

E103-W12 产品使用说明书

DA16200 超低功耗 WiFi 串口模块

目录

免责声明和版权公告	1
第 1 章 概述	2
1.1 简介	2
1.2 功能特点	2
1.3 应用场景	3
第 2 章 规格参数	3
2.1 射频参数	3
2.2 电气参数	3
2.3 硬件参数	4
第 3 章 机械尺寸与引脚定义	5
3.1 E103-W12X 引脚图	5
3.2 E103-W12C 引脚图	6
第 4 章 推荐连线图	7
第 5 章 功能说明	8
5.1 工作角色	8
5.2 ACCESS POINT (AP 模式)	8
5.3 STATION (STA 模式)	8
5.4 网络功能	8
5.4.1 TCP server	8
5.4.2 TCP client	8
5.4.3 UDP	9
5.4.4 MQTT	9
5.4.5 HTTP Client	9
5.5 低功耗	10
5.6 扫描附近 AP 信息	10
第 6 章 使用教程	10
6.1 基本的 TCP/UDP 数据传输	10
6.1.1 与 PC 端的通信	10
6.1.2 MQTT	16
6.1.3 阿里云	17
6.1.4 HTTP Client	24
6.2 低功耗使用	24
6.2.1 sleep1 模式	24
6.2.2 sleep2 模式	24
6.2.3 sleep3 模式	24
第 7 章 焊接作业指导	27
7.1 回流焊温度	27
7.2 回流焊曲线图	28
第 8 章 修订历史	28
关于我们	29

免责声明和版权公告

本文中的信息，包括供参考的 URL 地址，如有变更，恕不另行通知。文档“按现状”提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。本文档不负任何责任，包括使用本文档内信息产生的侵犯任何专利权行为的责任。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，不管是明示许可还是暗示许可。

文中所得测试数据均为亿佰特实验室测试所得，实际结果可能略有差异。

文中提到的所有商标名称、商标和注册商标均属其各自所有者的财产，特此声明。

最终解释权归成都亿佰特电子科技有限公司所有。

注意：

由于产品版本升级或其他原因，本手册内容有可能变更。亿佰特电子科技有限公司保留在没有任何通知或者提示的情况下对本手册的内容进行修改的权利。本手册仅作为使用指导，成都亿佰特电子科技有限公司尽全力在本手册中提供准确的信息，但是成都亿佰特电子科技有限公司并不确保手册内容完全没有错误，本手册中的所有陈述、信息和建议也不构成任何明示或暗示的担保。

第 1 章 概述

1.1 简介

E103-W12 系列是成都亿佰特电子有限公司基于 Dialog DA16200 SoC 芯片开发一款支持 IEEE 802.11b/g/n 超低功耗 WiFi 模块，内置高性能 ARM Cortex-M4 处理器，工作在 2.4~2.4835GHz 频段，无线最大发射功率可达 20dBm。

功能上，模块支持 IEEE802.11 b/g/n 标准，模块支持 AP 和 STA 模式，支持 TCP/UDP/HTTP/MQTT 多种网络传输协议，可在低功耗模式下保持 WIFI 连接。



E103-W12X



E103-W12C

1.2 功能特点

- ◆ 支持 IEEE802.11 /b/g/n 标准，工作在 2.4GHz 频段；
- ◆ 具备三种低功耗模式可选择，最低电流可达 0.2uA；
- ◆ 支持 AP、STA、AP+STA 3 种工作模式；
- ◆ 支持 WEP/WPA/WPA2/WPA3 多种 WIFI 安全认证方式；
- ◆ 支持 TCP/UDP/HTTP/MQTT 多种网络通信协议；
- ◆ 最大支持 7 路 Socket 连接；
- ◆ AP 接入点支持 6 路 STA 设备连接；
- ◆ 支持 SNTP 网络时间获取（可上网的前提下）；
- ◆ 支持 MQTT 和 MQTTS 网络协议（阿里云、百度云、OneNet，亚马逊云）
- ◆ 支持 HTTP Client；
- ◆ 支持 TCP SERVER/TCP CLIENT、UDP 通信方式；
- ◆ 支持 AT 指令配置；
- ◆ 支持静态 IP 地址分配和 DHCP 动态分配；
- ◆ 支持扫描附近 AP 信息；
- ◆ 支持 WIFI 快速连接；
- ◆ 支持保持 WIFI 连接的超低功耗模式；
- ◆ 支持 PWM 输出，ADC 采集，GPIO 控制；

1.3 应用场景

-医疗和保健

- 多参数病人监护仪
- 心电图 (ECG)
- 医院电子床及床控系统
- 远程医疗系统

-建筑和家庭自动化:

- 暖通空调系统和恒温器
- 视频监控, 视频门铃, 和低功耗的相机
- 建筑安全系统和低功耗电子锁

-智能电器

-智能穿戴

-资产跟踪

-工厂自动化

-电网基础设施

第 2 章 规格参数

2.1 射频参数

射频参数	单位	型号		备注
		E103-W12X	E103-W12C	
WiFi 协议	-	IEEE 802.11b/g/n		
天线形式	-	IPEX	陶瓷天线	特性阻抗约 50 欧姆
参考距离	M	500	300	天线增益 5dBi, 模块与设备端通信
频段	GHz	2.4		支持全球免许可 ISM 2.4GHz 频段

2.2 电气参数

射频参数	单位	型号		备注
		E103-W12X	E103-W12C	
工作电压	V	2.1-3.6V		超过 3.5V 永久烧毁模块
通信电平	V	3.3		使用 5V TTL 有风险烧毁
功耗	发射功率	dBm		20

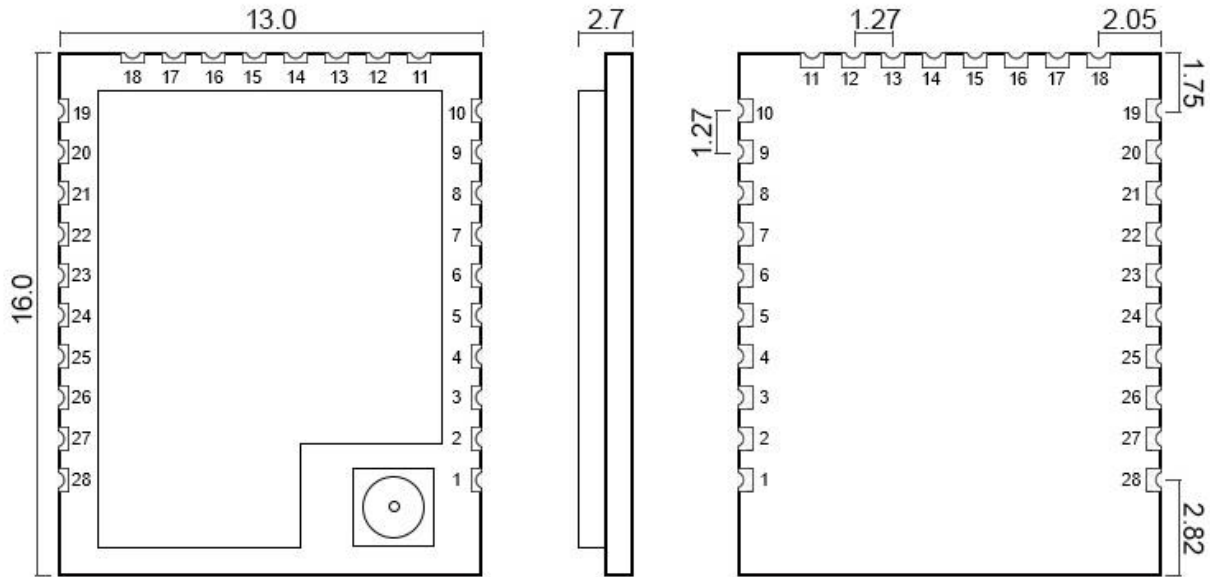
	发射电流	mA	200		
	接收电流	mA	46		
	休眠电流	Sleep1	uA	0.2	完全关断
		Sleep2	uA	3	可定时唤醒或引脚唤醒, 唤醒可在200ms 以内连接 WiFi。
Sleep3		uA	100-200	WIFI 保持连接, TCP/UDP/MQTT 可保持连接, 可以接收数据; 发送数据需要通过唤醒模块. 功耗与路由器和网络质量有关。	
温度	工作温度	℃	-40~+85	工业级	
	储存温度		-40~+125		

2.3 硬件参数

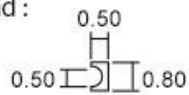
硬件参数	型号		备注
	E103-W12X	E103-W12C	
芯片	DA16200		
Flash	4MB		可定制 16MB/8MB
封装方式	贴片		
尺寸	16*13*2.7mm		误差尺寸为±0.1mm
重量	3.5g	3.4g	误差为±0.1g

第 3 章 机械尺寸与引脚定义

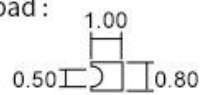
3.1 E103-W12X 引脚图



Top pad:



Bottom pad:



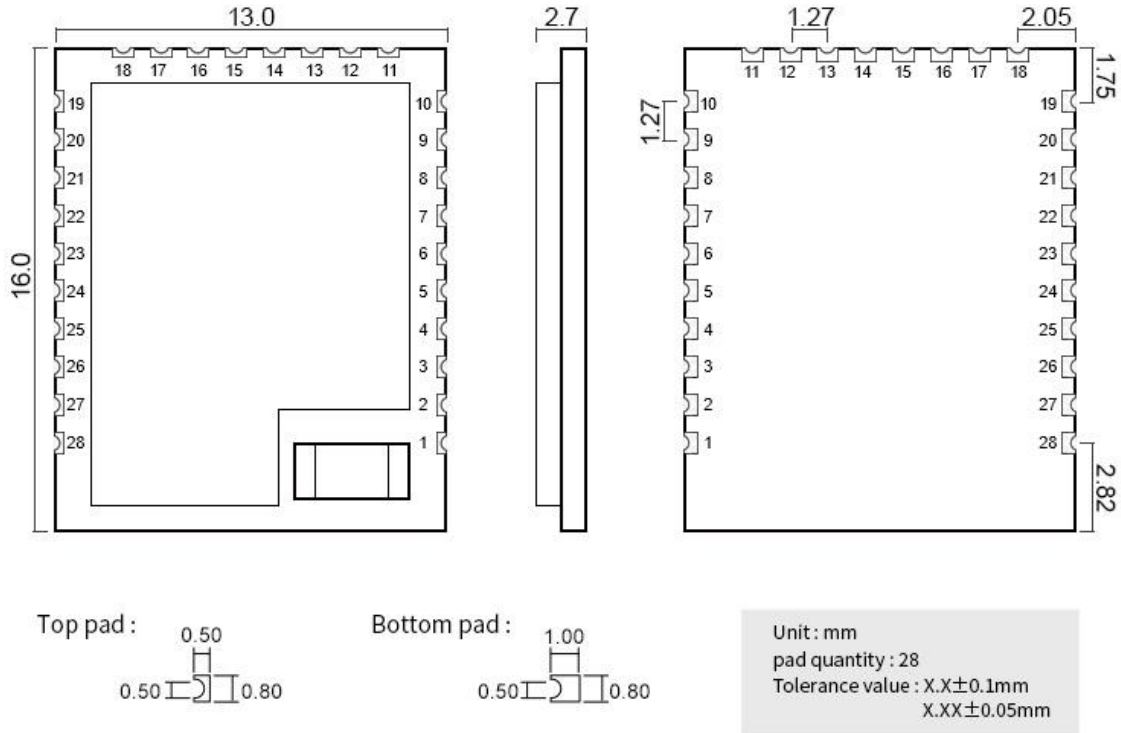
Unit : mm

pad quantity : 28

Tolerance value : X.X±0.1mm

X.XX±0.05mm

3.2 E103-W12C 引脚图

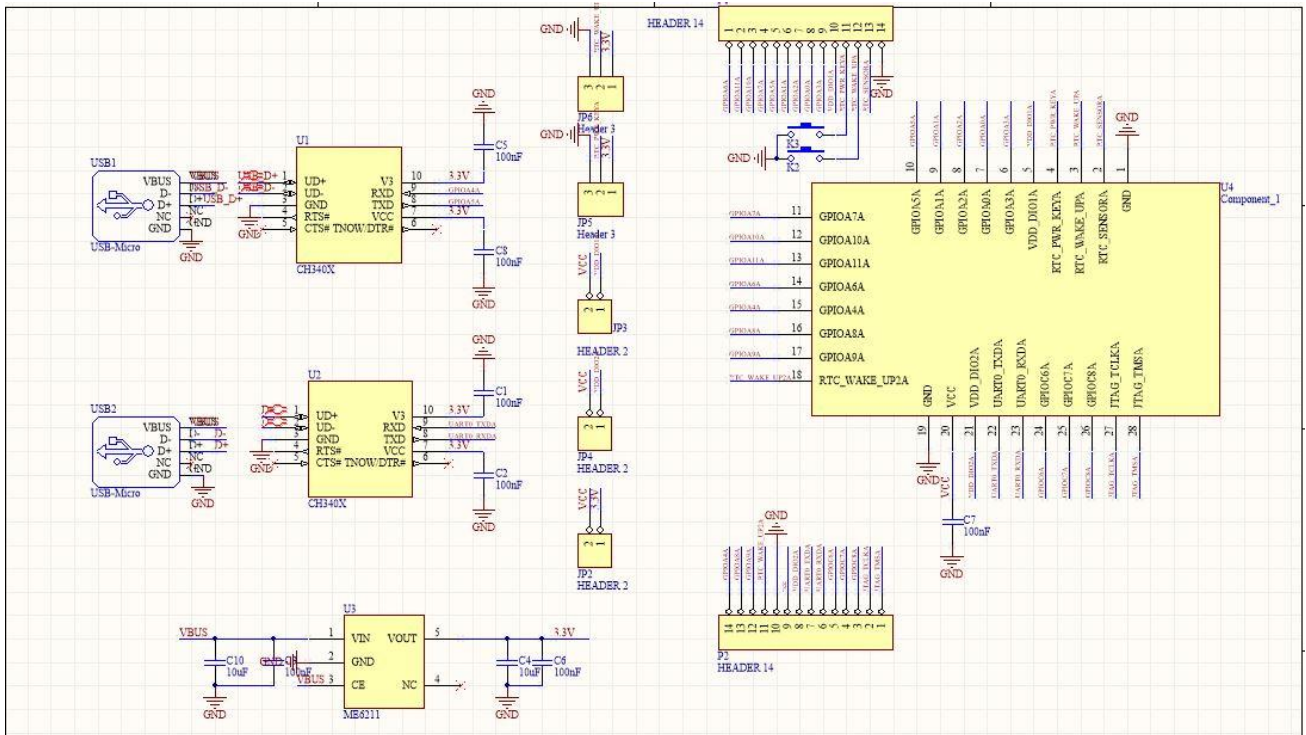


E103-W12C&E103-W12X 引脚定义:

引脚序号	引脚名称	引脚类型	引脚用途
1	GND	-	地线, 连接到电源参考地
2	NC	-	-
3	WAKE_UP	-	sleep2/sleep3 外部唤醒脚
4	PWR_KEY	-	供电使能脚, 高电平使能, 低电平关断处于 sleep1 模式
5	VDD_DIO1	I/O	GPIOA1-GPIOA11 供电 1.8v-3.3V, GPIO 输出电压与此一致
6	GPIOA3	I/O	通用 I/O 口, 可复用 ADC channel3
7	GPIOA0	I/O	通用 I/O 口, 可复用 ADC channel0
8	GPIOA2	I/O	通用 I/O 口, 可复用 ADC channel2
9	GPIOA1	I/O	通用 I/O 口, 可复用 ADC channel1
10	GPIOA5	I/O	AT 指令串口 UART_RX
11	GPIOA7	I/O	通用 I/O 口
12	GPIOA10	I/O	通用 I/O 口, 可复用 PWM 输出
13	GPIOA11	I/O	通用 I/O 口, sleep3 模式下收到数据通知引脚
14	GPIOA6	I/O	通用 I/O 口
15	GPIOA4	I/O	AT 指令串口 UART1_TX

16	GPIOA8	I/O	通用 IO 口
17	GPIOA9	I/O	通用 IO 口
18	RTC_WAKE_UP2	I/O	下拉接地
19	GND	I/O	地线, 连接到电源参考地
20	VCC	I/O	模块供电, 3.3V
21	VDD_DIO2	I/O	GPIOC6-GPIOC8 供电 1.8v-3.3V, GPIO 输出电压与此一致
22	UART0_TXD	I/O	调试串口 TX, 用户不用关心
23	UART0_RXD	I/O	调试串口 RX, 用户不用关心
24	GPIOC6	I/O	通用 IO 口
25	GPIOC7	I/O	通用 IO 口
26	GPIOC8	I/O	通用 IO 口
27	JTAG_TCLK	-	JTAG 仿真烧录引脚
28	JTAG_TMS	-	JTAG 仿真烧录引脚

第 4 章 推荐连线图



注意:

- 1、 供电电源必须保证在 2.1V~3.6V, 为保证模块稳定工作, 建议外部选择供电能力大于 500mA 的 LDO 供电。

第 5 章 功能说明

本模块从 WiFi 层的角色来说可以分为 Access Point 即通常所说的 AP 接入点, Station 即通常所说的站点从传输模从服务类别来说分为 TCP Server, TCP Client, UDP, MQTT, HTTP client。

5.1 工作角色

工作角色是基于 WiFi 物理层的角度来定义的。

5.2 Access Point (AP 模式)

Access Point 简称 AP 模式, 类似于路由器, 允许无线设备连入, 并建立基于 TCP/IP 的 server、client、UDP 通信。此模式下支持连接 6 个 station, 最大支持 7 路 TCP socket 传输。指令 AT+CWMODE=1 来配置模块工作在 AP 角色。

5.3 Station (STA 模式)

Station 模式简称 STA, 此角色下模块不提供连接, 只能连接到 Access Point 或者路由器上, 本模块在 Station 角色下的支持 TCP server, TCP client, UDP, 最大支持 8 路 socket。也支持 MQTT, HTTP client。指令 AT+CWMODE=0, S 设置处于 STA 模式。

。

5.4 网络功能

本模块支持 TCP server, TCP client, UDP, MQTT, HTTP client 等功能。所有网络功能可以同时使用, 列如可以在开启 TCP server 的同时同时做 TCP client 去连接其他的 Server。

5.4.1 TCP server

让模块作为 TCP 服务器工作。使用指令: AT+CIPSERVER 来设置对应的 ip 和端口的 TCP server。最大可以开启 8 个 server, 每个 server 能连接 7 路 client。开启 8 个 server 后无法使用 TCP Client 和 UDP 功能了。

5.4.2 TCP client

让模块作为 TCP 客户端工作。使用指令: AT+CIPSTART 开始 TCP client 并连接到对应的 server。最大能开启 8 个 client 去连接 server, 开启 8 个 client 后无法使用 TCP server 和 UDP 功能了。

5.4.3 UDP

模块使用 UDP 进行通信，使用 AT+CIPUDP=<local_port>指令开启本地 UDP 端口号。最大可以创建 8 个本地 udp 端口号，达到最大数量后无法使用 TCP Client 和 TCP server 了。

5.4.4 MQTT

模块在 MQTT 模式下，支持阿里云、百度云、OneNet 等物联网平台。将在平台创建的服务参数入到模块即可进行通信。

5.4.4.1 阿里云

基于阿里云平台的网络通信，需要登录阿里云获取相关参数，主要包含产品秘钥、设备名称、客户端 ID 等信息，详见第七章阿里云配置教程。

5.4.4.2 百度云

基于百度云平台的网络通信，需要登录百度云获取相关参数，主要包含设备名、用户名、密码等信心，详见第七章百度云配置教程。

5.4.4.3 Onenet

基于 OneNet 云平台的网络通信，需要登录 OneNet 获取相关参数，主要包含设备 ID、产品 ID、权鉴信息等，详见第七章 OneNet 配置教程。

5.4.4.4 亚马逊云

基于亚马逊云的网络通信，需要登录亚马逊云获取相关参数，主要包含 AmazonRootCA1.pem，xxxxxx-certificate.pem.crt，xxxxxx-private.pem.key 这三个证书文件。详见第七章亚马逊云配置教程。

5.4.5 HTTP Client

使用该功能时，只需要配置好对应的服务器资源符 URL，启动一次触发请求即可获得到服务器响应的资源，无需关心复杂的 HTTP 协议层，详见第七章 HTTP Client 配置教程。

5.5 低功耗

此模块有三种低功耗模式分别为：sleep1, sleep2, sleep3 模式。Sleep1 模式为关断模块供电，电流 0.2ua 左右。Sleep2 模式 RTC 是工作的，可以定时唤醒也可以引脚唤醒，电流 3ua 左右。Sleep3 模式 wifi 是保持连接的可以接收 tcp, udp, mqtt 数据；发送数据需要通过引脚和指令唤醒模块后发送数据，发送完成后发送指令进入休眠。

5.6 扫描附近 AP 信息

当本模块工作在 STA 模式下时，可以使用 AT 指令来扫描附近的 AP 信息，若存在则将 AP 信息通过串口打印出来，格式如下：

固定头	MAC(BSSID)	frequency	rsssi	flag	ssid
+CWSCAN:	70:5d:cc:32:15:32	2422	-30	[WPA2-PSKCCMP] [WPS] [ESS]	TEST

- 1、扫描附近 AP 只能在 STA 模式下。
- 2、每个参数以逗号 ‘\t’ 分隔，以<LF>结束。MAC 地址(BSSID)之间以分号 ‘:’ 分隔。

第 6 章使用教程

本模块的功能繁多，在使用之前需要配置相关参数，然后才能正常的工作。
在以下的使用教程中，加*的是必须执行的操作，其他模式使用时请根据自己的需求设置。

6.1 基本的 TCP/UDP 数据传输

6.1.1 与 PC 端的通信

6.1.1.1 模块做 STA 与 PC 的 TCP server 通信

模块在 STA 角色下要实现与 PC 通信需要保证 PC 和模块处于同一局域网。这里我们模块做 STA 与 PC 的 TCP 的 server 进行通信，下一小节模块做 AP 与 PC 的 TCP client 进行通信；其余组合也是类似的。

- 1、设置模块为 STA 模式，然后重启生效：AT+CWMODE=0；重启：AT+RST。
- 2、连接到电脑所在网段的路由器：AT+CWJAPA=TST-2.4G, TST12345678，连接成功后会打印连接的 wifi 名子和分配到的 ip 地址。

3、打开网络调试助手，开启一个 TCP Server。网络调试助手的本机地址需要与模块的 ip 地址在同一网段：



4、模块连接到 TCP Server：AT+CIPSTART=192.168.2.25,8080，返回：+CIPSTART:1 OK。

5、使用指令向 TCP server 发送数据：AT+CIPSEND=1,10,192.168.2.25,8080,123456789。

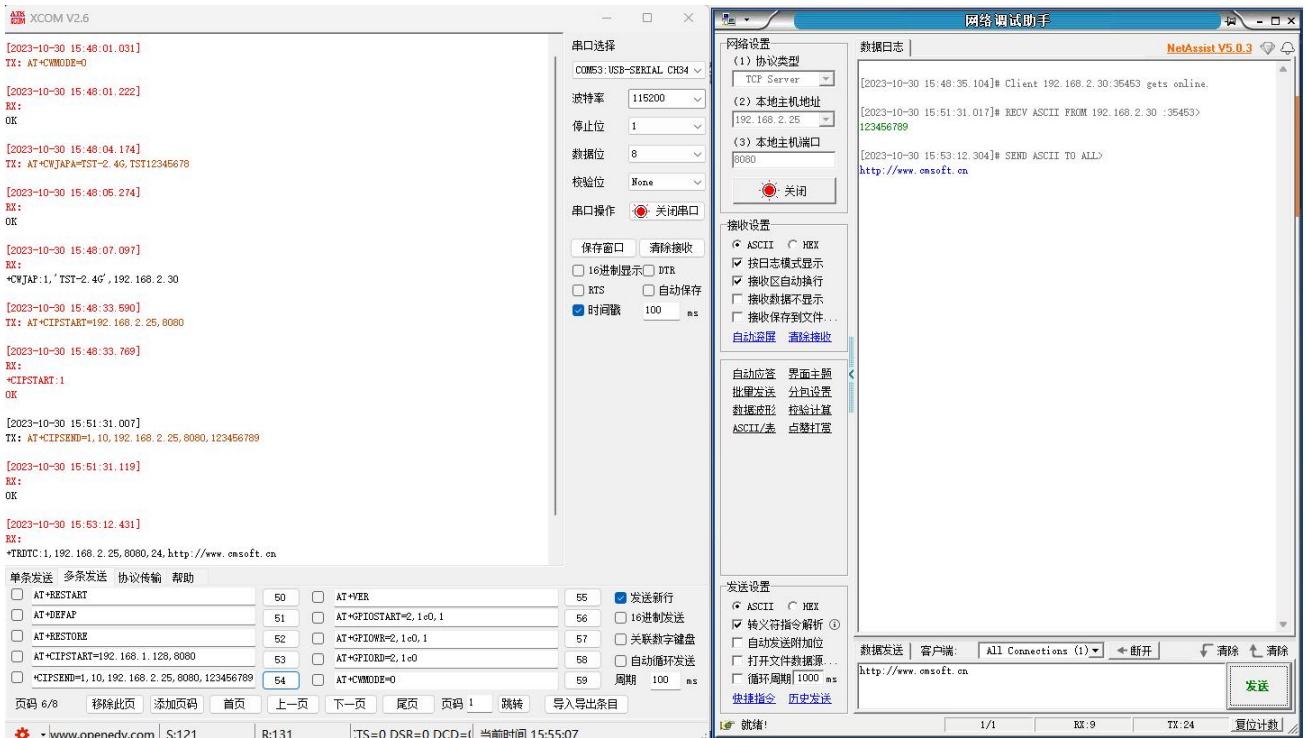
6、查看网络调试助手发现收到了数据：



7、网络调试助手向模块发送数据，模块收到数据打印：

```
[2023-10-30 15:53:12.431]
RX:
+TRDTC:1,192.168.2.25,8080,24,http://www.cmsoft.cn
```

8、整个操作步骤。



6.1.1.2 模块做 AP 与 PC 的 TCP client 通信

- 1、设置模块为 AP 模式：AT+CWMODE=1。
- 2、复位模块：AT+RST。
- 3、设置 AP 的 IP 地址：AT+CIPIP=1, 192.168.10.1, 255.255.255.0, 192.168.10.1。
- 4、设置 DHCP 服务器：AT+CWDHCP=1, 192.168.10.2, 192.168.10.10, 1800。
- 5、设置软 AP：AT+CWSAP=MY_SSID, 3, 1, 12345678, 1, CN。

6、电脑连接模块创建的 AP:



连接成功后模块打印电脑的 MAC 地址:

```
[2023-10-30 16:54:40.488]  
RX:  
+CWCST:f0:d4:15:92:3a:0b
```

7、模块创建 TCP Server: AT+CIPSERVER=8080。

8、网络调试助手做 TCP Client 连接模块，连接成功后模块打印 PC 的 IP 地址和端口号。

```
[2023-10-30 17:02:40.693]  
RX:  
+TRCTS:0,192.168.10.2,50004
```

9、模块向网络调试助手发送数据: AT+CIPSEND=0,10,192.168.10.2,50004,1234567890。

10、网络调试助手收到了数据，网络调试助手下发：



```
[2023-10-30 17:13:37.754]
RX:
+TRDTS:0,192.168.10.2,50004,22,http://www.cmsoft.cn
```

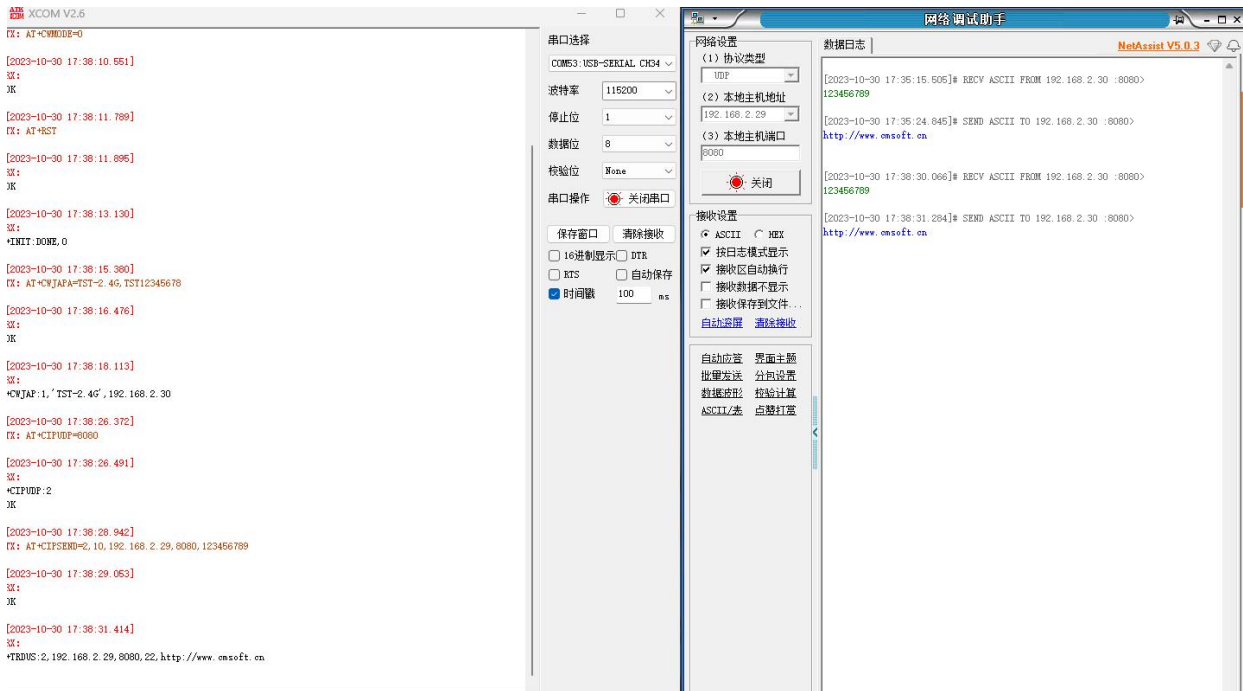
6.1.1.3 UDP 传输

上面已经介绍了模块做为 AP 或者 STA 连接 TCP 的使用方法了，这里我们就只以 STA 模式进行通信了。

- 1、设置模式为 STA 模式：AT+CWMODE=0。
- 2、复位：AT+RST。
- 3、连接到路由器（模块 ip 需要和电脑需要在同一网段）：AT+CWJAPA=TST-2.4G,TST12345678。
- 4、打开本地 UDP 端口：AT+CIPUDP=8081。
- 5、网络调试助手创建一个 UDP 会话。
- 6、模块向网络调试助手发送数据，网络调试助手向模块发送数据。



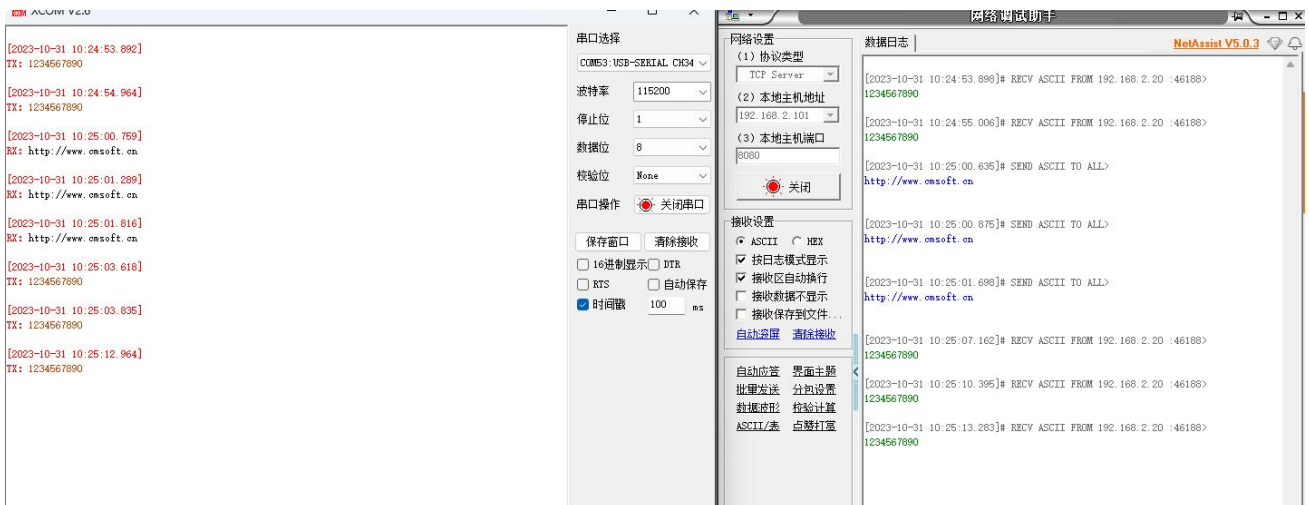
7、全部操作如下图



6.1.1.4透传模式

透传模式目前只能在非低功耗模式下使用，支持 TCP client 和 udp 模式，发送数据需要以换行结束。透传模式下只能建立一个 socket 连接，不能多连接。

- 1、连接到路由器 AP（模块 ip 需要和电脑需要在同一网段）：AT+CWJAPA=TST-2.4G,TST12345678。
- 2、打开网络调试助手，开启一个 TCP Server，网络调试助手的本机地址需要与模块的 ip 地址在同一网段。
- 3、模块连接到 TCP Server：AT+CIPSTART=192.168.2.101,8080。
- 4、进入透传：AT+CIPTRANSLINK=0 //0 是 tcp,1 是 udp。
- 5、发送数据，双方互发无问题，如：123456789。



10、发送+++，退出透传模式 //这个也要加换行。

6.1.1.5 高速传输指令

在高波特率，快速发送的时候使用 AT 指令或者透传都可能丢包。我们可以先发送指令进入数据模式，然后在发送数据。<ESC>的值是 0x1b，指令说明见 AT 指令手册。

发送格式：<ESC>H<cid>,<length>,<remote_ip >,<remote_port >

数据：<data>

- 1、连接到路由器 AP（模块 ip 需要和电脑需要在同一网段）：AT+CWJAPA=TST-2.4G,TST12345678。
- 2、打开网络调试助手，开启一个 TCP Server，网络调试助手的本机地址需要与模块的 ip 地址在同一网段。
- 3、模块连接到 TCP Server：AT+CIPSTART=192.168.2.101,8080。
- 4、进入数据模式：<ESC>H1,10,0,0。
- 6、发送上面指定长度的数据，长度不够时会一直等待，如：1234567890。

```
[2023-10-31 10:52:27.475]
TX: □H1,10,0,0

[2023-10-31 10:52:27.583]
RX:
OK

[2023-10-31 10:52:33.402]
TX: 1234567890

[2023-10-31 10:52:33.517]
RX:
OK
-
```

6.1.2 MQTT

在 MQTT 下支持 v3.1 和 v3.1.1 版本，只要支持这两种版本的 mqtt 云平台的都可以连接。本小结将以阿里云，百度云，ONNET, 亚马逊云为实例进行连接和首发数据。本节只讲解模块端的基本操作，云平台的注册和操作见另一个文档。

6.1.3 阿里云

- 1、模块做 STA 连接到能访问外网的路由器上面。
- 2、在阿里云->物联网平台->设备管理->设备信息->MQTT 连接参数获取我们需要的连接参数。



- 3、设置 mqtt 客户端 ID:

AT+MQTTCID=' *****.device1|securemode=2,signmethod=hmacsha256,timestamp=1698667627806|'

由于阿里云 client id 中有逗号需要用单引号或者双引号把整个 client id 扩起来。

- 4、设置 mqtt broker 地址和端口号:

AT+MQTTBR=a1***.iot-as-mqtt.cn-shanghai.aliyuncs.com, 1883。

- 5、订阅主题，可以在阿里云->物联网平台->设备管理->产品->Topic 列表->自定义 Topic 列表这里找到 topic 格式，这里我们在云平台把第二个 topic 属性改为自发自收方便测试。



AT+MQTTTS=1,/a*****/\${deviceName}/user/update

- 6、发布主题:

AT+MQTTTP=/a*****/\${deviceName}/user/update

- 7、设置设备名和登录密码:

AT+MQTTLI=username,password

- 8、启动连接 MQTT:

AT+MQTTCL=1

```
[2023-10-30 20:47:11.654]
TX: AT+MQTTCL=1

[2023-10-30 20:47:11.763]
RX:
JK

[2023-10-30 20:47:12.318]
RX:
+MQTTCL:1
```

响应+MQTTCL:1 连接成功，如果失败可能是 topic 设置不正确。

9、发布 mqtt 消息：

AT+MQTTMSG=hello, /a*****/\${deviceName}/user/update

由于我们发布者和订阅者主题相同，我们会收到发送的消息。

```
[2023-10-30 20:53:05.799]
TX: AT+MQTTMSG=hello, /k*****|C3UkL/device1/user/update

[2023-10-30 20:53:06.017]
RX:
OK

+MQTTMSGSEND:1

[2023-10-30 20:53:06.224]
RX:
+MQTTMSG:hello, /k*****|C3UkL/device1/user/update,5
```

6.1.3.1 百度云

模块做 STA 连接到能访问外网的路由器上面。

1、百度云注册完设备获取到连接参数是 IoTCoreId, DeviceKey, DeviceSecret 这三个。



与实际 mqtt 连接参数不一致，我们要通过百度云提供的这个在线工具计算出我们实际的连接参数：

🔗 <https://iotcore-dev-tool.gz.bcebos.com/mqtt-sign/>

填写设备信息 (*为必填项)

*IoTCoreId	<input type="text" value="必填项"/>
地域	<input style="border-bottom: 1px solid #ccc;" type="text" value="广州"/>
*DeviceKey	<input type="text" value="必填项"/>
*DeviceSecret	<input type="text" value="必填项"/>
时间戳 (秒)	<input style="width: 80%;" type="text"/> <input type="button" value="填入当前时间戳"/> <input type="button" value="填入 0"/>
加密方式	<input style="border-bottom: 1px solid #ccc;" type="text" value="MD5"/>

使用当前时间戳则密码有效期为10min, 更安全; 使用0则密码长期有效, 建议仅在测试环境使用。

结果 (MQTT 连接信息)

Broker 地址	<input type="text"/>
MQTT 用户名	<input type="text"/>
MQTT 密码	<input type="text"/>
ClientID	<input style="border-bottom: 1px solid #ccc;" type="text"/> 推荐值

这里输入信息就是上面我们获取的连接参数，点击计算就能生成我们实际的连接参数。需要注意这个时间戳时间只有 10 分钟。需要长久有效时间戳设置成 0 或者远大于当前时间。

2、其余操作和阿里云一样。

6.1.3.2 OneNet

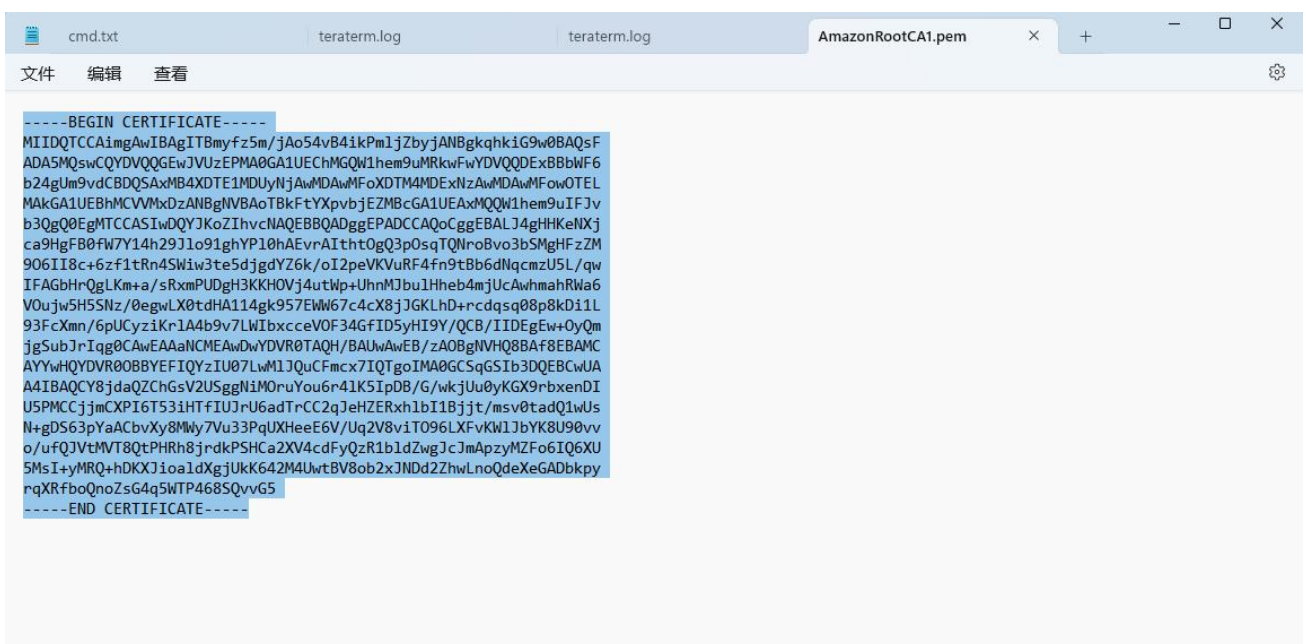
onenet 与阿里云和百度云是类似的，都是使用云平台的参数和计算工具生成最终的连接参数：broker 地址，用户名，密码，clientID。需要注意的是 onenet 现在的 mqtt 版本是 v3.1.1, 版本设置不正确无法连接成功。

6.1.3.3 亚马逊云

亚马逊云以证书方式进行连接，我们注册了亚马逊云的 MQTT 服务后会获得几个证书。我们需要用到的证书是下图中后缀这几个。

名称	修改日期
策略证书	2023/8
0a7b1d69255594606ed69a4424109783cf41cdbde9967ad412a1fc634a21ef18-certificate.pem.crt	2023/8
0a7b1d69255594606ed69a4424109783cf41cdbde9967ad412a1fc634a21ef18-private.pem.key	2023/8
0a7b1d69255594606ed69a4424109783cf41cdbde9967ad412a1fc634a21ef18-public.pem.key	2023/8
AmazonRootCA1.pem	2023/8
AmazonRootCA3.pem	2023/8

1、给模块烧录证书，我们需要烧录根证书，客户端证书和私钥，以记事本方式打开我们上面的证书，然后复制。



示例：

```
AT+MQTTCA=0, ' -----BEGIN CERTIFICATE-----
MIIDQTCCAimgAwIBAgITBmyfz5m/jAo54vB41kPm1jZbyjANBgkqhkiG9w0BAQsF
ADA5MQswCQYDVQQGEwJVUzEPMA0GA1UEChMGQW1hem9uMRkwFwYDVQQDExBBbWF6
b24gUm9vdCBDQSAxMB4XDTE1MDUyNjAwMDAwMFoXDTE4MDEwNzAwMDAwMFowOTEL
MAkGA1UEBhMCVWxkZDANBgNVBAoTBkFtYXNjZWZmZjEzZWZmZjEzZWZmZjEzZWZm
b3QgQ0EgMTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALJ4gHHKeNXj
ca9HgFB0fW7Y14h29Jl091ghYP10hAEvrAItht0gQ3p0sqTQNroBvo3bSMgHFzZM
906II8c+6zf1tRn4SWiw3te5dJgdYZ6k/oI2peVKVuRF4fn9tBb6dNqcmzU5L/qw
IFAGbHrQgLKm+a/sRxmPUDgH3KKHOVj4utWp+UhnMJbulHheb4mjUcAwhmahRwa6
V0ujw5H5SNz/0egwLX0tdHA114gk957EWW67c4cX8jJGKLhD+rcdqsq08p8kDi1L
93FcXmn/6pUCyziKr1A4b9v7LWIbxcceVOF34GfID5yHI9Y/QCB/IIDEgEw+OyQm
jgSubJrIqgOCAwEAaNCMEAwDwYDVR0TAQH/BAUwAwEB/zA0BgNVHQ8BAf8EBAMC
AYYwHQYDVR0OBBYEFIQYzIU07LwM1JQuCFmcx7IQTgoIMA0GCSqGSIB3DQEBCwUA
A4IBAQC8jdaQZChGsV2USggNiM0ruYou6r41K5IpDB/G/wkjuU0yKG9rbxenDI
U5PMCCjmmCXPI6T531HTfIUJrU6adTrCC2qJehZERxh1b1Bjtt/mvsv0tadQ1wUs
N+gDS63pYaACbvXy8MMy7Vu33PqUXHeeE6V/Uq2V8viT096LXFvKW1JbYK8U90vv
o/ufQJVtMT8QtPHRh8jrdkPSHCa2XV4cdFyQzR1b1dZwgJcJmApzyMZf06IQ6XU
5MsI+yMRQ+hDKXJioaldXgJukK642M4UwtBV8ob2xJNDd2ZhwLnoQdeXeGADbkpy
rqXRfboQnoZsG4q5WTP468SQvV65
-----END CERTIFICATE-----
```

```
A4IBAQC8Y8jdaQZChGsV2USggNiM0ruYou6r4lK5IpDB/G/wk jUu0yKGX9rbxenDI
U5PMCCj jmCXPI6T53iHTfIUJrU6adTrCC2qJeHZERxhlbi1Bj jt/msv0tadQ1wUs
N+gDS63pYaACbvXy8MWy7Vu33PqUXHeeE6V/Uq2V8viT096LXFvKWlJbYK8U90vv
o/ufQJVtMVT8QtPHR8jrdkPSHCa2XV4cdFyQzR1bl dZwgJcJmApzyMZFo6IQ6XU
5MsI+yMRQ+hDKXJioaldXg jUkK642M4UwtBV8ob2xJNDd2ZhwLnoQdeXeGADbkpy
rqXRfboQnoZsG4q5WTP468SQvvG5
-----END CERTIFICATE-----'
```

证书内容需要用单引号或者双引号括起来。

```
[2023-10-31 11:14:58.746]
TX: AT+MQTTCA=0, '-----BEGIN CERTIFICATE-----

[2023-10-31 11:14:58.898]
RX:
OK

[2023-10-31 11:15:31.901]
TX: AT+MQTTCA=0, '-----BEGIN CERTIFICATE-----
MIIDQTCCAimgAwIBAglTBmyfz5m/jAo54vB4ikPmljZbyjANB gkqhkiG9wOBAQsF
ADA5MQswCQYDVQGEwJVUzEPMAOGA1UEChMGQW1hem9uMRkwFwYDVQQDExBBbWV6
b24gUm9vdCBDQSAsMB4XDTE1MDUyNjAwMDAwMFoXDTE1MDUyNjAwMDAwMFowOTEL
MAkGA1UEBhMCVGVzZANBAGUwRjEwRjEwRjEwRjEwRjEwRjEwRjEwRjEwRjEwRjEw
b3QgQ0EgMTCCASlwdQYJKoZIhvcNAQEBBQADggEPADCCAQoCggEBALJ4gHhKeNXj
ca9HgFBOfw7Y14h29Jl091ghYPl0hAEvrAltht0gQ3p0sqTQNr0Bvo3bSMgHFzZM
906Ii8c+6zfl tRn4SWiw3te5djgdYZ6k/oI2peVKVuRF4fn9tBb6dNqcmzU5L/qw
IFAGbHrQgLkm+a/sRxmPUDgH3KKhOVj4utWp+UhmMjbulHheb4mjUcAwhmahRwa6
V0uJw5H5SNz/OegLX0tdHA114gk957EWW67c4cX8jJGKLhD+redqsq08p8kDi1L
93FcXmn/6pUCyziKrlA4b9v7LWtIbxcceVDF34GfID5yHl9Y/QCB/IIDEgEw+0yQm
jgSubJrIqgOCAwEAAaNCMEAwDwYDVROTAQH/BAUwAwEB/zAOBgnVHQ8BAf8EBAMC
AYYwHQYDVROOBbYEFiQYzIU07LwMLJQuCFmcx7IQTgoLMAOGCSqGSIb3DQEBCwUA
A4IBAQC8Y8jdaQZChGsV2USggNiM0ruYou6r4lK5IpDB/G/wk jUu0yKGX9rbxenDI
U5PMCCj jmCXPI6T53iHTfIUJrU6adTrCC2qJeHZERxhlbi1Bj jt/msv0tadQ1wUs
N+gDS63pYaACbvXy8MWy7Vu33PqUXHeeE6V/Uq2V8viT096LXFvKWlJbYK8U90vv
o/ufQJVtMVT8QtPHR8jrdkPSHCa2XV4cdFyQzR1bl dZwgJcJmApzyMZFo6IQ6XU
5MsI+yMRQ+hDKXJioaldXg jUkK642M4UwtBV8ob2xJNDd2ZhwLnoQdeXeGADbkpy
rqXRfboQnoZsG4q5WTP468SQvvG5
-----END CERTIFICATE-----'
```

- 2、接下来按此方法依次烧录客户端证书和私钥。
- 3、由于亚马逊要验证当前的时间，我们需要开启 SNTP 网络获取时间：AT+CIPSNTP=1, pool.ntp.org, 86400。
- 4、设置时区，默认是 UTC 时间与中国时间相差 8 小时：AT+TZONE=28800。
- 5、重启然后查询时间看一下：AT+TZONE=28800, AT+RST, AT+TIME。

```
[2023-10-31 11:27:35.353]
TX: AT+TZONE=28800

[2023-10-31 11:27:35.539]
RX:
OK

[2023-10-31 11:27:41.284]
TX: AT+RST

[2023-10-31 11:27:41.393]
RX:
OK

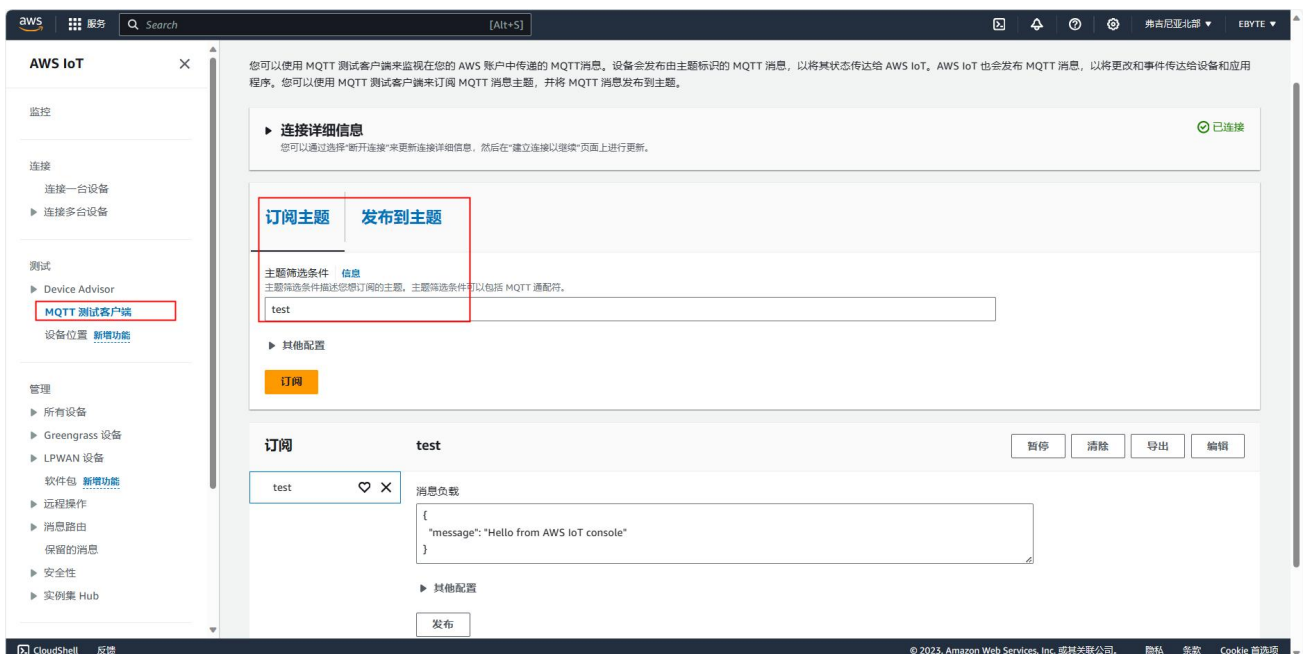
[2023-10-31 11:27:42.632]
RX:
+INIT: DONE, 0

[2023-10-31 11:27:44.177]
RX:
+CWJAP: 1, 'TP-LINK_6C35', 192.168.1.115

[2023-10-31 11:27:47.309]
TX: AT+TIME

[2023-10-31 11:27:47.427]
RX:
+TIME: 2023-10-31, 11:27:49
OK
```

- 10、设置 mqtt broker 地址和端口号：AT+MQTTBR=a*****-ats.iot.us-east-1.amazonaws.com，8883。
- 11、开启 TLS 功能：AT+MQTTTLS=1。
- 12、亚马逊云测试 MQTT 的时候，主题可以自定义，这里我们定义主题名字为 test。



- 13、模块订阅主题，模块发布主题：AT+MQTTTS=1, test; AT+MQTTTP=test1。

14、启动连接 MQTT: AT+MQTTCL=1。

```
[2023-10-31 11:44:32.851]
TX: AT+MQTTCL=1

[2023-10-31 11:44:32.959]
RX:
OK

[2023-10-31 11:44:35.877]
RX:
+MQTTCL: 1
```

连接成功。

14、模块给 AWS 发送消息,有换行或者逗号的消息都需要用双引号或者单引号括起来:

AT+MQTTMSG=' { "message": "hello"}',test

The screenshot shows a web-based MQTT client interface. On the left, under '其他配置' (Other Settings), there is a '发布' (Publish) button. Below it, a topic 'test1' is selected, and the message payload is a JSON object: { "message": "hello w12" }. On the right, a terminal window displays the following log output:

```
[2023-10-31 13:37:27.844]
TX: AT+MQTTMSG=' {"message": "hello"}', test1

[2023-10-31 13:37:28.010]
RX:
OK

+MQTTMSGSEND: 1

[2023-10-31 13:37:38.858]
TX: AT+MQTTMSG=' {"message": "hello w12"}', test1

[2023-10-31 13:37:39.009]
RX:
OK

+MQTTMSGSEND: 1
```

15、亚马逊 mqtt 下发

The screenshot shows the MQTT client interface with the '发布到主题' (Publish to Topic) tab selected. The topic name is 'test'. The message payload is a JSON object: { "message": "hello aws" }. Under '其他配置' (Other Settings), the '发布' (Publish) button is highlighted with a red box. On the right, the terminal window shows the following log output:

```
[2023-10-31 13:37:28.010]
RX:
OK

+MQTTMSGSEND: 1

[2023-10-31 13:37:38.858]
TX: AT+MQTTMSG=' {"message": "hello w12"}', test1

[2023-10-31 13:37:39.009]
RX:
OK

+MQTTMSGSEND: 1

[2023-10-31 13:38:41.138]
RX:
+MQTTMSG: { "message": "hello aws"}, test, 28

[2023-10-31 13:38:51.418]
RX:
+MQTTMSG: { "message": "hello aws"}, test, 28

[2023-10-31 13:38:52.075]
RX:
+MQTTMSG: { "message": "hello aws"}, test, 28

[2023-10-31 13:38:52.285]
RX:
+MQTTMSG: { "message": "hello aws"}, test, 28
```

6.1.4 HTTP Client

使用 AT+HTTPC 获取对应 http 信息，这里使用 httpbin.org 做示例：

AT+HTTPC=https://httpbin.org/get,get。

```
[2023-10-31 10:56:48.074]
TX: AT+HTTPC=https://httpbin.org/get,get

[2023-10-31 10:56:48.194]
RX:
OK

[2023-10-31 10:56:49.580]
RX: HTTP/1.1 200 OK
Date: Tue, 31 Oct 2023 02:56:50 GMT
Content-Type: application/json
Content-Length: 297
Connection: close
Server: gunicorn/19.9.0
Access-Control-Allow-Origin: *
Access-Control-Allow-Credentials: true

{ "args": {}, "headers": { "Accept": "*/*", "Host": "httpbin.org", "User-Agent":
"lwIP/2.1.2 (http://savannah.nongnu.org/projects/lwip)", "X-Amzn-Trace-Id": "Root=1-65406cf2-
653c5eb5453c413e410e19d8" }, "origin": "112.54.89.224", "url": "https://httpbin.org/get"}
+HTTPCSTATUS:0
```

6.2 低功耗使用

6.2.1 sleep1 模式

Sleep1 为关断模式，将 PWR_KEY 拉低就进入 sleep1 模式，拉高则退出。

6.2.2 sleep2 模式

Sleep2 模式 RTC 是工作的，唤醒方式是时间到了或者 WAKE_UP 下降沿唤醒；

AT+SETDPMSLPEXT=10,0 //进入sleep2 模式，10s时间到了自动唤醒，也可以通过WAKE_UP脚进行唤醒。

6.2.3 sleep3 模式

sleep3 模式下保持 wifi 连接，sleep3 模式只能在 sta 模式下，并且需要连接到 AP，未连接成功多次失败后会进入异常低功耗模式需要重新上电（可以通过 PWR_KEY 实现），未唤醒的时候无法向串口发送数据。这里我们以低功耗模式下作为 tcp client 为示例。

1、连接到 AP:

AT+CWJAPA=TST-2.4G,TST12345678

2、电脑的 ip 与模块在同一网段，打开网络调试助手开启一个 TCP server.



3、模块连接到 PC 端的 TCP Server 上:

AT+CIPSTART=192.168.2.101,8080

4、保存当前状态，准备进入低功耗:

AT+CIPSAVE

5、进入低功耗:

AT+DPM=1

```
[2023-10-31 09:44:54.155]
TX: AT+CIPSTART=192.168.2.101,8080

[2023-10-31 09:44:54.330]
RX:
+CIPSTART:1
OK

[2023-10-31 09:47:22.939]
TX: AT+CIPSAVE

[2023-10-31 09:47:23.126]
RX:
OK

[2023-10-31 09:47:57.048]
TX: AT+DPM=1

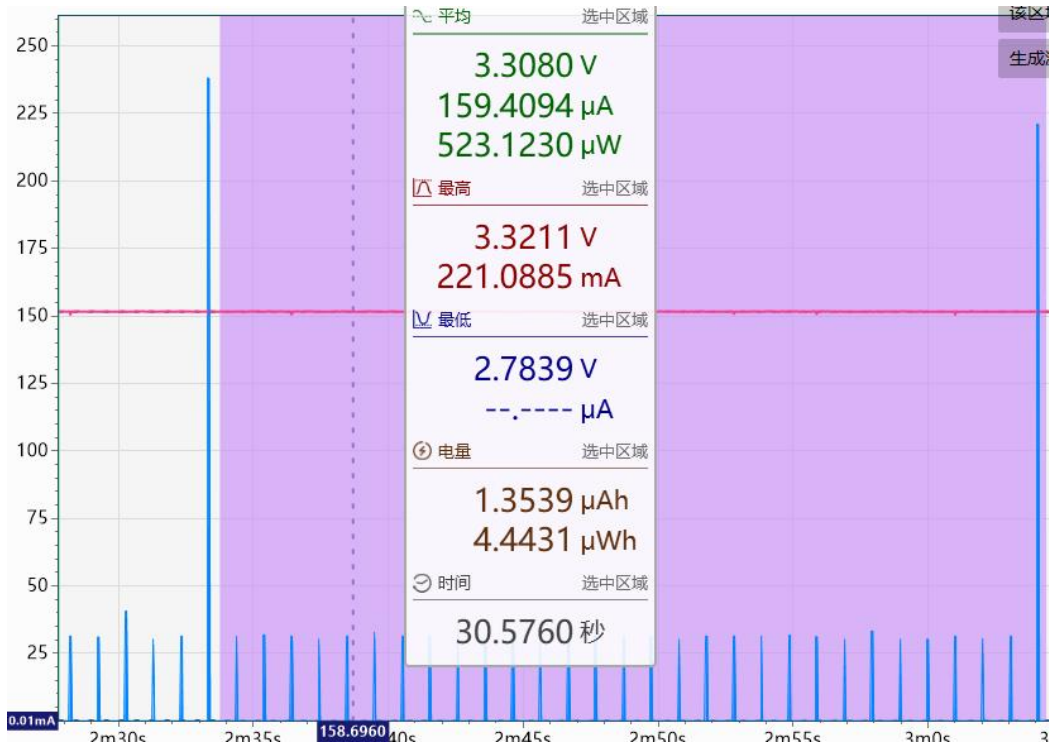
[2023-10-31 09:47:57.415]
RX:
OK

[2023-10-31 09:47:58.693]
RX:
+INIT: DONE, 0

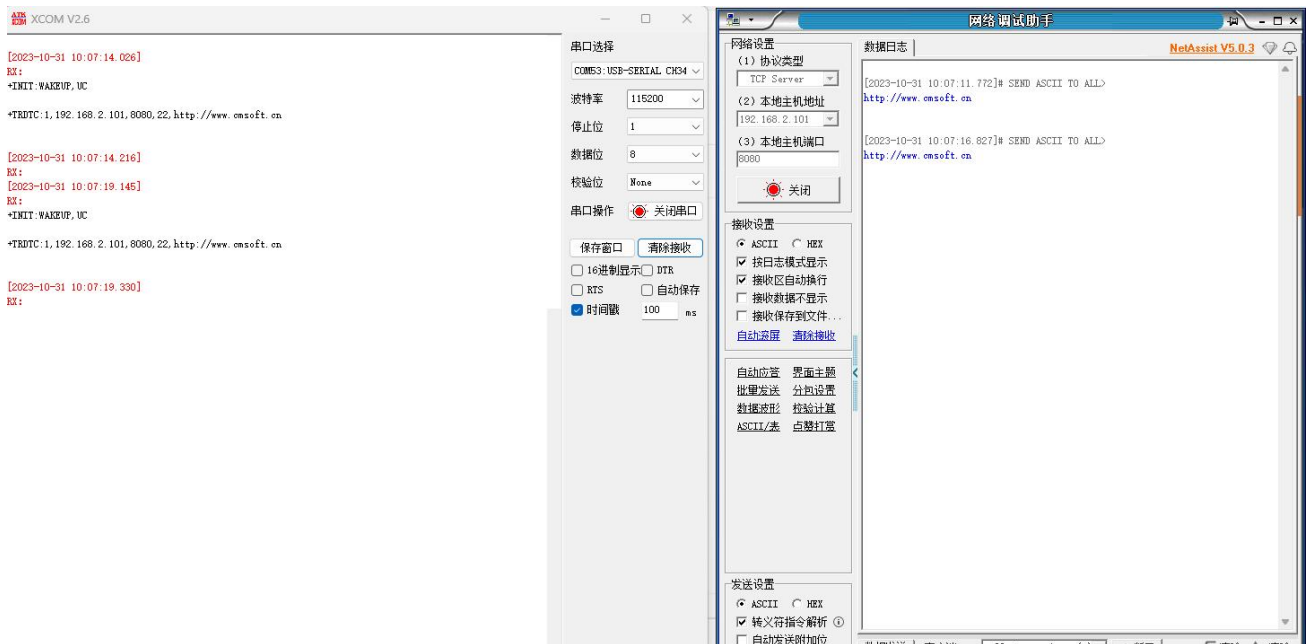
[2023-10-31 09:48:00.741]
RX:
+CWJAP:1,'TST-2.4G',192.168.2.24

[2023-10-31 09:48:01.928]
RX:
+CIPINFOALL:1,TCP,192.168.2.101,8080,32493
```

6、进入 sleep3 模式下，模块会周期性的醒来接收数据，一段时间上报一次。正常情况下平均功耗在 150ua 左右，但部分路由器的 wifi 帧会让模块应答导致一直回复功耗会变的很大，大概 1ma 左右；ap 网络质量不好的时候也会导致平均功耗变大。这里我们是使用 TP-LINK 的路由器，平均功耗在 150ua 左右，也可使用手机开热点验证。



7、在低功耗模式下，我们网络调试助手向模块发送数据是可以接收的，如下图：

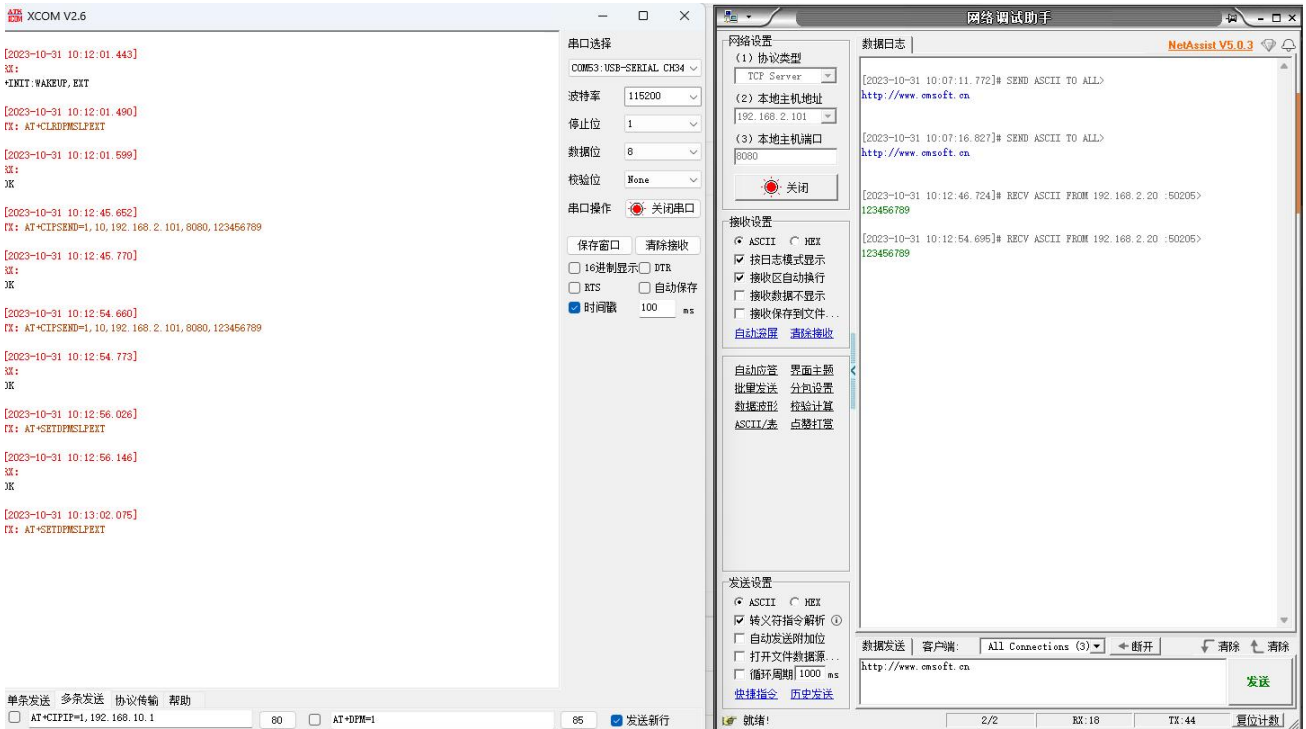


8、模块发送数据需要唤醒，WAKE_UP 脚下降沿唤醒，需要在 1s 内发送 AT+CLRDPSLPEXT 指令展示退出低功耗不然又会进入低功耗状态，串口是无法使用的。

9、模块唤醒后向 pc 端的 tcp server 发送数据，发送完成后通过 AT+SETDPSLPEXT 进入低功耗，示例：

AT+CIPSEND=1, 10, 192. 168. 2. 101, 8080, 123456789

AT+SETDPSLPEXT

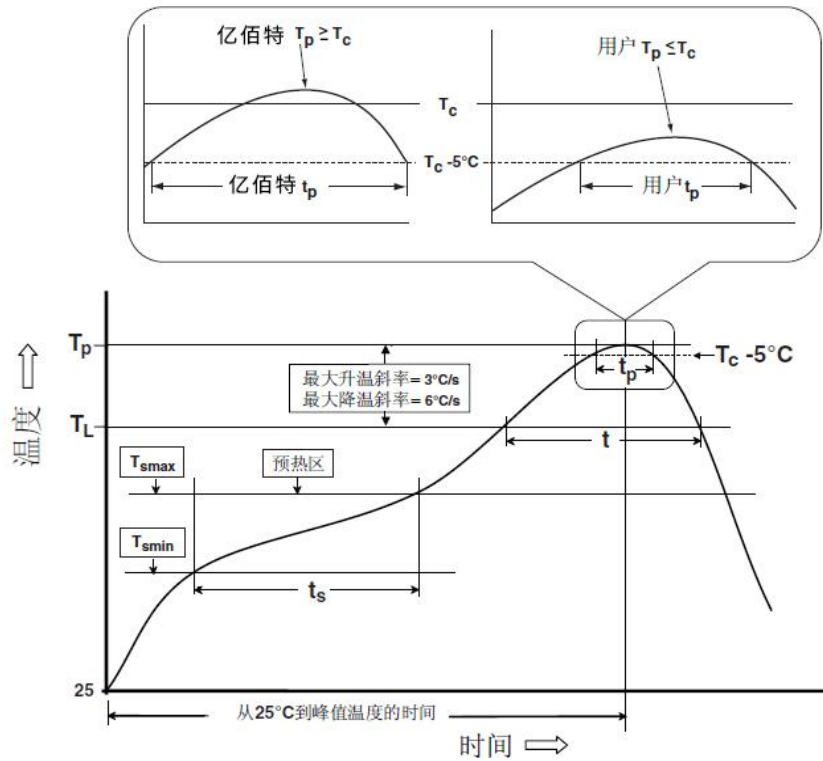


第 7 章 焊接作业指导

7.1 回流焊温度

回流焊曲线特征		有铅工艺组装	无铅工艺组装
预热/保温	最低温度 (T _{smin})	100°C	150°C
	最高温度 (T _{smax})	150°C	200°C
	时间 (T _{smin} ~T _{smin})	60-120 秒	60-120 秒
升温斜率 (T _L ~T _p)		3°C/秒, 最大值	3°C/秒, 最大值
液相温度 (T _L)		183°C	217°C
T _L 以上保持时间		60~90 秒	60~90 秒
封装体峰值温度 T _p		用户不能超过产品“潮湿敏感度”标签标注的温度。	用户不能超过产品“潮湿敏感度”标签标注的温度。
在指定分级温度 (T _c) 5°C 以内的时间 (T _p), 见下图		20 秒	30 秒
降温斜率 (T _p ~T _L)		6°C/秒, 最大值	6°C/秒, 最大值
室温到峰值温度的时间		6 分钟, 最长	8 分钟, 最长
※温度曲线的峰值温度 (T _p) 容差定义是用户的上限			

7.2 回流焊曲线图



第 8 章 修订历史

版本	修订日期	修订说明	维护人
1.0	2023-10-31	初始版本	luo

关于我们



销售热线：4000-330-990

公司电话：028-61543675

技术支持：support@cdebyte.com

官方网站：www.ebyte.com

公司地址：四川省成都市高新西区西区大道 199 号 B5 栋

 **成都亿佰特电子科技有限公司**
EBYTE Chengdu Ebyte Electronic Technology Co.,Ltd.

X-ON Electronics

Largest Supplier of Electrical and Electronic Components

Click to view similar products for [WiFi Modules - 802.11 category](#):

Click to view products by [Ebyte manufacturer](#):

Other Similar products are found below :

[7265.NGWG.SW](#) [HDG204-DN-3](#) [FXX-3061-MIX](#) [EWM-W190H02E](#) [ESP32-S3R8](#) [SIM800C](#) [Ai-WB2-01F](#) [Ai-M62-32S](#) [DFR0654-F](#)
[ESP32-WROVER-IB-N16R8](#) [Ai-M62-M2-I-Kit](#) [Ai-WB2-01M](#) [Ai-M62-13U](#) [NodeMCU-VB-01-Kit](#) [Ai-M62-32S-Kit](#) [Ai-M62-13](#) [Ai-M61-](#)
[32S\(ALL\)](#) [Ai-M61-32S](#) [Ai-WB2-12F](#) [Ai-M62-M2-I](#) [Ai-WB2-32S](#) [Ai-M62-13-Kit](#) [Ai-M62-32S-I](#) [Ai-M61-32S-I](#) [E103-W12X](#) [E103-W04B](#)
[E103-W05A](#) [E103-W20\(7628\)](#) [E103-W04](#) [E103-W05C](#) [E103-W03](#) [ML307S](#) [MN316-DBRD](#) [ML305](#) [M5310-E](#) [ESP32-SOLO-1-N49](#)
[ESP32-SOLO-1-N16](#) [ESP-WROOM-02D-N2](#) [ESP-WROOM-02-N4](#) [ESP32-S2-WROVER-I-N8R2](#) [ESP32-S2-SOLO-N4R2](#) [HLK-7621](#)
[HLK-RM08S](#) [A7670G](#) [A7670G-LABE](#) [A7680C-LANY](#) [SIM5360A](#) [A7600C-TXBA](#) [A7670C-LASL](#) [A7670E-LASE](#)