

口红串口服务器 USR-DR13X 系列

说明书



联网找有人，靠谱

可信赖的智慧工业物联网伙伴

产品特点

- 支持 RS232 和 RS485 两种串口信号，可以按需选择不同规格设备，节省项目投入成本；
- 串口波特率可设 600bps 到 230.4Kbps；
- 工作模式可选择 TCP Client、TCP Server、UDP Client、UDP Server、HTTPD Client、MQTT；
- 支持 Modbus 网关功能，轻松实现 ModbusRTU 和 ModbusTCP 协议互转；
- 支持自动轮询 Modbus 指令，毫秒级采集数据，可选透传或者 JSON 协议上报至服务器；
- 支持边缘计算功能，MQTT/TCP 通道下，通过 JSON 格式采集、读写数据；
- 支持 MQTT 功能，可接入标准 MQTT 协议云平台，实现透传功能；
- 支持快速设置连接阿里云，实现数据透传功能
- TCP Server 连接客户端数目可达到 16 个，已连接 IP 可显示，按连接计算发送/接收数据；
- TCP Server 模式下，当连接数量达到最大值时，新连接是否踢掉旧连接可设置；
- 支持 TCP Client 短连接功能，短连接断开时间自定义；
- 支持超时重启（无数据重启）功能，重启时间自定义；
- 自定义注册包和心跳包机制，保证连接真实可靠，杜绝死连接；
- DHCP 功能，能够自动获取 IP；域名解析功能，DNS 服务器地址可自定义设置；
- 10/100Mbps 自适应以太网接口，支持 AUTO-MDIX 网线交叉直连自动切换；
- 支持虚拟串口，配套有人自主开发的 USR-VCOM 软件；
- 多种参数设置方式：WEB 设置、AT 指令设置、串口协议配置、网络协议设置；
- 出厂烧写全球唯一 MAC，同时支持软件和硬件恢复出厂设置；
- 工业级设计，硬件看门狗，超高可靠性。

目 录

产品特点	1
1. 产品概述	4
1.1. 技术参数	4
2. 硬件参数	5
2.1. 产品尺寸	5
2.2. 硬件接口	5
2.3. 指示灯说明	6
2.4. 恢复出厂设置	6
3. 产品功能	6
3.1. 基础功能	6
3.1.1. 当前状态	6
3.1.2. IP 地址/DHCP/DNS	6
3.1.3. 固件升级	8
3.2. 串口设置	8
3.2.1. 串口基本参数	8
3.2.2. 串口成帧机制	9
3.2.3. 虚拟串口工具 VCOM	10
3.3. SOCKET 模式	11
3.3.1. TCP Client 模式	11
3.3.2. TCP Server 模式	13
3.3.3. UDP Client 模式	15
3.3.4. UDP Server 模式	18
3.3.5. Httpd Client 模式	19
3.4. MQTT 网关	21
3.4.1. 通用 MQTT	21
3.4.2. 阿里云	23
3.4.3. 主题发布配置	23
3.4.4. 主题订阅配置	24
3.5. Modbus 网关	25
3.5.1. Modbus TCP/RTU 协议转换	25
3.5.2. Modbus 多主机轮询	27
3.5.3. Modbus 自定义轮询功能	28
3.6. 边缘计算	29
3.6.1. 边缘计算参数配置	29
3.6.2. Json 交互格式	31
3.6.3. MQTT+Json 组包上报示例	33
3.7. 通用功能	36
3.7.1. 心跳包功能	36
3.7.2. 注册包功能	37
3.7.3. Reset 功能	38
3.7.4. INDEX 功能	39
3.7.5. 类 RFC2217 功能	40
3.7.6. 清除缓存数据	41
3.7.7. 有人云功能	42
3.7.8. 自定义网页功能	42
3.7.9. 无数据重启	43

4. 联系方式	44
5. 免责声明	45
6. 更新历史	45

1. 产品概述

USR-DR13X 是有人物联网推出的超小体积导轨式工业级串口服务器，可实现 RS232/RS485 串口与 RJ45 网口之间双向快速的数据传输，内部集成了 TCP/IP 协议栈，拓展 Modbus 网关属性，可直接将设备集成应用到行业场景中，从而快速完成业务框架搭建，节省人力资源投入。

该系列产品采用工业级设计标准，支持宽电压端子供电，RS232/RS485 通信方式支持波特率 600~230400bps，耐高低温，可以 7×24H 稳定运行，无惧任何严苛环境影响。本系列产品经过多年市场严格考验，已经在金融、交通、大型公司网络、工厂制造业等众多行业被使用者认可并大量应用在自己的项目方案中。

USR-DR13X 系列串口服务器，型号包括 USR-DR132 和 USR-DR134，以下简称为 DR132 和 DR134。为了满足更多客户的需求，提供 2 种规格按需选择。

表 1 产品选型

型号	版本	具体描述
USR-DR132	RS232 版本	串口使用 RS232 方式进行通信，按压式端子接线。
USR-DR134	RS485 版本	串口使用 RS485 方式进行通信，按压式端子接线。

1.1. 技术参数

表 2 技术参数表

分类	参数项	数值
供电参数	工作电压	DC5.0~24.0 V
	工作电流	97mA@5V
	供电方式	按压式端子接线
硬件接口	网口规格	RJ45、10/100Mbps、交叉直连自适应
	串口数	DR132: RS232*1 DR134: RS485*1
	串口波特率	600-230.4K (bps)
	数据位	7、8
	校验位	数据位为 7 位时，支持 Odd、Even、Mark、Space; 数据位为 8 位时，支持 None、Odd、Even
	其他	一键恢复出厂设置按键、指示灯 (POW、WORK、COM、LINK)
产品尺寸	规格 (L*W*H)	74*24*22mm (长* 宽* 高，不含端子)
	安装方式	导轨式安装
工作环境	工作温度	-40~85℃
	存储温度	-40~105℃
	工作湿度	5%~95% RH(无凝露)
	存储湿度	5%~95% RH(无凝露)
	防护等级	静电：接触放电 4KV，空气放电 4KV 浪涌：电源 1KV，网口串口 1KV 电快速脉冲群：电源 1KV，网口串口 500V
	网络协议	IP、TCP/UDP、ARP、ICMP、IPV4、HTTP
	工作模式	TCP Server/TCP Client/UDP Server/UDP Client/Httpd Client
	MQTT 模式	支持, 可快速连入阿里云
	Modbus 网关	支持 ModbusTCP/RTU 协议转换、Modbus 多主机轮询、Modbus 自定义轮

软件功能		询
	Modbus 自定义轮询	支持指令自动轮询，最多 5 条，可选透传/JSON 协议上报
	边缘计算	支持，最多 25 点位
	UDP 组播	支持
	短连接	支持
	连接清除缓存	支持
	网络缓存	发送：6Kbyte；接收：6Kbyte
	串口缓存	接收：1Kbyte

2. 硬件参数

2.1. 产品尺寸

产品尺寸：74*24*22mm（长* 宽* 高，不含端子）



图 1. USR-DR13X 尺寸图

2.2. 硬件接口

USR-DR13X 系列采用按压式端子接线，方便快捷。端子接线定义如下所示。

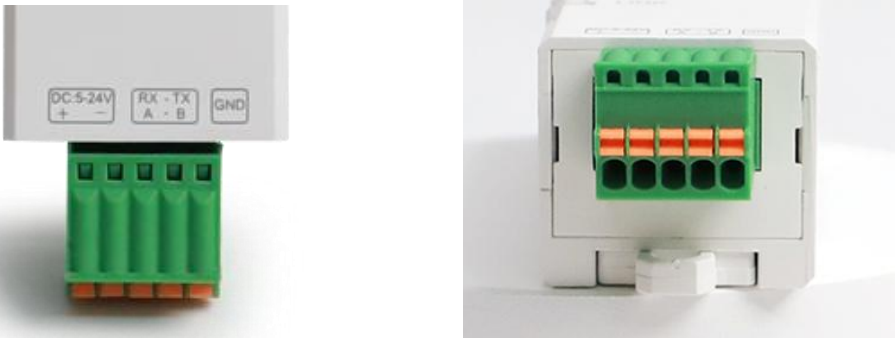


表 3 USR-DR13X 端子接线定义

管脚	名称	信号类型	功能说明
1	DC 5-24V +	P	DC 5-24V 电源正极
2	DC 5-24V -	P	DC 5-24V 电源负极
3	RX/A	I/O	串口数据的接收端
4	TX/B	I/O	串口数据的发送端
5	GND	P	数字地

<说明>

➤ P：表示电源类引脚；I：表示输入引脚；O：表示输出引脚

2.3. 指示灯说明

USR-DR13X 系列具有正面指示灯显示，方便观察工作状态。下面主要描述各种灯的状态和作用。

表 4 指示灯说明

指示灯名称	指示灯功能	状态
POW	电源指示灯	红色 LED，供电后常亮，断电后熄灭
WORK	工作指示灯	绿色 LED，软件正常启动，work 灯每秒闪烁一个周期。如果进入升级状态，work 灯则每 200ms 闪烁一个周期（即快闪）
COM	串口数据指示灯	绿色 LED，串口 TX、RX 有数据收发时快闪
LINK	连接指示灯	绿色 LED，socket A 正常连接后常亮，socket A 断开连接后熄灭（默认开启 link 功能，UDP 模式下常亮）

2.4. 恢复出厂设置

硬件恢复出厂设置：模块能够通过硬件恢复出厂设置，按下 Reload 按键，保持 Reload 按下状态并在 3-15s 后松开，即可硬件恢复出厂设置。

软件恢复出厂设置：通过设置软件或者网络配置协议恢复出厂设置。

AT 指令恢复出厂设置：进入 AT 指令模式，通过指令 AT+CLEAR 恢复出厂。

3. 产品功能

3.1. 基础功能

本小节主要介绍网络基础功能，所有涉及的参数主要用于网络设备和所连接的网络进行正常的通信连接和数据交互。

3.1.1. 当前状态

用户可以通过 DR13X 内置的网页服务器进行参数设置，也可以通过网页查看设备的相关状态。其中网页服务器的端口号可设置，默认为 80。使用方式如下：

- 1) 打开浏览器，在地址栏输入 DR13X 的 IP 地址，如 192.168.0.7（IP 地址和电脑在同一网段）。
- 2) 在弹出的登录界面输入用户名和密码，默认用户名和密码均为 admin，点击“确定”后即可进入内置网页主界面。
- 3) 主界面为状态界面。可以显示当前连接的设备的 IP，以及每条连接发送数据和接收数据的字节数，同时还有服务器收发数据的总字节数统计。状态界面每 10s 自动刷新一次。

3.1.2. IP 地址/DHCP/DNS

点击“网络参数”，可进行 IP 地址、子网掩码、网关、DNS 服务器等网络基本参数的设置。

当前状态

网络参数

端口参数

MQTT网关

边缘网关

点表参数

通用功能

系统参数

模块管理

参数

IP获取方式: 静态IP

本机IP: 192 . 168 . 0 . 7

子网掩码: 255 . 255 . 255 . 0

网关地址: 192 . 168 . 0 . 1

DNS服务器: 208 . 67 . 222 . 222

保存 取消

帮助

- IP地址获取方式: StaticIP表示使用静态的IP地址; DHCP表示动态获取IP地址, 子网掩码, 网关等。
- 本机IP地址: 模块自身的IP地址。
- 子网掩码: 用来划分子网范围大小 (一般是 255.255.255.0) 。
- 网关地址: 访问外网的必经之路 (一般填路由器的IP地址)。
- DNS地址: 域名解析的网关或路由器的地址。

图 2. 网页设置示例

其中 IP 地址可以有两种设置方式，一种是静态 IP，一种是 DHCP。

模块的静态 IP 是需要用户手动设置，需同时写入 IP、子网掩码和网关，静态 IP 适合于需要对 IP 和设备进行统计并且要一一对应的场景。使用静态 IP 需要对每个模块进行设置，并且确保 IP 地址在该局域网内部和其他网络设备不重复。

DHCP 主要作用是从网关主机动态的获得 IP 地址、Gateway 地址、DNS 服务器地址等信息，从而免去设置 IP 地址的繁琐步骤。适用于对 IP 没有什么要求，也不强求要 IP 跟模块一一对应的场景。

表 5 DHCP 静态 IP 优缺点对比

IP 方式	优点	缺点
静态 IP	能够使 IP 和 DR13X 一一对应，无论什么环境都能快速搜索到模块	需要根据接入网络参数不同而调整，需要设置正确的网关，对设置 IP 地址的人有一定的技术要求。
DHCP	直接接入有 IP 分配能力的网络内，就能获得正确的参数进行通讯，可以做到即插即用。	不能与电脑直连使用，必须挂载在具有 IP 分配能力设备下。

注：DR13X 在直连电脑时不能设置为 DHCP，一般电脑不具备 IP 地址分配的能力，如果 DR13X 设置为 DHCP 直连电脑，会导致 DR13X 处于等待分配 IP 地址的状态。设置为 DHCP 时，必须连接一个可以分配 IP 地址的路由器或交换机。

DNS 服务器地址可设，能够在本地域名服务器不完善的情况下实现域名解析，用户也可以根据需求设置特定的 DNS 服务器的地址，DR13X 需要域名解析时就会向设定的 DNS 服务器提交解析请求。使用更加灵活可靠。

静态 IP 模式下，DNS 服务器地址默认为 208.67.222.222，在 DHCP 模式下，DNS 服务器地址为自动获取，域名服务器地址可以设置。

3.1.3. 固件升级

如果模块需要升级更高版本的固件，用户可以向业务人员咨询升级固件。通过升级固件，可以获得更多前沿的运用。DR13X 支持设置软件升级固件。具体升级方法如下：

- 1) 固件升级使用有线网络升级，模块和电脑同时连接路由器或者模块直接连接电脑两种方式均可。升级过程中保持电脑仅有一个 IP。
- 2) 硬件连接好之后，通过设置软件升级固件，IP 和 MAC 地址不要更改，设置好固件路径，然后点击升级固件。



图 3. 固件升级示例

升级时有进度条，当进度条达到 100%时，会提示升级成功。此时点击退出，重新搜索模块。

注意：

- 用配置软件升级时建议关闭电脑的防火墙和杀毒软件（一般在控制面板里）
- 在 work 灯恢复正常闪烁之前请不要进行其他操作及断电

3.2. 串口设置

3.2.1. 串口基本参数

串口基础参数是串口正常通信的保证，主要包括：波特率，数据位，停止位，校验位。

- 波特率：串口通讯速率，支持自定义数值，可设置范围为：600~230.4Kbps；
- 数据位：数据位的长度；范围为 7~8；
- 停止位：可设置范围为：1~2；
- 校验位：数据位为 8 支持 NONE、Odd、Even；数据位为 7，支持 Odd、Even、Mark、Space。



图 4. 端口参数

3.2.2. 串口成帧机制

由于网络端的数据都是以数据帧为单位进行数据传输的，因此需要经串口的数据组成帧数据发送到网络端，这样可以更加高效快捷的传输数据。DR13X 在数据透传过程中，按照设定的打包长度和打包时间，对串口数据进行打包。

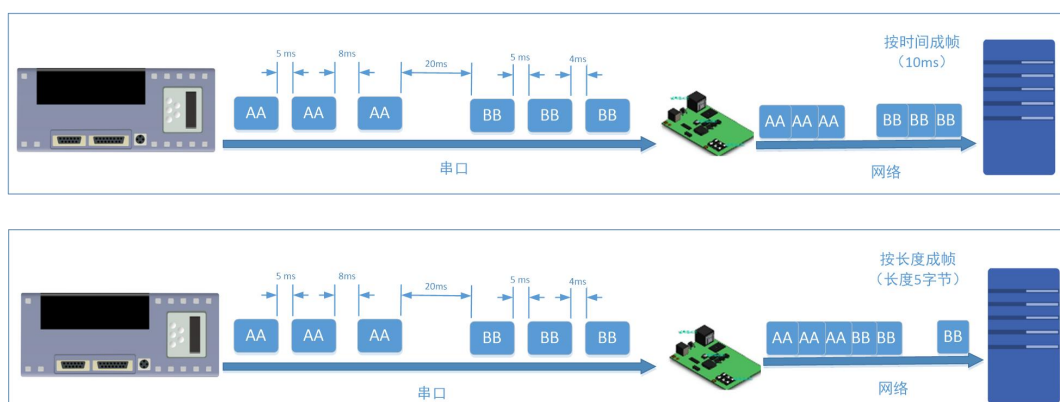


图 5. 端口参数

打包时间和打包长度判定举例：

(1) 打包时间和打包长度均不为 0，则满足打包时间或打包长度任意一条，则打包发送。

举例：打包时间设置 10ms，打包长度为 1024。

当串口收到数据，如果串口接收数据间隔时间超过 10ms，或者数据长度大于 1024，DR13X 打包发到网络中去。

(2) 若打包时间和打包长度其中有一项为 0，打包规则只有不为零的一项有效。

(3) 打包时间设置 0ms，打包长度为 0。

当打包时间设置成 0ms 时，DR13X 执行默认打包时间，即串口收到数据间隔时间超过发送四个字节的打包时间时，DR13X 便打包发送出去。比如波特率为 115200 时，四个字节的打包时间为：

$T=0.4\text{ms}$ ，当计算数值小于 1ms 时，打包时间按照 1ms 计算。

$$T = \frac{1}{\text{波特率}} * 10 * 4$$

注意：当 DR13X 从网络端接收数据，然后再发送到串口端时，由于串口速度的限制，需要用户控制好发送流量，否则会出现串口端数据溢出的问题，所以送网络透传数据到串口时，需要计算数据流量问题。

计算举例：

网络数据每隔 n 秒，发送 m 个字节数据。检查是否有可能溢出的方法为：（假设网络情况良好，而且网络数据传输时间忽略不计）

如果不出现溢出情况，在 n 秒内必须传输完毕 m 个字节的数据，则 M 字节数据 传输时间：

$$T = \frac{1}{\text{波特率}} * 10 * m$$

如果 n>2T 表明数据不会溢出，DR13X 服务器能够正常工作，如果波特率在 9600 以下，保持 n>T 即可。



图 6. 串口打包参数配置

3. 2. 3. 虚拟串口工具 VCOM

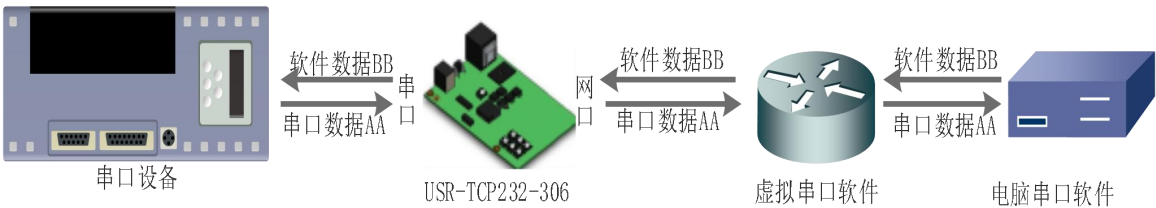


图 7. VCOM 应用示意图

通过 VCOM 应用模式，可以解决传统设备 PC 端软件为串口方式通讯的问题，通过虚拟串口软件，接收指定串口的数据，然后把串口的数据以网络的形式发送出去，进而实现传统串口设备联网通讯的目的，方便用户使用，实现用户不用修改用户的软件进而实现远程数据传输和通讯的能力。

本节重点讲解 DR13X 怎样和虚拟串口建立连接：

- (1) 设置 DR13X 为 TCP Server 模式(DR13X 设置为 Server，方便用户更换电脑，依然能够连接设备)。
- (2) 手动设置虚拟串口软件方式和 DR13X 连接：
 - 打开虚拟串口，设置点击添加串口，选择串口号为 COM2(选择 COM 号一定要避开已经存在的 COM 口)。网络协议选择 TCP Client，目标域名/IP 设置成 DR13X 的 IP，目标端口号设置为 DR13X 的端口号，备注填写自己的设备名称。
 - 点击确定，观察连接是否建立，显示连接建立表明已建好连接，然后就能双向透传数据。



图 8. 应用模式添加串口

手动设置虚拟串口软件方式和 DR13X 建立链接。虚拟串口更多用法介绍和实例：

USR-VCOM 软件下载地址 <http://www.usr.cn/Download/31.html>

3.3. SOCKET 模式

DR13X 的 Socket 工作模式共分为 TCP Client、TCP Server、UDP Client、UDP Server、Httpd Client、MQTT。工作模式设置界面如下：

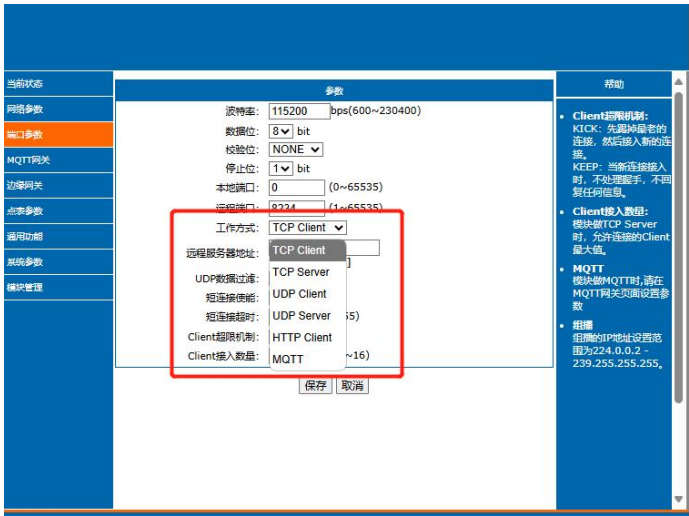


图 9. 工作模式设置示例

3.3.1. TCP Client 模式

TCP Client 为 TCP 网络服务提供客户端连接。主动向服务器发起连接请求并建立连接，用于实现串口数据和服务器数据的交互。通常用于设备与服务器之间的数据交互，是最常用的联网通信方式。

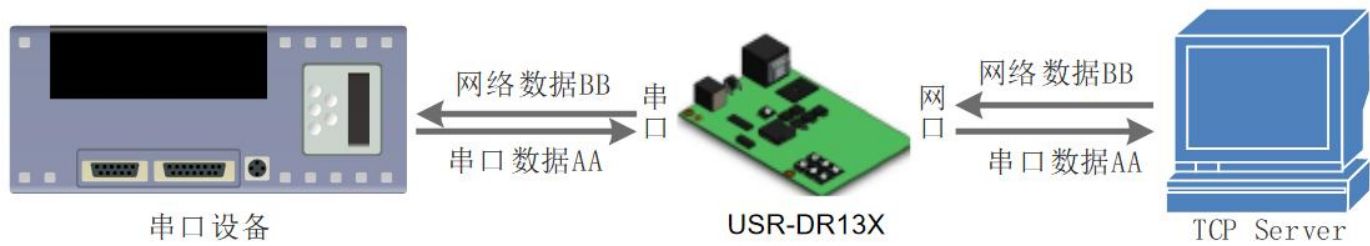


图 10. TCP Client 模式说明

- 1) TCP Client 为 TCP 网络服务提供客户端连接。主动发起连接并连接服务器，用于实现串口数据和服务器数据的交互。根据 TCP 协议的相关规定，TCP Client 是有连接和断开的区别，从而保证数据的可靠交换。
- 2) DR13X 做 TCP Client，需要连接 TCP Server。需要关注的参数：目标 IP/域名和目标端口号，目标 IP 可以是本地同一局域网的设备，也可以是不同局域网的 IP 地址或者跨公网的 IP，如果连接跨公网的服务器，那么要求服务器具有公网 IP 或者是域名。
- 3) DR13X 做 TCP Client 会主动连接目标 IP 的目标端口，不会接受其他连接请求。
- 4) DR13X 做 TCP Client，建议把 DR13X 的本地端口号设置成 0，这样 DR13X 就能以随机端口号访问服务器，可以解决因服务器判断连接状态异常，屏蔽 DR13X 发出的重连请求而导致重连失败的情况。
- 5) 本模式具备主动识别连接异常的功能，当连接建立后，会有以大约 15s 的间隔发送的 KeepAlive 保活探查包，如果连接有异常中断等情况，则会被立即检测到，并促使 DR13X 断开原先的连接并重连。
- 6) 在同一局域网下，如果 DR13X 设为静态 IP，请保持 DR13X 的 IP 和网关在同一网段，并且正确设置网关 IP，否则将不能正常通信。

TCP Client 通讯实例：

- 1) 设置 DR13X 的工作方式为 TCP Client。可以通过网页设置，设置目标 IP 和远程端口号。需要连接的 PC 端为 TCP Server，该 PC 的 IP 为 192.168.0.95，监听的端口号为 20108。则设置目标 IP 192.168.0.95，远程端口号 20108。设置完成后点击“保存”，并重启模块。



图 11. TCP Client 网页设置

- 2) 打开网络调试助手，设置需要连接 PC 端为 TCP Server，该 PC 的 IP 为 192.168.0.95，监听的端口号为 20108。
- 3) 打开串口调试助手，设置正确的串口参数。点击打开串口，点击发送，接收到双向透传的数据。

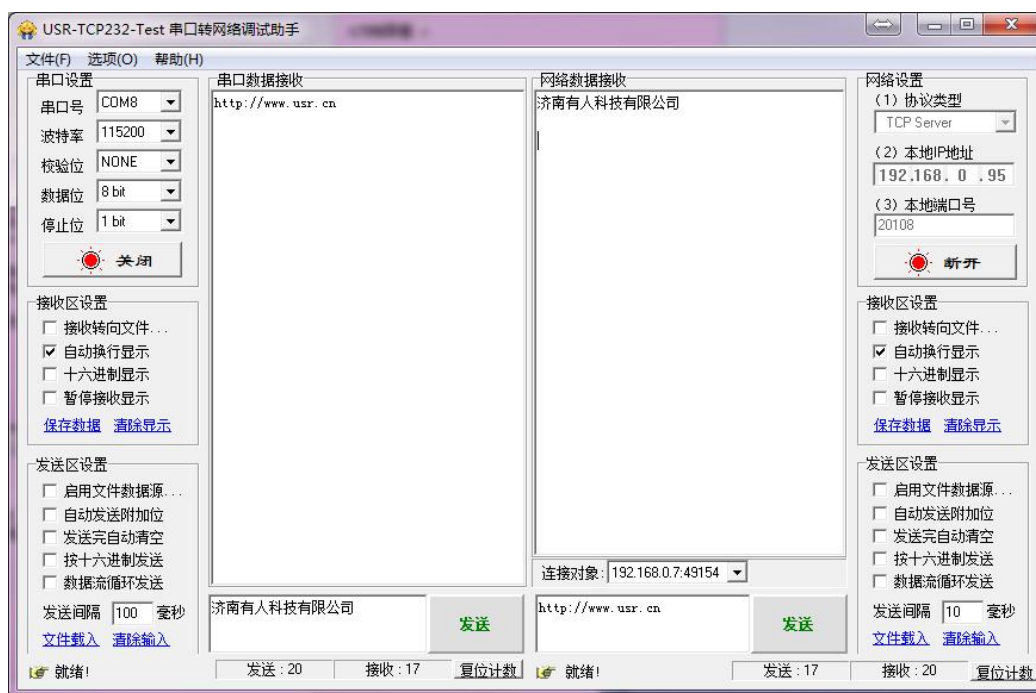


图 12. TCP Client 软件设置

3.3.1.1. 短连接

TCP 短连接的使用主要是为了节省服务器资源，一般应用于多点对一点的场景。使用短连接，可以保证存在的连接都是有用的连接，不需要额外的控制手段进行筛选。

TCP 短连接功能应用于 TCP Client 模式下，开启短连接功能后，发送信息，如果在设定的时间内串口或网口再无数据接收，将会自动断开连接。该功能默认关闭，断开时间可在功能开启后设置，设置范围为 2~255s，默认为 3s。设置示意图如下：

当前状态	参数	帮助
网络参数	波特率: 115200 bps(600~230400)	Client超限机制: KICK: 先踢掉最老的连接, 然后接入新的连接。 KEEP: 当新连接接入时, 不处理握手, 不回复任何信息。 Client接入数量: 模块做TCP Server时, 允许连接的Client最大值。 MQTT 模块做MQTT时, 请在MQTT网关页面设置参数。 组播 组播的IP地址设置范围为224.0.0.2 - 239.255.255.255。
端口参数	数据位: 8 bit	
MQTT网关	校验位: NONE	
边缘网关	停止位: 1 bit	
点表参数	本地端口: 0 (0~65535)	
通用功能	远程端口: 20108 (1~65535)	
系统参数	工作方式: TCP Client	
模块管理	远程服务器地址: 192.168.0.95 [192.168.0.95]	
	UDP数据过滤: Close	
	短连接使能: Close	
	短连接超时: 3 (2~255)	
	Client超限机制: KICK	
	Client接入数量: 4 (1~16)	
	保存 取消	

图 13. 短连接设置图示

3.3.2. TCP Server 模式

TCP Server 即 TCP 服务器。在 TCP Server 模式下, DR13X 监听本机端口, 有连接请求发来时接受并建立连接进行数据通信, 当 DR13X 串口收到数据后会同时将数据发送给所有与 DR13X 建立连接的客户端设备, 同样 TCP Server 模式也有 KeepAlive 功能用于实时监测连接的完整。

通常用于局域网内与 TCP 客户端的通信。适合于局域网内没有服务器并且有多台电脑或是手机向服务器请求数据的场景。同 TCP Client 一样有连接和断开的区别, 以保证数据的可靠交换。本模式支持有人自主的同步波特率功能 (RFC2217) 功能。

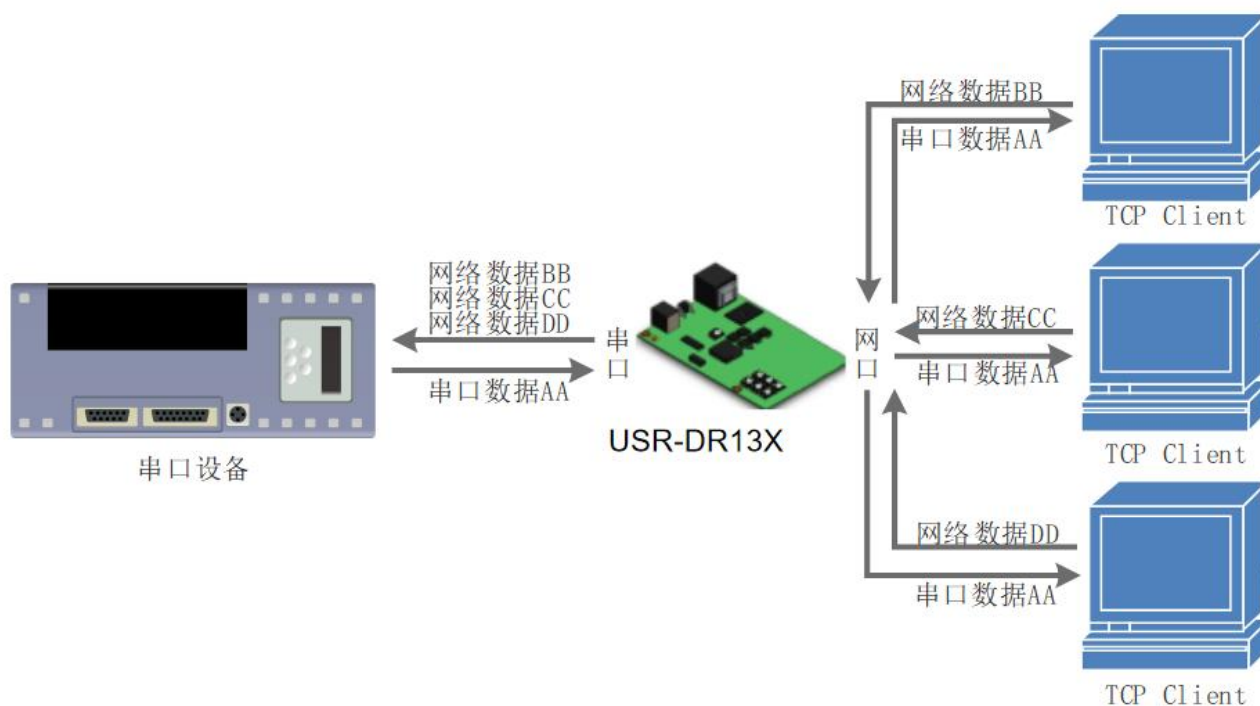


图 14. TCP Server 模式说明

通讯实例:

- 1) 设置 DR13X 工作方式为 TCP Server, 本地端口号为 23。
- 2) 打开网络调试助手, 模式为 TCP Client。写入 DR13X 的 IP 和端口号, 点击连接, 进行透传测试。
- 3) 打开串口调试助手, 设置正确的串口参数。点击打开串口, 点击发送, 接收到双向透传的数据。

当前状态	参数	帮助
网络参数	波特率: 115200 bps(600~230400)	Client超限制制: KICK: 先踢掉最老的连接, 然后接入新的连接。 KEEP: 当新连接接入时, 不处理握手, 不回复任何信息。 Client接入数量: 模块做TCP Server时, 允许连接的Client最大值。 MQTT 模块做MQTT时, 请在MQTT网关页面设置参数 组播 组播的IP地址设置范围为224.0.0.2 - 239.255.255.255。
端口参数	数据位: 8 bit	
MQTT网关	校验位: NONE	
边缘网关	停止位: 1 bit	
点表参数	本地端口: 23 (0~65535)	
通用功能	远程端口: 20108 (1~65535)	
系统参数	工作方式: TCP Server	
模块管理	远程服务器地址: 192.168.0.95 [192.168.0.95]	
	UDP数据过滤: Close	
	短连接使能: Close	
	短连接超时: 3 (2~255)	
	Client超限制制: KICK	
	Client接入数量: 4 (1~16)	
	保存 取消	

图 15. TCP Server 网页设置



图 16. TCP Server 测试截图

3.3.2.1. 自定义 Client 连接数

DR13X 做 TCP Server 时, 能够连接 Client 的数量为最多 16 个, 默认值为 4 个, 最大值可根据客户需求自行设置, 方便客户使用。当连接 Client 的数量超过 4 个时, 收发同时进行的情况下, 总体数据流量需控制在 2.5KB/s 以内 (即每条连接每秒发送信息不超过 200 个字节)。

当 Client 连接数量大于用户设定的最大值时, 默认新连接会替换掉旧连接, 也可设置为新连接不能踢掉旧连接。

超出连接数量的处理:

KICK: 踢掉旧的连接, 接入新的连接。

KEEP: 保持现有的连接, 踢掉新接入的连接。

通过网页设置示意图如下:

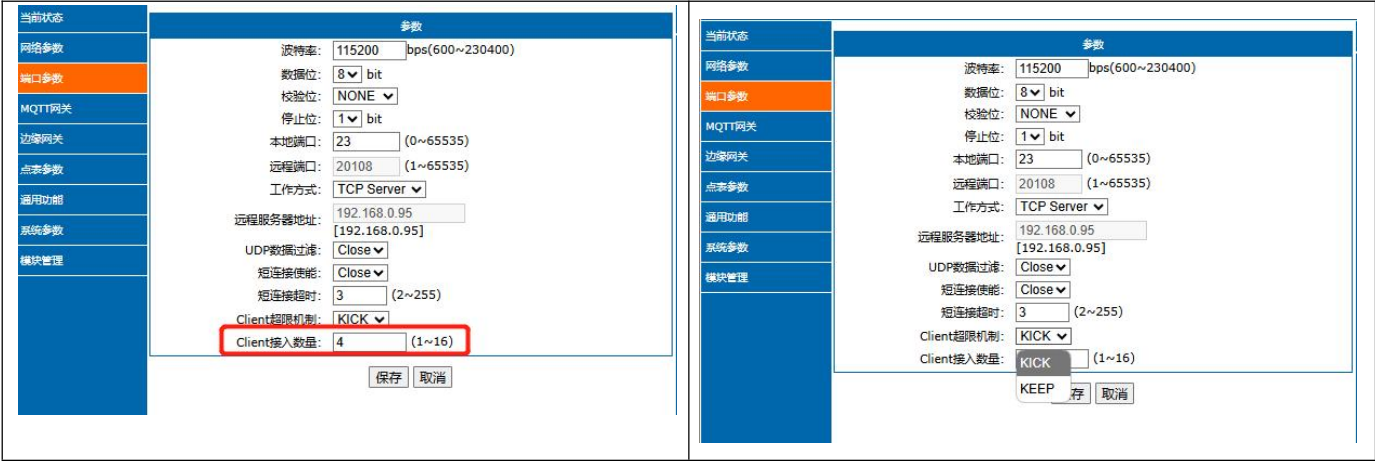


图 17. 设置示意图

3.3.3. UDP Client 模式

本工作模式从属于 UDP 协议。UDP Client 一种无连接的传输协议，提供面向事务的简单不可靠信息传送服务，没有连接的建立和断开，只需要制定 IP 和端口即可将数据发向对方。通常用于对丢包率没有要求，数据包小且发送频率较快，并且数据要传向指定的 IP 的数据传输场景。

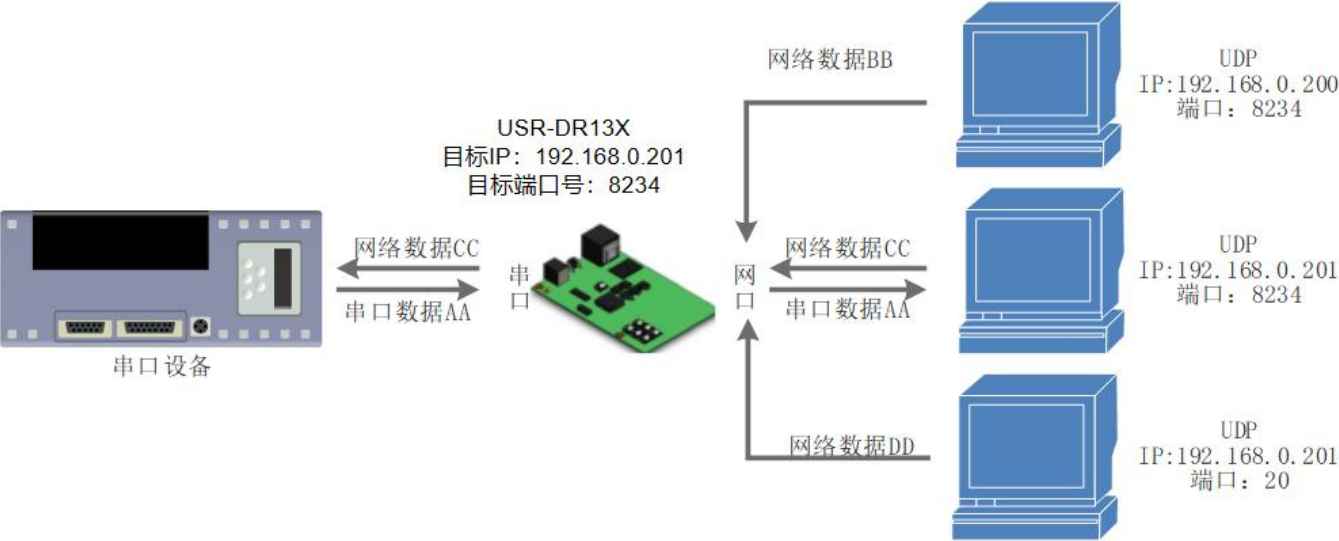


图 18. UDP Client 模式说明

- 1) UDP Client 模式下，DR13X 串口到网络方向只会与目标 IP 的目标端口通讯。
- 2) UDP Client 模式支持数据过滤功能，该功能需要手动开启，开启后能根据配置选择接收来自全网段数据、段内数据、指定 IP 透传向串口。
- 3) 在本模式下，目标地址设置为 255.255.255.255，则可以达到 UDP 全网段广播的效果；同时也可以接收广播数据；支持网段内的广播，比如 xxx.xxx.xxx.255 的广播方式。**支持组播。**

通讯案例：

- 1) 打开网络连接助手，建立 UDP，该 PC 的 IP 为 192.168.0.95，监听的端口号为 23；
- 2) 设置 DR13X 为 UDP Client 模式，远程服务器地址为 192.168.0.95，目标端口为 23；
- 3) 必须先点击串口发送，接收到数据后，测试软件的目标 IP 和目标端口号变为 DR13X 的 IP 和端口号。然后点击网络发送，发送数据到串口。

当前状态	参数	帮助
网络参数	波特率: 115200 bps(600~230400)	Client超限制制: KICK: 先踢掉最老的连接, 然后接入新的连接。 KEEP: 当前连接接入时, 不处理握手, 不回复任何信息。 Client接入数量: 模块做TCP Server时, 允许连接的Client最大值。 MQTT 模块做MQTT时, 请在MQTT网页页面设置参数 组播 组播的IP地址设置范围为224.0.0.2 - 239.255.255.255。
端口参数	数据位: 8 bit	
MQTT网关	校验位: NONE	
边缘网关	停止位: 1 bit	
点表参数	本地端口: 0 (0~65535)	
通用功能	远程端口: 23 (1~65535)	
系统参数	工作方式: UDP Client	
模块管理	组播设置: None	
	远程服务器地址: 192.168.0.95 [192.168.0.95]	
	UDP数据过滤: Close	
	短连接使能: Close	
	短连接超时: 3 (2~255)	
	Client超限制制: KICK	
	Client接入数量: 4 (1~16)	

保存 取消

图 19. UDP Client 网页设置

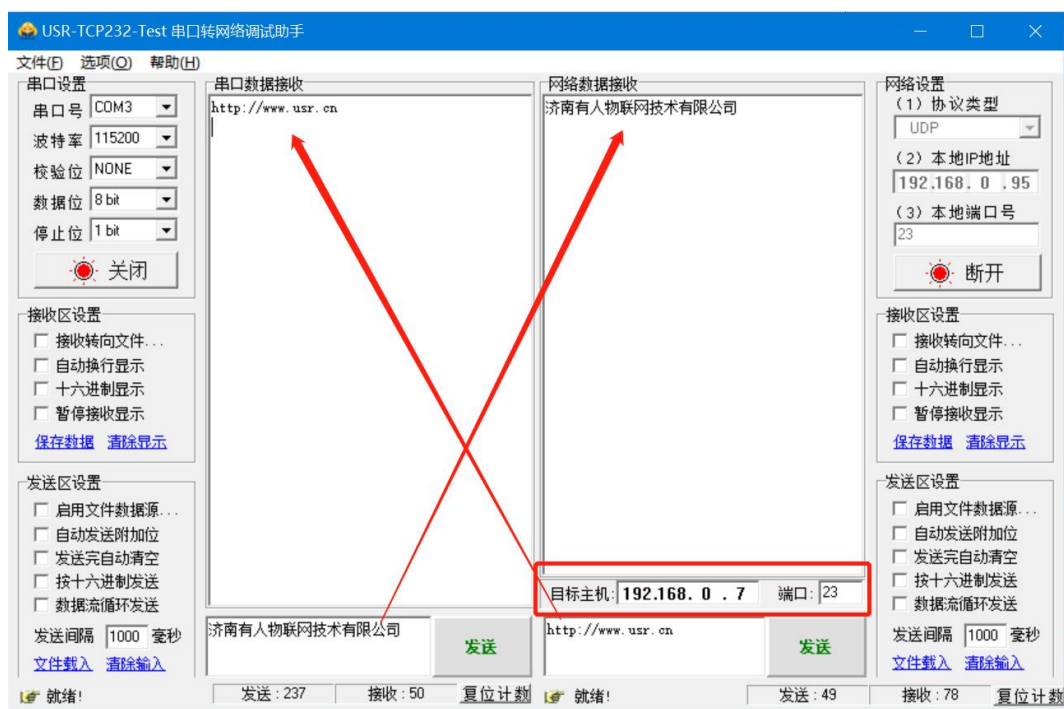


图 20. UDP Client 测试截图

3.3.3.1. UDP 组播

使用组播可以实现数据发送者和接收者之间一对多点的连接方式, 多个接收者加入同一个组播组, 共享同一个 IP 地址, 同时组播组中的成员是动态的, 某个成员的加入和退出并不影响原有的组播组。组播组的有效地址范围是 224.0.0.2 - 239.255.255.255。

通讯案例:

- 1) UDP client 模式下, 打开组播。设置组播 IP 地址、远程端口、本地端口

当前状态	参数	帮助
网络参数	波特率: 115200 bps(600~230400)	Client超限机制: KICK: 先踢掉最老的连接, 然后接入新的连接。 KEEP: 当前连接接入时, 不处理握手, 不回复任何信息。 Client接入数量: 模块做TCP Server时, 允许连接的Client最大值。 MQTT 模块做MQTT时, 请在MQTT网关页面设置参数 组播 组播的IP地址设置范围为224.0.0.2 - 239.255.255.255。
端口参数	数据位: 8 bit	
MQTT网关	校验位: NONE	
边缘网关	停止位: 1 bit	
点表参数	本地端口: 123 (0~65535)	
通用功能	远程端口: 456 (1~65535)	
系统参数	工作方式: UDP Client	
模块管理	组播设置: 组播	
	远程服务器地址: 224.0.0.15 [192.168.0.95]	
	UDP数据过滤: Close	
	短连接使能: Close	
	短连接超时: 3 (2~255)	
	Client超限机制: KICK	
	Client接入数量: 4 (1~16)	
	保存 取消	

图 21. 组播参数设置

2) 打开 TCP&UDP 测试工具, 创建 UDP 组播连接。填写目标 IP、端口。

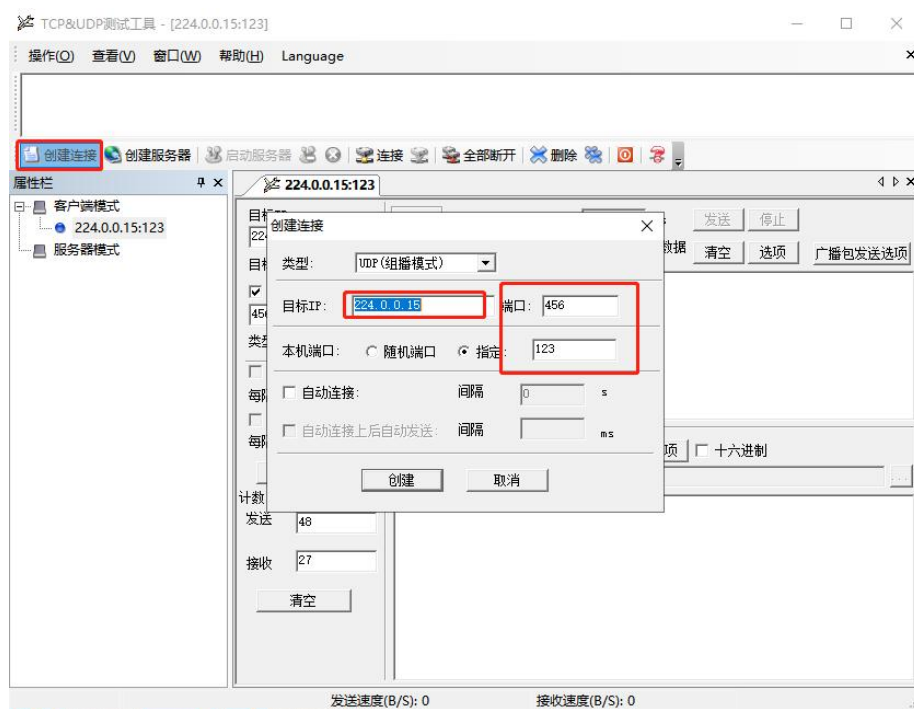


图 22. UDP 组播连接

3) 可以看到建立起了 UDP 组播通讯。

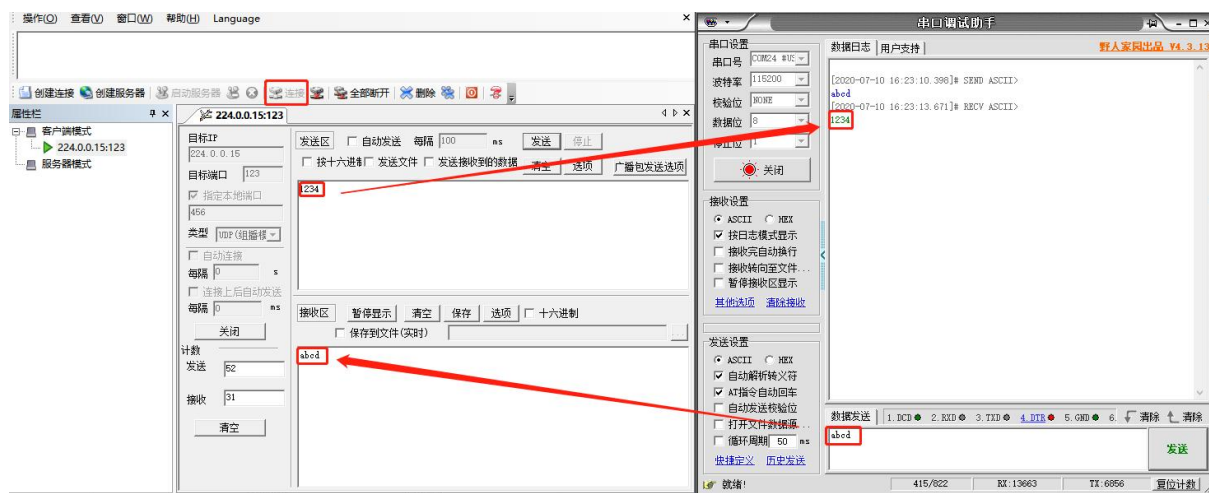


图 23. UDP 组播通讯

3.3.3.2. UDP 数据过滤功能

UDP 数据过滤功能可以在 UDP Clinet 功能接收网络数据时验证数据来源，过滤掉不符合条件的 IP 以及端口号。

数据过滤功能可以通过网络配置协议、网页打开，通过配置目标 IP 格式为全网段广播(例：255.255.255.255)、段内广播(例：192.168.0.255)、常规 IP(例：192.168.0.201)便可实现，以下为配置详解：

- 全网段广播不判断数据来源 IP，只判断来源端口是否和目标端口相同，相同时网络数据从串口输出，不同时，网络数据丢弃。
- 段内广播判断数据来源端口与目标端口是否相同，同时判断 IP 是否为段内 IP，符合条件的数据从串口输出，否则丢弃。
- 常规 UDP 通信，判断来源端口和 IP，与目标端口和 IP 相同的数据从串口输出，否则丢弃。

3.3.4. UDP Server 模式

UDP Server 是指在普通 UDP 的基础上不验证来源 IP 地址，每收到一个 UDP 数据包后，都将目标 IP 改为数据来源 IP 和端口号，发送数据时，发给最近通讯的那个 IP 和端口号。该模式通常用于多个网络设备都需要跟模块通信并且由于速度频率较快不想使用 TCP 的数据传输场景。

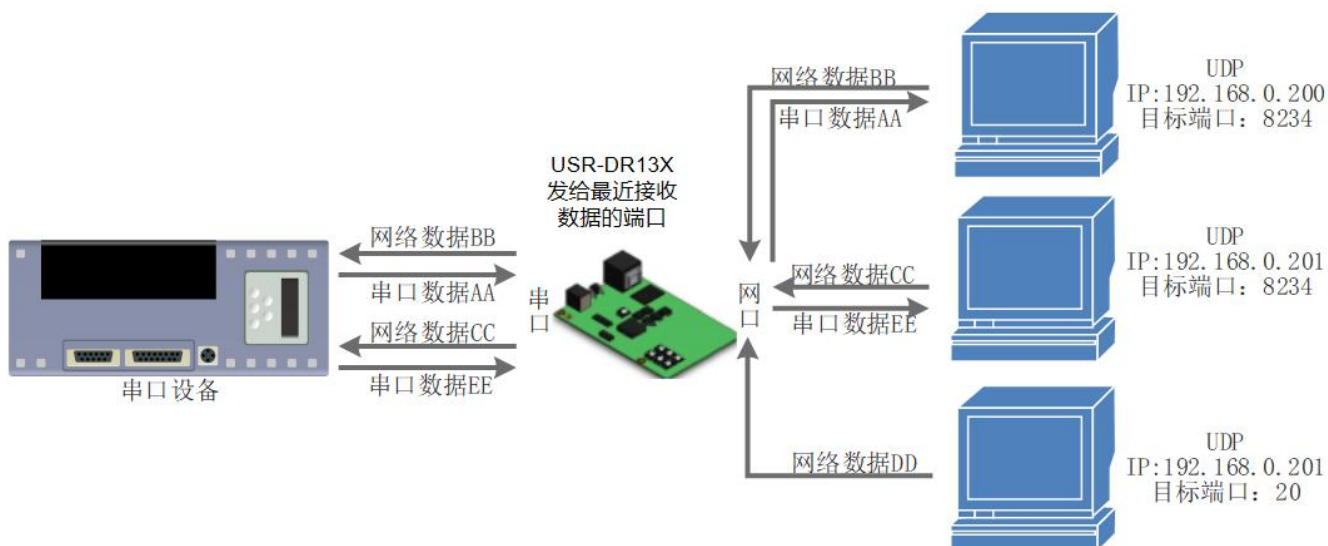


图 24. UDP Server 模式

通讯实例：

- 1) 设置 DR13X 为 UDP Server 模式，本地端口为 23
- 2) 打开两个测试软件，工作方式设置为 UDP，目标 IP 设为 DR13X 的 IP，目标端口改为 DR13X 的本地端口号，依次点击发送，串口会收到发送的数据，点击串口发送，测试软件只会有最近一个和 DR13X 的通信的软件收到数据
- 3) 启动后当没有任何 UDPC 通讯时，串口的数据会发往预先配置的远程服务器地址和端口

当前状态	参数	帮助
网络参数	波特率: 115200 bps(600~230400)	Client超限机制: KICK: 先踢掉最老的连接, 然后接入新的连接。 KEEP: 当新连接接入时, 不处理握手, 不回复任何信息。 Client接入数量: 模块做TCP Server时, 允许连接的Client最大值。 MQTT 模块做MQTT时, 请在MQTT网关页面设置参数 组播 组播的IP地址设置范围为224.0.0.2 - 239.255.255.255。
端口参数	数据位: 8 bit	
MQTT网关	校验位: NONE	
边缘网关	停止位: 1 bit	
点表参数	本地端口: 23 (0~65535)	
通用功能	远程端口: 456 (1~65535)	
系统参数	工作方式: UDP Server	
模块管理	远程服务器地址: 192.168.0.95 [192.168.0.95]	
	UDP数据过滤: Close	
	短连接使能: Close	
	短连接超时: 3 (2~255)	
	Client超限机制: KICK	
	Client接入数量: 4 (1~16)	
	保存 取消	

图 25. UDP Server 配置

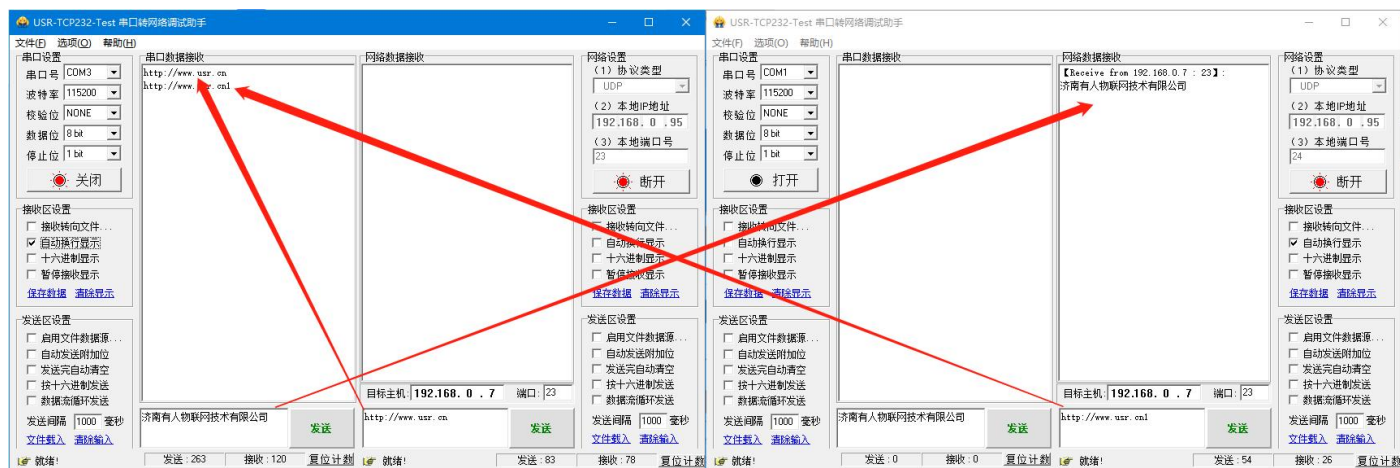


图 26. UDP Server 测试截图

3.3.5. Httpd Client 模式

在此模式下, 用户的终端设备, 可以通过 DR13X 串口服务器发送请求数据到指定的 HTTP 服务器, 然后 DR13X 接收来自 HTTP 服务器的数据, 对数据进行解析并将结果发至串口设备。

用户不需要关注串口数据与网络数据包之间的数据转换过程, 只需通过简单的参数设置, 即可实现串口设备向 HTTP 服务器的数据请求。Httpd Client 模式具体工作示意图和设置示例图如下:



图 27. HTTPD Client 模式说明

通讯实例：

- 1) 设置 DR13X 为 Httpd Client 模式，目前支持 GET/POST 方式请求 HTTP 服务器
- 2) 打开网页设置 Httpd 包头、URL 等信息。GET/POST 包头为预设包头，客户无需自己添加，其他包头客户自行添加
- 3) 填写远程服务器地址和网页服务器端口号。网页端口号一般为 80
- 4) 设置完成点击保存参数，然后重启 DR13X 生效
- 5) 打开串口发送数据，然后串口的数据就提交到网页服务器上了

当前状态	波特率: 115200 bps(600~230400)	Client超限机制: KICK: 先踢掉最老的连接, 然后接入新的连接。 KEEP: 当新连接接入时, 不处理握手, 不回复任何信息。 Client接入数量: 模块做TCP Server时, 允许连接的Client最大值。 MQTT 模块做MQTT时, 请在MQTT网关页面设置参数 组播 组播的IP地址设置范围为224.0.0.2 - 239.255.255.255。
网络参数	数据位: 8 bit	
端口参数	校验位: NONE	
MQTT网关	停止位: 1 bit	
边缘网关	本地端口: 0 (0~65535)	
点表参数	远程端口: 80 (1~65535)	
通用功能	工作方式: HTTP Client	
系统参数	远程服务器地址: test.usr.cn	
模块管理	UDP数据过滤: Close	
	短连接使能: Close	
	短连接超时: 3 (2~255)	
	Client超限机制: KICK	
	Client接入数量: 4 (1~16)	
	HTTP 协议: GET	
	Http URL(<100byte): /1.php?	
	HTTP 包头(<200byte): User-Agent: Mozilla/4.0 Connection: close	
	HTTP 超时: 10 (2~255)	
	HTTP 去包头: Close	
	保存 取消	

图 28. Httpd Client 设置图示

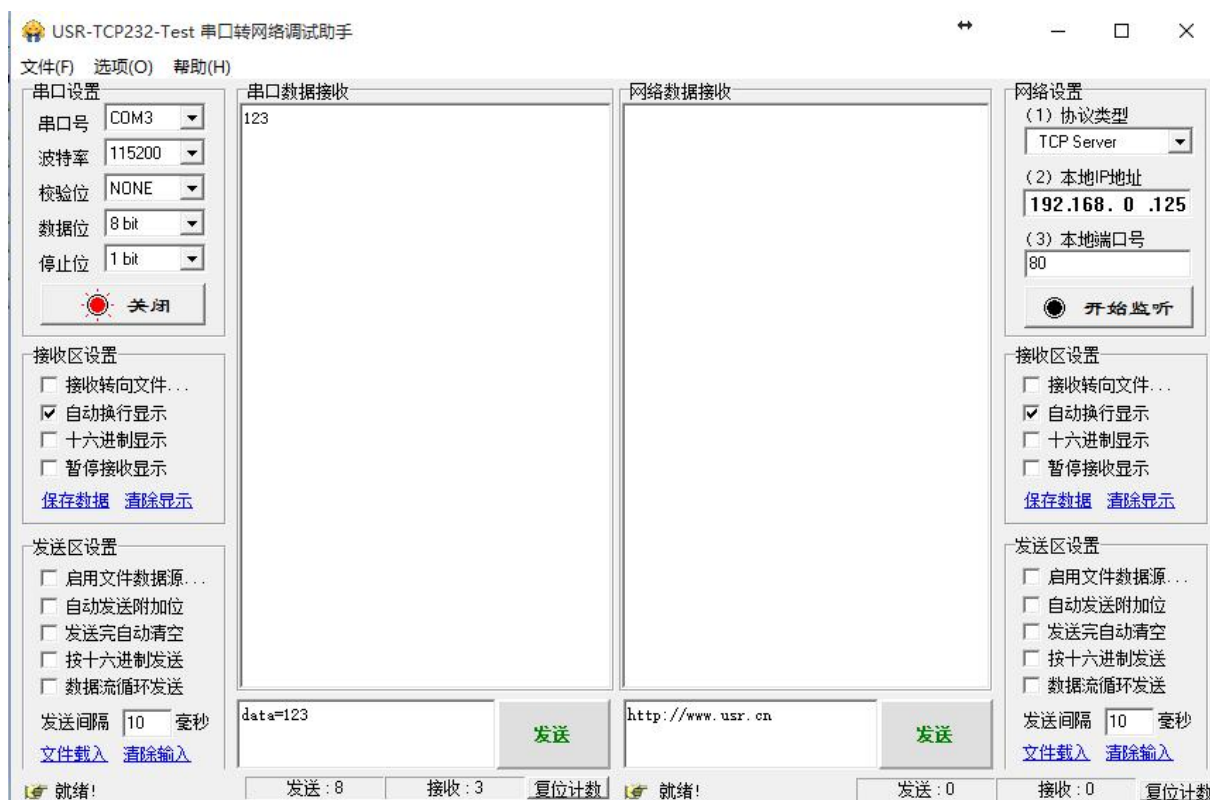


图 29. HTTPD Client 测试截图

3.4. MQTT 网关

MQTT 是一个基于客户端-服务器的消息发布/订阅传输协议。MQTT 协议具有轻量、简单、开放和易于实现的特点, 这使得它适用范围非常广泛。在此模式下, 本产品作为 MQTT Client, 可帮助用户快速接入搭建的私有 MQTT 服务器或公有 MQTT 物联网云平台。用户不需要关注串口数据与网络数据包之间的数据转换过程, 只需通过简单的参数设置, 即可实现串口与服务器的数据透传。

本产品支持通用 MQTT 接入和阿里云快捷接入, 数据发布和数据订阅均支持多主题添加配置。

3.4.1. 通用 MQTT

本产品以通用 MQTT 接入的方式, 连接标准 MQTT 协议物联网平台, 如百度云、腾讯云、华为云、AWS 云等, 同时支持重连间隔配置, 以适应不同 MQTT 服务器。

配置方式如下：

(1) 工作模式选择 MQTT 模式



图 30. MQTT 模式配置

(2) MQTT 网关中配置参数

- 工作模式：设备可选择 MQTT 模式或阿里云模式。选择 MQTT 模式
- MQTT 协议版本：设备可选择 MQTT-3-1 版本和 MQTT-3-1-1 版本
- 客户 ID：MQTT 客户端标识符。可自定义客户 ID，也可选择设备 SN 或 MAC
- 服务器域名（IP）：MQTT 服务器域名（IP）地址
- 本地端口号：设备 mqtt socket 绑定的端口号，填写 0 表示设备 mqtt socket 随机绑定本地端口号
- 远程端口号：MQTT 服务器连接端口号
- keepalive：MQTT 协议心跳时间
- 无数据重连时间：一直收不到 MQTT 协议帧直至无数据时间到了，会触发设备重连 MQTT 服务器
- 重连间隔时间：MQTT 断连后下次重连间隔
- 清理会话使能：MQTT 协议连接标志位，用于控制会话状态的生存时间
- 连接验证：打开可设置 MQTT 连接用户名和密码，关闭后 MQTT 连接的用户名和密码会设置为空
- 用户名：MQTT 连接的用户名
- 密码：MQTT 连接的密码
- 遗嘱：MQTT 连接标志，网络连接关闭时，服务端必须发布这个遗嘱消息，订阅这个遗嘱主题的客户端会收到设置的遗嘱



图 31. MQTT 参数配置

3.4.2. 阿里云

阿里云物联网平台是目前非常流行的公有云平台，设备支持 MQTT 协议接入阿里云物联网平台，支持设备密钥认证（三元组）证书认证接入阿里云。设备密钥认证，设备需要填写阿里云平台设备的三元组等信息即可接入阿里云。此外主题透传功能与 MQTT 网关功能一致，配置简单，使用方便。

配置方式如下：

(1) 工作模式选择 MQTT 模式

图 32. MQTT 模式配置

(2) MQTT 网关中配置基本参数：

- 工作模式：设备可选择 MQTT 模式或阿里云模式。选择 MQTT 模式
- ProductKey: 阿里云添加设备中的三元组之 ProductKey
- 区域代码：阿里云区域代码，例如华东 2（上海）填写：cn-shanghai
- 客户 ID：支持自定义客户 ID，用于拼接 MQTT clientid，不填默认拼接设备 SN
- DeviceName：阿里云添加设备中的三元组之 DeviceName
- DeviceSecret：阿里云添加设备中的三元组之 DeviceSecret
- 远程端口号：阿里云物联网平台服务器端口号
- Keepalive：建立 mqtt 连接的心跳时间
- 无数据重连时间：一直收不到 MQTT 协议帧直至无数据时间到了，会触发设备重连阿里云物联网平台服务器
- 重连间隔：重连阿里云物联网平台服务器等待时间（MQTT 连接断开后到重新建立连接的时间）

图 33. 阿里云基本参数配置

3.4.3. 主题发布配置

主题发布的目的是为了其他 MQTT 客户端可以订阅到该主题，并进行数据传输。最多可支持 4 个主题的发布。主题发布支持透传模式、带主题下发模式、自定义模式。

- 透传模式：将串口接收到的数据以透传的方式传输到相关联的主题，最多支持 4 个主题发布，每个主题都可以单独进行配置，

串口收到数据后，设备会把数据推送到绑定该串口的主题。

- Topic 字符串：发布主题名
- QOS：发布主题的消息质量
- 保留消息：MQTT 发布消息保留消息标志位，用于服务端是否存储这个应用消息和它的服务质量等级（QoS）

图 34. 发布配置——透传模式

- 带主题下发模式：配置主题的时候可以定义此主题的识别标识符，设备收到串口数据后会根据识别标识符，推送至与之关联的主题。设备允许不同的主题拥有相同的识别标识符，这样数据会推送到识别标识符相同的所有主题，识别标识符与 payload 之间用逗号分隔。
 - Topic 字符串：发布主题名
 - 主题标识：该主题识别的标识符
 - QOS：发布主题的消息质量
 - 保留消息：MQTT 发布消息 保留消息标志位，用于服务端是否存储这个应用消息和它的服务质量等级（QoS）

图 35. 发布配置——主题分发模式

- 自定义模式：自定义模式使用简单快捷，用户无需在网页配置数据发布的相关参数，仅需要开启自定义模式开关，点击保存重启设备后即可实现 MQTT 数据透传。

图 36. 发布配置——自定义模式

自定义数据发布模式消息格式：<top>,<qos>,<retained>,<payload>

<top>:发布主题名称。

<qos>:发布主题消息质量，可填写 0 或 1 或 2，0:QOS0, 1:QOS1, 2:QOS2。

<retained>:发布主题保留消息标志位，可填写 ON 或 OFF, ON:保留消息置位 1, OFF:保留消息置位 0。

<payload>: 发布主题消息内容。

例如：publishtopicname, 0, on, payloaddata

3.4.4. 主题订阅配置

主题订阅的目的是为了订阅其他客户端的主题，以达到两个或者多个客户端进行数据通讯的目的，最多支持 4 个主题订阅。主题订阅支持透传模式和带主题下发模式，其他详细参数配置下面将进行详细介绍。

- 透传模式：将接收到的数据以透传的方式传输到相关联的主题，最多支持 4 个主题发布，每个主题都可以单独进行配置，主题收到数据后会发送数据到绑定串口。
 - Topic 字符串：发布主题名
 - QOS：发布主题的消息质量

订阅配置

订阅主题

透传模式

☒ 主题1

/SubTopic1

QOS0

☐ 主题2

☐ 主题3

☐ 主题4

保存

取消

图 37. 订阅配置——透传模式

- 带主题下发模式：带主题下发是指将订阅收到的消息前面加上主题，随 payload 一起下发，数据默认格式为：topic,payload。主题与数据之间用分隔符隔开。每个主题都可以单独进行配置，且都可以绑定不同的串口，主题收到数据后会发送数据到绑定串口。
 - Topic 字符串：发布主题名
 - 分隔符：主题与应用数据之间的分隔符
 - QOS：发布主题的消息质量

订阅配置

订阅主题

带主题下发模式

☒ 主题1

/SubTopic1

QOS0

分隔符

,

☐ 主题2

☐ 主题3

☐ 主题4

图 38. 订阅配置——带主题下发模式

3. 5. Modbus 网关

3. 5. 1. Modbus TCP/RTU 协议转换

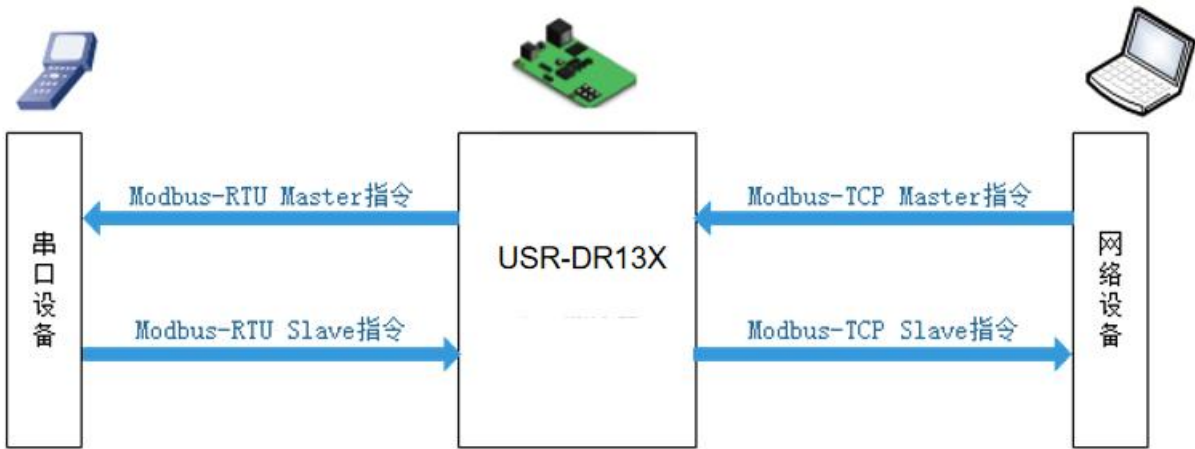


图 39. Modbus 协议转换功能框图

在网络透传模式下（TCP、UDP），采用 Modbus 标准的 TCP 协议和 RTU 协议进行开发，主要功能逻辑有两点：

- 网络端下发的 Modbus TCP 协议数据转换为 Modbus RTU 协议数据，并将该数据通过串口发出
- 串口接收到的 Modbus RTU 协议数据转换为 Modbus TCP 协议数据，并将改数据通过网络发送到服务器

设置示意图如下：



图 40. Modbus 协议转换功能设置

功能测试:

- 打开网页，把 DR13X 设置成 TCP Server 模式，开启 modbus TCP/RTU 转换功能，点击设置保存参数
- 通过 Modbus Poll 和 Modbus Slave 去查询验证 Modbus TCP 转 Modbus RTU，设置 Modbus 软件如图所示设置
- 完成后点击 OK，更新 Modbus Slave 的数据，Modbus Poll 的数据也跟着更新。并且可以看到 RTU 数据转换为 TCP 数据。

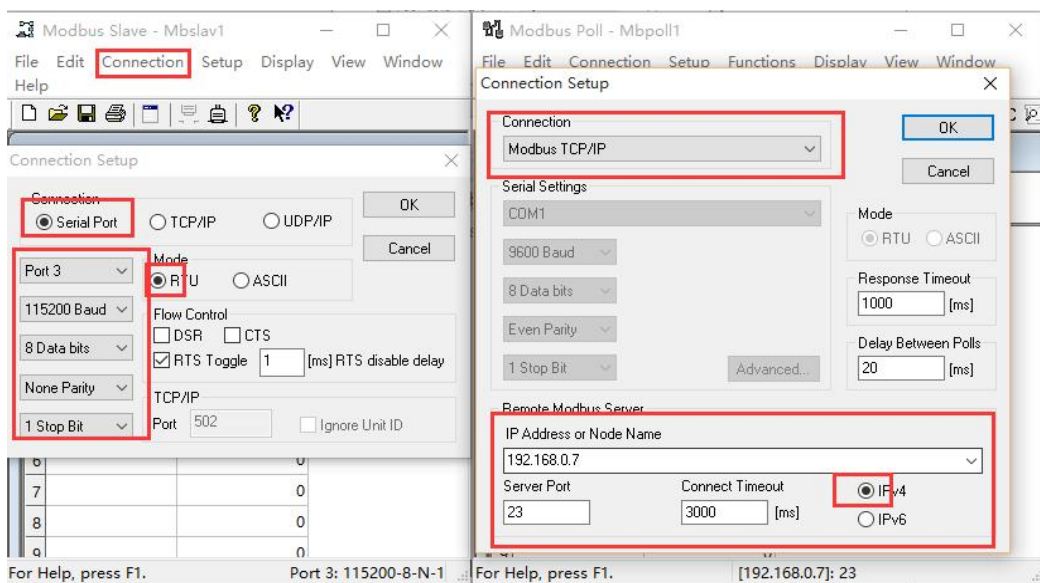


图 41. Modbus RTU 转 Modbus TCP 软件设置截图

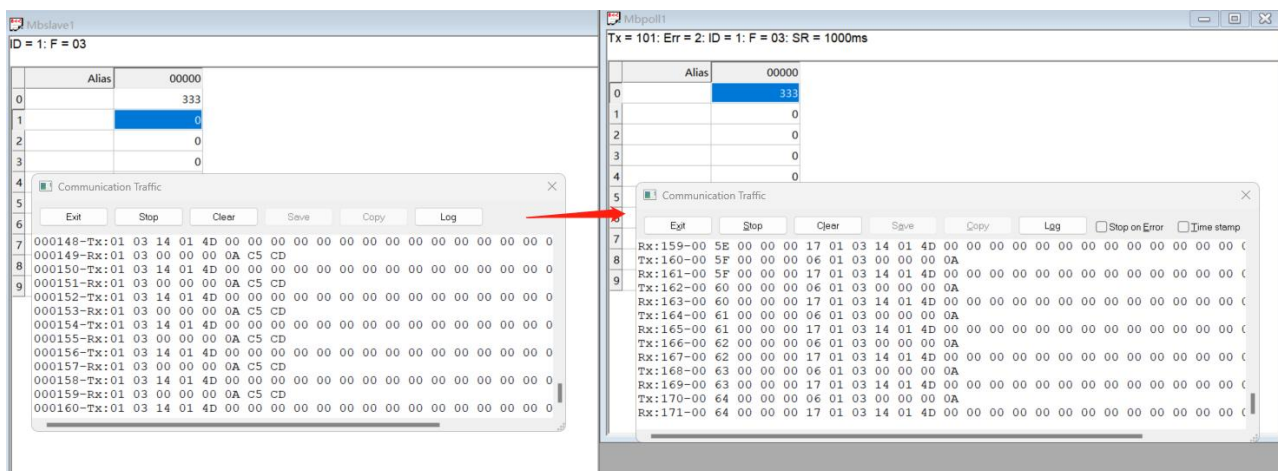


图 42. Modbus RTU 转 Modbus TCP 转换结果

3. 5. 2. Modbus 多主机轮询

多主机轮询功能主要是为了解决多个网络设备同时发送数据后串口设备回复错位的问题，可识别不同网络设备发送的数据并进行对应回复，防止数据错位。

设置多主机轮询参数：

- 工作模式：开启多主机轮询功能
- 使能 Modbus 异常响应：该功能可根据需求自主选择开启或关闭，功能开启后 DR13X 在下端串口回复超时或接收到错误的 Modbus RTU 协议帧时，向主机回复携带错误码的异常帧。功能关闭，DR13X 在下端串口回复超时或接收到错误的 Modbus RTU 协议帧时不做异常帧回复。
- 轮询超时：从 DR13X 向下端串口设备发送完一帧数据，到串口设备向 DR13X 回复数据的时间间隔；默认为 200ms。
- 轮询间隔：从 DR13X 把接收到的下端串口设备回复数据成功抛向网络端，到 DR13X 把缓存的轮询数据再次下发到串口设备的时间间隔；默认值 100ms。



图 43. MODBUS 多主机轮询使能设置图示

功能测试：

- 配置 DR13X 多主机轮询参数
- 打开 Modbus Slave，串口选择与 DR13X 连接的串口，打开多个 Modbus Poll 选择网络模式，和 DR13X 建立连接
- 修改 Modbus Slave 的值，多个 Modbus Poll 也跟着改变。

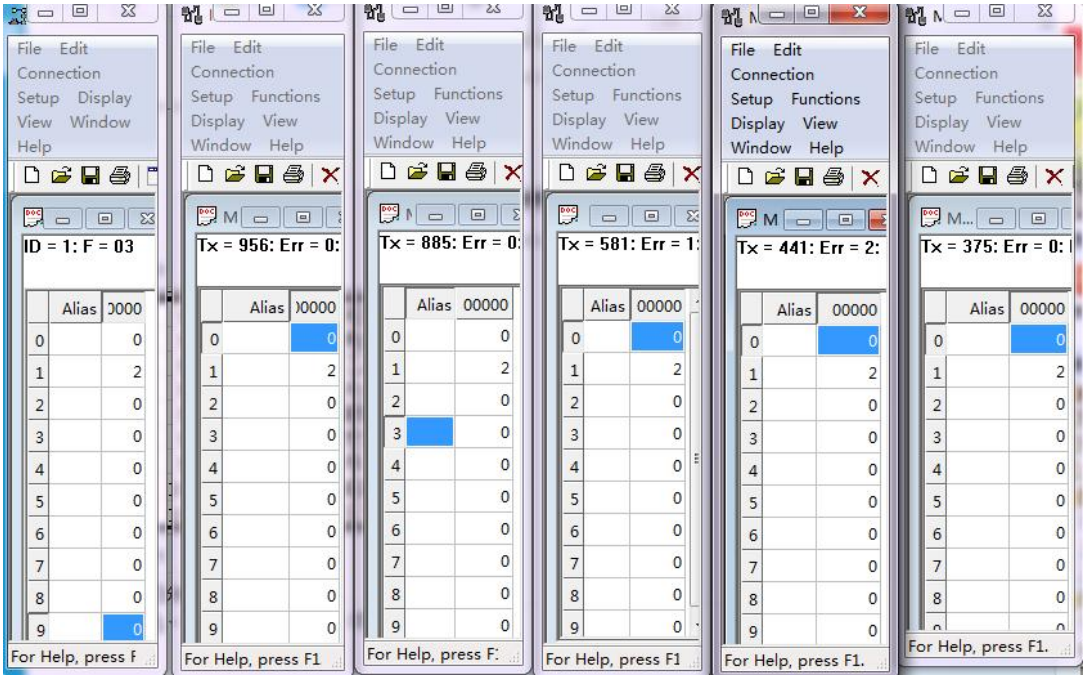


图 44. MODBUS 轮询测试

3. 5. 3. Modbus 自定义轮询功能

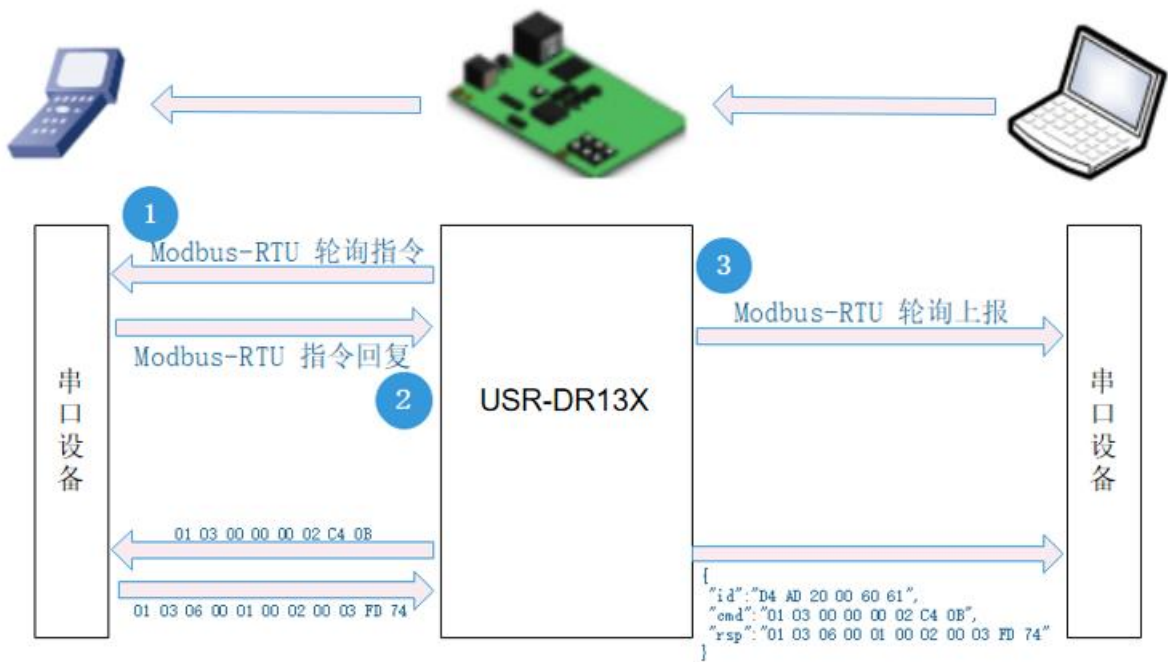


图 45. Modbus 采集轮询功能框图

开启轮询采集功能，设备在上电后，将会按照存储的命令顺序依次从串口发送 Modbus-RTU master 命令进行轮询，每条命令下发后，按照轮询超时等待回复，超时后，将会按照轮询间隔时间进行下一条采集命令的发送。
设置示意图如下：

当前状态	参数	帮助
网络参数	工作模式: Modbus 自定义轮询	• Modbus 简单协议转换: 网络端接收的Modbus TCP协议数据转换为 Modbus RTU协议数据。 串口接收到的Modbus RTU协议数据转换为 Modbus TCP协议数据。 • Modbus 多主机轮询: 可识别不同网络设备发送的数据并进行对应回复，防止数据错位。 • Modbus 自定义轮询: 设备将会按照存储的命令顺序依次从串口发送采集命令进行轮询。
端口参数	Modbus 上报协议: JSON	
MQTT网关	轮询超时: 200 (ms)(50~65535)	
边缘网关	轮询间隔: 100 (ms)(1~65535)	
点表参数	异常响应: Open	
通用功能	保存 取消 添加 清空	
系统参数		
模块管理		

图 46. Modbus 协议轮询采集功能设置

轮询采集功能只支持完整的 Modbus 采集命令设置，最多可设置 5 条采集命令。每数据令最大 16 字节。
采集命令可以通过 AT 指令设置，指令设置格式为：AT+MODCMD=1, cmd （其中 cmd 为设置和查询的 Modbus 轮询指令），也可以通过网页直接写入。

当前状态	参数	帮助
网络参数	工作模式: Modbus 自定义轮询	• Modbus 简单协议转换: 网络端接收的Modbus TCP协议数据转换为Modbus RTU协议数据。串口接收到的Modbus RTU协议数据转换为Modbus TCP协议数据。 • Modbus 多主机轮询: 可识别不同网络设备发送的数据并进行对应回复,防止数据错位。 • Modbus 自定义轮询: 设备将会按照存储的命令顺序依次从串口发送采集命令进行轮询。
端口参数	Modbus 上报协议: JSON	
MQTT网关	轮询超时: 200 (ms)(50~65535)	
边缘网关	轮询间隔: 100 (ms)(1~65535)	
点表参数	异常响应: Open	
通用功能	Modbus 指令1: 010300000000AC5CD 删除	
系统参数	保存 取消 添加 清空	
模块管理		

图 47. Modbus 协议轮询指令写入

Modbus 自定义轮询采集功能主要参数包括:

- (1) 工作模式: 可选择开启 Modbus 自定义轮询功能。
- (2) 上报协议: 可选 Modbus RTU 透传上报, 也可选择 Json 上报。

Json 数据格式如下:

```
{
  "id": "设备 MAC",
  "cmd": "串口轮询命令",
  "rsp": "串口接收的 modbus 内容"
}
```

- (3) Modbus 轮询超时: Modbus 轮询超时时间设置, 范围 0~65535。
- (4) Modbus 轮询间隔: Modbus 轮询间隔时间设置, 范围 0~65535。
- (5) Modbus 异常响应: 设置异常响应功能使能状态。
- (6) Modbus 指令: 选择“添加”, 可开启 Modbus 指令, 最多可写入 5 条

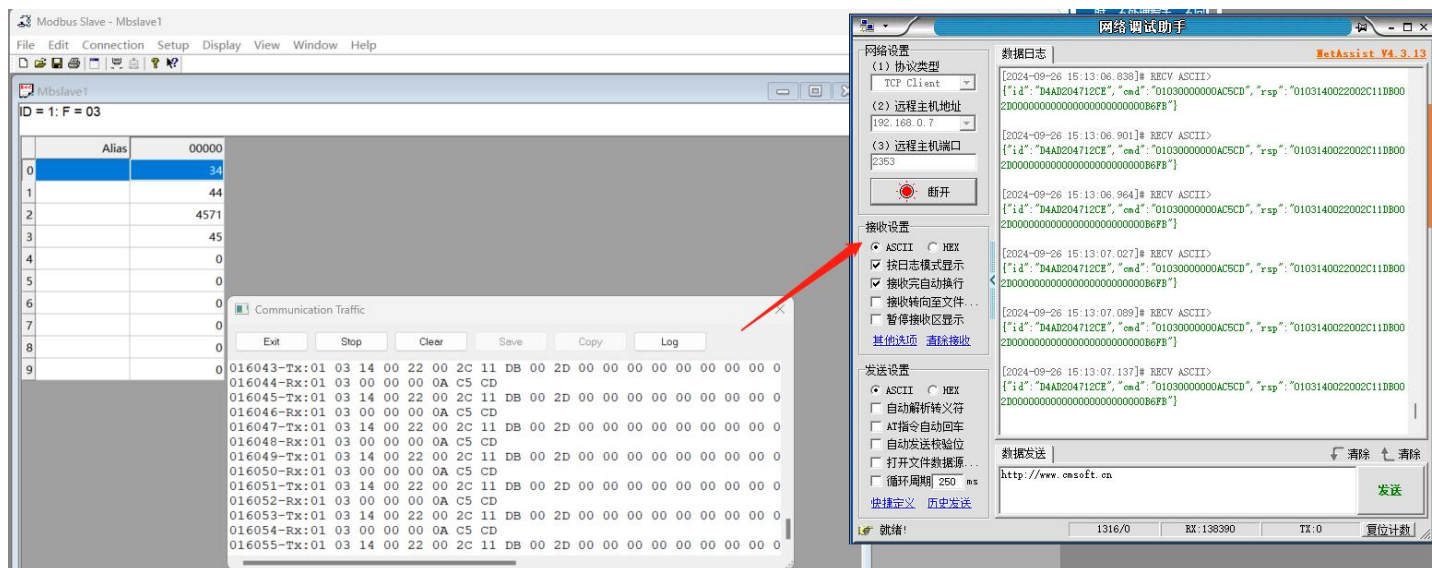


图 48. Modbus 协议轮询示意图

3.6. 边缘计算

边缘计算网关功能中, 网关设备作为轮询的发起方, 自主地、周期性地从终端设备上读取、解析、计算用户预置的数据信息。同时网关设备可以根据用户预设的上报逻辑, 有选择性地数据进行数据上报, 无需远端服务器主动参与, 即可完成数据采集任务, 极大地提升了云端服务器可以介入管理的终端设备的数量。

DR13X 支持预先配置从机数据点位, 边缘计算后, 支持 MQTT/TCP 通道下, JSON 格式采集、读写数据。

3.6.1. 边缘计算参数配置

- (1) 选择边缘网关, 工作模式选择轻边缘网关, 即可开启边缘计算功能。

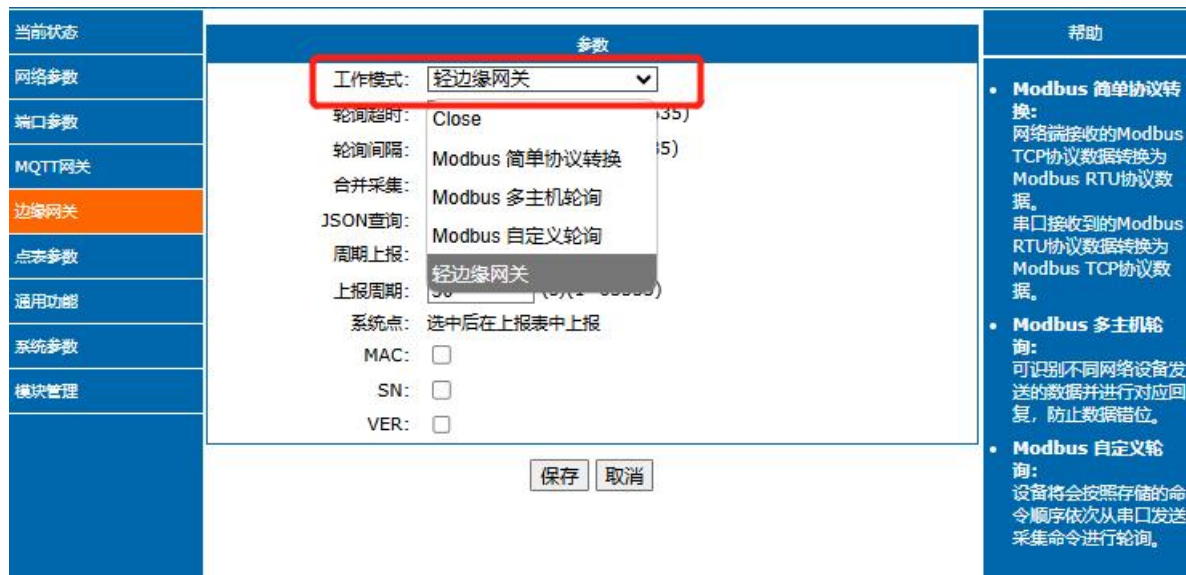


图 49. 轻边缘网关示意图

(2) 基本参数设置

- 轮询超时：单条 RTU 命令轮询，等待回复的最长时间，如果超时没有回复，则执行下一条轮询命令。范围：50~65535ms，默认 200ms
- 轮询间隔：同一轮采集周期内，两个采集点之间的采集。范围：1~65535ms，默认 100ms
- 周期轮询时间：设备上一轮完整轮询采集到下一轮采集之间的间隔。范围：1~65535s，默认 30s
- 轮询上报方式：可选 JSON 组包上报或透传上报。若选择透传上报，采集到的数据直接以 modbusRTU 格式透传上报。若选择 Json 组包上报，则采集的数据以 Json 格式上报，也可通过 Json 格式下发指令控制终端设备。
- 合并采集：开启合并采集后，会自动识别点位信息（从机地址、寄存器类型、寄存器地址），如点位信息连续，则合并采集上报。**注意：点位信息必须顺序连续，例 40001、40002、40003，如果顺序不连续，如 40002、40001、40003，则不会合并采集**
- JSON 查询：开启 JSON 查询，可发送 JSON 数据包，对设备点位信息进行读写。
- 周期上报：开启后，设备定时进行主动上报。时间可选 1~65535s。默认 30s
- 系统点
 - MAC：选择后将设备 MAC 上报
 - SN：选择后将设备 SN 上报
 - VER：选择后将设备版本上报



图 50. 轻边缘网关基础信息配置

(3) 点表参数设置

在点表参数栏中，预先配置采集点位信息。最多可支持 25 条点位配置。配置内容如下：

- 设备名：数据点在表内的唯一标识。变量名称不可重复，每次更新点位配置时，变量名称不可与上次配置相同。
- 从机地址：采集从机的 modbus 地址。范围：1~247
- 寄存器类型：数据点采集时使用的 modbus 功能码。支持功能码：01（线圈）、02（离散量输入）、03（保持寄存器）、04（输

入寄存器) 功能码

- 寄存器地址: 数据点的 modbus 寄存器地址。从 0 地址开始, 范围: 0~65534
- 数据类型: 数据点的数据类型, 表明数据长度与预期边缘计算网关对其解析的方式。01、02 功能码仅允许设置 Bool 数据类型, 03、04 功能码不允许设置 Bool 数据类型
- 变化上报: 基于倍率计算后的数据变化范围, 如果超过设定的幅值, 即上报。
- 倍率: 倍率计算, 范围 0.000001~9999999。计算结果保留三位小数上报。
- 幅值: 在变化上报下生效。支持小数, 精度最大三位小数

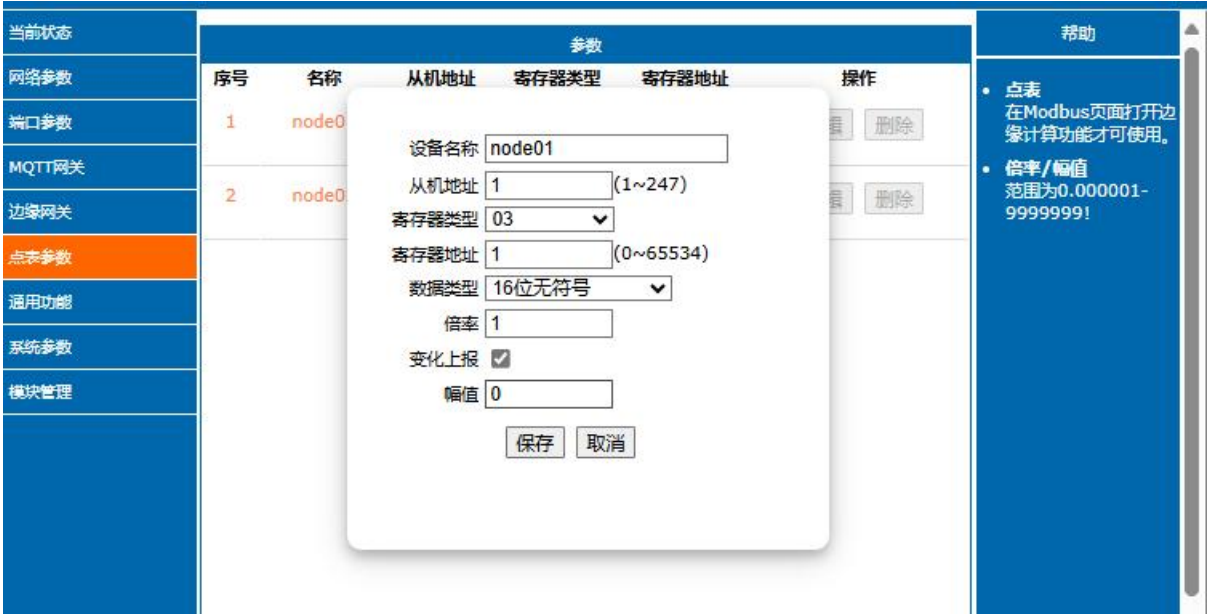


图 51. 轻边缘网关点表参数配置

3.6.2. Json 交互格式

(1) 当设备轮询完毕后会根据用户设定的规则主动上报数据, 格式如下:

```
{
  "params": {
    "dir": "up",
    "id": "00501521042000019454",
    "r_data": [{
      "name": "node0101",
      "value": "35",
      "err": "0"
    },
    {
      "name": "node0102",
      "value": "35",
      "err": "0"
    }
  ]
}
```

(2) 也可以通过云端发送读写命令进行交互。DR13X 支持一个点位的读命令或一个点位的写命令。可通过写命令来改变变量值。或存在还没到下一次轮询, 用户立刻需要查看最新数据的情况, 云端可以发送读命令来更新数据。格式如下:

```
读命令下发:
{
  "rw_prot": {
    "Ver": "1.0.1",
    "dir": "down",
```

```
      "id": "00501521042000019454",
      "r_data": {
        "name": "node0101"
      }
    }
  }
```

设备上报:

```
{
  "rw_prot": {
    "Ver": "1.0.1",
    "dir": "rsp",
    "id": "00501521042000019454",
    "r_data": {
      "name": "node0101",
      "value": "35",
      "err": "0"
    }
  }
}
```

写命令下发:

```
{
  "rw_prot": {
    "Ver": "1.0.1",
    "dir": "down",
    "id": "00501521042000019454",
    "w_data": {
      "name": "node0101",
      "value": "35"
    }
  }
}
```

设备上报:

```
{
  "rw_prot": {
    "Ver": "1.0.1",
    "dir": "rsp",
    "id": "00501521042000019454",
    "w_data": {
      "name": "node0101",
      "value": "35",
      "err": "0"
    }
  }
}
```

字段说明:

表 6 字段说明

字段	说明
Ver	协议版本。由服务器下发，回复时保持一致即可。如服务器不下发或者仅上报时，则本项默认无
dir	命令类型，包括下发和上报标识。下发为 down, 下发回复为 rsp，轮询采集上报为 up

id	信息 ID 号，由服务器下发，设备回复保持一致即可。如服务器不下发，则忽略。上报中，默认采用设备 SN 作为 ID
r_data	读操作字段
w_data	写操作字段
name	点位名称
value	点位有效数据 注： err=0 时，value 上报有效数值 err=1、2 或其他错误码时，value 值为空
err	0 表示成功，1 表示从机无响应；2 表示 CRC 校验不通过；其他值表示 Modbus 错误码（如 0x81）

错误字段：JSON 格式错误、r_data 错误、w_data 错误或其他字段错误，回复错误协议。

错误协议：

```
{
  "rw_prot": {
    "Ver": "1.0.1",
    "dir": "rsp",
    "err": "1"
  }
}
```

3.6.3. MQTT+Json 组包上报示例

以 Modbus Slave 模拟从机设备，MQTT 通道+Json 组包上报为例，详细介绍边缘计算使能。

1、配置 Modbus Slave 用以模拟从机设备。

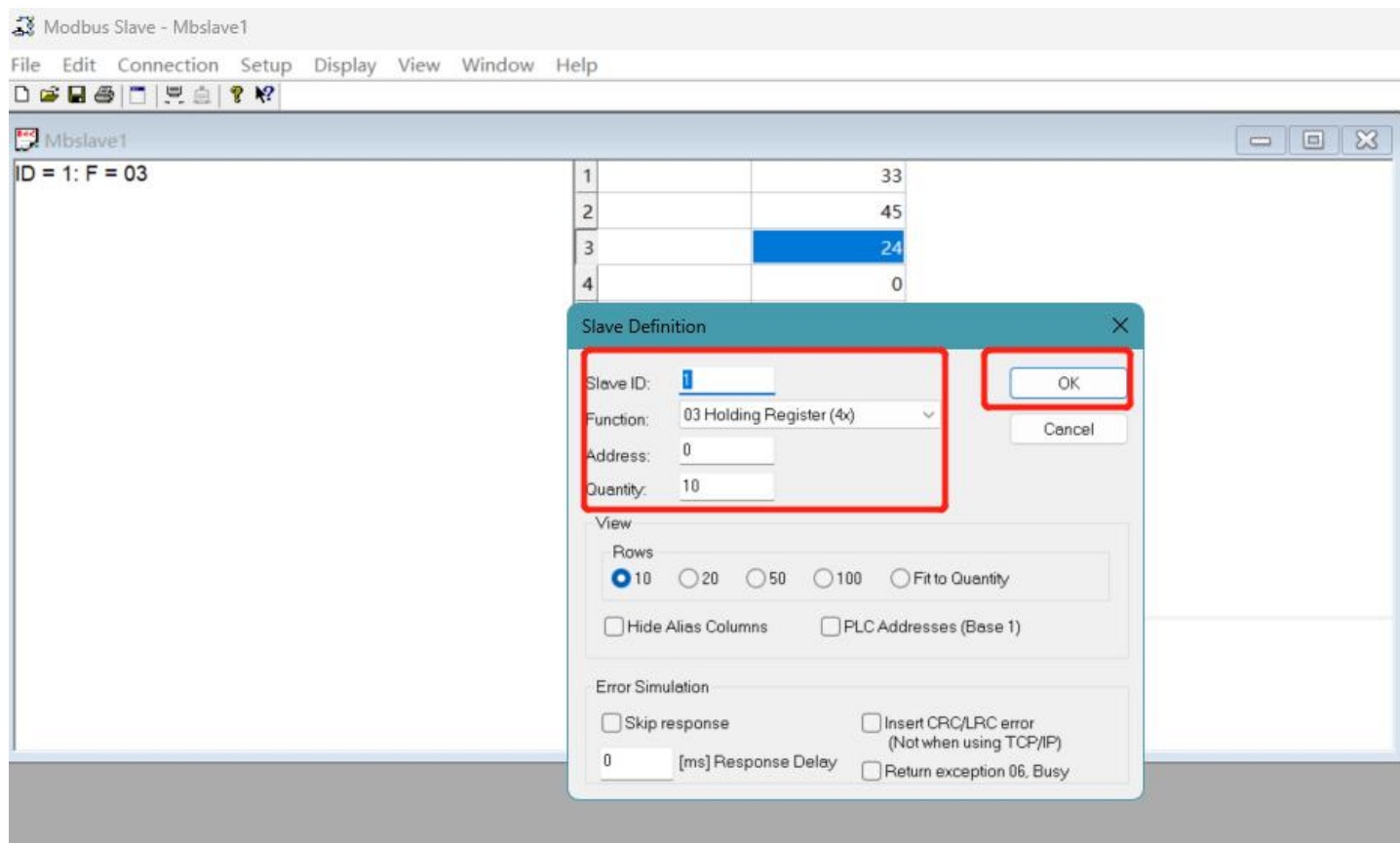


图 52. slave 模拟从机配置

2、对 DR13X 进行参数配置。

(1) 根据 3.4 章节，配置 MQTT 参数

当前状态

网络参数

端口参数

MQTT网关

边缘网关

点表参数

通用功能

系统参数

模块管理

参数

工作模式:MQTT

MQTT协议:MQTT-3.1.1

客户ID:SYS_SN

服务器域名(IP):mqtt.usr.cn

本地端口:0(0~65535)

远程端口:1883(1~65535)

keepalive:60(s)(0~65535)

无数据重连时间:0(s)(0~65535)

重连间隔时间:5(s)(1~65535)

清除会话:☐

连接验证:☒

用户名:usr.cn

密码:usr.cn

遗嘱使能:☐

帮助

MQTT网关

支持清理会话、遗嘱、QOS等功能,支持对接第三方MQTT服务器

发布/订阅主题

最多支持4个发布主题和4个订阅主题

发布配置

发布主题:透传模式

☒主题1/PubTopic123QOS0保留消息☐

☐主题2

☐主题3

图 53. MQTT 参数配置

(2) 按照模拟的 slave 从机，配置边缘计算参数。注意从机地址、功能码、寄存器地址要与从机保持一致。

当前状态

网络参数

端口参数

MQTT网关

边缘网关

点表参数

通用功能

系统参数

模块管理

参数

工作模式:轻边缘网关

轮询超时:200(ms)(50~65535)

轮询间隔:100(ms)(1~65535)

合并采集:☐

JSON查询:☒

周期上报:☒

上报周期:30(s)(1~65535)

系统点:选中后在上报表中上报

MAC:☒

SN:☐

VER:☐

保存取消

帮助

Modbus 简单协议转换:

网络端接收的Modbus TCP协议数据转换为Modbus RTU协议数据。

串口接收到的Modbus RTU协议数据转换为Modbus TCP协议数据。

Modbus 多主机轮询:

可设置不同网络设置发送的数据并进行对应回复,防止数据错位。

Modbus 自定义帧:

设备符合按照存储的命令顺序依次从串口发送采集命令进行轮询。

当前状态

网络参数

端口参数

MQTT网关

边缘网关

点表参数

通用功能

系统参数

模块管理

序号名称从机地址寄存器类型寄存器地址操作

1node01131编辑删除

2node02132编辑删除

3node03133编辑删除

添加数据点

帮助

点表

在Modbus页面打开边缘计算功能才可使用。

设备/地址

范围为0.000001-99999999

图 54. 轮询参数配置

3、配置完网关参数后，保存重启生效。将串口连接到 Modbus Slave。

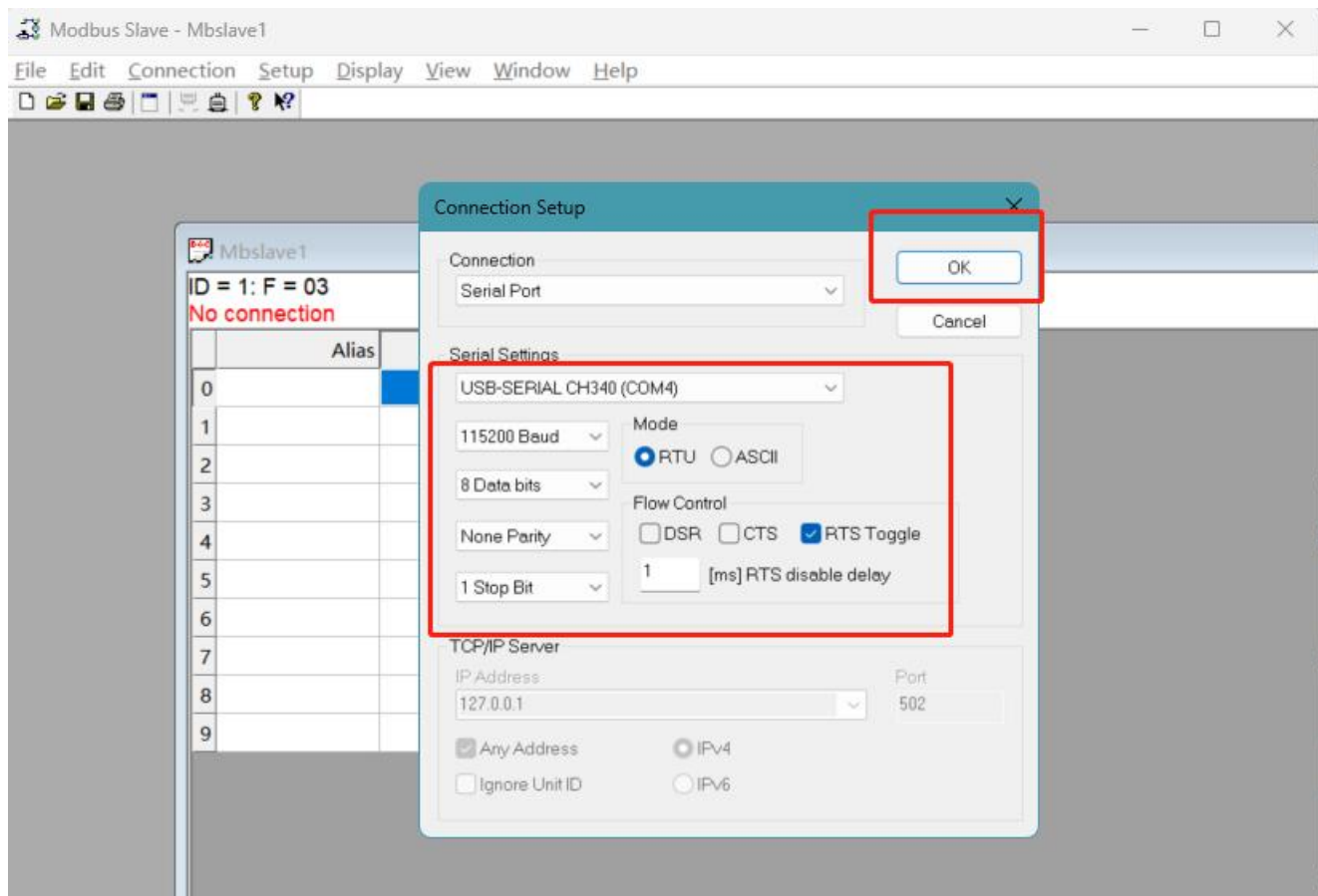


图 55. 串口连接到 Modbus Slave

4、网关会按照设定的轮询规则，主动采集从机的数据，并以 JSON 格式上报数据给 MQTT 服务器。需要接收数据的 MQTT 客户端订阅网关模块的发布主题就能收到设备的上报信息。

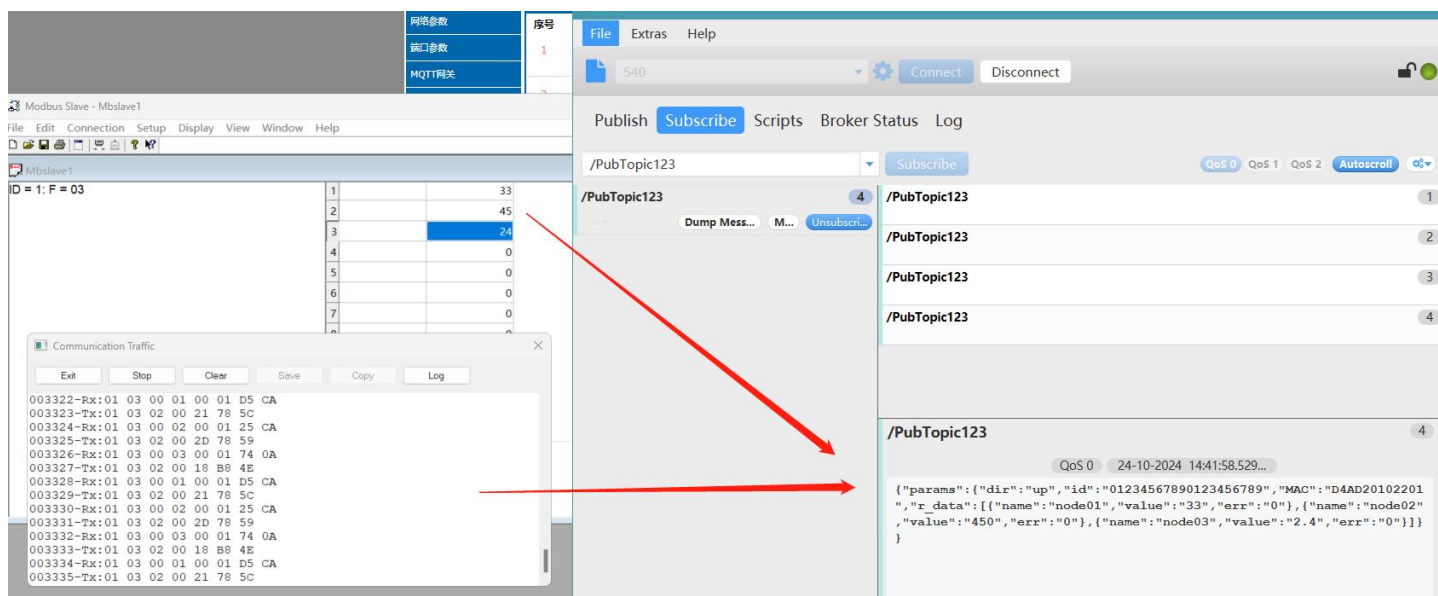


图 56. MQTT 服务器收到轮询信息

5、云端可以往网关模块订阅的主题发送命令来实现交互，例如往从机写数据。可以看到 slave 从机收到指令进行了回复，同时参数发生了改变。

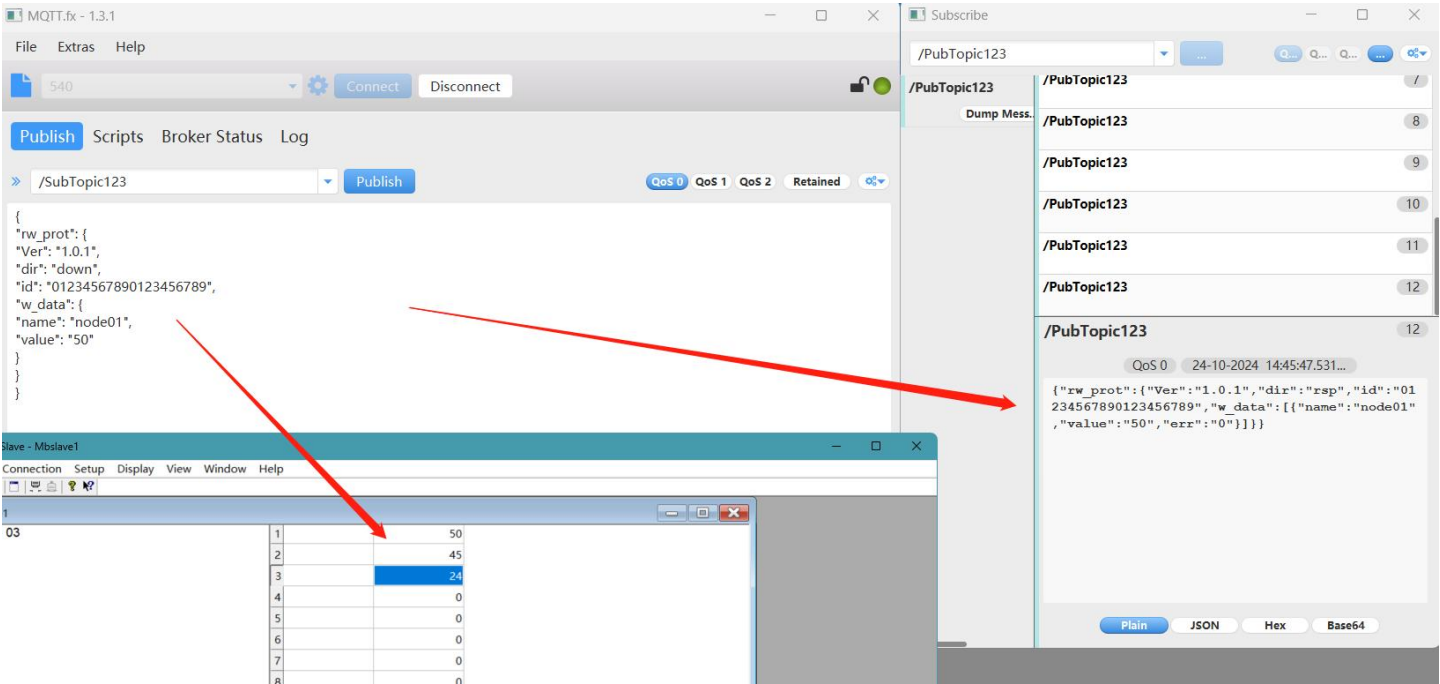


图 57. Json 云端读写

3. 7. 通用功能

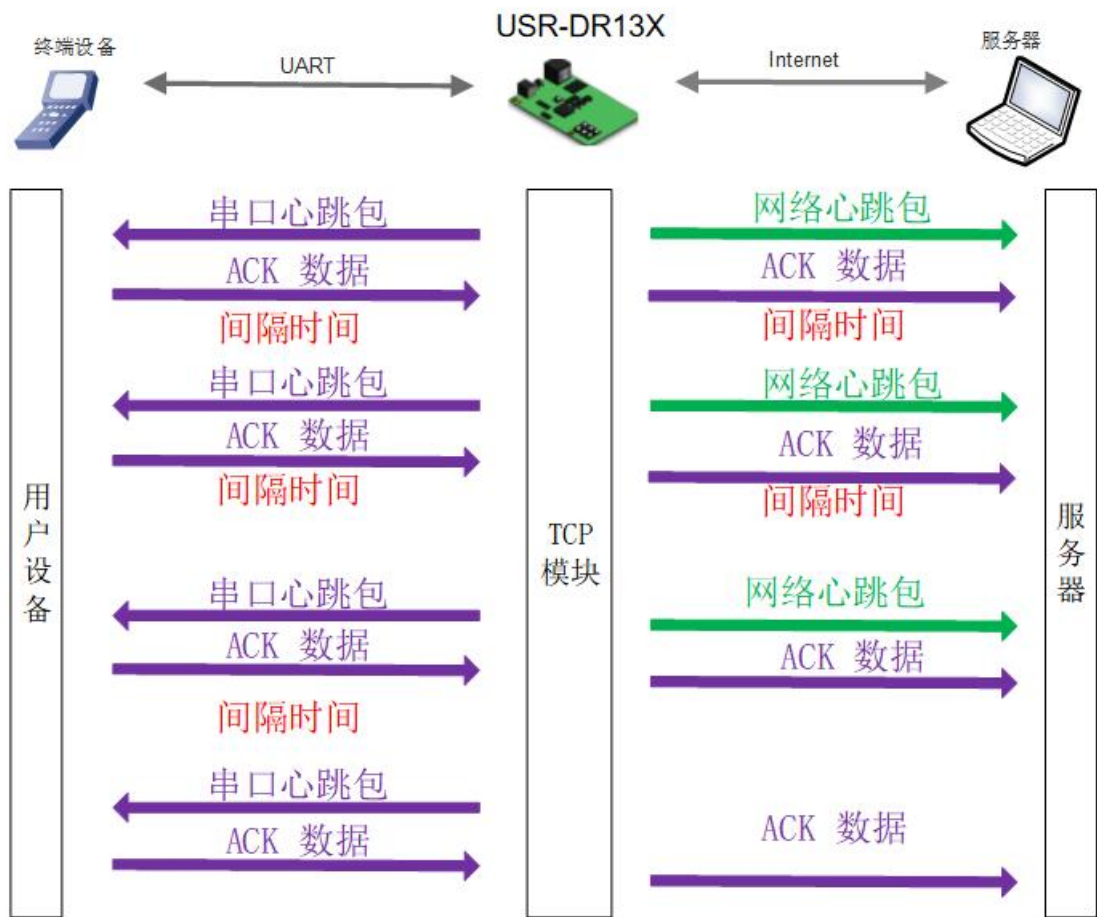
3. 7. 1. 心跳包功能

在网络透传模式下，用户可以配置心跳包功能。心跳包功能可配置网络心跳包、串口心跳包两个方向，两个方向不可同时运行。向网络端发送主要目的是为了与服务器保持连接，仅在 TCP Client 和 UDP Client 模式下生效。

在服务器向设备发送固定查询指令的应用中，为了减少通信流量，用户可以选择，用向串口设备端发送心跳包（查询指令），来代替从服务器发送查询指令。心跳包格式支持 MAC 心跳包、自定义心跳包，自定义心跳包内容最长 40 字节。

当前状态	参数	帮助
网络参数	心跳包使能: 串口心跳包	- 自定义心跳包: 默认心跳时间为30s, 支持16进制输入, 不支持汉字, 发送方向可随意选择, 最长40字节。 - 自定义注册包: 默认不开启, 网页支持16进制输入, 不支持汉字, 最长40字节。
端口参数	心跳时间: 30 (s)(1~65535)	
MQTT网关	心跳包类型: 自定义	
边缘网关	心跳包数据格式: ASCII	
点表参数	心跳包数据: heartbeat	
通用功能	注册包使能: 注册包关闭	
系统参数	Reset: ☐	
模块管理	Link: ☐	
	Index: ☐	
	类RFC2217: ☒	
	清除缓存数据: ☐	
	保存 取消	

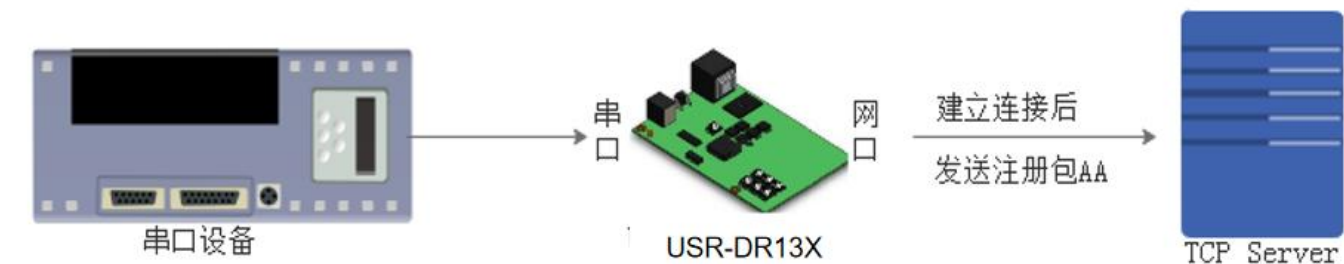
图 58. 心跳包设置



3.7.2. 注册包功能

在网络透传模式下，用户可以配置注册包功能。注册包可以作为服务器识别数据来源设备的标识，也可以作为获取服务器功能授权的密码。

注册包可以配置连接发送注册包或携带发送注册包，可以同时生效。连接发送指在 TCP 建立连接或 UDP 建立时发送，携带发送指在每个数据包的最前端拼接入注册包数据，作为一个数据包。注册包的数据可选 MAC 地址、自定义注册数据或有人云，其中自定义注册包设置内容最长为 40 字节。应用示意图如下：



数据携带注册包：发送数据在数据最前端接入注册包，主要用于协议传输，应用示意图如下：

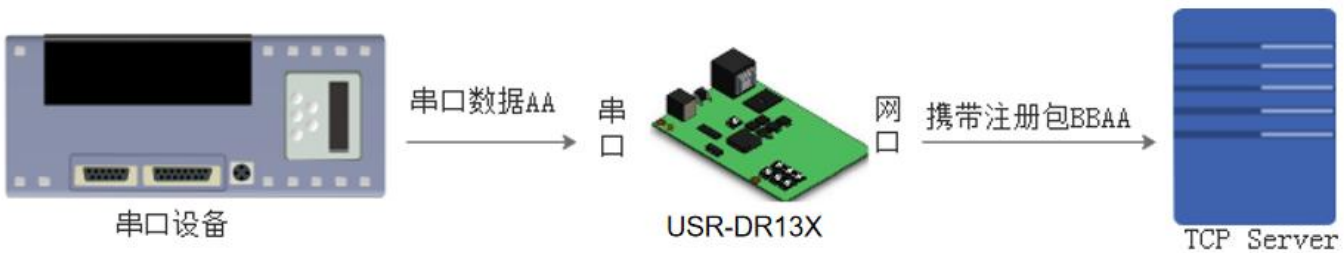


图 61. 数据携带注册包

当前状态

网络参数

端口参数

MQTT网关

边缘网关

点表参数

通用功能

系统参数

模块管理

参数

心跳包使能：

串口心跳包

心跳时间：

30

 (s)(1~65535)

心跳包类型：

自定义

心跳包数据格式：

ASCII

心跳包数据：

heartbeat

注册包使能：

自定义

注册包发送方式：

连接发送

注册包数据格式：

ASCII

注册包数据：

register

Reset：☐

Link：☐

Index：☐

类RFC2217：☒

清除缓存数据：☐

保存

取消

帮助

自定义心跳包：

默认心跳时间为30s，支持16进制输入，不支持汉字，发送方向可随意选择，最长40字节。

自定义注册包：

默认不开启，网页支持16进制输入，不支持汉字，最长40字节。

图 62. 注册包设置

3.7.3. Reset 功能

Reset 功能主要用于当 DR13X 程序跑飞或者死机情况下无法正常建立链接而无法正常工作 的情况下，通过重启来实现模块初始化，从而使 DR13X 在一定情况下恢复正常工作。当 DR13X 作为 TCP Client 时，DR13X 会主动连接 TCP Server。当启动 Reset 功能后，DR13X 尝试连接 TCP Server 端 30 次，仍无法建立连接时，DR13X 会自动重启。功能开关如下：

当前状态

网络参数

端口参数

MQTT网关

边缘网关

点表参数

通用功能

系统参数

模块管理

参数

心跳包使能：

串口心跳包

心跳时间：

30

 (s)(1~65535)

心跳包类型：

自定义

心跳包数据格式：

ASCII

心跳包数据：

heartbeat

注册包使能：

自定义

注册包发送方式：

连接发送

注册包数据格式：

ASCII

注册包数据：

register

Reset：☐

Link：☐

Index：☐

类RFC2217：☒

清除缓存数据：☐

保存

取消

帮助

自定义心跳包：

默认心跳时间为30s，支持16进制输入，不支持汉字，发送方向可随意选择，最长40字节。

自定义注册包：

默认不开启，网页支持16进制输入，不支持汉字，最长40字节。

图 63. Reset 功能设置

有人物联网

可靠 易用 价格合理

联网找有人，靠谱

38

3. 7. 4. INDEX 功能

注：Index 功能无法和 Modbus 功能同时使用，在使用 Modbus 网关功能下，该功能失效。

Index 功能主要是解决 TCP Sever 模式下，用户有多个客户端连接 DR13X 并同时收发数据时，不能区分数据来源或不能发送到特定客户端的情况。

DR13X 作为 TCP Server 时，最多可以同时建立 16 个连接，最大值可在 1-16 范围内任意设置，此处以默认值 16 个为例。Server 同时向 16 个 Client 发送数据，或者 Server 接收 Client 数据时，开启 Index 功能后，将会对不同的客户端进行标号区分，客户可以根据信息标号来区分接收数据的数据源，同时也可以根据协议将不同的信息发送到特定的客户端。

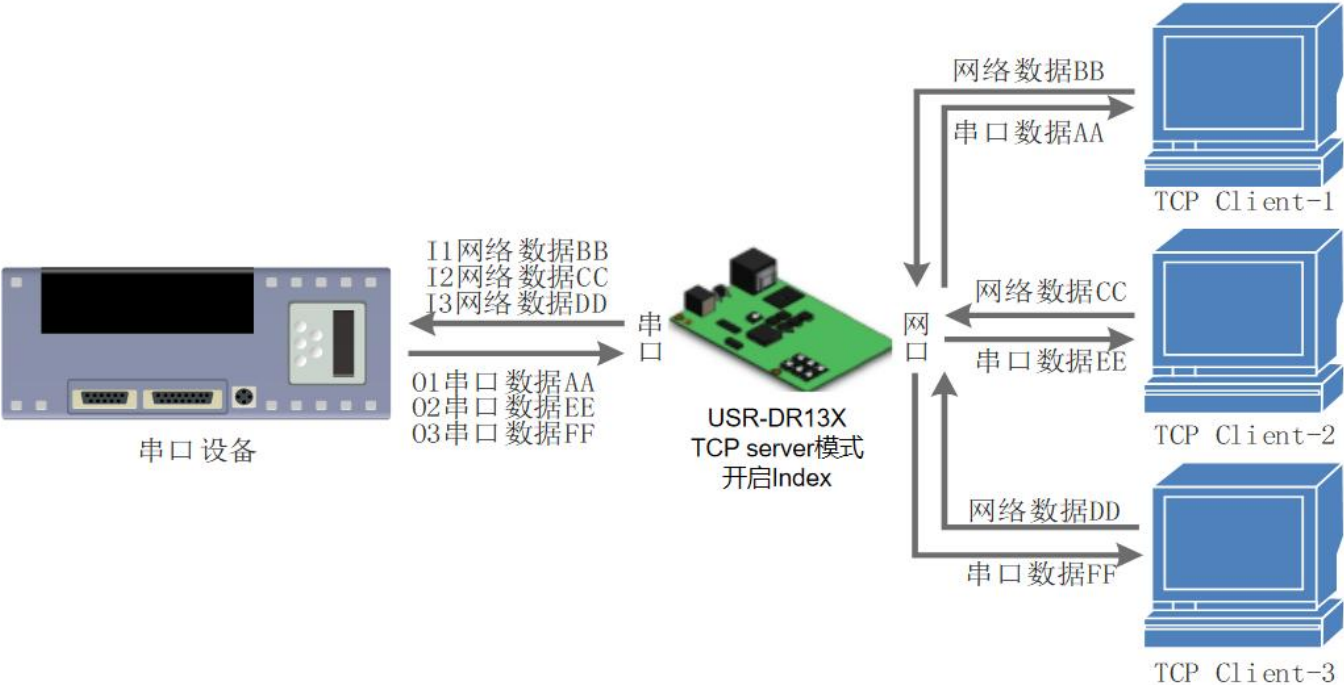


图 64. Index 功能示例图

设置示意图如下图：

当前状态

网络参数

端口参数

MQTT网关

边缘网关

点表参数

通用功能

系统参数

模块管理

参数

心跳包使能：

串口心跳包

心跳时间：

30

 (s)(1~65535)

心跳包类型：

自定义

心跳包数据格式：

ASCII

心跳包数据：

heartbeat

注册包使能：

自定义

注册包发送方式：

连接发送

注册包数据格式：

ASCII

注册包数据：

register

Reset：☐

Link：☐

Index

：☐

类RFC2217：☒

清除缓存数据：☐

保存

取消

帮助

自定义心跳包：

默认心跳时间为30s，支持16进制输入，不支持汉字，发送方向可随意选择，最长40字节。

自定义注册包：

默认不开启，网页支持16进制输入，不支持汉字，最长40字节。

图 65. Index 设置示意图



图 66. Index 传输示意图

3.7.5. 类 RFC2217 功能

类 RFC2217，是简化版的 RFC2217 协议，配合虚拟串口能够动态改变 DR13X 的串口参数，实现和变串口参数（波特率、数据位、校验位等）的设备通讯。

开启该功能后，使用 USR-VCOM 虚拟串口软件也开启类 RFC2217 功能，则可以实现计算机上的应用软件串口波特率与 DR13X 的串口波特率自动匹配适应，而无需关注串口的波特率设置，设置示意图如下：

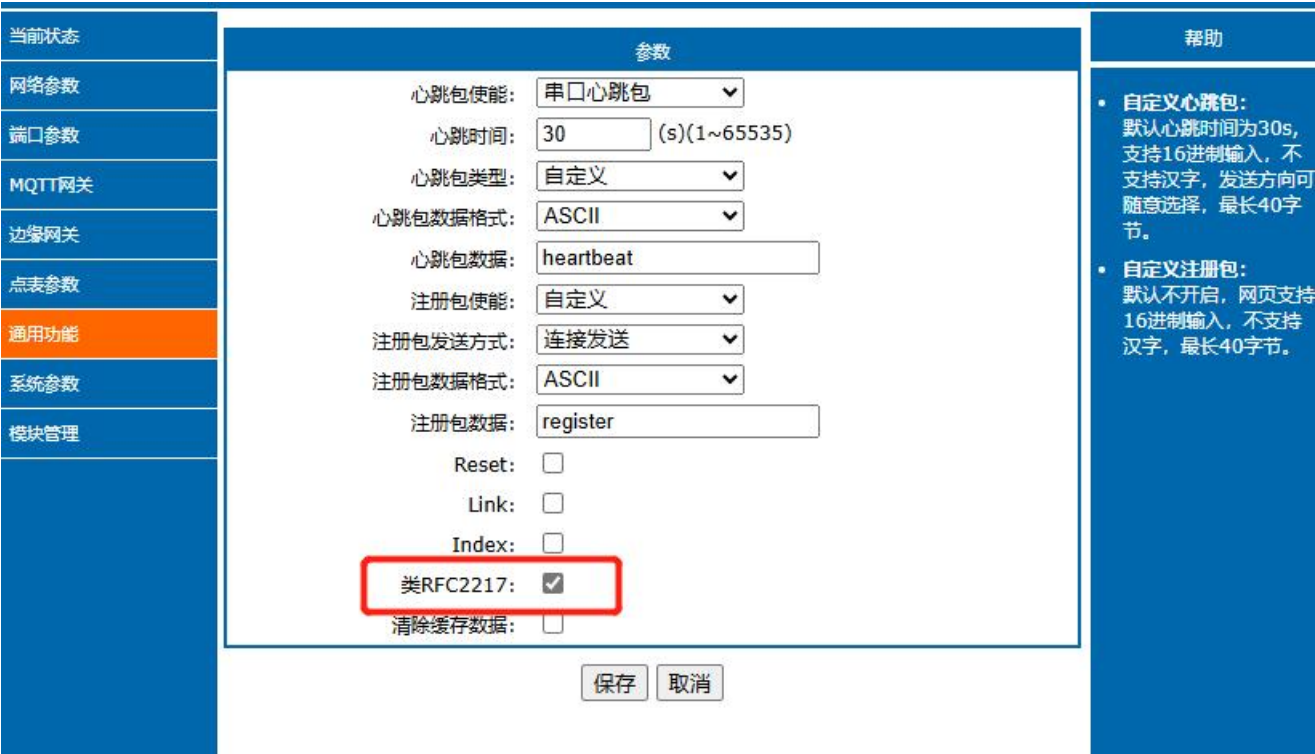


图 67. 类 RFC2217 功能设置

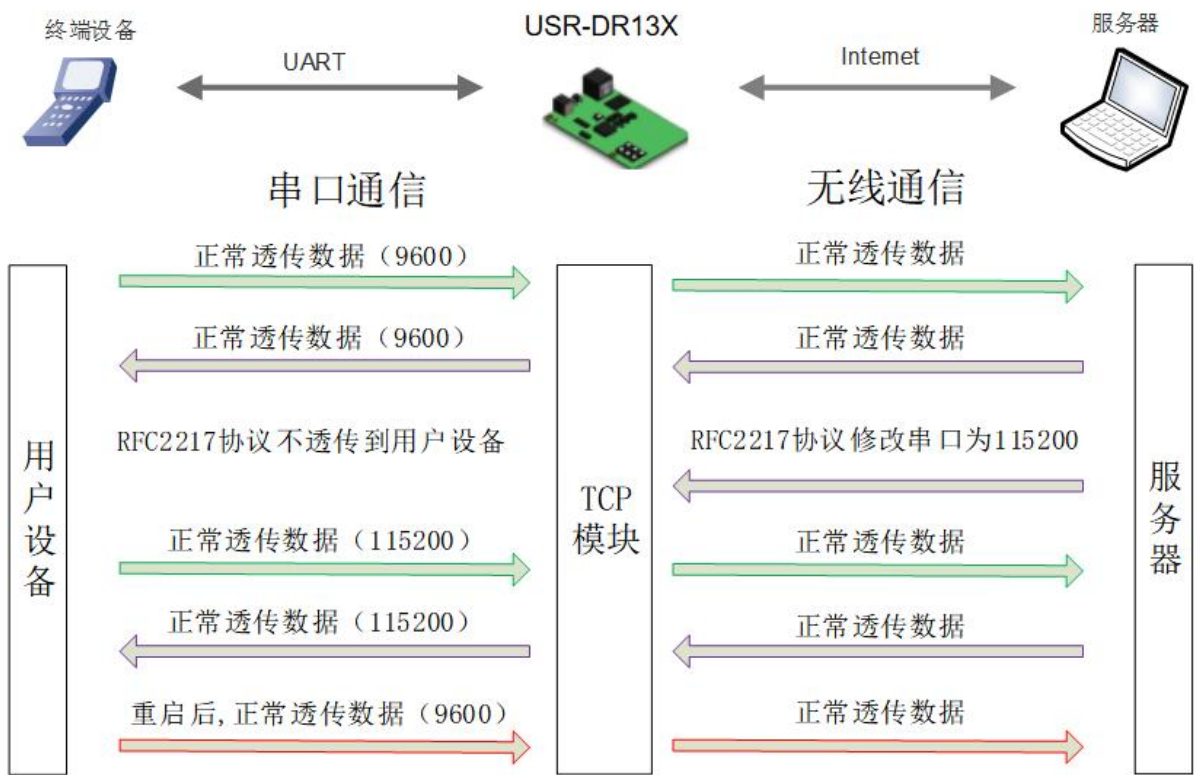


图 68. RFC2217 功能框图

3. 7. 6. 清除缓存数据

当 TCP 连接未建立时，串口接收的数据将会被放在缓存区，DR13X 串口接收缓存是 1Kbyte，当 TCP 连接建立后，串口缓存数据可以根据客户需求设置是否清理。该功能默认为不清理。Httpd Client 模式下或者 TCP Client 模式下开启短连接功能时，清除缓存数据功能失效。设置示意图如下：

当前状态	参数	帮助
网络参数	心跳包使能: 串口心跳包	- 自定义心跳包: 默认心跳时间为30s, 支持16进制输入, 不支持汉字, 发送方向可随意选择, 最长40字节。 - 自定义注册包: 默认不开启, 网页支持16进制输入, 不支持汉字, 最长40字节。
端口参数	心跳时间: 30 (s)(1~65535)	
MQTT网关	心跳包类型: 自定义	
边缘网关	心跳包数据格式: ASCII	
点表参数	心跳包数据: heartbeat	
通用功能	注册包使能: 自定义	
系统参数	注册包发送方式: 连接发送	
模块管理	注册包数据格式: ASCII	
	注册包数据: register	
	Reset: ☐	
	Link: ☐	
	Index: ☐	
	类RFC2217: ☒	
	清除缓存数据: ☐	
	保存 取消	

图 69. 清除缓存数据设置图示

3.7.7. 有人云功能



图 70. 有人云功能

有人云主要是为解决设备与设备、设备与上位机（Android、IOS、PC）之间相互通信而开放的平台。有人云主要用来透传数据，接入设备几乎不需做修改便可接入实现远程透传数据。有人云适用于远程监控、物联网、车联网、智能家居等领域，因此 USR-TCP232-DR13X 接入有人云功能。关于有人云的相关信息请浏览 cloud.usr.cn 获取更多资料。设置示意图如下：

图 71. 有人云设置图示

3.7.8. 自定义网页功能

自定义网页功能可以将自己风格的网页代码升级到设备中。方便用户使用。该功能的使用需要借助网页升级包，其中包括两种网页代码（DR13X 默认的网页代码和 DR13X 中性网页代码）、网页升级工具。

下载网址：<https://www.usr.cn/Download/460.html>



图 72. 自定义网页升级包

升级自定义网页时需要注意，Destination IP 设置为 DR13X 的 IP 地址，最终的网页文件名必须为“fs”，产品选择“M0”，确认好之后点击“Upgrade”即可升级。

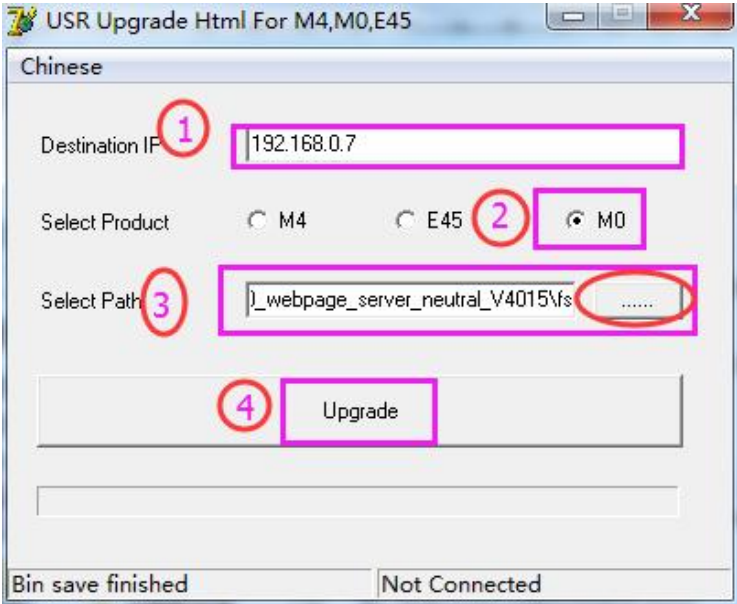


图 73. 自定义网页升级软件使用说明

3. 7. 9. 无数据重启

无数据重启（超时重启）功能主要用于保证 DR13X 长期稳定工作，当网口长时间接收不到数据，或者网络长时间未接收到数据时，DR13X 将在超出设定时间后重启，从而避免异常情况对通信造成影响。无数据重启的时间可以通过网页设置，该功能的正常工作时间设置为 60~65535s，默认值为 3600s。设置时间小于 60s 时，默认置零，即关闭该功能。设置示意图如下：

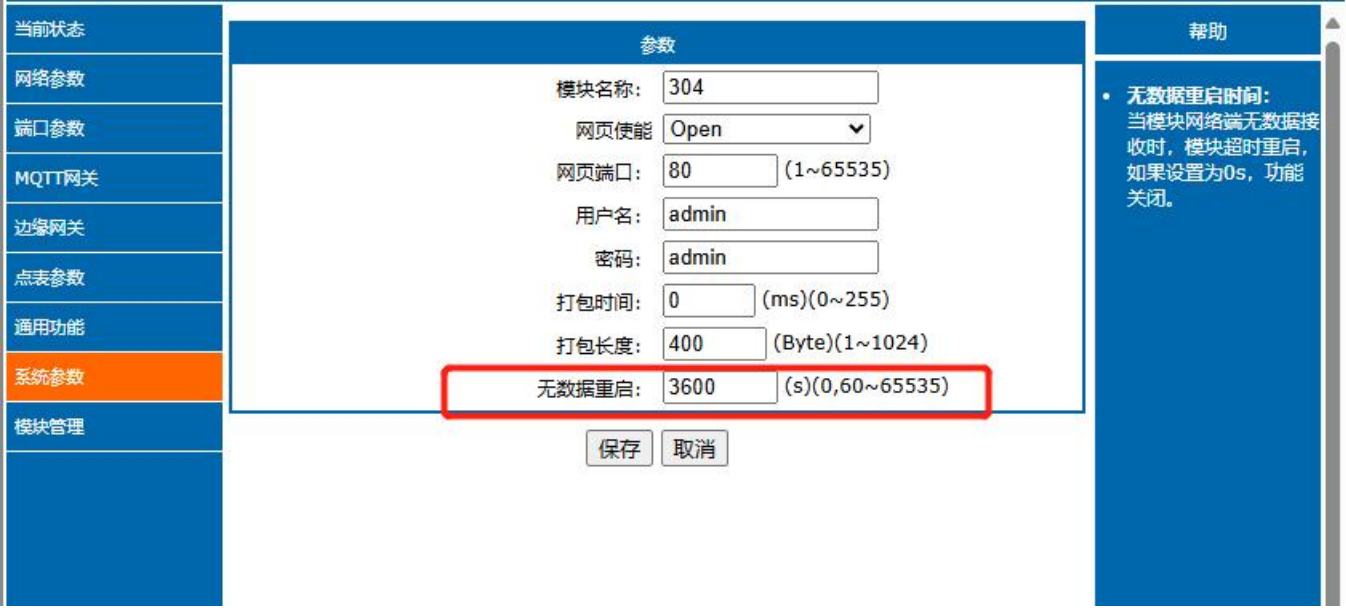


图 74. 无数据重启功能

4. 联系方式

公 司：济南有人物联网技术有限公司

地 址：山东省济南市历下区茂岭山三号路中欧校友产业大厦 12、13 层有人物联网

网 址：<http://www.usr.cn>

用户支持中心：<http://im.usr.cn>

邮 箱：sales@usr.cn

电 话：4000-255-652 或者 0531-66592361

有人定位：可靠的智慧工业物联网伙伴

有人愿景：成为工业物联网领域的生态型企业

有人使命：连接价值 价值连接

产品理念：可靠 易用 价格合理

企业文化：有人在认真做事！

5. 免责声明

本文档提供有关 USR-DR13X 产品的信息，本文档未授予任何知识产权的许可，并未以明示或暗示，或以禁止发言或其它方式授予任何知识产权许可。除在其产品的销售条款和条件声明的责任之外， 我公司概不承担任何其它责任。并且，我公司对本产品的销售或使用不作任何明示或暗示的担保，包括对产品的特定用途适用性，适销性或对任何专利权，版权或其它知识产权的侵权责任等均不作担保。本公司可能随时对产品规格及产品描述做出修改，恕不另行通知。

6. 更新历史

文件版本	更新内容	更新时间
V1.0.0	创建文档	2023-08-17
V1.0.1	修改描述勘误	2023-11-30
V1.0.2	修改描述勘误	2024-03-08
V2.0.0	版本迭代，增加 MQTT、边缘计算等功能	2024-11-08

可信赖的智慧工业物联网伙伴

天猫旗舰店: <https://youren.tmall.com>

京东旗舰店: <https://youren.jd.com>

官方网站: www.usr.cn

技术支持工单: im.usr.cn

战略合作联络: ceo@usr.cn

软件合作联络: console@usr.cn

电话: 0531-66592361

地址: 济南市历下区茂岭山三号路中欧校友产业大厦 12、13 层



关注有人微信公众号

登录商城快速