOTASUBFTEST=<mode>	OK
	ERROR

- <mode>: 0表示关闭测试, 1表示开启测试, 2表示清除测试结果, 3表示查询测试结果。

实例说明如下表:

输入AT指令	输出结果	输出结果说明
AT+WIOTASUBFTEST=1	OK	开启子帧模式测试成功
	ERROR	指令执行失败

3.58 AT+WIOTABCUTC 设置广播世界时间功能

v3.1版本新增指令，用于同步世界时间给终端，依赖授时功能。

Command	Possible response(s)
+WIOTABCUTC=<is_bc_utc>	OK
	ERROR

- <is_bc_utc>：0不开启，1开启。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTABCUTC=1	OK	开启广播世界时间功能
	ERROR	指令执行失败

3.59 AT+WIOTABL05 设置开关boost_level0.5的接收

v3.2版本新增指令，用于开关bl0.5的接收，默认关闭。

Command	Possible response(s)
+WIOTABL05=<is_open>	OK
	ERROR

- <cycle_min>：默认15分钟，范围5~60分钟。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTABL05=1	OK	开启接收bl0.5的消息
	ERROR	指令执行失败

3.60 AT+WIOTABNACK 设置开关bnack策略

v3.3版本新增指令，bnack策略能有效解决同一子帧多终端的业务冲突问题，默认开启。

Command	Possible response(s)
+WIOTABNACK=<is_open>	OK
	ERROR

- <cycle_min>：默认15分钟，范围5~60分钟。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTABNACK=0	OK	关闭bnack冲突解决策略
	ERROR	指令执行失败

3.61 T+WIOTAREGNEWCB 注册新回调函数

v3.2版本新增指令，用于注册新的接收和授时信息回调，注册新的后旧的将不再生效，新回调返回的信息更多。

Command	Possible response(s)
+WIOTAREGNEWCB=<new_rcv_cb>, <new_ts_cb>	OK
	ERROR

- <new_rcv_cb>：使用新的接收回调替换旧版回调。
- <new_ts_cb>：使用新的授时信息接收回调替换旧版回调。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAREGNEWCB =1,1	OK	替换回调成功
	ERROR	指令执行失败

3.62 AT+WIOTAFN 注册帧号刷新回调

v3.2版本新增指令，可实时监控帧号刷新，有需要用到帧号的业务（如：指定帧收发业务）注册。

Command	Possible response(s)
+WIOTAFN	OK
	ERROR

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAFN	OK	注册帧号刷新回调成功
	ERROR	指令执行失败

3.63 AT+WIOTARSBYFN 安排指定帧收发任务

v3.2版本新增指令，可安排某一帧开始安排接收任务并发送短消息，且发送只能在接收之后。

Command	Possible response(s)
+WIOTARSBYFN=<data_id>,<len>, <user_id>,<start_rcv_fn>,<rcv_fns>, <send_fns>	OK
	ERROR

- <data_id>：一般为数据地址，相当于某段数据的标记，当发送结束后，该参数会在发送结束后返回，可以让应用层感知哪段数据发送成功了或失败了，如果不需要可填任意值。
- <len>：发送数据的长度，为0时表示只安排接收不安排发送。
- <user_id>：安排定帧收发的终端ID。
- <start_rcv_fn>：接收任务开始的帧号，表示从这一帧开始安排短消息接收任务，开始帧号必须大于等于当前帧号。
- <rcv_fns>：安排短消息接收任务的帧数。
- <send_fns>：发送短消息的帧数，为0时表示只安排接收不安排发送。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTARSBYFN=123,10,63c8,500,3,1	OK	终端63c8从500帧开始连续3帧安排接收短消息任务，并发送10字节数据
	ERROR	指令执行失败

3.64 AT+WIOTAQC 快速连接指令

v3.2版本新增指令，可一键配置，实现终端和AP的快速连接。

Command	Possible response(s)
+WIOTAQC=<onoff>,<freq>,<mode>	OK
	ERROR

- <onoff>：0表示退出快速连接，1表示进入快速连接。
- <freq>：快速连接的频点，onoff为1时有效。
- <mode>：快速连接的模式，onoff为1时有效，范围0~4，分别表示短距离高并发、中距离高速率、中距离高并发、远距离通信、超远距离通信。

实例说明如下表：

输入AT指令	输出结果	输出结果说明
AT+WIOTAQC=1,145,0	OK	配置频点为145，模式为短距离高并发的快速连接
	ERROR	指令执行失败

4. 正常启动流程

启动流程	输出结果	输出结果说明
AT+WIOTAINIT	OK	初始化WIoTa协议栈
AT+WIOTARUN=1	OK	启动WIoTa协议栈

以上两个命令便可启动默认配置的WIoTa协议栈。