EP-E300 串口服务器用户手册

版本：V1.0.0

EP-E300 手册适用产品

型号	产品简述
EP-E300-4LD	以太网转串口（RS485）服务器
EP-E300-2LD	以太网转串口（RS232）服务器

产品特点

- 超小体积，尺寸与打火机相当
- 双核 240MHz 高性能 MCU
- 自切换 10/100Mbps 网口
- 5~36V 宽电压供电
- 万次 PING 延时均值小于 1ms
- 双看门狗守护
- 每个串口支持双路 Socket 通信
- 支持 TCP/UDP 客户端、服务器，HTTP 客户端，MQTT
- 支持对接阿里云、百度云、移动 OneNET
- 支持 ModbusTCP/RTU 协议转换
- 支持参数批量导出、导入
- 内置网页
- 支持网页升级、远程升级

目　　录

第一章 产品简介 1
1.1 概述 1
1.2 规格参数 2
1.3 尺寸图 3
1.4 硬件接口 4
1.5 快速上手 5
1.5.1 硬件连接 5
1.5.2 使用配置工具搜索设备 5
1.5.3 设置网口设置为 DHCP 6
1.5.4 设置 Socket 参数 6
1.5.5 重启设备 6
1.5.6 测试通信 7
第二章 产品功能 8
2.1 串口 8
2.1.1 串口参数 8
2.1.2 打包机制 9
2.1.3 串口工作模式 9
2.2 网口 10
2.3 数据传输 10
2.3.1 TCP Client/Server 10
2.3.2 UDP Client/Server 11
2.3.3 HTTP Client 12
2.3.4 点对点传输 13
2.3.5 MQTT 14
2.3.6 心跳包、注册包 14
2.4 辅助功能 15
2.4.1 网络搜索 15
2.4.2 网络 AT 16
2.4.3 固件升级 16
2.5 异常处理 17
第三章 参数设置 18
3.1 网页设置 18
3.1.1 状态页面 18
3.1.2 系统管理页面 18
3.1.3 端口设置页面 19
3.2 AT 指令 19
3.2.1 AT 指令格式 19
3.2.2 AT 指令错误码信息 20
3.2.3 AT 指令快速掌握 20
3.2.4 AT 指令详解 21
免责声明 31
联系方式 32

第一章 产品简介

1.1 概述

EP-E300 系列产品为可实现串口与网口互传的数据终端。采用主频高达 240MHz 的双核高性能 MCU，支持宽电压（5~36V）供电，搭载两路可同时工作的串口，每路串口对应两路 Socket，每路 Socket 均支持 TCP/UDP 透传、HTTP Client、MQTT 等协议。

优异的硬件性能、丰富的软件功能可使本产品应用各种行业和复杂的场景，如电网、交通、消防、工业生产、气象环境、农林、矿产等等。

产品典型应用如图 1：

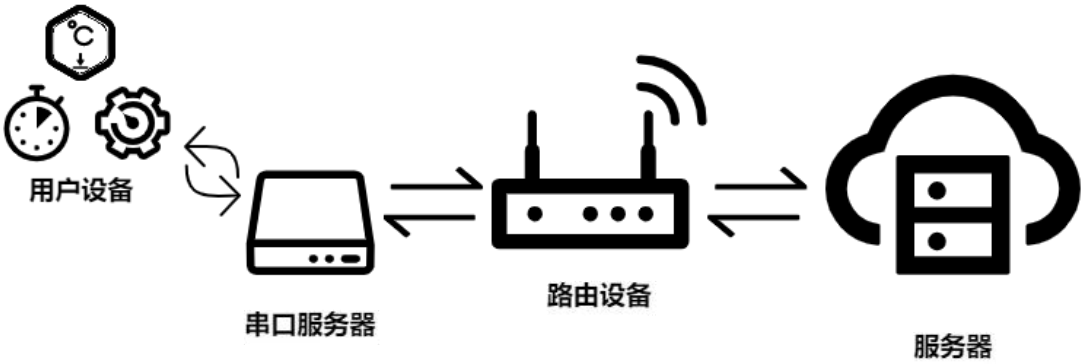


图 1 网络拓扑图

1.2 规格参数

表格 1 规格参数

	条目	参数
电气参数	尺寸	74.6*22.6*16.0mm（不含侧耳及端子）
	工作温度	-40 ~ 85℃
	工作湿度	5% ~ 95%
	供电接口	插拔式端子(2.54mm)
	供电电压	5~36V
	供电电流	65mA@12V
	网口 * 1	RJ45 10/100M 自适应
串口（2 路）	接口类型	串口：RS485 或 RS232 3P 拔插式端子(2.54mm)
	波特率	2400 ~ 460800bps
	校验位	NONE/ODD/EVEN
	数据位	7、8
	停止位	1/2
	硬件流控	NFC 或 485
LED	Power	电源指示
	Work	工作状态指示灯
	TX	串口 Tx 指示灯
	RX	串口 Rx 指示灯
按键	Reload	长按 3 秒以上恢复默认参数
软件功能	网络协议	DHCP/DNS/TCP/UDP/HTTP/ICMP/SSL/TLS/MQTT
	透传协议	TCP Client/Server（Server 限制最多接入 8 个客户端） UDP Client/Server MQTT HTTP
辅助功能	内置网页	支持参数设置，固件升级
	AT 指令	串口 AT/网络 AT
	局域网搜索	搜索设备，可执行网络 AT 命令
	设备维护	串口升级、网络升级、异常处理

1.3 尺寸图

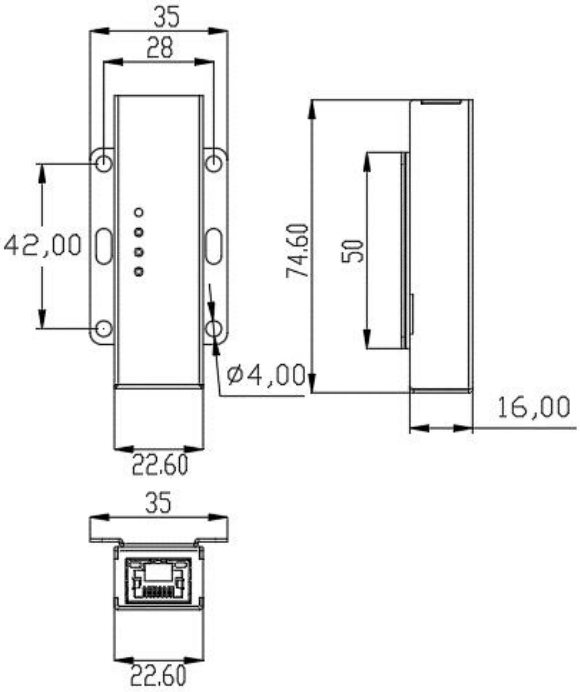


图 2 尺寸图（单位：mm）

1.4 硬件接口

图 3 硬件接口

默认串口参数：

串口	选项	参数
RS232/RS485	波特率	115200
	校验位	NONE
	数据位	8
	停止位	1
	接口	RS232 或 RS485

默认网口参数：

网口	选项	参数
RJ45	DHCP	静态 IP
	IP	192.168.10.8
	网关	192.168.10.1
	子网掩码	255.255.255.0

1.5 快速上手

本节介绍如何快速使用本产品实现基本的透传功能。最终效果实现设备作为 TCP 客户端与电脑上的 TCP 服务互传数据，整体流程分为以下几个步骤：

1. 硬件连接
2. 设置软件配置设备参数
3. 重启设备使新参数生效
4. 测试通信

1.5.1 硬件连接

1. 使用串口线连接设备和电脑
2. 使用网线连接设备和路由器（电脑和设备要接入同一台路由器）
3. 设备上电

1.5.2 使用配置工具搜索设备

打开 `HuayunIOT Ethernet Setup Too` 配置工具，点击**搜索设备**可获取到设备列表，鼠标左键双击设备或通过右键菜单打开参数配置页面。

图 4 搜索设备

1.5.3 设置网口设置为 DHCP

系统页面设置启用 DHCP 并保存

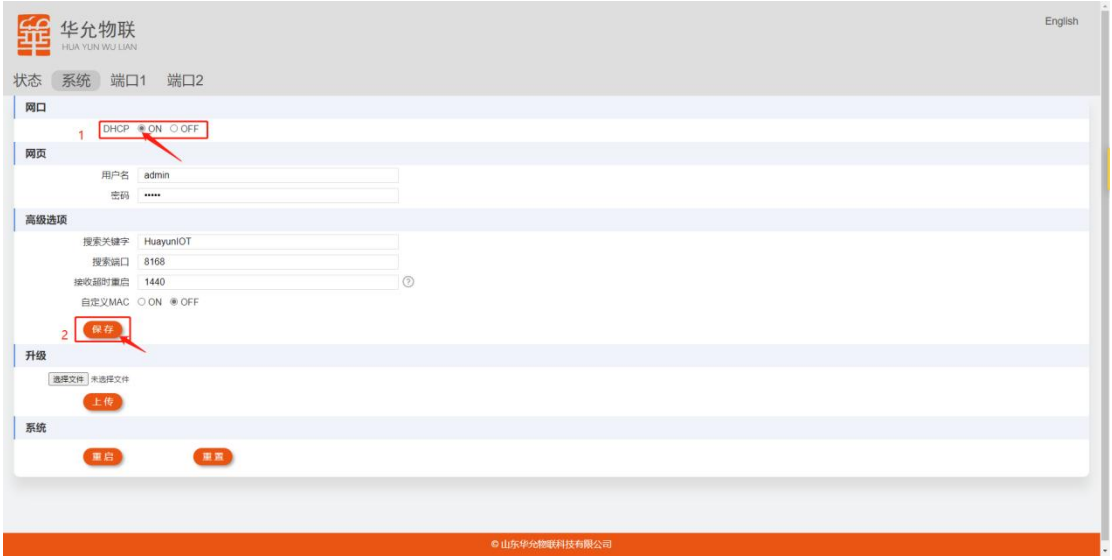


图 5 配置网口

1.5.4 设置 Socket 参数

电脑端作为 TCP 服务器，IP 和端口为 192.168.10.168:7788，设备端需作为 TCP 客户端。进入设备端口页面，修改 SocketA 参数，然后点击页面下方**保存**按钮。

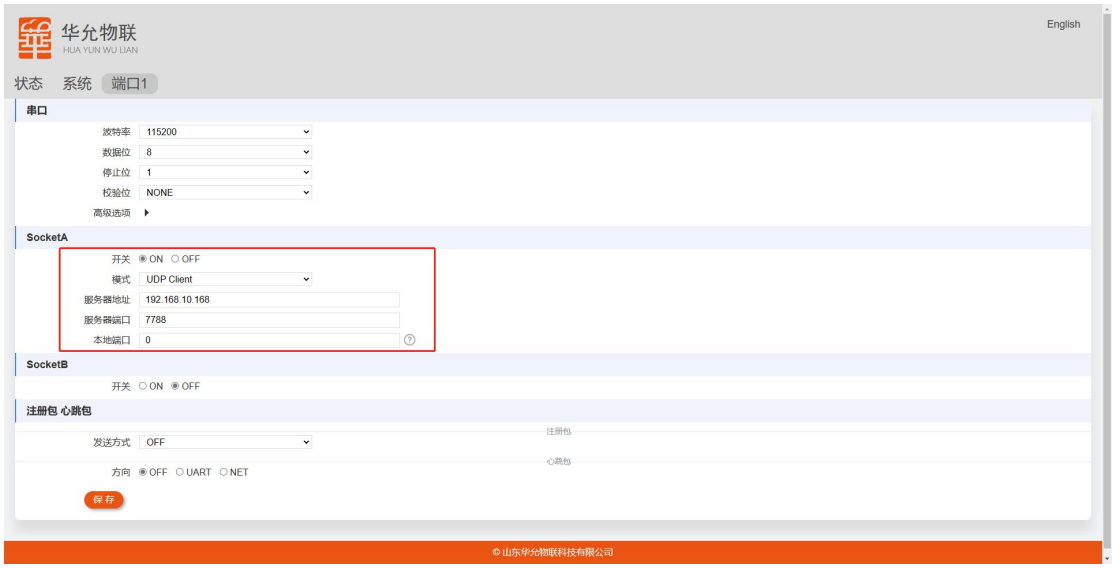


图 6 网络通信参数

1.5.5 重启设备

系统页面重启设备使新参数生效

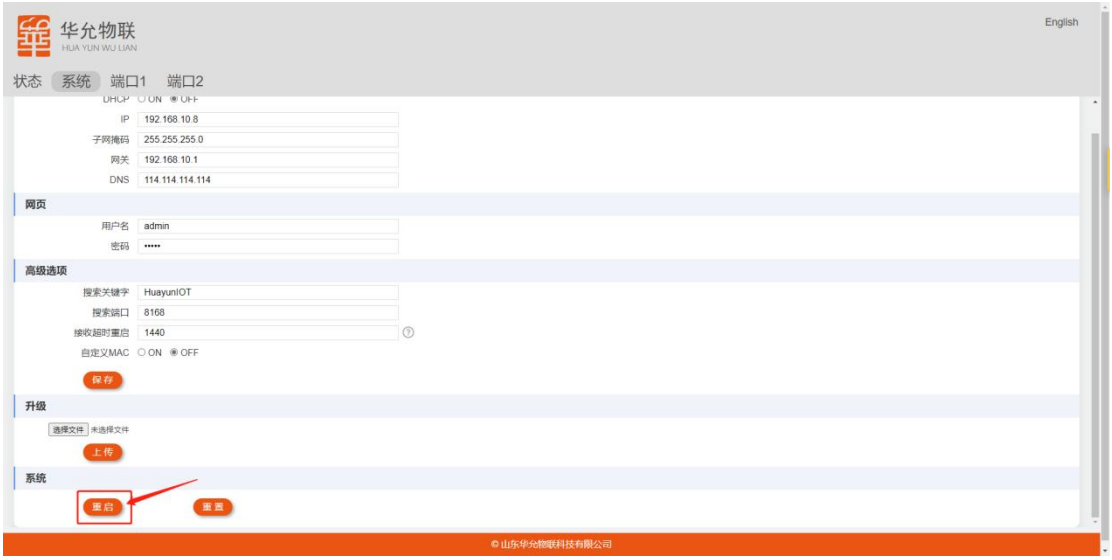


图 7 重启设备

1.5.6 测试通信

设备重启后自动连接电脑的 TCP 服务器，此时串口与网络实现数据互传

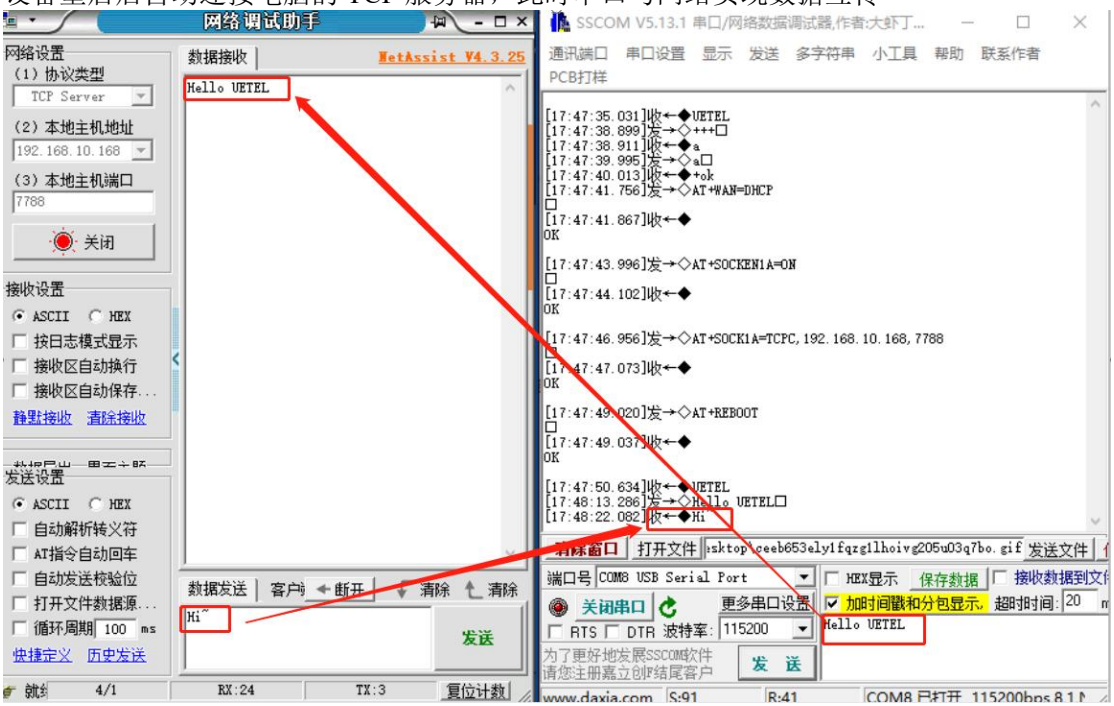


图 8 数据收发测试

第二章 产品功能

产品主要功能框图如下：

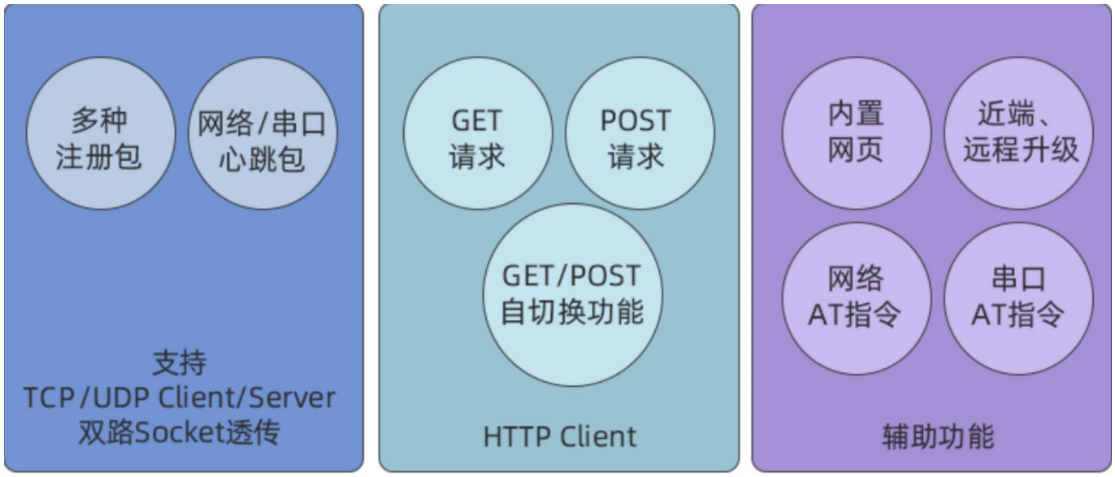


图 9

2.1 串口

本系列产品可选的串口链路个数为 1 路、4 路或者 8 路，每路串口均支持 RS232 和 RS485（为提升效率默认关闭 RS485）接口。各路串口之间功能一致且相互独立，可分别设置串口参数、切换工作模式和进行 Socket 通信。本文仅介绍单路串口的使用，多路串口功能完全一致。下文涉及的 AT 指令中若含有字母 n 则表示该指令设置的参数与设备第 n 号串口有关。

2.1.1 串口参数

表格 2 串口参数

串口	选项	参数
RS232/RS485	工作模式	指令模式
		透传模式（默认）
	波特率	2400～460800bps（默认 115200）
	校验位	NONE/ODD/EVEN（默认 NONE）
	数据位	7/8（默认 8）
	停止位	1/2（默认 1）

2.1.2 打包机制

为提高网络网络传输性能，串口接收数据后先打包成一帧数据再转发到网络。本产品支持通过数据长度或者数据接收间隔两种方式进行打包，两种方式只要满足其一即进行数据转发。两种打包条件如下：

- 数据长度打包：数据长度≥打包长度（默认 1024，支持 1～1024）
- 数据间隔打包：相邻字符间隔≥打包间隔（默认 5ms，支持 1～300ms）

指令示例：`AT+UARTTL1=30,1024` 将打包时间设为 30ms，长度 1024

注意：TCP 协议下数据会有连包现象，如对数据包长度要求严格，请务必在应用层增加拆分包的机制。

2.1.3 串口工作模式

串口支持两种工作模式：

- AT 指令模式：串口收到数据作为指令来执行，网络下发的数据被丢弃。指令模式下可查询、设置参数。
- 透传模式：串口收到数据后通过 Socket 进行转发，是设备开机的默认模式。

用户可通过电脑或者 MCU 的串口通信发送 AT 指令来查询、设置本产品的参数。每条命令行中只能包含一条 AT 指令，单条命令最大 256 字节。设置新参数后自动保存，新参数重启生效 。

2.1.3.1 进入指令模式

进入 AT 模式时序如图 10 其中“UART”表示用户串口设备，“华允终端”表示本系列产品。

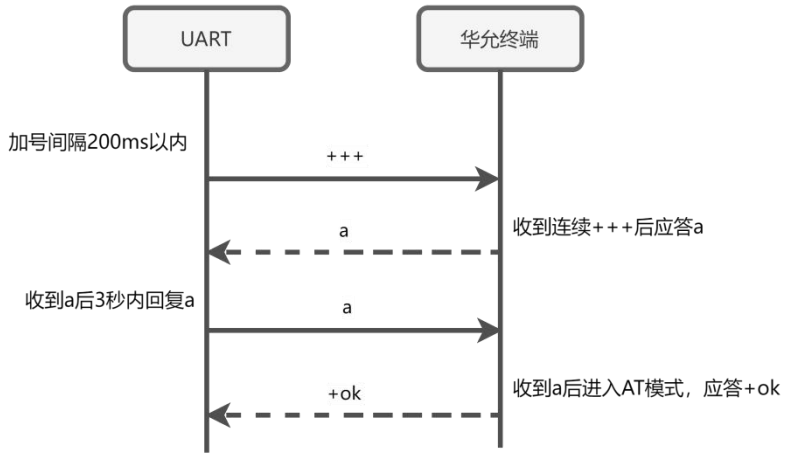


图 10

若发送进入指令的动作中止，期间发送的数据则会转发至 Socket。若用户串口收到了“+ok”表示成功进入了 AT 指令模式，指令模式掉电不保存。

为减少网络下发的数据对上述过程造成干扰，本产品收到 `+++` 后暂停网络数据输出最长持续 3 秒。即便如此‘a’还是有可能与其他数据掺杂在一起，推荐的方法是：

1. 用户设备发送 `+++`
2. 用户设备延时（大于打包时间，小于 3 秒，推荐 500ms）后直接发送 `a`
3. 用户设备判断是否收到了 `+ok` （切记收到+ok 才能收发命令）

有关 AT 指令的详细说明请参阅 *AT 指令*章节。

2.1.3.2 退出指令模式

AT 指令模式下可通过 AT 指令 `AT+EXIT` 或者重启设备切换为透传模式，注意命令以回车换行即转义字符 `\r\n` 结尾。

2.2 网口

网口支持 10/100Mbps 自切换，自适应交叉线直连线。网口默认为静态 IP，参数如下表。

网口参数	
DHCP	关闭 DHCP
IP	192.168.10.8
网关	192.168.10.1
子网掩码	255.255.255.0

无论哪种方式获取 IP，设备的主选 DNS 均为网关，同时设备支持修改备选 DNS。相关设置指令示例如下：

动态 IP： `AT+WAN=DHCP`

静态 IP： `AT+WAN=STATIC,192.168.10.8,192.168.10.1,255.255.255.0` （静态，IP，网关，子网掩码）

备用 DNS： `AT+DNS=114.114.114.114`

2.3 数据传输

本系列产品每路串口支持两条 Socket 链路，每条 Socket 支持的功能完全一样。默认状态下每路串口只启用一条 Socket，当两条 Socket 同时开启时，串口接收到的数据会分别转发到两条 Socket 上；而当两条 Socket 收到数据时也会依次由同一个串口输出。

当 Socket 作为客户端（Client）时，本地端口默认采用随机端口也可以设置为固定端口，需要注意协议相同时不可使用同一端口。

2.3.1 TCP Client/Server

TCP 协议是一种面向连接的可靠传输协议，在对数据完整性要求苛刻的场景下建议使用 TCP 协议。

TCP 协议为 C/S 架构，Server 和 Client 需要先建立连接才能进行数据交互。通信流程为：Server 被动的监听某个端口，而 Client 可主动向服务器发起连接请求，连接建立后双方可互发数据。本系列产品在使用 TCP 协议时会开启 Keepalive 检测功能，可有效避免出现死链接。

2.3.1.1 TCP Client

启用 TCP Client 时，当设备获取到 IP 后 Client 会自动向服务器发起连接，若连接失败或者端口本产品则会自动发起重连，重连间隔为 1 秒。

设置示例，使用串口 3 的 SocketB 的 TCP Client 通信：

1. 开启 Socket3B： `AT+SOCKEN3B=ON`
2. 设置 Socket 参数：
   `AT+SOCK3B=TCPC,192.168.1.16,7788`
3. 重启生效： `AT+REBOOT`

重启后设备开机自动连接 `192.168.1.16:7788` 的 TCP 服务器，连接成功后串口 3 可以与服务器互发数据。

2.3.1.2 TCP Server

设备作为 Server 时会监听用户设定的端口，当收到 Client 接入请求时则与 Client 建立连接。当前每路 Server 限制最多接入 3 个 Client，若 Client 数量超限后则 Server 会将最早接入的 Client 踢下线再接入新 Client。当 Server 接入多个 Client 时，串口接收到的数据会转发给每一个 Client。

设置示例，使用串口 1 的 SocketA 的 TCP Server 通信：

1. 开启 Socket1A： `AT+SOCKEN1A=ON`
2. 设置 Socket 参数： `AT+SOCK1A=TCPS,192.168.1.16,7788`
3. 重启生效： `AT+REBOOT`

重启设备接入 Client 后，串口 1 和 Client 之间可以互发数据。

2.3.2 UDP Client/Server

UDP 协议是一种无连接的传输层协议，提供不可靠信息传送服务，但 UDP 也有其优势：因不需要连接和传输控制，UDP 可实现更加高效的数据传输，也不需要消耗额外的资源维护连接。

严格意义上来讲 UDP 数据收发双方完全对等不需要区分 Client/Server 也不用建立连接。一方只需要知道对方 IP 和端口便可向对方发送数据。为方便用户使用本产品人为的将 UDP 分成了 Client 和 Server 两种模式。

2.3.2.1 UDP Client

UDP Client 模式下目标 IP 和端口始终不变，即可向固定目标发送数据。UDP 模式下建议固定本地端口，否则需要本设备先给对端发一包数据后对端才能发送数据至本设备。

设置示例，使用串口 1 的 SocketB 的 UDP Client 通信，本地端口为 5678，远程 UDP 服务器为 192.168.1.16:7788：

1. 开启 Socket1B： `AT+SOCKEN1B=ON`
2. 设置 Socket 参数： `AT+SOCK1B=UDPC,192.168.1.16,7788,5678`
3. 重启生效： `AT+REBOOT`

2.3.2.2 UDP Server

UDP Server 模式下绑定用户设定的本地端口，当收到某个“Client”数据时 Server 将此“Client”作为发送目标。也就是说 Server 模式下可接收任意多个“Client”发过来的数据，而发送目标则是最近通信的单个“Client”。

设置示例，使用串口 1 的 SocketB 的 UDP Server 通信：

1. 开启 Socket1B： `AT+SOCKEN1B=ON`
2. 设置 Socket 参数： `AT+SOCK1B=UDPS,192.168.1.16,7788`
3. 重启生效： `AT+REBOOT`

重启后设备开启 UDP 监听 7788 端口，设备未收到数据前没有发送目标，当收到 UDP Client 数据后设备会将 Client 作为目标，此时双方可以互传数据。

2.3.3 HTTP Client

HTTP 是一个简单的请求-响应协议，简单来说 HTTP 由 Client 发出请求，然后服务器予以响应，我们平时浏览网页使用的便是 HTTP 协议。

本产品支持的协议版本为 HTTP1.0 和 HTTP1.1，可以实现 HTTP 的 GET/POST 请求。本产品支持 AUTO 模式允许用户在发起 HTTP 请求时灵活的选择请求方式，下面分别举例说明 GET/POST/AUTO 的使用。

测试接口说明：

- GET 接口：http://www.rt-thread.com/service/rt-thread.txt 请求成功返回 rt-thread 简介。
- POS 接口：http://www.rt-thread.com/service/echo 提交数据后自动返回相同的数据。

注：HTTP 与 Socket 共用地址和端口参数，使用 HTTP 之前确保对应的 Socket 功能已使能。地址上的 `HTTP://` 可以省略。

2.3.3.1 GET 请求

1. 切换 HTTPC 模式并设置地址和端口 `AT+SOCKnA=HTPC,www.rt-thread.com,80`
2. 设置包头 `AT+HTPHDn=Connection: keep-alive` （若有多个包头以“|”分割）
3. 设置 GET 请求方式 `AT+HTPREQn=GET`
4. 重启生效 `AT+REBOOT`
5. 串口 n 发送 `/service/rt-thread.txt` 请求成功后服务器返回 rt-thread 简介

注： `AT+HTPURLn` 命令设置的 URL 仅对 `POST` 请求有效

2.3.3.2 POST 请求

1. 切换 HTTPC 模式并设置地址和端口 `AT+SOCKnA=HTPC,www.rt-thread.com,80`
2. 设置 URL `AT+HTPURLn=/service/echo`

3. 设置包头 `AT+HTPHDn=Connection: keep-alive` （若有多个包头以“|”分割）
4. 设置 POST 请求方式 `AT+HTPREQn=POST`
5. 重启生效 `AT+REBOOT`
6. 串口 n 发送任意数据，服务器自动回复相同数据

2.3.3.3 AUTO

AUTO 模式下用户仅需要切换为 HTTPC 模式和设置包头即可，其他参数如服务器地址、端口、URL、请求方式、数据等在发送数据时可灵活改变。

1. 切换 HTTPC 模式并 `AT+SOCKnA=HTPC,null,80` （地址和端口任意设置）
2. 设置包头 `AT+HTPHDn=Connection: keep-alive` （若有多个包头以“|”分割）
3. 设置 AUTO 请求方式 `AT+HTPREQn=AUTO`
4. 重启生效 `AT+REBOOT`

GET 请求：串口 n 发送完整的 URI 即可，例如以下任意格式（非指定端口时默认使用 80 端口连接）

- http://www.rt-thread.com/service/rt-thread.txt
- http://www.rt-thread.com:80/service/rt-thread.txt
- www.rt-thread.com/service/rt-thread.txt

POST 请求格式：URI 与数据 n 以回车换行 `转义字符\r\n` 分割，例如以下任意格式：

- http://www.rt-thread.com/service/echo\r\nTEST
- http://www.rt-thread.com:80/service/echo\r\nTEST
- www.rt-thread.com/service/echo\r\nTEST

2.3.4 点对点传输

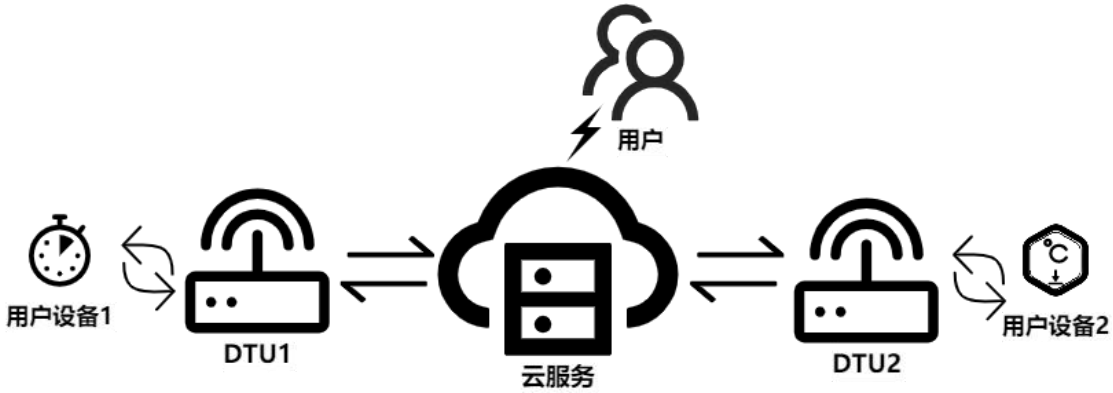


图 7 点对点传输网络拓扑图

点对点可以实现设备间互传输数据，同时用户可在云端对本设备进行数据监控以及远程管理维护。我司点对点功能支持一对一，一对多，多对一以及多对多，此外不同系列的产品比如以太网与 4G DTU 也可以实现点对点传输。

点对点功能流程和示例：

1. 联系我司客服获取对传 ID 号（一机一号），假设设备 1 为 `111` ，设备 2 为 `222`
2. 配置设备 1

- 配置 Socket 为点对点模式： `AT+SOCK1A=EDP,huayuniot,888`
- 配置将点对点目标 ID 设置为设备 2 的 ID： `AT+EDP1=222,OFF`
- 保存重启

3. 操作设备 2，过程同设备 1，需将目标 ID 改成设备 1 的 111，操作完成后设备 1 与设备 2 即可实现数据互发。

2.3.5 MQTT

MQTT 是一种基于发布/订阅（publish/subscribe）模式的“轻量级”通讯协议，使用 TCP/IP 提供网络连接，能够对负载内容实现消息屏蔽传输，开销小，可以有效降低网络流量。

DTU 设备的 MQTT 功能是类透传机制，当与服务器建立连接会自动订阅预设的 topic，同时预设一条发布使用的 topic，DTU 内部处理 MQTT 订阅、发布的事务，用户终端设备只需要接收、发送消息内容即可。

MQTT 参数繁多，用户根据自己需求自行设定，更为详尽的参数请参考相关 AT 指令

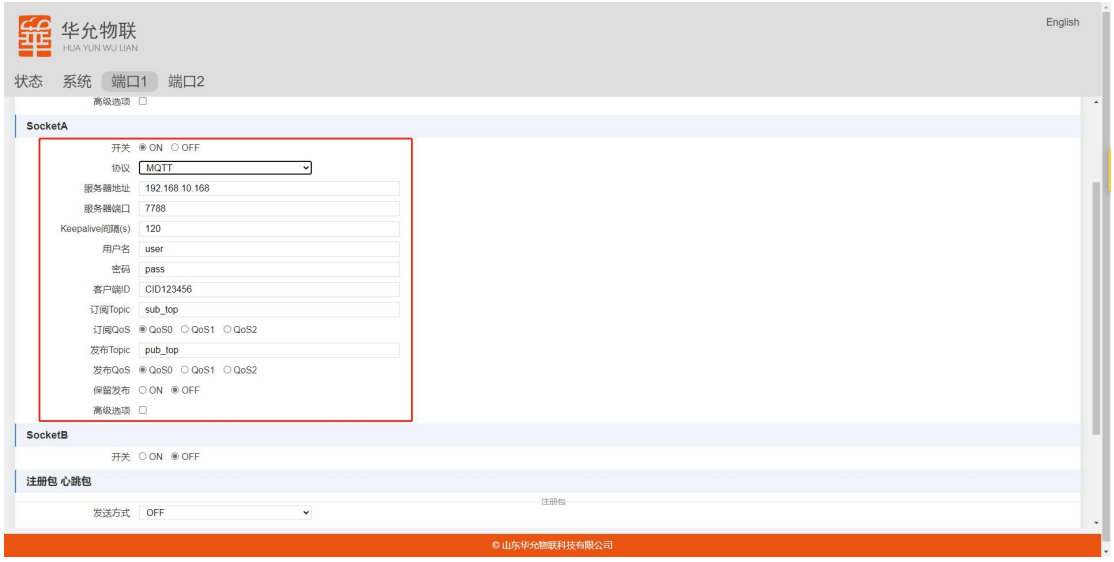


图 8 MQTT 参数设置

2.3.6 心跳包、注册包

心跳包和注册包属于数据透传的附加功能，每路串口均可独立设置心跳包功能和注册包功能，默认关闭。

2.3.6.1 心跳包

心跳包功能是一种应用层上的保活机制，可以定时的将数据发送到网络或者串口以方便用户确认设备是否正常工作。

- 网络心跳包支持 TCP/UDP Client 模式
- 串口心跳包不受网络模式的限制，当设备进入 AT 指令状态时心跳包会暂停发送。

设置示例：开启串口 2 的串口心跳包，间隔 30s 发送一包数据：

1. 串口 2 心跳模式设置为串口： `AT+HEARTMD2=UART`
2. 设置心跳间隔： `AT+HEARTTM2=30`
3. 重启生效： `AT+REBOOT`

2.3.6.2 注册包

注册包的功能是当设备与用户服务器通信时设备主动发一些特定的数据，服务器可据此区分客户端。注册包支持 TCP/UDP Client，可以设置三种发送方式：

- 首次发送：TCP Client 每次连接成功后上报；UDP Client 联网后只上报一次
- 数据携带：作为数据包头和数据同时发送
- 首次发送+数据携带

设置示例：开机串口 2 的 MAC 心跳，仅发送一次：

1. 使用 MAC 注册包： `AT+REGTP2=MAC`
2. 开启注册包，仅第一次发送： `AT+REGMD2=FIRST`
3. 重启生效： `AT+REBOOT`

2.4 辅助功能

2.4.1 网络搜索

网络搜索功能便于用户维护局域网内的设备，用户可以搜索局域网内设备的列表，同时还能向设备发送 AT 指令。

搜索方法是使用 UDP 协议向端口 `8168` 广播发送搜索关键字 `HuayunIOT`，设备收到指定的广播包后会回复自身的 `IP,MAC,型号,版本`。例如：`192.168.1.4,0080E1134527,HY-ETH,V1.0.0`

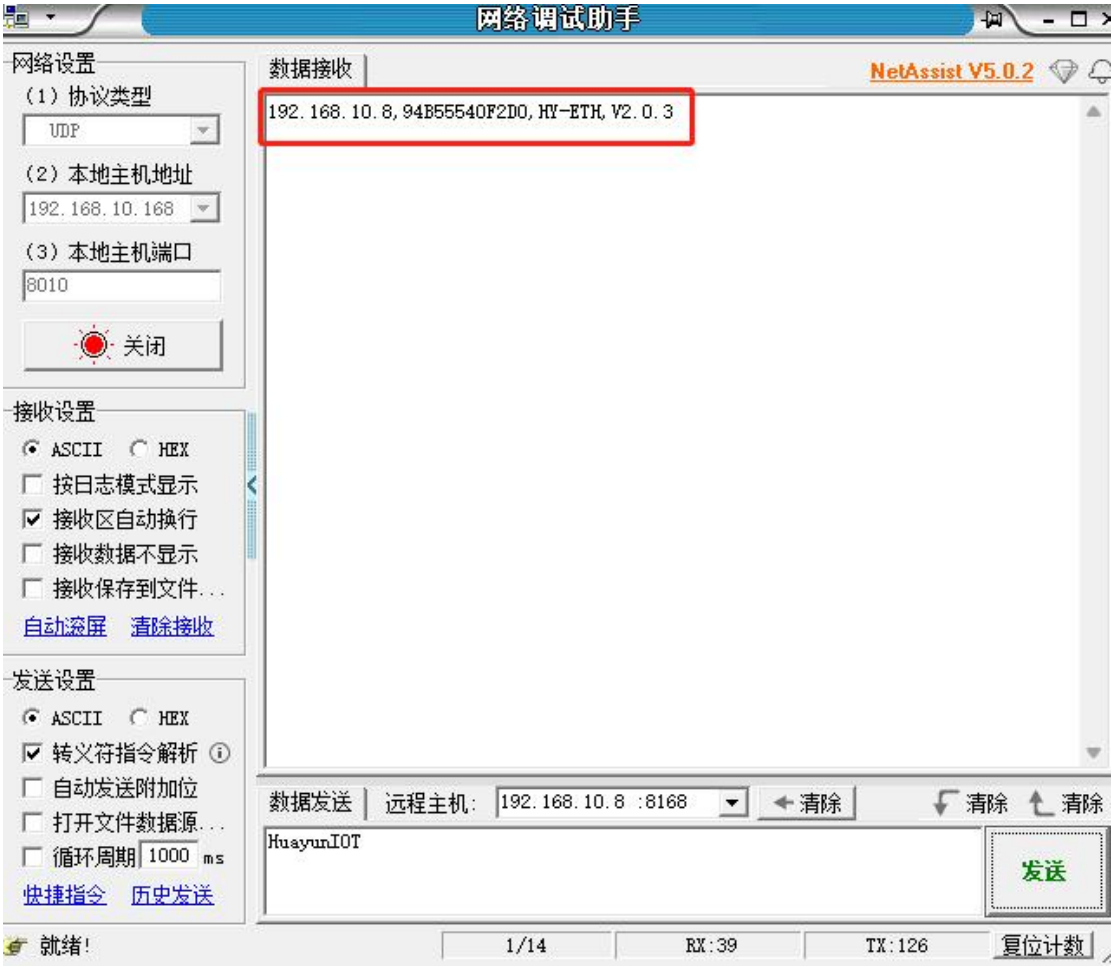


图 10 网络搜索

修改端口和关键字指令： `AT+SEARCH=8168,HuayunIOT`

2.4.2 网络 AT

通过“网络搜索”找到的设备可以在 1 分钟内执行网络 AT 指令，用户向设备的 IP 和端口直接发送 AT 指令即可，设备执行指令后计时刷新。需要注意的是网络 AT 模式不影响串口的工作模式。

有关 AT 指令的详细说明请参阅 *AT 指令*章节。

2.4.3 固件升级

本系列设备支持 3 种方式升级固件，一般情况下升级不影响设备之前的参数配置。

2.4.3.1 串口升级

先按下 `Reload` 按键再开机设备会进入串口升级状态，此时 `Work` 灯快闪，通过我司提供的工具加载固件可进行串口升级。

2.4.3.2 HTTP 升级

设备支持 HTTP 协议升级，用户使用 AT 指令触发升级动作，升级过程如下：

1. 将固件放在 HTTP 服务器的目录下，假设服务器端口为 8080，固件 URI 为 update.xxx.com:8080/firmware.bin
2. 通过串或网络发送 AT 命令：

   `AT+DOWNLOAD=update.xxx.com:8080/firmware.bin`
3. 下载成功设备会应答 OK，否则为 FAIL
4. 下载成功后需要手动重启才能进行升级

2.4.3.3 网页升级

登录设备页面，加载进入 xxx 页面，

2.5 异常处理

设备支持无数据重启功能，当一定时间内收不到网络下发的数据，设备会自动重启。本功能默认开启，时间为 24 小时。设置指令为：

`AT+SOCKRTO=time` （0≤time≤4320，0 表示关闭此功能，单位：min）

第三章 参数设置

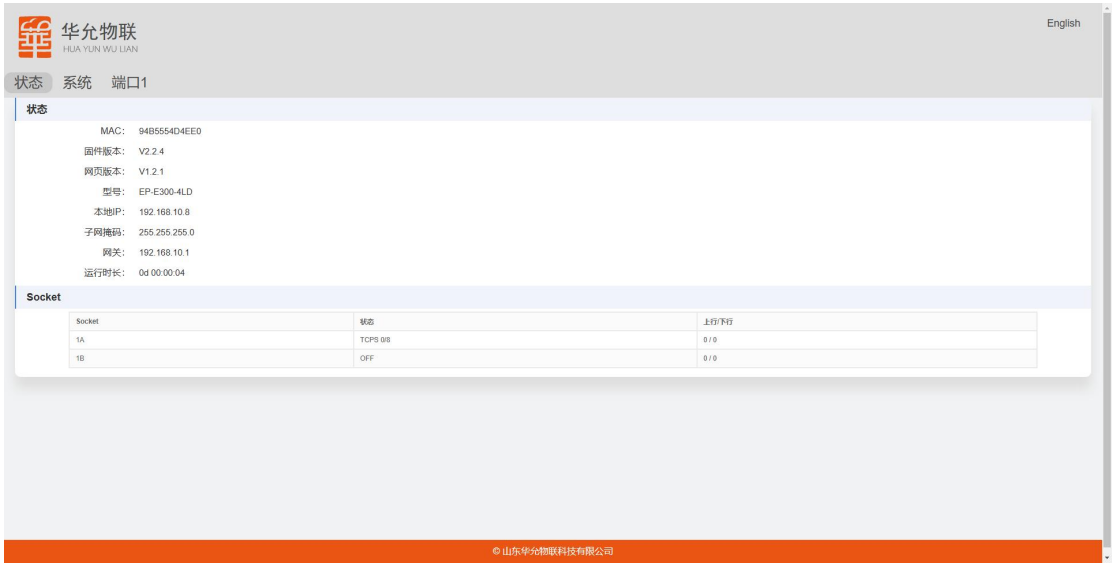
3.1 网页设置

内置网页便于客户直观的对设备进行参数查询、设置以及固件升级。

在浏览器中输入设备的 IP 后可打开设备内置页面，默认用户名为 `admin`，密码为 `admin`

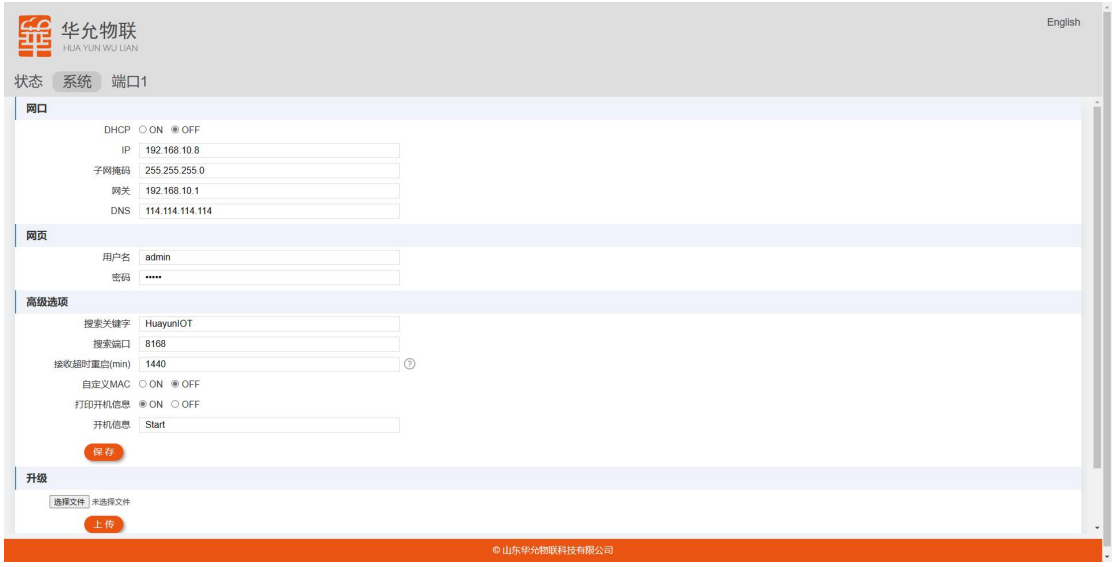
3.1.1 状态页面

状态页为设备基本信息



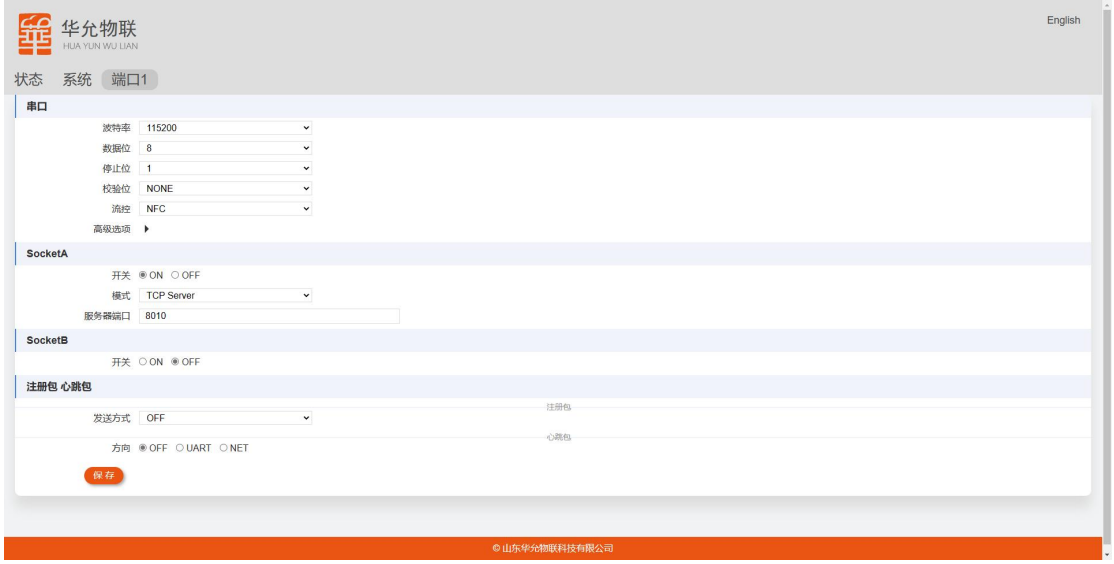
3.1.2 系统管理页面

管理页面为设备系统管理相关，可以配置网页用户名和密码，网络搜索端口、关键字等，同时也支持升级固件。



3.1.3 端口设置页面

端口设置页面为每路串口以及其对应的网络 Socket 参数、心跳包、注册包等。



3.2 AT 指令

AT 指令可用来查询、设置参数，本产品支持串口 AT 和网络 AT。关于串口 AT 模式切换方法请参阅 2.1.3 串口工作模式，网络 AT 进入方式请参阅 *2.4.2 网络 AT*。

设置参数后自动保存，重启生效。

3.2.1 AT 指令格式

AT 指令遵循下列格式：

1. 以 `AT+` 开头，以 `\r\n` 结尾
2. 对指令大小写不敏感，建议使用大写字母

3. 命令行中只能包含一条 AT 指令，单条命令最长 256 字节
4. 多个参数时以英文半角逗号 `,` 分割
5. 要等待前一条命令返回结果后才能发送新命令（命令最大超时时间 5s）

指令有查询、设置、帮助 3 类形式，每类形式的格式如下：

- 查询
  - 发送：`AT+CMD\r\n` 或 `AT+CMD?\r\n`
  - 返回：`\r\n+CMD:value\r\nOK\r\n`
- 设置
  - 发送：`AT+CMD=value1,value2…\r\n`
  - 返回：`\r\nOK\r\n`
- 帮助（对于可设参数的指令才有效）
  - 发送：`AT+CMD=?\r\n`
  - 返回：`\r\n+CMD:(参数 1:范围),(参数 2:范围)…\r\nOK\r\n`

下文最指令的描述如未特殊说明则省略 `\r\n` 。

3.2.2 AT 指令错误码信息

指令执行失败时设备会发出错误码，错误码格式为：`\r\n+ERROR:Error_Code\r\n` 错误码对应的描述见表格 3 错误码

表格 3 错误码

Error Code	错误类型	原因
ARGS	参数不合法	参数长度、大小、格式等不合法
ARGC	参数个数不合法	参数个数不对
CMD_UNKNOWN	未知指令	指令不存在
CMD_FORMAT	格式错误	未以 AT+开头
CMD_LENGTH	长度错误	超过最大命令长度
DEV_MEMORY	内存错误	内存错误
DEV_SAVE	保存失败	保存失败

3.2.3 AT 指令快速掌握

产品庞大数量的 AT 指令给用户带来灵活性的同时也增加了用户对 AT 指令的上手难度，本节介绍几条常用指令可让用户在短时间内掌握大部分指令。

- 切换透传模式：`AT+EXIT`
- 获取指令列表：`AT+LIST`
- 重启设备：`AT+REBOOT`
- 恢复默认参数：`AT+RSTCFG`
- 查询参数格式：`AT+CMD=?` 例如：发送 `AT+ECHO=?` 返回 `+ECHO: (echo:OFF/ON)`

3.2.4 AT 指令详解

本节详细介绍设备的 AT 指令。注意指令含有小写字母 n 时表示指令为第 n 路串口所有，发送指令时请将 n 替换为串口号。

例如查询串口 2 的参数： AT+UART2

指令中有小写字母 s 时表示 SocketA 或 SocketB，发送指令时请将 s 替换为 A 或 B 。

例如设置串口 1 的 SocketA 参数： AT+SOCK1A=TCPC,192.168.1.8,8888

3.2.4.1 LIST 展示命令列表

AT+LIST 展示命令列表	
查询 AT+LIST	+LIST:(命令列表) OK

3.2.4.2 EXIT 退出命令模式

AT+EXIT 退出命令模式	
AT+EXIT	OK

3.2.4.3 MAC 查询设备 MAC

AT+MAC 查询设备 MAC	
AT+MAC	+MAC:<mac> OK
参数	
<mac>	MAC 地址，示例：0080E1134527

3.2.4.4 VER 查询固件版本

AT+VER 查询固件版本	
AT+VER	+MAC:<ver> OK
参数	
<ver>	版本，示例：V1.0.0

3.2.4.5 DEVINFO 查询设备信息

AT+ DEVINFO 查询固件版本	
AT+DEVINFO	+MODULE:<value> +VERSION:<value> +MAC:<value> +DECRYPT:<value> +BUILD:<value> +PRODUCT TIME:<value> +SN:<value>
参数	
<MODULE>	产品型号
<VERSION>	固件版本
<MAC>	MAC
<DECRYPT>	固件解密状态，未解密功能受限
<BUILD>	编译时间
<PROTIME>	生产时间
<SN>	产品序列号

3.2.4.6 REBOOT 重启设备

AT+REBOOT 重启设备	
AT+REBOOT	OK

3.2.4.7 RSTCFG 恢复备份参数并自动重启

AT+RSTCFG 恢复备份参数并自动重启(效果等同于 Reload 按键)	
AT+RSTCFG	OK

3.2.4.8 BKCFG 备份当前运行参数

AT+BKCFG 备份当前运行参数	
AT+BKTCFG	OK

3.2.4.9 CLRCFG 恢复出厂固化参数并自动重启

AT+CLRCFG 恢复厂家固化参数并自动重启	
AT+CLRCFG	OK

设备有 3 个参数分区，三者关系如下：

- 参数区：设备开机该分区读取运行参数；用户查询、修改参数的分区。
- 备份区：用于“参数区”的备份和恢复。
- 固化区：厂家固化参数，用来彻底恢复出厂参数。

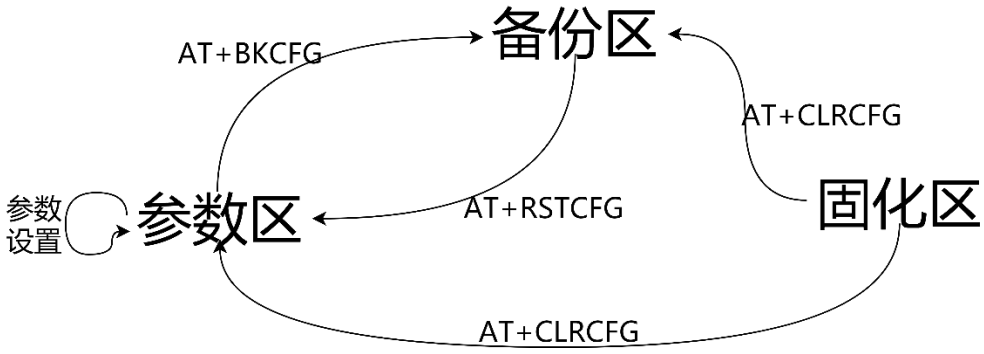


图 12 参数转换

推荐的配置流程：

1. 用户根据自身需求配置参数
2. 将参数通过 BKCFG 保存为默认设置
3. 有需要时通过 RSTCFG 恢复参数

3.2.4.10 SOCKLK 查询 TCP Client 连接状态

AT+SOCKLK 查询 TCP Client 连接状态（UDP 和 TCP Server 查询结果为 OFF）	
查询所有 Socket 连接状态 AT+SOCKLK	+SOCKLK:ns,<state> +SOCKLK:ns,<state> … OK
查询某路 TCP Client 连接状态 n 为串口号，s 为 A 或者 B AT+SOCKLK=ns	+SOCKLK:<state> OK
参数	
<state>	OFF：断开 ON：已连接

3.2.4.11 DOWNLOAD 网络升级

AT+DOWNLOAD HTTP 协议网络升级命令	
AT+DOWNLOAD=uri	<state>

参数	
<uri>	固件地址，不指定端口时默认为 80，例如以下地址： update.xxx.com/firmware.bin 192.168.1.56:8080/firmware.bin
<state>	OK：固件下载成功，需要手动重启进行升级 +ERROR: FAIL：下载失败

3.2.4.12 PING 命令

AT+PING	
AT+PING=addr	+PING:<result>
参数	
<addr>	IP 或者域名
<result>	Network not available：网络未连接 Timeout：超时 Unknown host：未知的地址 Number：ping 延时，单位 ms

以下指令支持 AT+CMD=? 查询帮助信息

3.2.4.13 ECHO 指令回显开关

AT+ECHO 指令回显开关	
AT+ECHO	+ECHO:<state>
AT+ECHO=<state>	OK
参数	
<state>	ON：开启 OFF：关闭（默认值）

3.2.4.14 BOOTINFO 开机启动信息

AT+BOOTINFO 启动信息	
AT+BOOTINFO	+BOOTINFO:<info>
AT+BOOTINFO=<info>	OK
参数	
<info>	1~16 字节字符串，默认值：Start

3.2.4.15 SOCKRTO 网络无数据接收超时重启间隔

AT+SOCKRTO 网络无数据接收超时重启间隔	
AT+SOCKRTO	+SOCKRTO:<time>
AT+SOCKRTO=<time>	OK
参数	
<time>	取值范围：0~65535，默认 1440（24 小时）0 表示关闭此功能 单位：min

3.2.4.16 WEBU 网页用户名和密码

AT+WEBU 网页用户名和密码	
AT+WEBU	+WEBU:<user>,<pwd>
AT+WEBU=<user>,<pwd>	OK
参数	
<user>	用户名，1~16 字节字符串，默认值：admin
<pwd>	密码，1~16 字节字符串，默认值：admin

3.2.4.17 LANG 网页语言

AT+LANG 网页语言	
AT+LANG	+LANG:<lang>
AT+LANG=<lang>	OK
参数	
<lang>	CN：中文（默认） EN：英文

3.2.4.18 WAN 网口参数

AT+WAN 网口参数	
AT+WAN	+WAN:<dhcp_en>,<ip>,<gw>,<mask>
AT+WAN=<dhcp_en>,<ip>,<gw>,<mask>	OK
参数	
<dhcp_en>	DHCP：动态 IP，设置动态 IP 时可以省略后面的参数 STATIC：静态 IP（默认）
<ip>	静态 IP 下的 IP（默认 192.168.10.8）
<gw>	静态 IP 下的网关（默认 192.168.10.1）

<mask>	静态 IP 下的子网掩码（默认 255.255.255.0）

3.2.4.19 DNS 备用 DNS 地址

AT+DNS 备用 DNS 地址	
AT+DNS	+DNS:<addr>
AT+DNS=<addr>	OK
参数	
<addr>	备用 DNS 地址，默认为 114.114.114.114 (首选 DNS 为网关地址)

3.2.4.20 SEARCH 网络搜索端口和关键字

AT+SEARCH 备用 SEARCH 地址	
AT+SEARCH	+SEARCH:<port>,<key>
AT+SEARCH=<port>,<key>	OK
参数	
<port>	端口，取值范围 1~65535，默认 8168
<key>	搜索关键字，1~16 字节

3.2.4.21 UARTn 串口参数

AT+UARTn 查询设置第 n 号串口参数	
AT+UARTn	+UARTn:<baudrate>,<databits>,<stopbits>,<parity>,<fc>
AT+UARTn=<baudrate>,<databits>,<stopbits>,<parity>,<fc>	OK
参数	
<baudrate>	波特率，默认值 115200 可选 2400/4800/9600/19200/38400/57600/115200/230400/460800
<databits>	数据位，7 或 8(默认值) 注：E31xx 系列不支持 7 位数据位
<stopbits>	停止位，1(默认值)或 2
<parity>	校验位：NONE(默认值)/EVEN/ODD
<fc>	流控，NFC(默认值)/485

3.2.4.22 UARTTLn 串口打包间隔和长度

AT+UARTTLn 串口 n 的打包间隔和长度	
AT+UARTTLn	+UARTTLn:<tm>,<len>
AT+UARTTLn=<tm>,<len>	OK
参数	
<tm>	打包间隔，取值范围 1~300ms，默认 5
<len>	打包长度，取值范围 64~1024，默认 1024

3.2.4.23 SOCKENns 串口 n 对应的 SocketA/B 开关

AT+SOCKENns 串口 n 对应的 SocketA/B 开关（n 为串口号，s 为 A 或者 B）	
AT+SOCKENns	+SOCKENns:<state>
AT+SOCKENns:<state>	OK
参数	
<state>	ON：开启，默认仅开启串口 1 的 A 路 Socket OFF：关闭

3.2.4.24 SOCKns Socket 参数

AT+SOCKns Socket 参数（n 为串口号，s 为 A 或者 B）	
AT+SOCKns	+SOCKns:<type>,<addr>,<port>[,localport]
AT+SOCKns:<type>,<addr>,<port>[,localport]	OK
参数	
<type>	协议类型： TCPC：TCP 客户端 TCPS：TCP 服务器 UDPC：UDP 客户端 UDPS：UDP 服务器 HTPC：HTTP 客户端
<addr>	远程服务器地址，64 字节以内，作 Server 时无效
<port>	Client 模式下为远程服务器端口，Server 模式为本地端口 取值范围：1~65535
<localport>	本地端口，此参数可省略。 仅在 TCPC 或 UDPC 下有效，推荐在 TCPC 下使用随机端口，UDPC 时使用固定本地端口。 注：相同协议不可重复使用同一端。 取值范围：0~65535，0 表示使用随机端口（默认值：0）

3.2.4.25 HTPURLn HTTP URL 参数

AT+HTPURLn HTTP URL 参数	
AT+HTPURLn	+HTPURLn:<url>
AT+HTPURLn:<url>	OK
参数	
<url>	1~64 字节字符串，URL 参数仅在 POST 请求模式有效

3.2.4.26 HTPHDn HTTP Header 参数

AT+HTPHDn HTTP Header 参数	
AT+HTPHDn	+HTPHDn:<hd>
AT+HTPHDn:<hd>	OK
参数	
<hd>	HTTP 协议包头，多个包头时以竖线 | 分割，长度范围 1~64 字节

3.2.4.27 HTPFTn HTTP 接收的数据否过滤包头

AT+HTPFTn 串口 n 接收的 HTTP 数据否过滤包头	
AT+HTPFTn	+HTPFTn:<state>
AT+HTPFTn:<state>	OK
参数	
<state>	ON：开启过滤（默认） OFF：关闭过滤

3.2.4.28 HTPREQn HTTP 请求方式

AT+HTPREQn HTTP 请求方式	
AT+HTPREQn	+HTPREQn:<request>
AT+HTPREQn:<request>	OK
参数	
<request>	GET：GET 请求 POST：POST 请求 AUTO：自动模式，发送请求时根据数据自动区分请求方式

3.2.4.29 HTPTOn HTTP 请求超时时间

AT+HTPTOn HTTP 请求超时时间	
AT+HTPTOn	+HTPTOn:<to>
AT+HTPTOn:<to>	OK
参数	
<to>	取值范围 1~30s，默认 6

3.2.4.30 REGTPn 注册包类型

AT+REGTPn 注册包类型	
AT+REGTPn	+REGTPn:<type>
AT+REGTPn:<type>	OK
参数	
<type>	MAC：使用 MAC 地址作为注册包，格式为 6 字节 HEX 数组 CUS：自定义注册包

3.2.4.31 REGMDn 注册包发送模式

AT+REGMDn 注册包类型	
AT+REGMDn	+REGMDn:<mode>
AT+REGMDn:<mode>	OK
参数	
<mode>	OFF：关闭注册包功能（默认） FIRST：TCP Client 连接发送或 UDP Client 第一次联网发送 EVERY：数据携带 ALL：FIRST+EVERY

3.2.4.32 REGDATn 自定义注册包内容

AT+REGDATn 自定义注册包内容	
AT+REGDATn	+REGDATn:<data>
AT+REGDATn:<data>	OK
参数	
<data>	用户自定义注册包，HEX 字符串格式，2~64 字节

3.2.4.33 HEARTMDn 心跳模式

AT+HEARTMDn 心跳模式	
AT+HEARTMDn	+HEARTMDn:<mode>
AT+HEARTMDn=<mode>	OK
参数	
<mode>	OFF：关闭（默认） UART：串口心跳 NET：网络心跳

3.2.4.34 HEARTTMn 心跳间隔

AT+HEARTTMn 心跳间隔	
AT+HEARTTMn	+HEARTTMn:<time>
AT+HEARTTMn:<time>	OK
参数	
<time>	取值范围 1~86400s，默认 60

3.2.4.35 HEARTDATn 心跳包内容

AT+HEARTDATn 心跳内容	
AT+HEARTDATn	+HEARTDATn:<data>
AT+HEARTDATn:<data>	OK
参数	
<data>	用户自定义注册包，HEX 字符串格式，2~64 字节

免责声明

山东华允物联科技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。本公司对产品规格及产品描述做出修改时恕不另行通知。

联系方式

公司：山东华允物联科技有限公司
地址：中国（山东）自由贸易试验区济南片区济南药谷 1 号楼 B 座 18 层
官网：http://www.huayuniot.com