

Smart Gateway

【智嵌物联】智能网关型串口服务器使用手册



【智嵌物联】智能网关型串口服务器产品系列

修订历史

版本	日期	原因
V1.0	2022.03.01	发布文档
V1.1	2022.06.11	1、增加 4G 联网介绍 2、增加 GPS/北斗定位介绍 3、增加 TLS 证书加密介绍
V1.2	2022.09.19	1、完善主动轮询，JSON 上报格式描述

目 录

1. 关于手册	1
2. 快速使用说明	2
2.1 功能简介	2
2.2 功能特点	2
2.3 快速使用说明	3
2.3.1 使用前硬件准备.....	3
2.3.2 使用前软件准备.....	3
2.3.3 设备默认参数测试步骤.....	4
3. 设备工作模式	9
3.1 设备网络工作模式.....	9
3.1.1 TCP SERVER	9
3.1.2 TCP CLIENT	11
3.1.3 UDP SERVER	12
3.1.4 UDP CLIENT	14
3.1.5 HTTP CLIENT	15
3.1.6 MQTT.....	16
3.1.7 虚拟串口.....	18
3.2 设备串口工作模式.....	19
3.2.1 透传	19
3.2.2 ModBus TCP/RTU 互转（从）	19
3.2.3 ModBus TCP/RTU 互转（主）	21
3.2.4 一问一答.....	23
3.2.5 数据分发.....	25
4. 4G 联网	28
举例一：办公电脑通过设备联网.....	29
举例二：远程传输网络摄像头数据.....	31
举例三：主动采集网口设备（如 PLC）数据，并将数据上云	32
举例四：通过 4G 读取串口设备的数据	37
举例五：通过 4G 采集并控制设备的 DI/DO	39
5. 设备特色功能	47
5.1 支持 8 DI、8 路 DO	47
5.2 灵活设置设备的数据通道.....	47
5.3 可同时支持 24 个 socket、http、MQTT 连接.....	47
5.4 网络心跳包	47
5.5 网络注册包	48
5.6 网页配置	48
5.7 RTC 时钟	49
5.8 TF 卡保存数据.....	49
5.9 GPS/北斗定位.....	50
5.10 支持 TLS 加密及证书加密	50
5.11 网络无数据设备自动重启.....	51
5.12 自设波特率	51

5.13	NTP 校时	52
5.14	跨路由搜索设备	53
5.15	支持配置参数导入、导出	54
5.16	调试日志	54
5.17	二次开发	55
6.8	8 路 DI、8 路 DO 控制协议	56
6.1	地址域 (Additional address)	56
6.2	功能码 (Function code)	56
6.3	8 路 DI/DO 控制举例	57
6.3.1	通过配置软件测试 DI/DO 功能	57
6.3.2	通过 Modbus Poll 工具测试 DI/DO 功能	57
7.	ModBus 主动轮询	61
7.1	Modbus 主动轮询原理	61
7.2	Modbus 主动轮询—数据上报/下发举例	63
7.2.1	ModBus 格式上传与下发	63
7.2.2	JSON 格式上传与下发	68
8.	设备恢复出厂	71
9.	固件升级	73
	销售网络	74

1. 关于手册

用户可以通过该手册了解怎样配置及使用智嵌物联的 Smart Gateway 网关型串口服务器系列产品。本手册内介绍的设备功能，除非特别说明，均适用于 Smart Gateway 系列产品。

本手册涉及到的测试演示实验，均以 ZQWL-GW1800M 为例，其他型号的产品用法一样，只需要配置不同串口（PORT）的参数即可。

1. 命名规则：

智嵌物联 Smart Gateway 产品线命名规则如图 1.1 所示。

网关型串口服务器（GW）系列带外壳产品命名规则

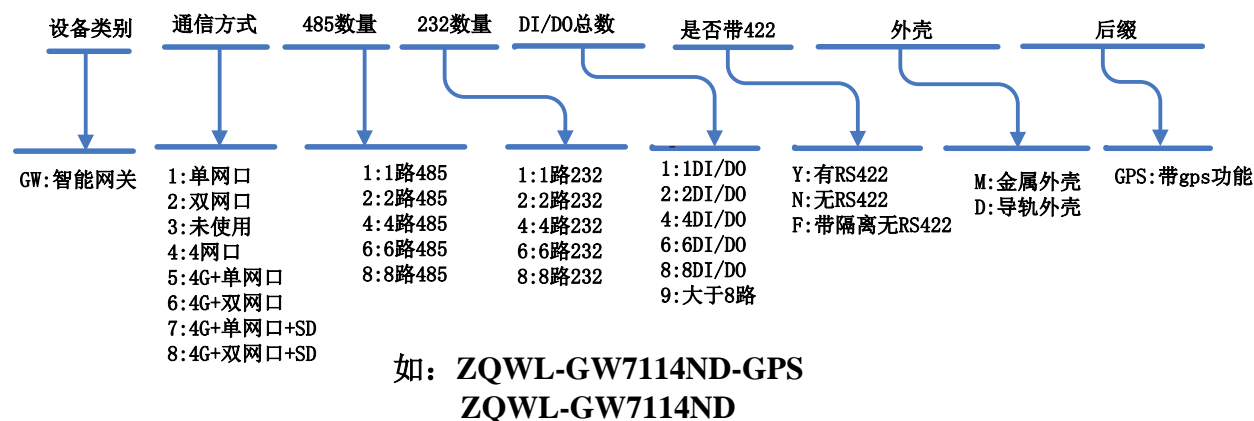


图 1.1 智嵌 Smart Gateway 网关型串口服务器产品线命名规则

2. 相关资料下载：

二次开发资料下载：[点击下载](#)

配置工具及测试软件下载：[点击下载](#)

MQTTX 工具下载：[点击下载](#)

虚拟串口软件及使用说明下载：[点击下载](#)

3. 产品价格查询及购买连接：

天猫商城：[点击查看](#)

京东商城：[点击查看](#)

2. 快速使用说明

2.1 功能简介

智嵌物联自主研发的 Smart Gateway（智能网关）系列产品，内部集成了 TCP/IP 协议栈，可实现串口到网络数据的双向透明传输、ModBus 协议转换等功能。设备通过简单的配置，即可轻松实现嵌入式用户串口设备的联网、上云等功能。广泛应用于机房监控、环境监控、智能交通、道闸控制、智能快递柜等行业。

2.2 功能特点

- ◆ 产品系列型号丰富，1 路串口~24 路串口、4G、网口；
- ◆ 最多支持 16 个开关量输入输出接口（部分型号支持）；
- ◆ 支持 TF 卡（部分型号支持）；
- ◆ 支持 RTC 时钟（部分型号支持）；
- ◆ 支持 GPS/北斗定位（部分型号支持）；
- ◆ 支持 TLS 加密及证书加密；
- ◆ 支持 NTP 授时，只要设备能连接外网，设备就可以从网络上获取时间；
- ◆ 支持无数据时，设备自动重启；
- ◆ 灵活配置数据通道，简单配置即可实现多个数据通道，且相互独立；
- ◆ 支持 24 个网络通道同时连接，每个网络通道之间相互独立，每个网络通道可支持 64 个 TCP 客户端连接；
- ◆ 支持标准的 MQTT 协议、HTTP、TCP SERVER、TCP CLIENT、UDP SERVER、UDP CLIENT 等网络工作模式；
- ◆ 支持透传、Modbus TCP/RTU 互转（从）、Modbus TCP/RTU 互转（主）、一问一答、数据分发等串口工作模式；
- ◆ 支持 modbus 主动轮询，定时/变化上报，上报格式可配置（自定义 JSON、阿里云 ALINK、modbus 等格式）；
- ◆ 支持自设波特率；
- ◆ 提供虚拟串口软件，可动态修改串口参数，真正实现虚拟串口；
- ◆ 支持接入智嵌云、阿里云、私有云；
- ◆ 支持心跳包、注册包功能；
- ◆ 可使用配置工具进行参数配置，支持跨路由搜索设备；
- ◆ 可使用网页浏览器进行参数配置；
- ◆ 丰富的 LED 状态指示灯、调试日志，快速定位问题；
- ◆ 支持固件升级；
- ◆ 支持二次开发；
- ◆ 支持 OEM、ODM。

2.3 快速使用说明

2.3.1 使用前硬件准备

智嵌物联不同型号的网关设备使用方法是一样的，下面以 ZQWL-GW1800NM 为例，简要介绍下串口数据与网络数据透传测试步骤。

为了测试 ZQWL-GW1800NM，需要以下硬件：

- PC 机一台；
- ZQWL-GW1800NM（或其他型号的网关设备）一台；
- DC12V 1A 电源适配器一个；
- 网线一条；
- USB 转 RS232 串口线一条（测试 RS232 功能时用）；
- USB 转 RS485 串口线一条（测试 RS485 功能时用）；



图 2.1 硬件准备

2.3.2 使用前软件准备

在智嵌物联官网下载串口调试助手“UartAssist”、网络调试助手“网络调试助手 V3.8”、设备的配置软件“智嵌物联 gateway 参数配置软件”，如图 2.2 所示。

调试工具下载地址：[点击下载](#)

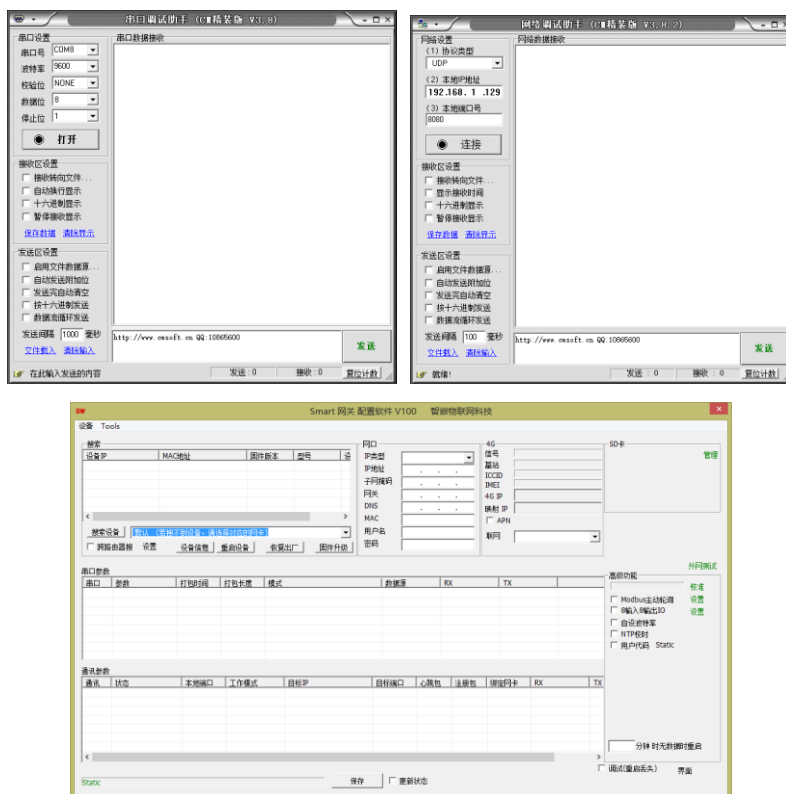


图 2.2 软件准备

2.3.3 设备默认参数测试步骤

不同串口（PORT）的 RS232、RS485、RS422 测试方法一样，下面以设备 PORT1 接口测试为例来说明智能网关设备的测试方法。

4. 连接硬件



图 2.3 硬件连接

- 用网线连接网关设备网口和电脑网口。
- 用 USB 转 RS485 串口线连接电脑的 USB 口和网关设备的 RS485 接口。

用电源适配器为设备上电，上电后观察指示灯是否正常，如

➤ 表 2.1 所示。

表 2.1 设备指示灯意义

指示灯	设备正常时
电源指示灯（PWR）	常亮
运行指示灯（RUN）	闪烁（频率约 1HZ）
网口灯	一个灯常亮，一个灯有数据时会闪烁
串口数据指示灯（PORT1~8）	串口有数据时会闪烁或常亮

5. 设备参数配置

为了能使用户快速的对网关设备有个简单的认识,我们使用网关设备的默认参数进行数据透传测试。智嵌物联网网关设备的默认参数如表 2.2 所示。

表 2.2 串口服务器默认参数

	项目	默认参数
所有路串口参数	波特率	9600bps
	数据位	8
	校验位	NONE
	停止位	1
	串口工作模式	透传
	数据通道	U1~U8 与 N1~N8 数据通道一一对应 N9~N24 默认不启用
网络参数	网络工作模式	TCP SERVER
	设备 IP	192.168.1.253
	端口号	N1~N8 端口号分别是 1030~1037 N9~N24 默认不启用
	网页登录用户名及密码	用户名: admin 密码: admin

6. 确保电脑 IP 与网关设备 IP 在同一网段且不能冲突。检查方法如图 2.4 所示。

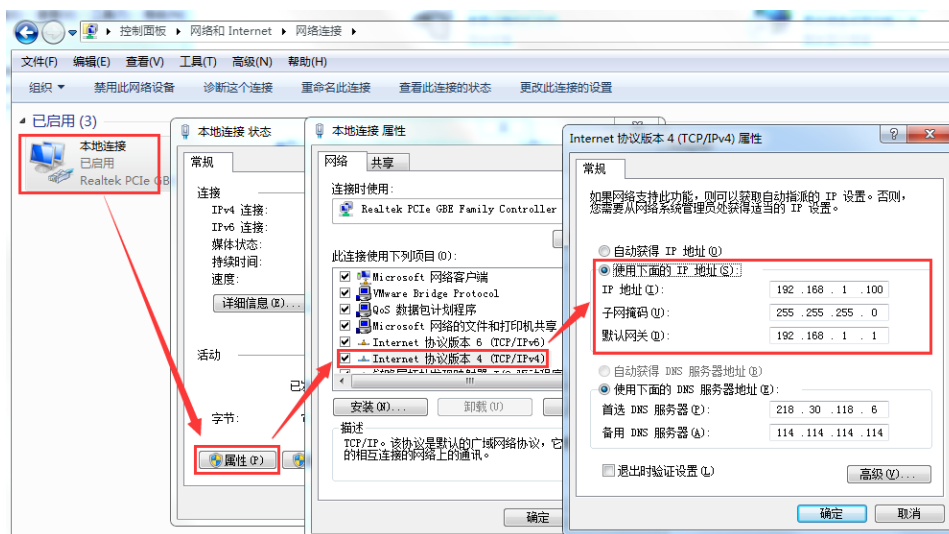


图 2.4 电脑 IP 设置方法

7. 关闭电脑防火墙

如果通信不成功，用户可尝试将电脑关闭防火墙后，再尝试。



图 2.5 关闭电脑防火墙

8. 打开“串口调试助手”

具体设置如图 2.6 所示。串口号可在电脑->管理->设备管理器->端口（COM 和 LPT）中查询 USB 转串口线的 COM 号。

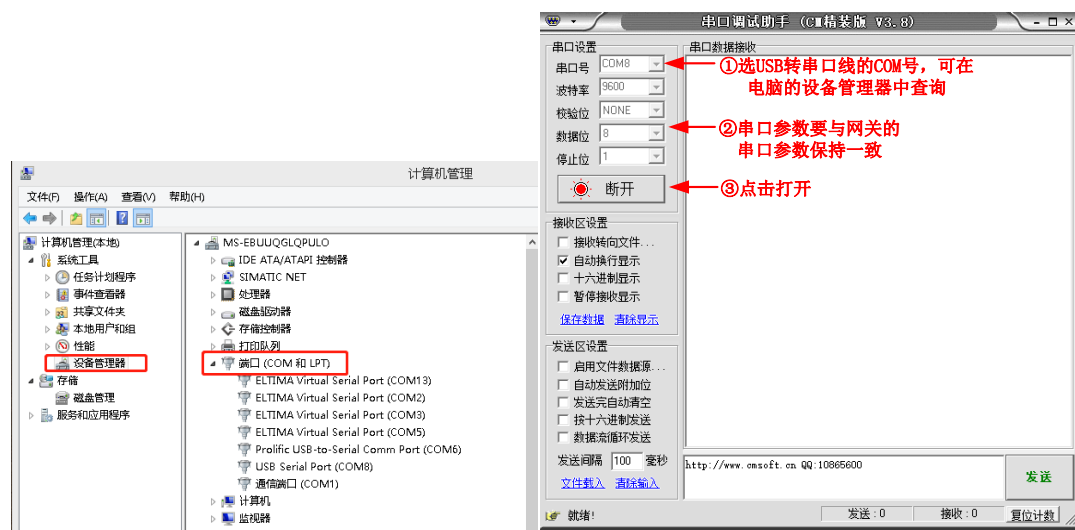


图 2.6 串口调试助手参数设置

9. 打开“网络调试助手”

具体设置如图 2.7 所示。

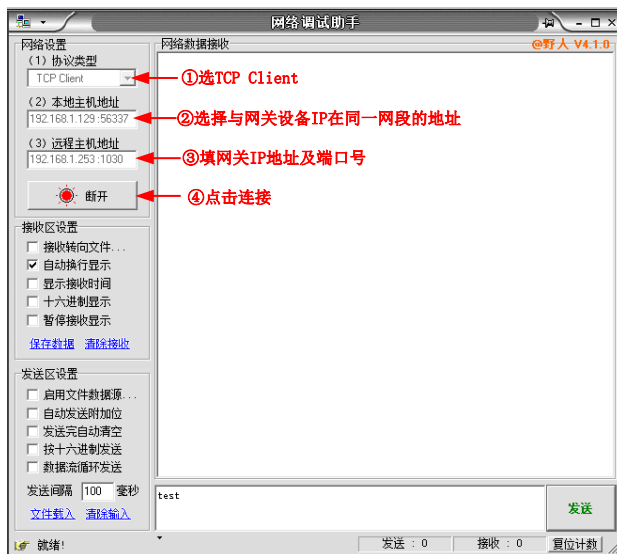


图 2.7 网络调试助手参数设置

10. 收发数据测试

分别点击“网络调试助手”和“串口调试助手”上的【发送】按键，即可实现网络和串口数据的透明传输，如图 2.8 所示。

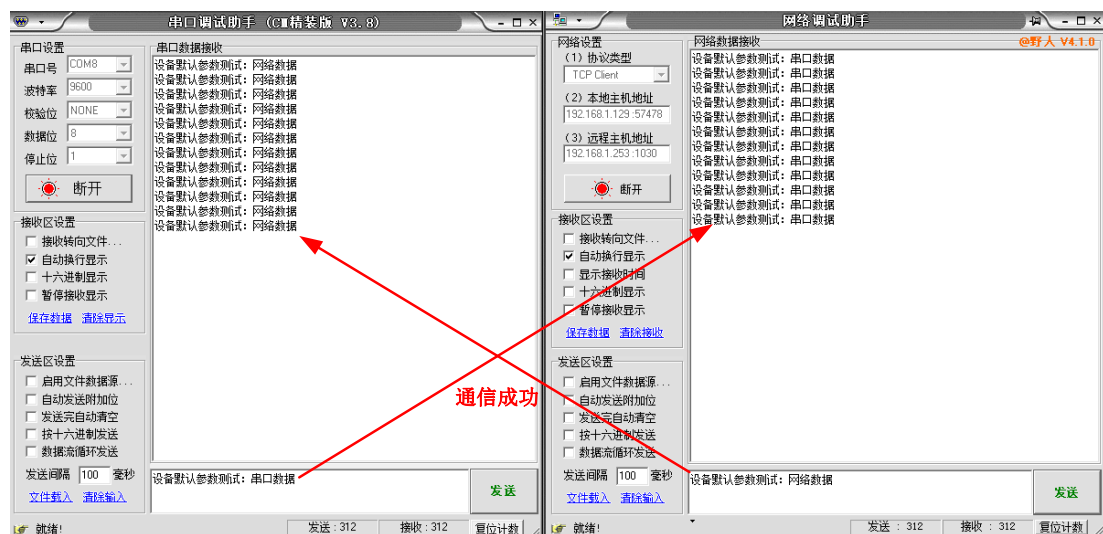


图 2.8 默认参数透传测试

- ① 如按以上步骤操作后，通信失败，请检查电脑防火墙是否关闭，如未关闭，请先关闭防火墙之后再测试。

11. 通过配置软件查看调试数据

为方便用户快速定位问题，设备支持打印所有的数据信息。具体调试方法如图所示。

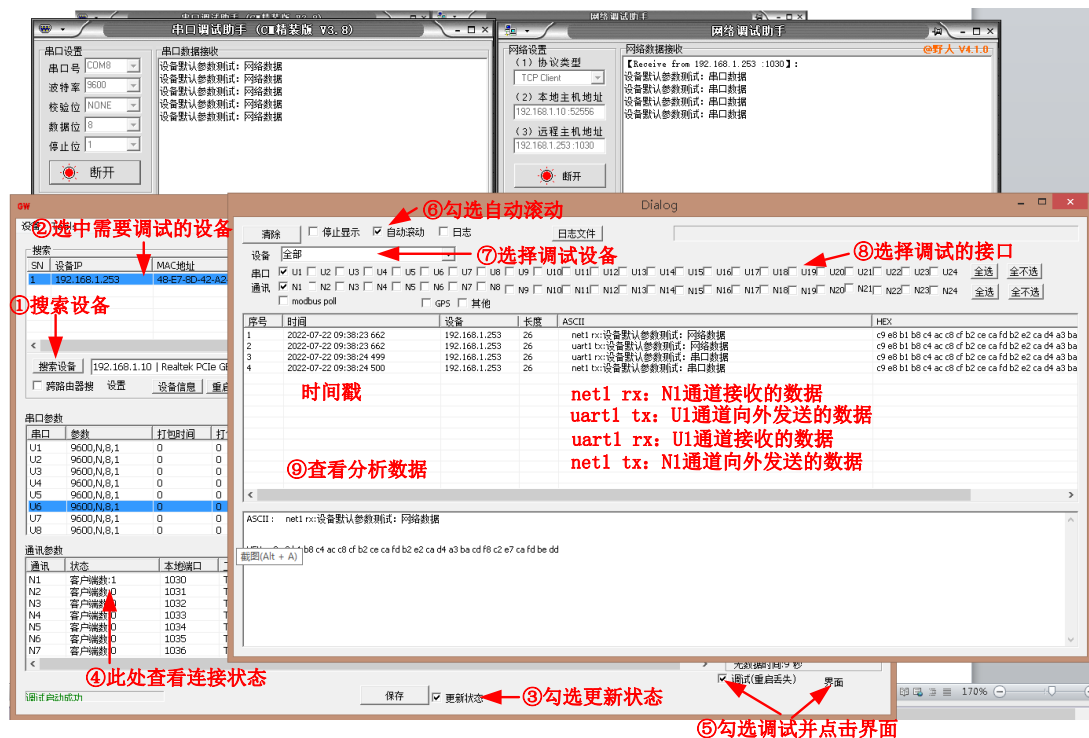


图 2.9 调试信息

3. 设备工作模式

3.1 设备网络工作模式

设备硬件上具有 1 个 LAN 口，1 个 4G 接口。网关设备支持 24 个网络通道（N1~N24）同时连接，且每个网络通道之间相互独立，网络工作模式既可以是 TCP/UDP socket 模式，也可以是 HTTP、MQTT 等。当配置为 TCP SERVER 模式时，该网络通道可同时支持 64 个客户端连接。

设备的每个网络通道的工作模式均支持 TCP_SERVER、TCP_CLIENT、UDP_SERVER、UDP_CLIENT、MQTT、HTTP CLIENT、虚拟串口模式。每个网络通道的工作模式可以设置不同。用户在选择设备网络工作模式时，可参考图 3.1 所示的流程图。

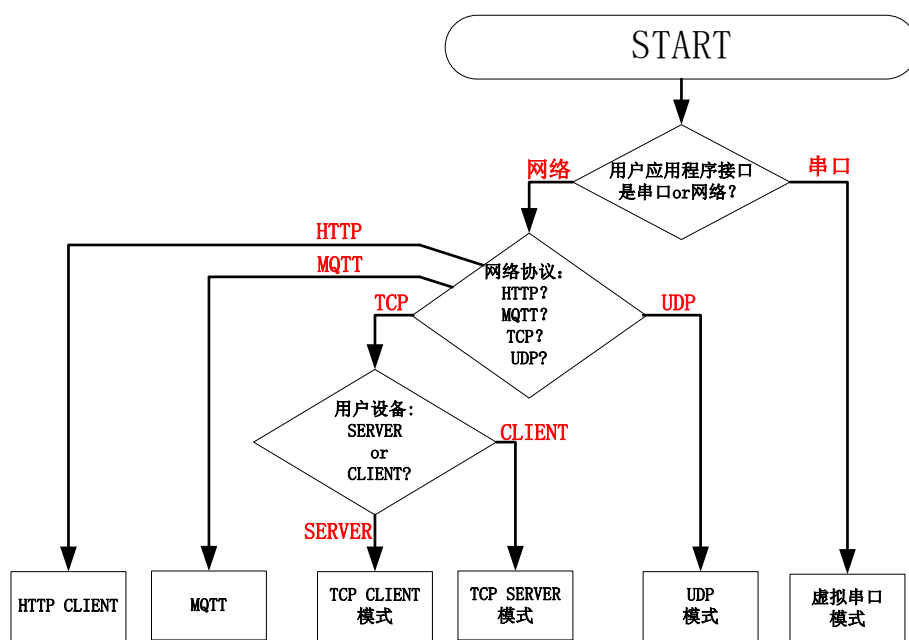


图 3.1 网络工作模式选择流程

3.1.1 TCP SERVER

在 TCP 服务器（TCP Server）模式下，设备始终等待用户的客户端（TCP Client）设备的连接，在与客户端建立 TCP 连接后即可进行双向数据通信。如图 3.2 所示，在 TCP_SERVER 工作模式下，数据传输步骤如下：

1. 建立连接。用户的 TCP 客户端向网关设备发送建立连接请求。
2. 连接一旦建立，TCP 客户端与网关设备之间即可进行数据的双向传输。



图 3.2 TCP SERVER 模式工作原理

- ① 设备可同时支持 24 个网络通道（N1~N24），每个网络通道之间相互独立，可设置不同的参数，且每个网络通道在 TCP SERVER 模式下支持 64 个客户端连接。
- ② 在 TCP_SERVER 模式下，当有客户端连接成功后，设备自动开启“keep alive”保活机制：如果 TCP 没有数据收发后，每隔 20 秒向客户端发送一个“keep alive”数据包来探测 TCP 客户端是否还在，如果客户端不在（没有收到回复），则断开该 TCP 连接，释放资源以待客户端重新连接。

以设备的第一路网络通道 N1 为例，来说明 TCP Server 模式的配置步骤，其他网络通道（N2~N24）的配置方法一样。

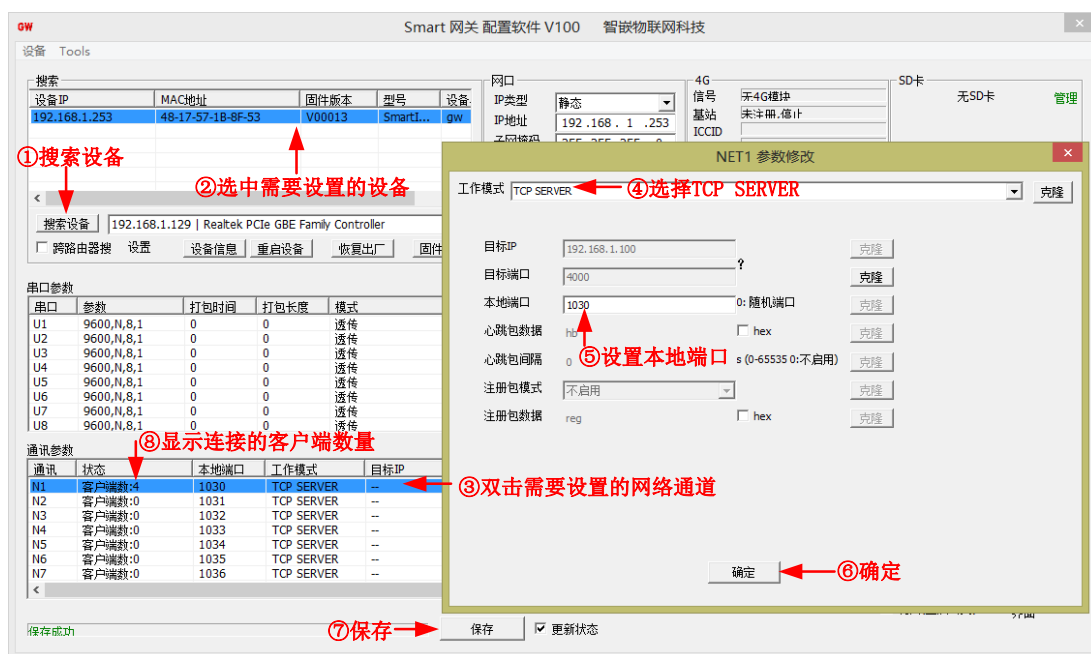


图 3.3 网络通道 N1 参数配置方法

打开 4 个或更多个网络调试助手，模拟用户的客户端设备或软件，同时连接网关设备的第一个网络通道 N1（默认参数：192.168.1.253:1030）；打开一个串口调试助手，模拟用户的串口设备（默认参数：9600,8,N,1）。

配置完之后，分别发送数据，接收框会收到对应的数据。

③选USB转串口线的COM号，可在
电脑的设备管理器中查询

同一个网络通道可支持64个客户端连接



图 3.4 TCP SERVER 模式下连接多个客户端

3.1.2 TCP CLIENT

在 TCP 客户端（TCP Client）模式下，网关设备作为 TCP 客户端，将主动向“目标 IP”和“目标端口”所指定的 TCP 服务器发起连接请求。如果连接不成功，TCP 客户端将会根据设置的连接条件不断尝试与 TCP 服务器建立连接。在与 TCP 服务器端建立 TCP 连接后即可进行双向数据通信。

1. 被配置为 TCP Client 模式的网关设备向 TCP 服务器发送建立连接请求。
2. 连接一旦建立，串口服务器与 TCP 服务器之间即可进行数据传输。



图 3.5 TCP CLIENT 模式工作原理

④ 连接成功后，设备自动开启“keep alive”保活机制：如果 TCP 没有数据收发后，每隔 20 秒向服务器发送一个“keep alive”数据包来探测 TCP 服务器是否还在，如果服务器不在（没有收到回复），则断开该 TCP 连接，并向服务器重连。

以设备的第一路网络通道 N1 为例，来说明 TCP CLIENT 模式的配置步骤，其他网络通道（N2~N24）的配置方法一样。



图 3.6 N1 通道配置为 TCP CLIENT 模式

打开 1 个网络调试助手，模拟用户的服务端设备或软件，设备的第一个网络通道 N1 作为客户端，会主动连接网络调试助手。打开一个串口调试助手，模拟接到设备 PORT1 上的串口设备。

④选USB转串口线的COM号，可在
电脑的设备管理器中查询

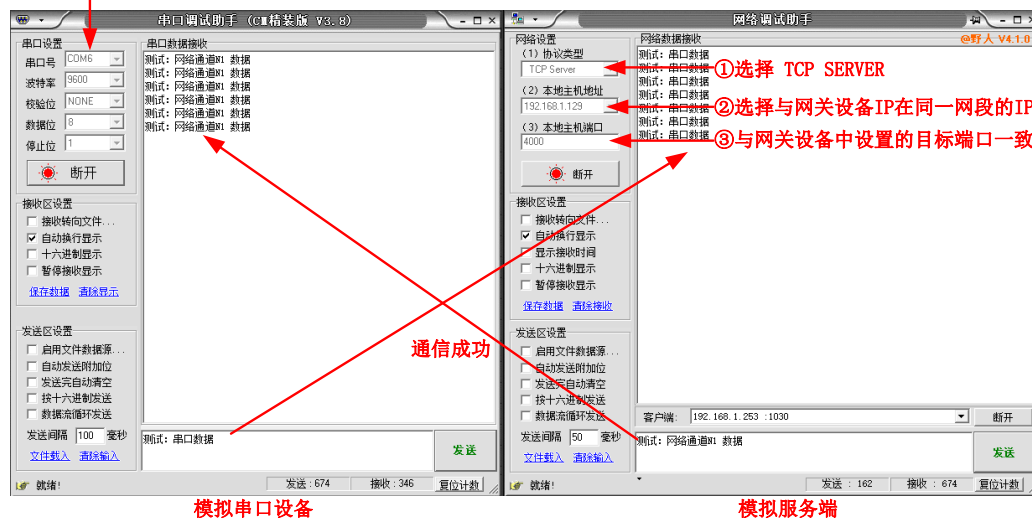


图 3.7 TCP CLIENT 模式通信成功

3.1.3 UDP SERVER

在 UDP SERVER 模式下，串口数据总是发往最后一个与设备通讯的 UDP 对象(IP 和端口)，并且必须先有对方向设备发数据后，设备才能记录这个 IP 和端口号。

该模式的特点是，串口数据可以和不同的 UDP 对象 (IP 和端口) 通讯。

1. 在该模式下，UDP 客户端首先向串口服务器发送一包数据。
2. 串口服务器收到 UDP 客户端的第一包数据后，两者即可进行双向数据传输。



以设备的第一路网络通道 N1 为例，来说明 UDP SERVER 模式的配置步骤，其他网络通道（N2~N24）的配置方法一样。

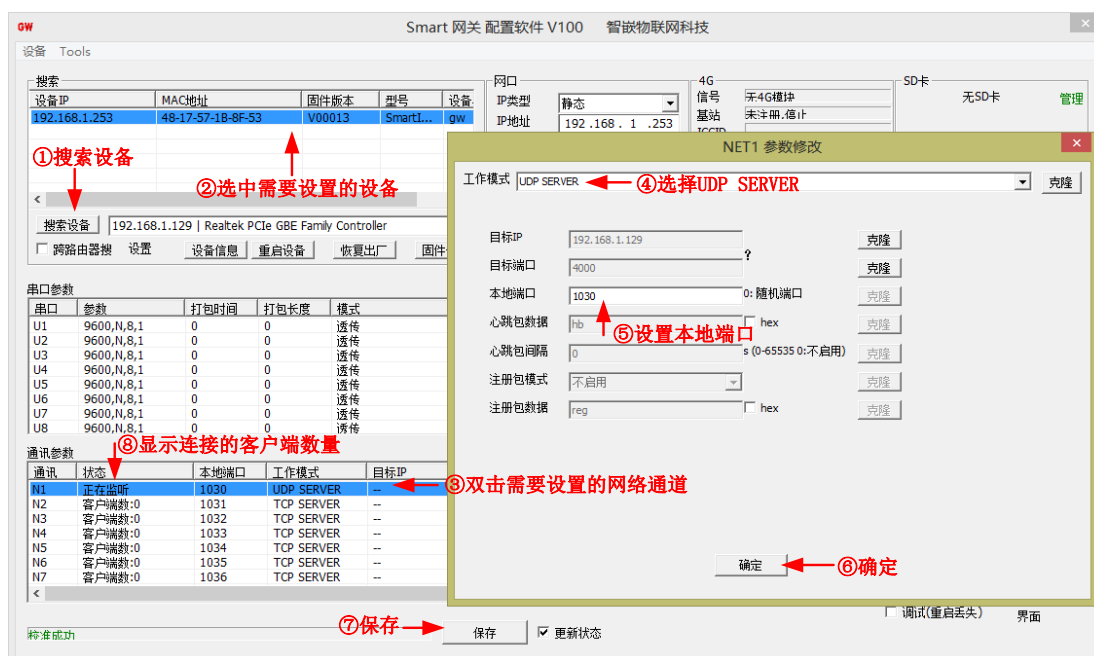


图 3.9 N1 通道配置为 UDP SERVER 模式

打开 1 个网络调试助手，模拟用户的 UDP 客户端设备或软件，同时连接网关设备的第一个网络通道 N1（默认参数：192.168.1.253:1030）。打开一个串口调试助手，模拟接到设备 PORT1 上的串口设备（默认参数：9600,8,N,1）。

⑤选USB转串口线的COM号，可在
电脑的设备管理器中查询

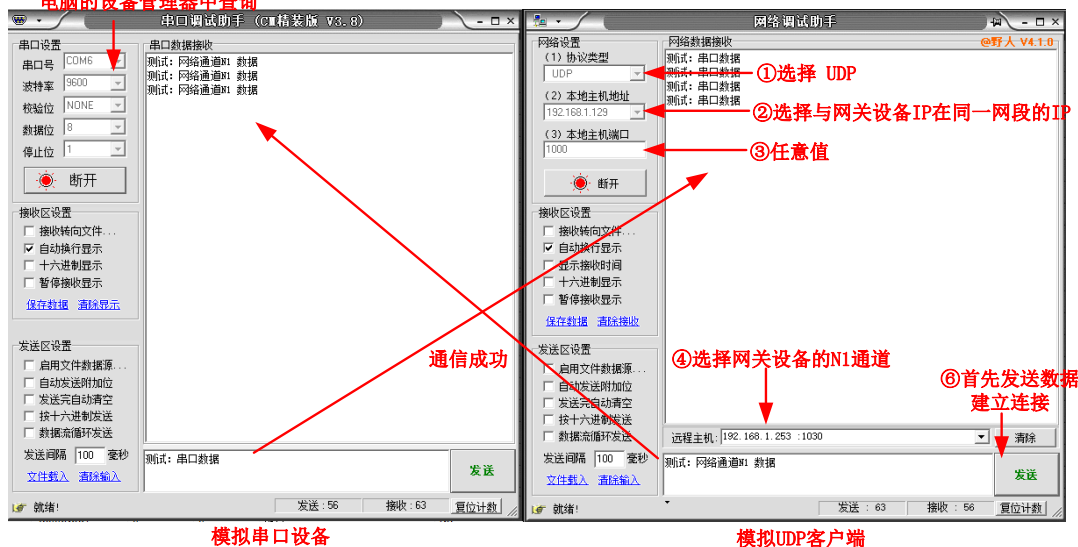


图 3.10 UDP SERVER 模式通信成功

3.1.4 UDP CLIENT

该模式下串口数据总是发往预先设置的“目标 IP”和“目标端口”，并且多个 UDP 对象（IP 和端口）都可以将数据发到串口服务器 IP 和该 PORT 的“本地端口”上，从而转发到该 PORT 口上。

该模式下，支持“目标 IP/域名”为“255.255.255.255”的广播发送。

以设备的第一路网络通道 N1 为例，来说明 UDP CLIENT 模式的配置步骤，其他网络通道（N2~N24）的配置方法一样。

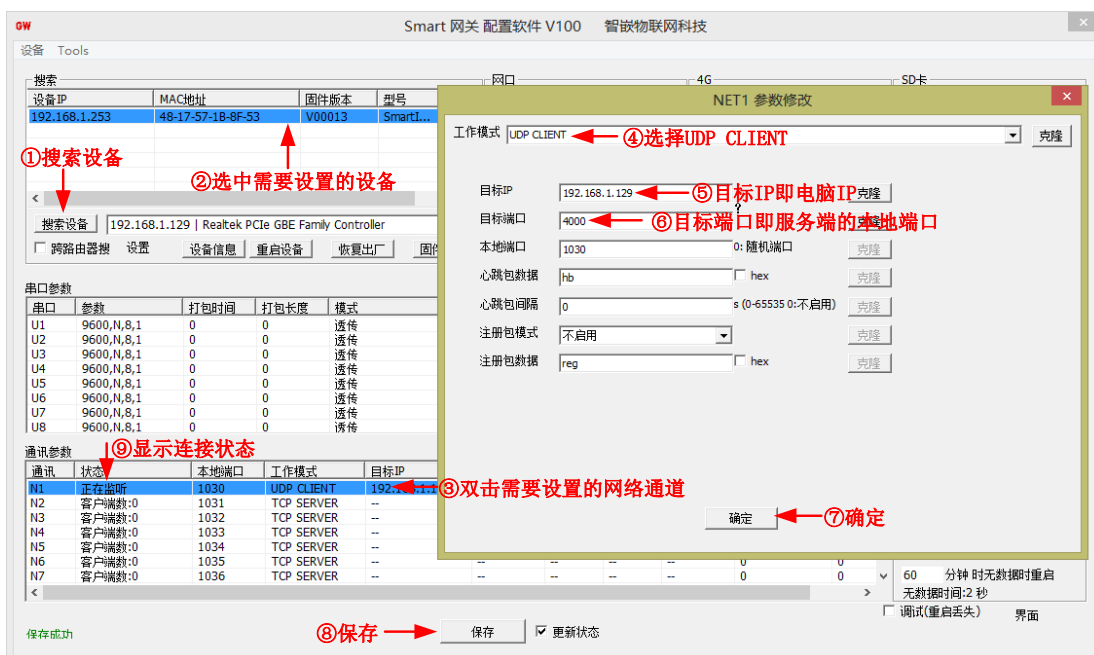


图 3.11 N1 通道配置为 UDP CLIENT 模式

打开一个网络调试助手，模拟用户的 UDP 服务端设备或软件，同时连接网关设备的第一个网络通道 N1。打开一个串口调试助手，模拟接到设备 PORT1 上的串口设备。

⑤选USB转串口线的COM号，可在
电脑的设备管理器中查询

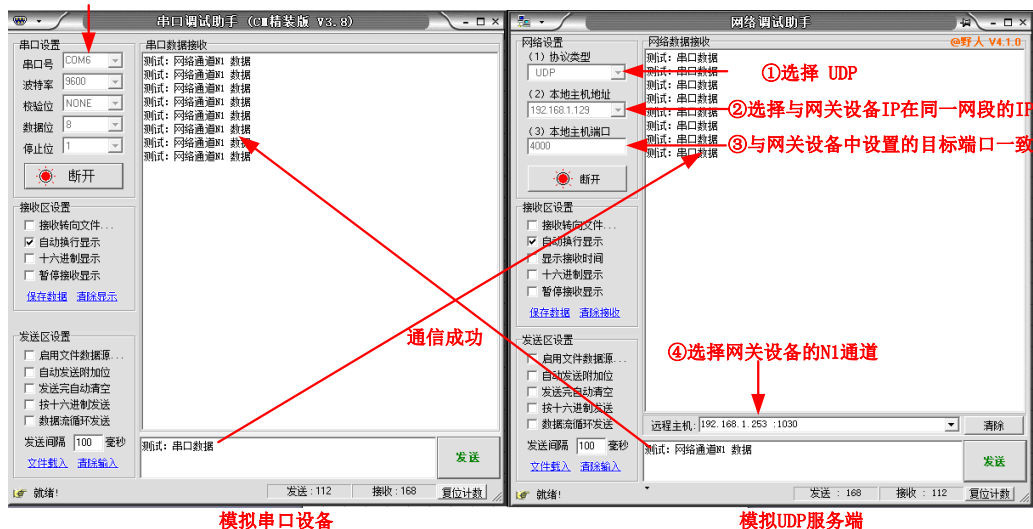


图 3.12 UDP CLIENT 模式通信成功

3.1.5 HTTP CLIENT

支持 HTTP POST/GET。

设备可同时支持连接 24 个 HTTP 服务器，即网络通道 N1~N24 均配置为 HTTP CLIENT 工作模式。

在此模式下，用户的终端设备，可以通过本设备发送请求数据到指定的 HTTP 服务器，然后设备接收来自 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。

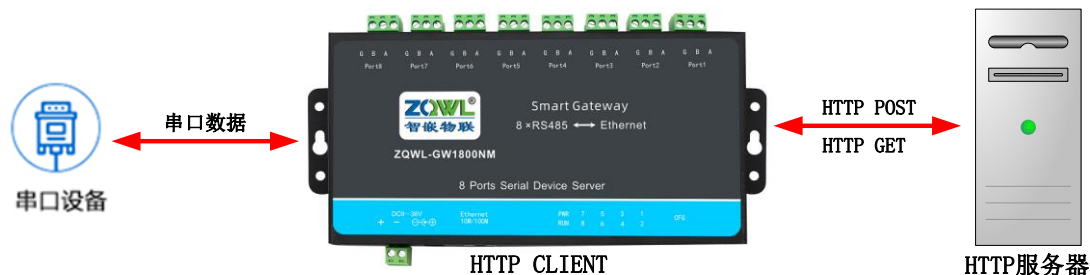


图 3.13 HTTP POST/GET

以设备的第一路网络通道 N1 为例，来说明 HTTP CLIENT 模式的配置步骤，其他网络通道（N2~N24）的配置方法一样。

为方便用户测试 HTTP CLIENT 模式，智嵌物联创建了一个测试专用的 HTTP 服务器：服务器将收到的数据，原路返回。服务器 IP: 47.106.128.28，端口号：80。

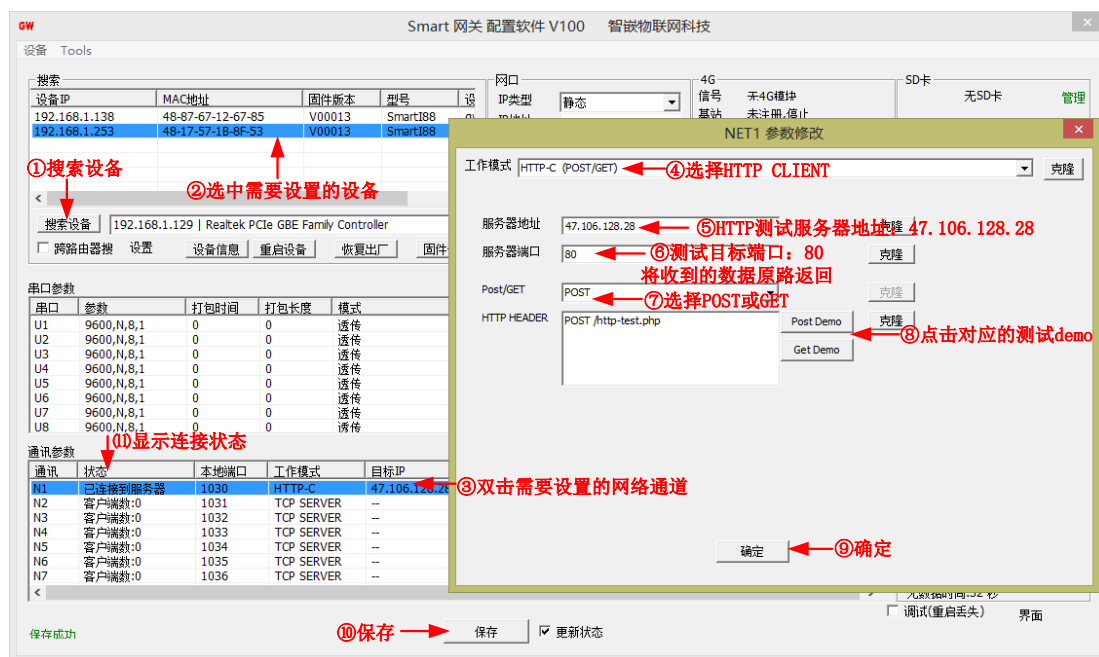


图 3.14 N1 通道配置为 HTTP 模式

打开一个串口调试助手，模拟接到设备 PORT1 上的串口设备。串口调试助手向 HTTP 服务器发送数据，测试服务器将收到的数据原路返回。



图 3.15 HTTP 模式通信成功

3.1.6 MQTT

设备可同时支持连接 24 个 MQTT 服务器，即网络通道 N1~N24 均配置为 MQTT 模式。

设备使用 MQTT 协议连接上 MQTT 服务器后，用户往设备串口发数据，设备把数据转发到用户配置的发布 Topic 里；设备也会接收用户配置的订阅 Topic 里的数据，当接收到订阅 Topic 的数据时，设备将数据转发到串口。

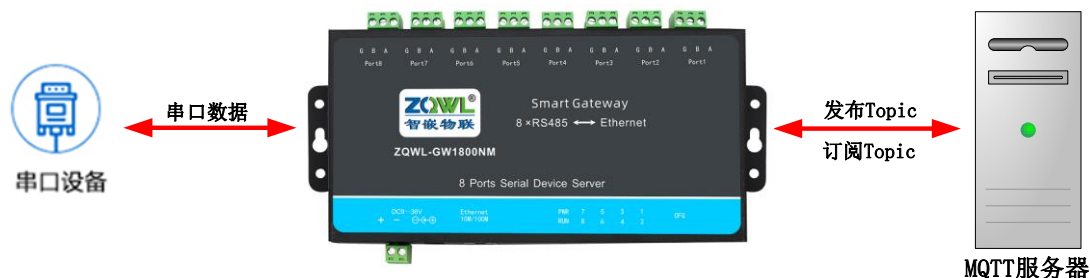


图 3.16 MQTT 应用拓扑

以设备的第一路网络通道 N1 为例，来说明 MQTT 模式的配置步骤，其他网络通道（N2~N24）的配置方法一样。

为方便用户测试设备的 MQTT 功能，智嵌物联创建了一个 MQTT 测试服务器，服务器 IP：39.108.220.80，端口号：8883。

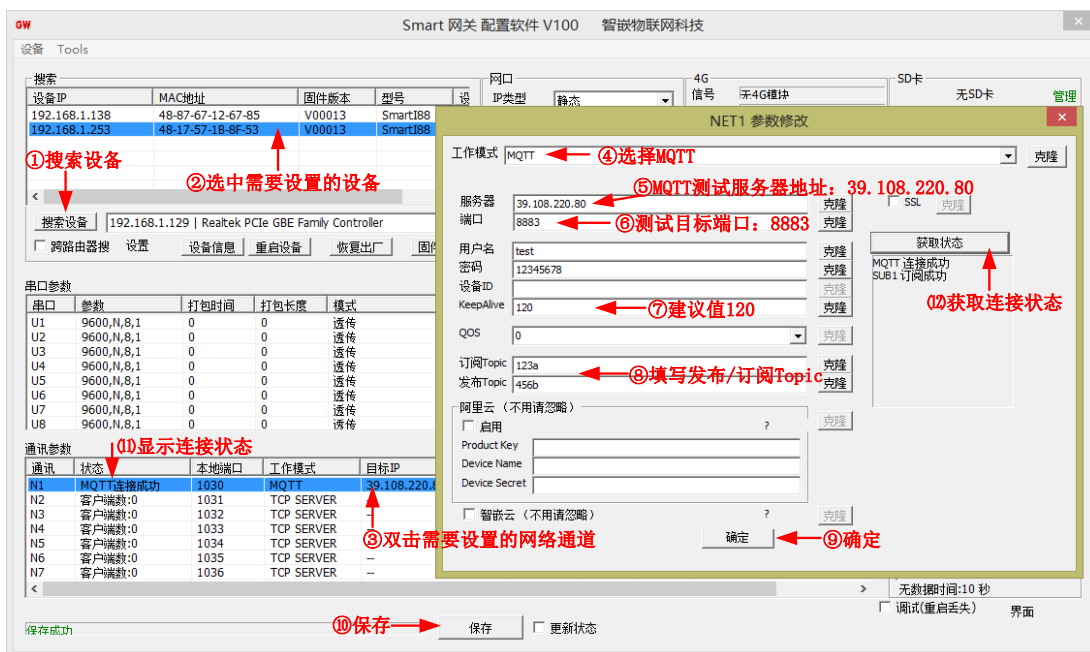


图 3.17 N1 通道配置为 MQTT 模式

打开一个 MQTTX 客户端软件，连接到 MQTT 测试服务器，并创建发布/订阅 Topic。打开一个串口调试助手，模拟接到网关设备 PORT1 上的串口设备。

MQTTX 软件下载地址：[点击下载](#)

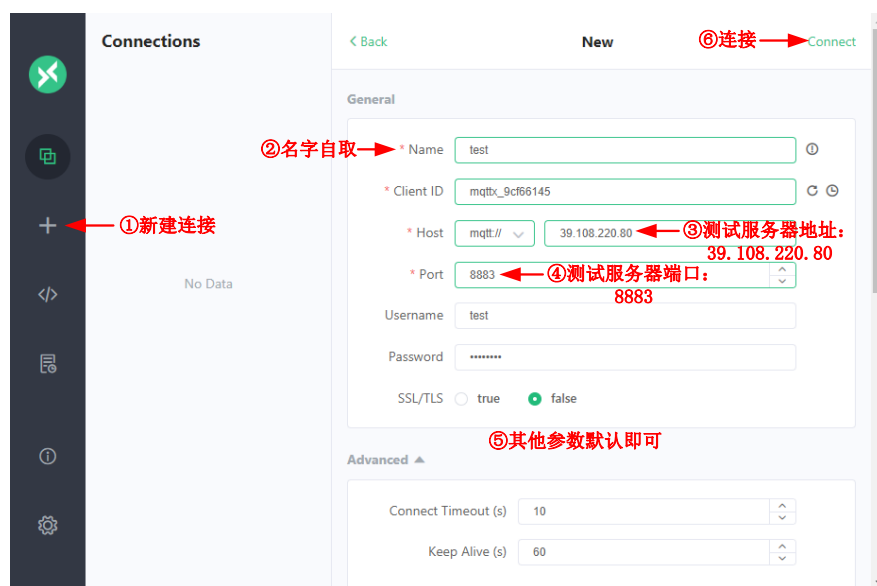


图 3.18 MQTTX 连接测试服务器

④选USB转串口线的COM号，可在
电脑的设备管理器中查询



图 3.19 Topic 收发数据成功

3.1.7 虚拟串口

虚拟串口工作模式是将网口模拟成串口通信，相当于两端都是使用串口通信，所以在使用中，我们要使用网络虚拟串口软件创建虚拟串口。从而 PC 机可以直接操作该串口来完成和串口设备的通讯，减少了 PC 软件的开发难度以及开发时间。虚拟串口软件及使用说明下载地址：[点击下载](#)



图 3.20 虚拟串口工作原理

3.2 设备串口工作模式

设备具有 1~8 个串口（PORT1~PORT8），即分别对应 1~8 个串口通道（U1~U8）。设备的每个串口通道之间相互独立，串口参数、串口工作模式等均可独立配置、独立工作。

设备的每个串口通道的工作模式均支持透传、ModBus TCP/RTU 互转（从）、ModBus TCP/RTU 互转（主）、一问一答、数据分发等，用户根据自己的应用场景，选择合适工作模式。

3.2.1 透传

设备的透传模式是指数据透明传输，不做任何协议的转换。例如在快速使用说明小节中的演示，建立数据通道 U1↔N1，设备的第一路串口 PORT1 上的接收到的数据会原封不动的转发到网络通道 N1 上；网络通道 N1 收到的数据也会原封不动的转发到设备的第一路串口 PORT1 上。

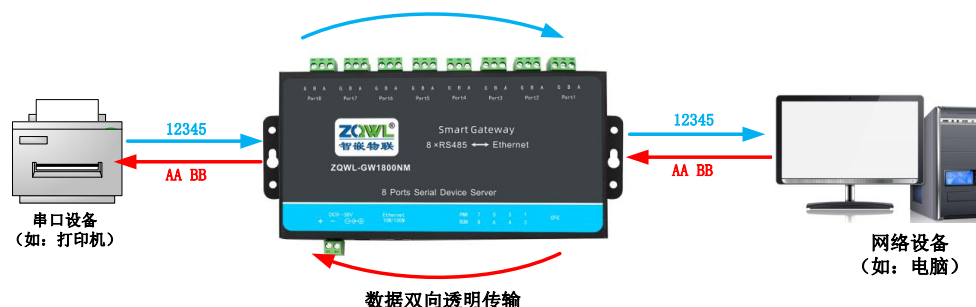


图 3.21 设备透传应用框图

3.2.2 ModBus TCP/RTU 互转（从）

若用户的串口设备是 Modbus RTU 从站，网络端的设备或组态软件是 Modbus TCP 主站，则可以使用 ModBus TCP/RTU 互转（从）模式。

设备内部具有智能的消息排队机制，因此在该模式下，支持多个 Modbus TCP 主站同时轮询 Modbus RTU 从站设备。

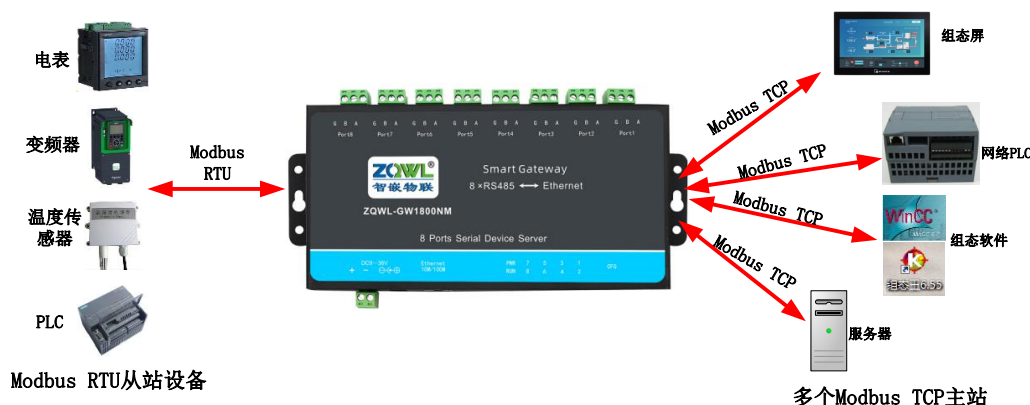


图 3.22 ModBus TCP/RTU 互转（从）模式下，多主机轮询

以设备的第一路串口 PORT1（U1）为例，来说明 ModBus TCP/RTU 互转（从）模式的配置步骤。设备支持自定义数据通道，本测试选择的数据通道为 U1↔N1。

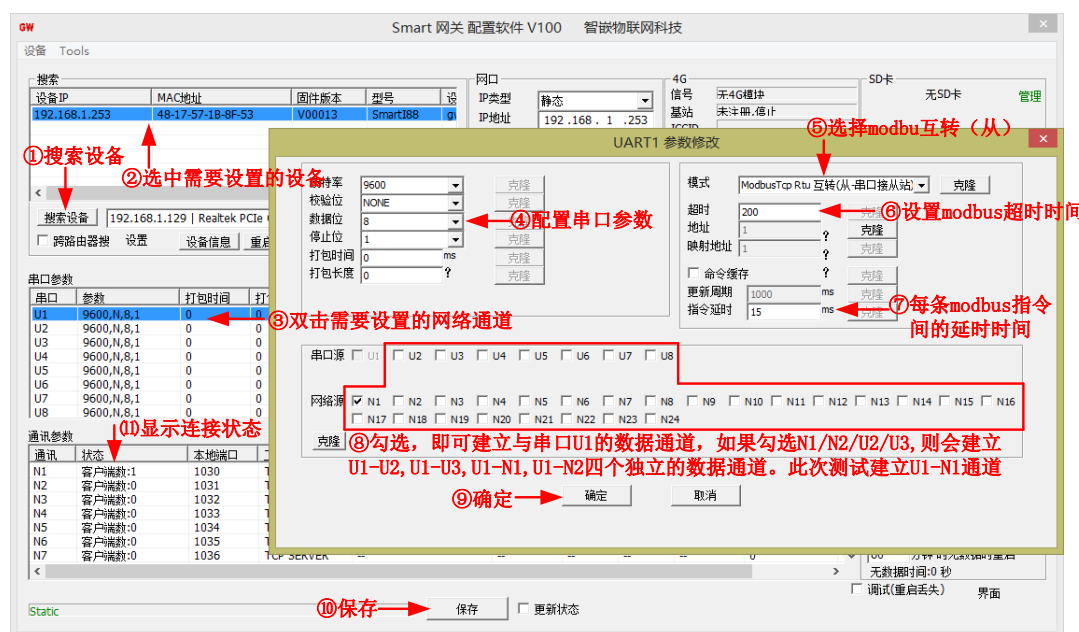


图 3.23 ModBus TCP/RTU 互转（从）模式配置

- ❶ 超时时间：当设备查询指令发出后，从站设备没有应答，网关设备需要等待该超时时间后，才会发送下一条指令。
- ❷ 指令延时时间：每条轮询指令之间的时间间隔。
- ❸ 命令缓存：勾选启用，该功能是将主站发过来的指令记忆下来，然后网关设备主动轮流下发所有记忆的指令，并将应答数据缓存起来。当主站再次查询时，网关设备会将缓存的数据直接应答给主站。

举例：

打开一个 modbus Slave 软件，模拟用户的 Modbus RTU 从站设备；打开 4 个或更多的 Modbus Poll 软件（配置参数完全一样），模拟用户 Modbus TCP 主站。实现多个主站同时轮询一个从站的目的。

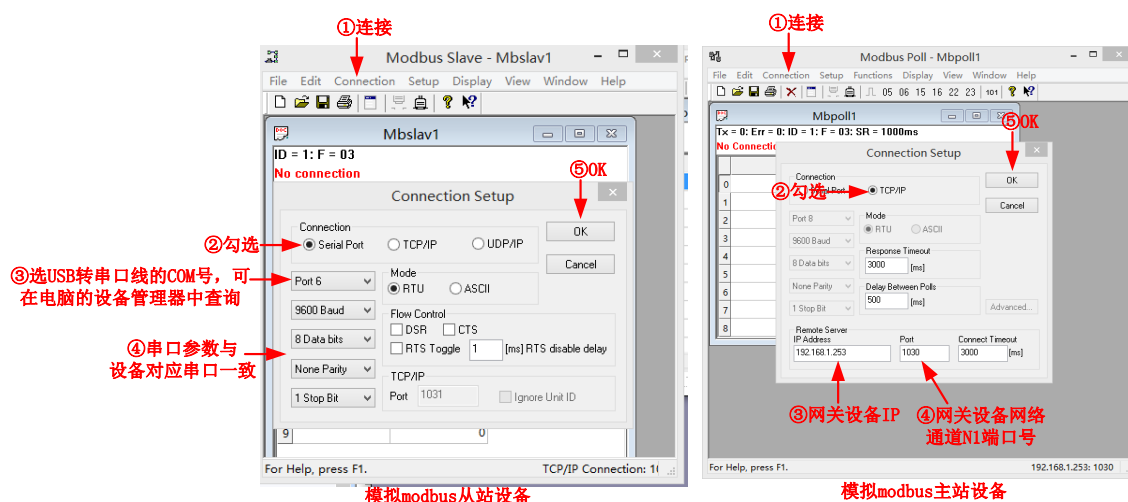


图 3.24 Modbus Poll 和 Slave 配置

配置完成之后，通信成功后的截图如图所示。

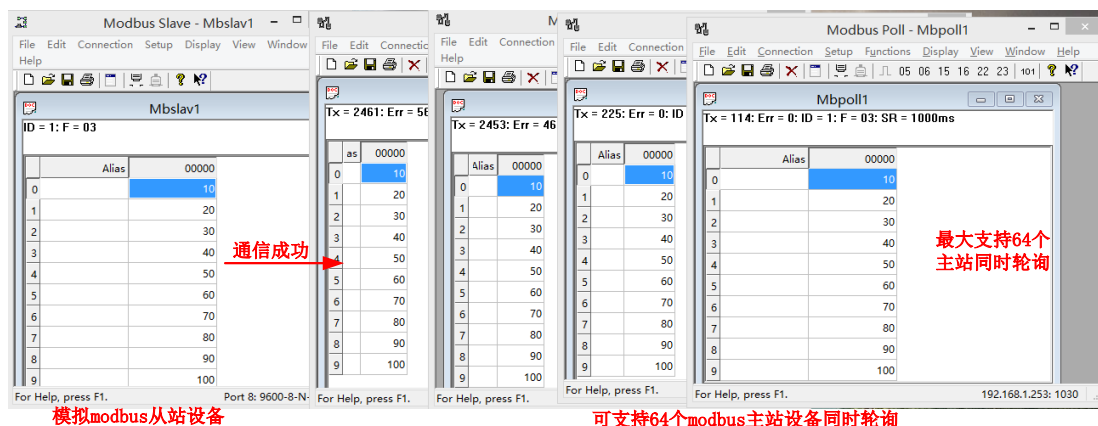


图 3.25 多主站同时查询从站通信成功

- ① 本测试设置的数据通道是 U1↔N1，用户可以尝试配置其他的数据通道测试下，比如同时勾选 U2、U3、N1、N2，此时设置的数据通道即为：U1↔N1；U1↔N2；U1↔U2；U1↔U3，每个数据通道相互独立。

3.2.3 ModBus TCP/RTU 互转（主）

若用户的串口设备是 Modbus RTU 主站，网络端的设备或组态软件是 Modbus TCP 从站，则可以使用 ModBus TCP/RTU 互转（主）模式。

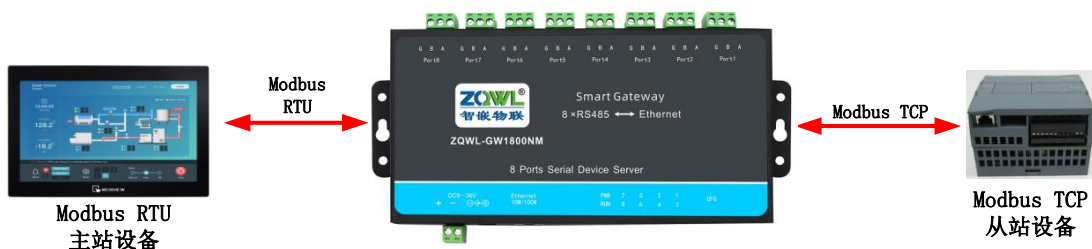


图 3.26 ModBus TCP/RTU 互转（主）模式应用框图

以设备的第一路串口 PORT1（U1）为例，来说明 ModBus TCP/RTU 互转（主）模式的配置步骤。设备支持自定义数据通道，本测试选择的数据通道为 U1↔N1。

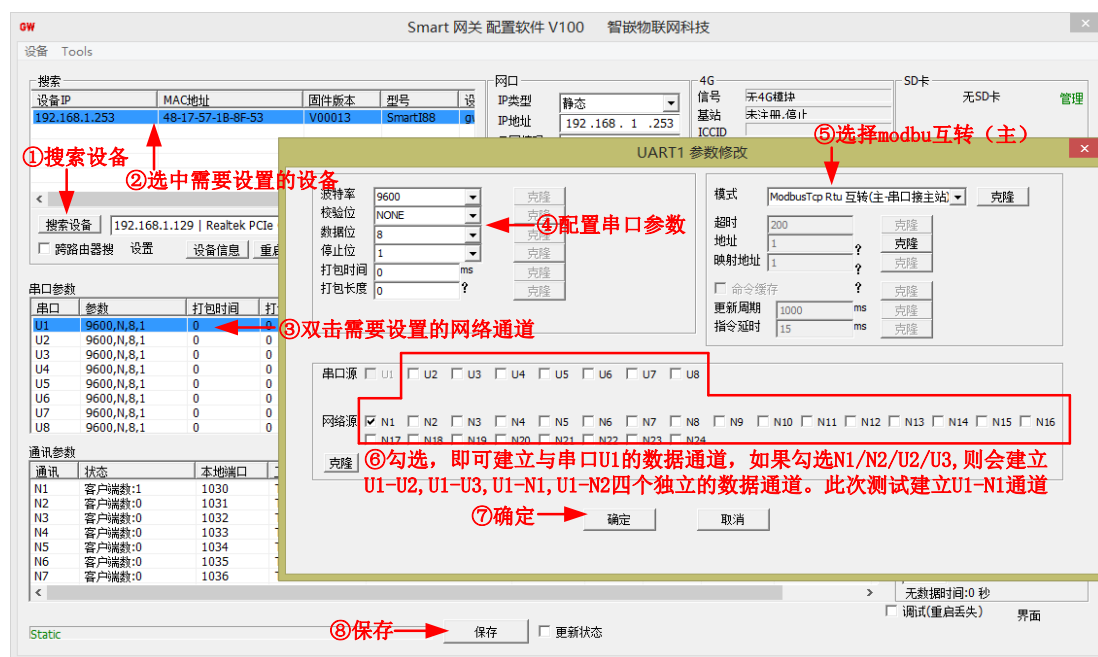


图 3.27 通道 U1 配置为 ModBus TCP/RTU 互转（主）模式

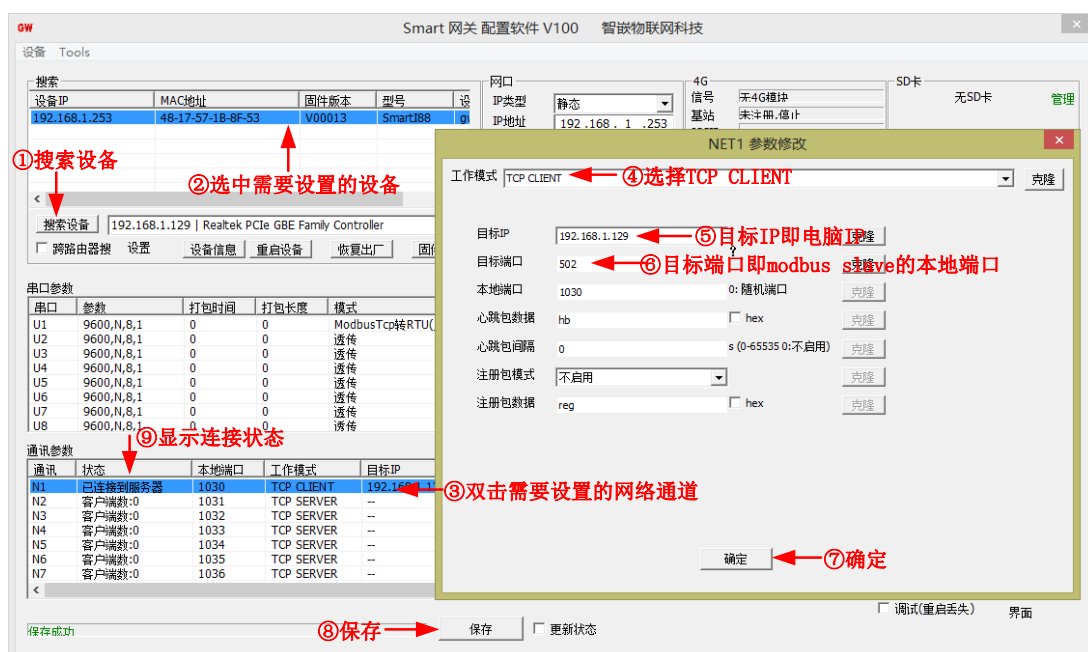


图 3.28 通道 N1 连接服务器配置

打开一个 modbus Poll 软件，模拟用户的 Modbus RTU 主站设备；打开一个 Modbus Slave 软件，模拟用户 Modbus TCP 从站。实现 Modbus RTU 主站查询 Modbus TCP 从站数据的目的。

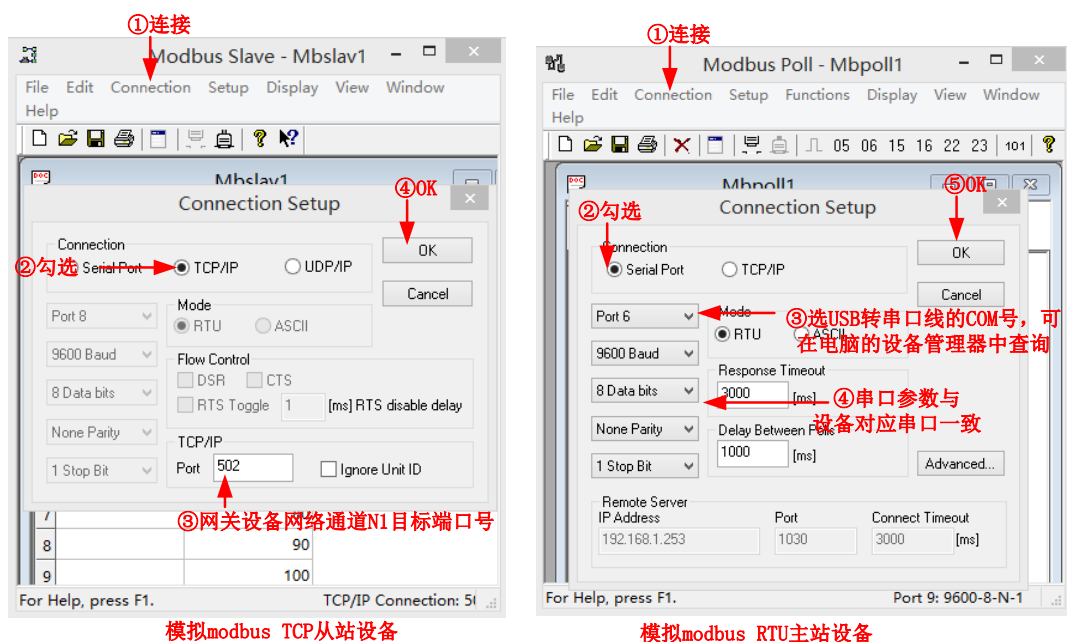


图 3.29 Modbus Poll 和 Slave 配置

配置完成之后，通信成功后的截图如图所示。

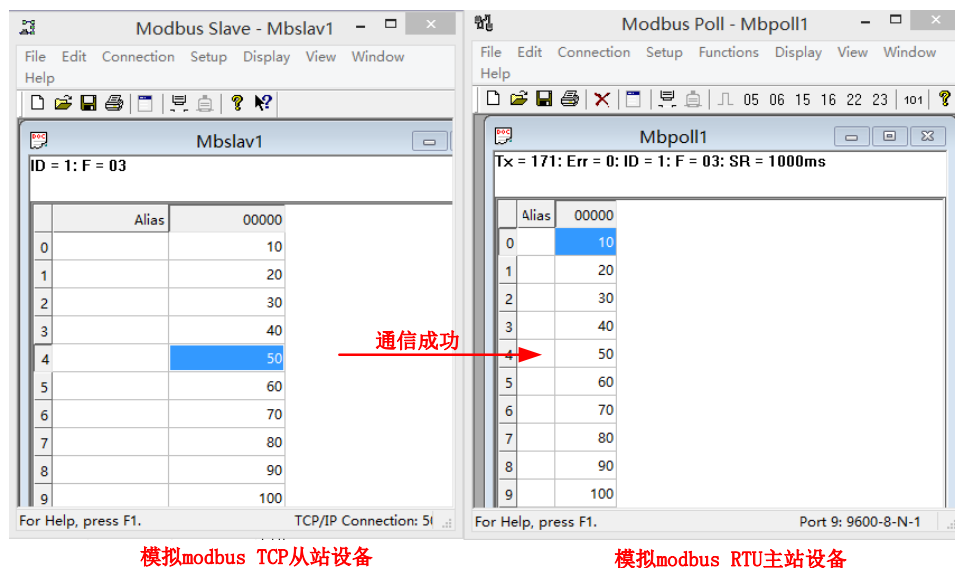


图 3.30 通信成功

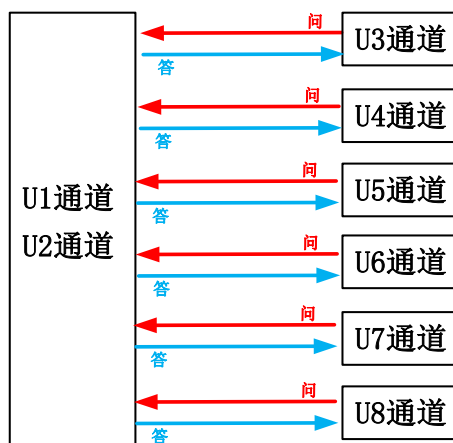
3.2.4 一问一答

在一问一答模式下，设备可以实现多个串口通道或网络通道对 1 个或多个串口数据的查询。设备会对所有串口通道或网络通道的数据进行排队处理，不会造成数据的冲突。配置方法如图 3.31 所示。



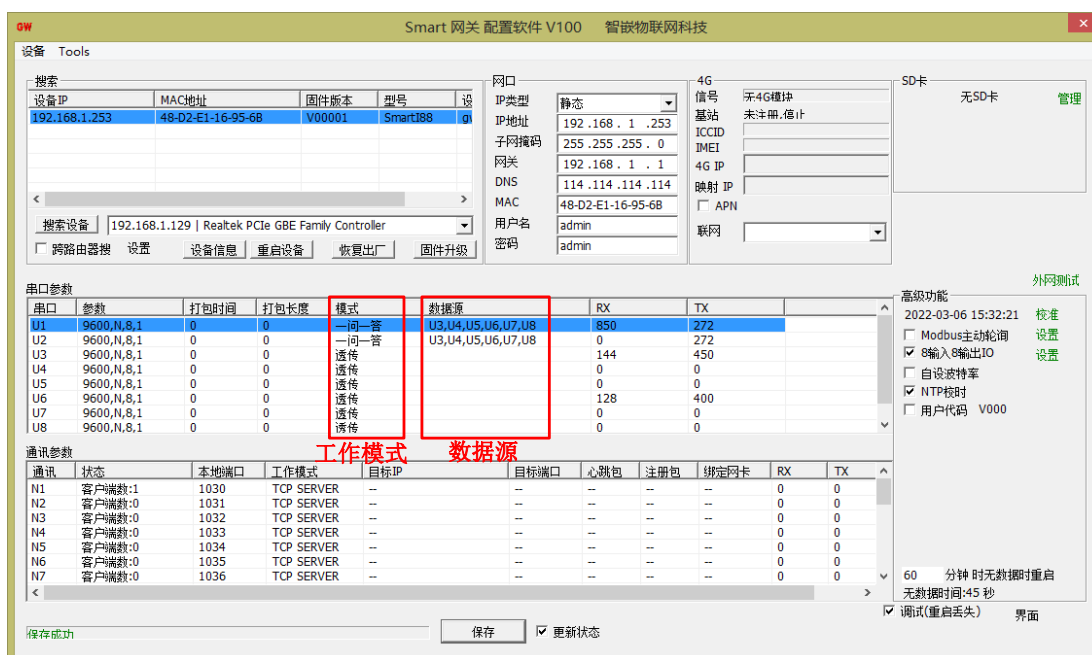
举例：

实现效果：设备的六个串口通道 U3~U8 同时查询两个串口通道 U1/U2 的数据。



U3~U8可同时采集U1、U2数据

为实现上图中的效果，需将 U1~U8 通道的配置如图所示：



① 用户测试可使用 Modbus Poll 模拟主站，发送数据请求；用 Modbus Slave 模拟从站，对请求做出数据应答。

3.2.5 数据分发

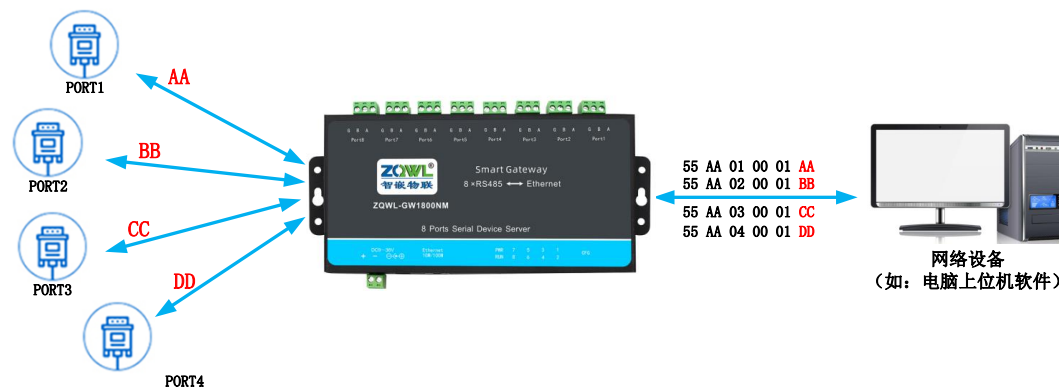


图 3.32 分发模式原理框图

在分发模式下，设备的 8 个串口 PORT1~PORT8，均会被分配一个不同的串口地址，这样用户可以按照设备的分发协议，向指定的串口发送数据，同时用户也可以根据串口地址来区分数据来自哪一路串口。设备的分发协议如表所示。

图 3.33 分发协议

	帧头	串口地址	数据长度	数据
字节数	2 byte	1 byte	2 byte	数据长度
说明	55 AA	01~08	数据长度	用户数据
举例：向 PORT2 发	55 AA	02	00 01	FF

送数据 0xFF			
----------	--	--	--

以网络通道 N1 向设备的 8 个串口 PORT1~PORT8 为例，来说明设备的分发模式的配置步骤以及分发协议的使用方法。

将 8 个串口通道的工作模式均设置为分发，网络通道均选择 N1，配置方法如图所示。

这样就建立了 N1↔U1、N1↔U2、N1↔U3……N1↔U8，共 8 个数据通道。

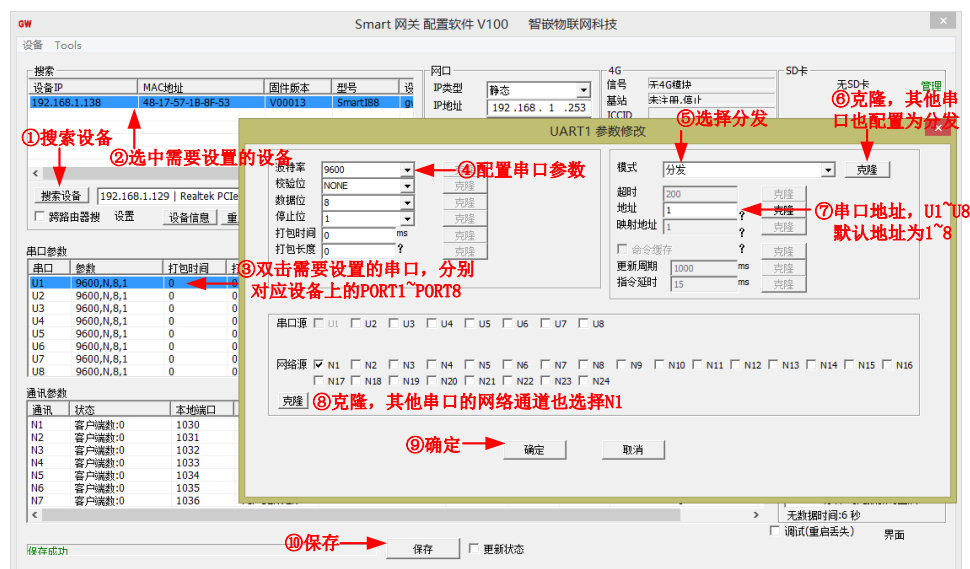


图 3.34 分发配置

1. N1↔U1 数据通道分发测试

打开 1 个串口调试助手，模拟用户的串口设备，通过 USB 转 RS485 串口线连接设备的第一路串口 PORT1，从而与设备的串口通道 U1 建立连接；打开一个网络调试助手，模拟用户的网络设备或上位机软件，与设备的网络通道 N1 建立连接。

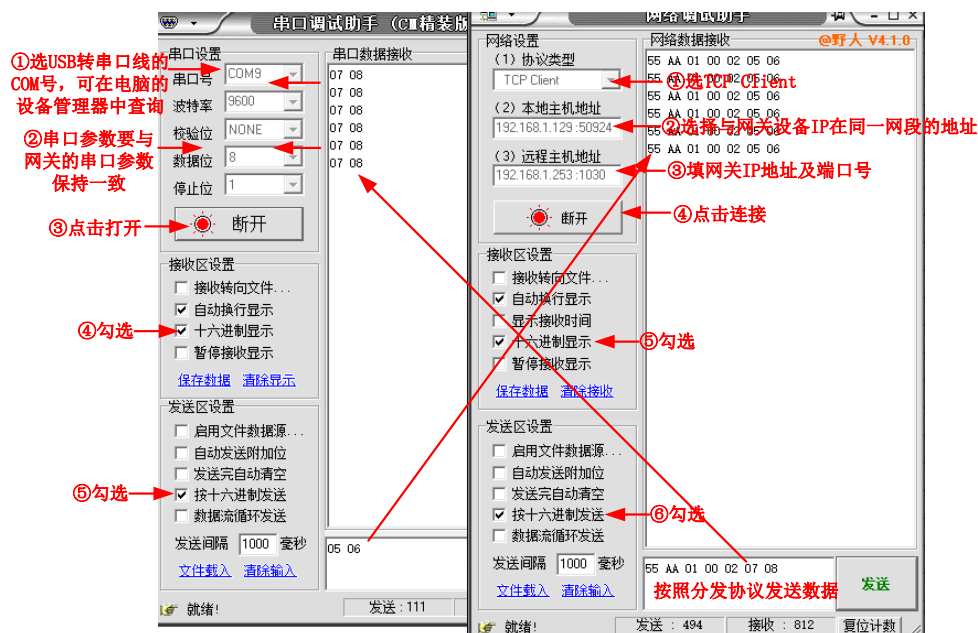


图 3.35 N1↔U1 通道按分发协议收发数据

2. N1↔U2 数据通道分发测试

将 USB 转 RS485 线接到设备的 PORT2 上，网络调试助手发送的数据中的串口地址改成 02，则即可实现 N1↔U2 数据的分发转换。其他通道数据分发测试方法一样。



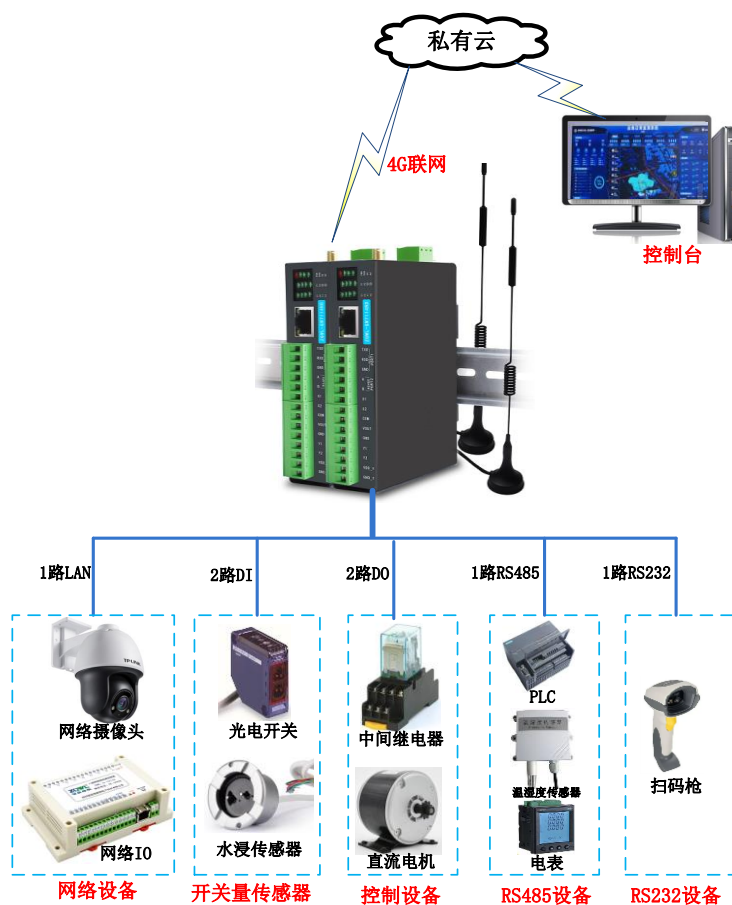
图 3.36 N1↔U2 通道按分发协议收发数据

4. 4G 联网

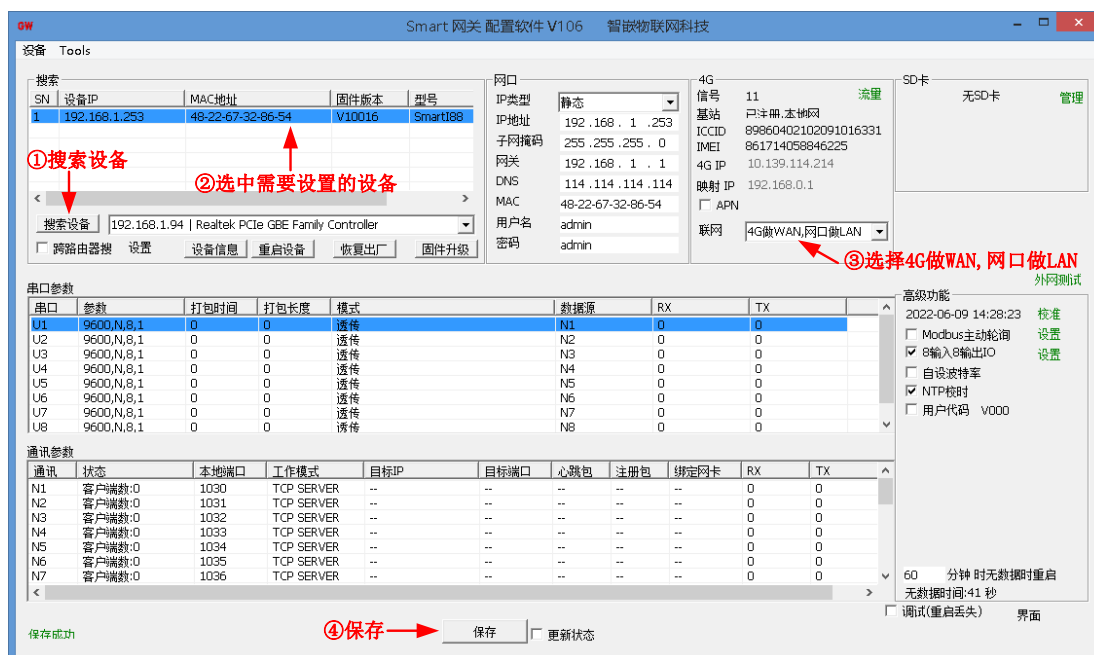
设备支持 1 路 LAN，插上 SIM 卡即可实现网络设备与 4G 网络之间的双向数据传输。如网络摄像头联网、办公电脑联网等。

设备支持 RS485/RS232 串口与 4G 网络之间的双向数据透明传输，支持多种网络协议，可轻松实现串口数据联网。

支持 2~8 路 DI 开关量输入、2~8 路 DO 开关量输出，可远程采集开关量传感器的状态（如光电开关），也可远程控制用户设备的运转。



设备 4G 联网的配置方法如所示。



举例一：办公电脑通过设备联网

实验效果：将一台带网口的电脑，通过该设备之后，能够连接互联网。

硬件连接：用网线连接电脑和设备的网口，设备插上 SIM 卡，然后为设备上电。

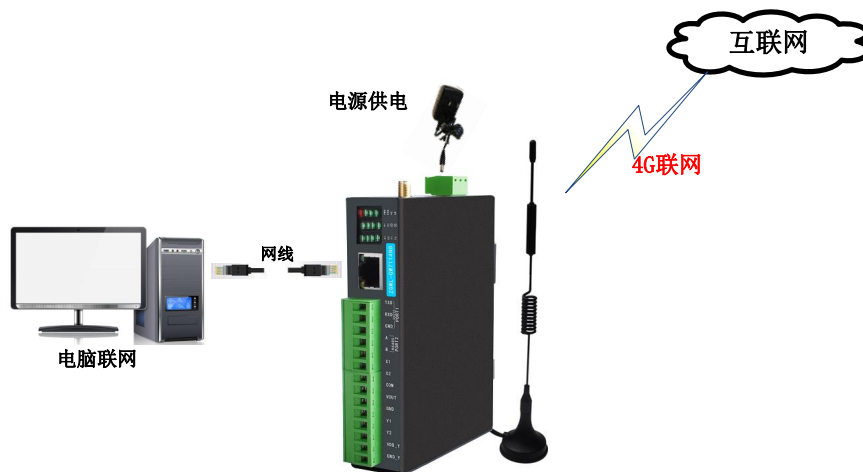


图 4.1 办公电脑联网示意图

实验步骤：

1. 打开配置软件，并对设备进行联网配置，配置完之后，保存参数。

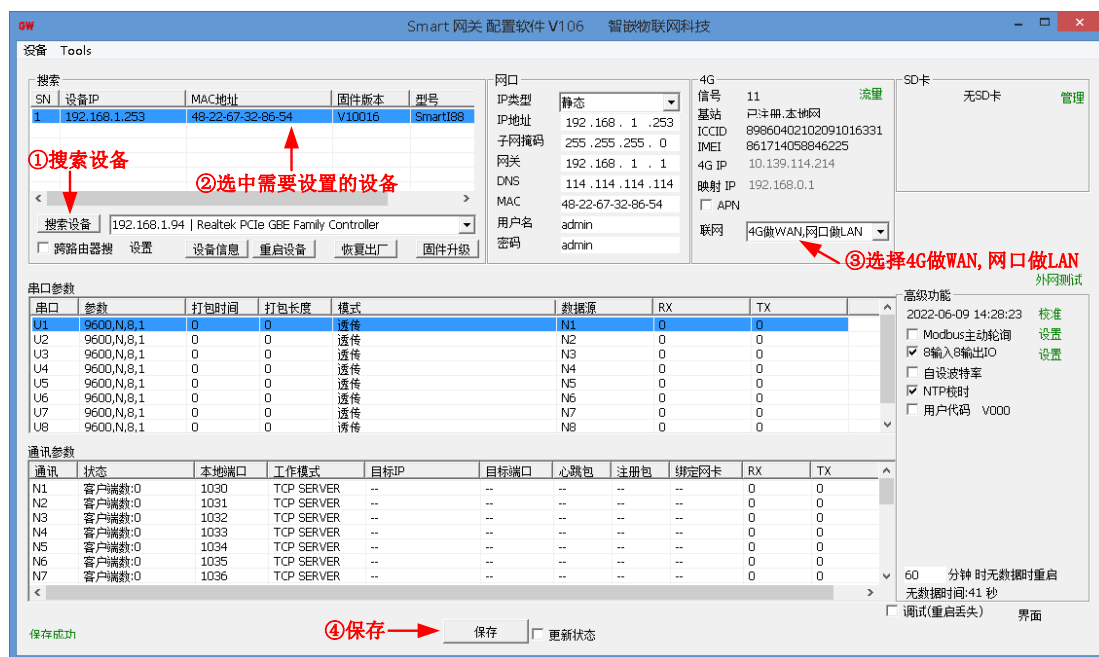


图 4.2 4G 做 WAN，网口做 LAN

2. 电脑 Ipv4 属性配置。将电脑 IP 改成与网关设备在同一个网段，网关 IP 改为设备的 IP 地址。

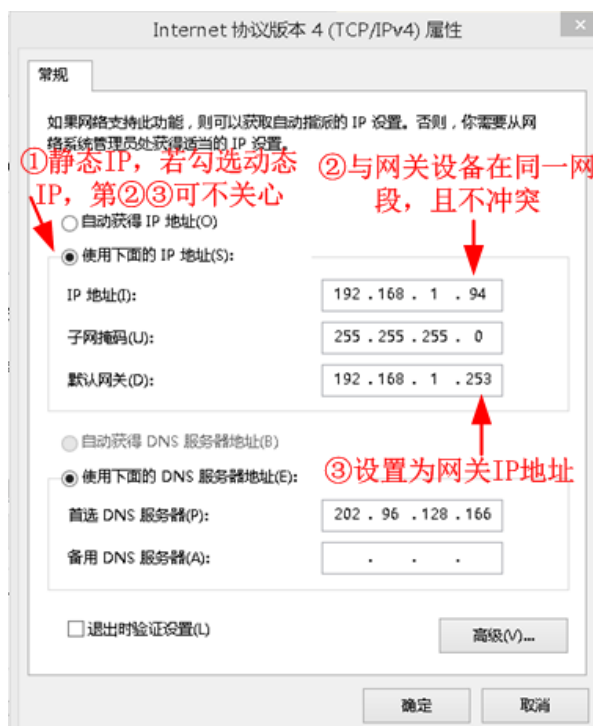


图 4.3 电脑 IP 配置

3. 外网测试

用配置软件 ping 百度测试。



图 4.4 Ping 百度测试

接下来就可以用电脑来看视频、浏览网页了。

举例二：远程传输网络摄像头数据

实验效果：网络摄像头接入网关设备后，用户可远程查看网络摄像头的视频。

硬件连接：用网线连接网络摄像头和网关设备；设备插上 SIM 卡，然后为设备上电。

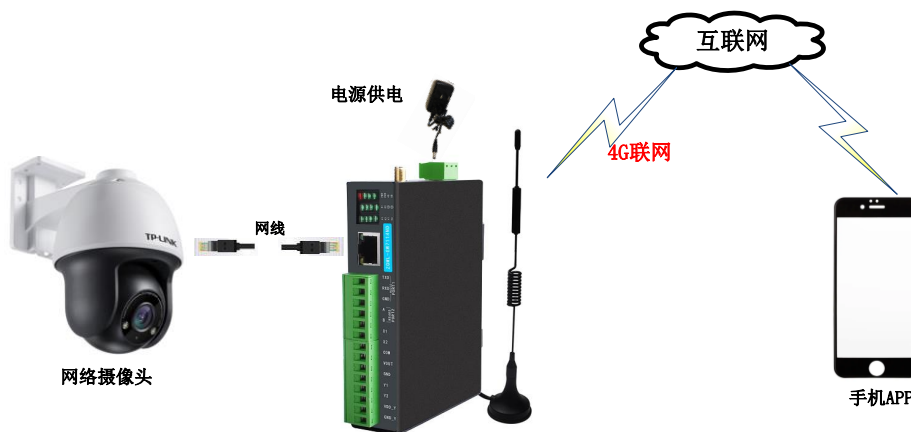


图 4.5 远程传输网络摄像头数据示意图

实验步骤：

1. 将网关设备的网口与电脑的网口用网线连接，对网关设备进行参数配置。
打开配置软件，并对设备进行联网配置，配置完之后，保存参数。

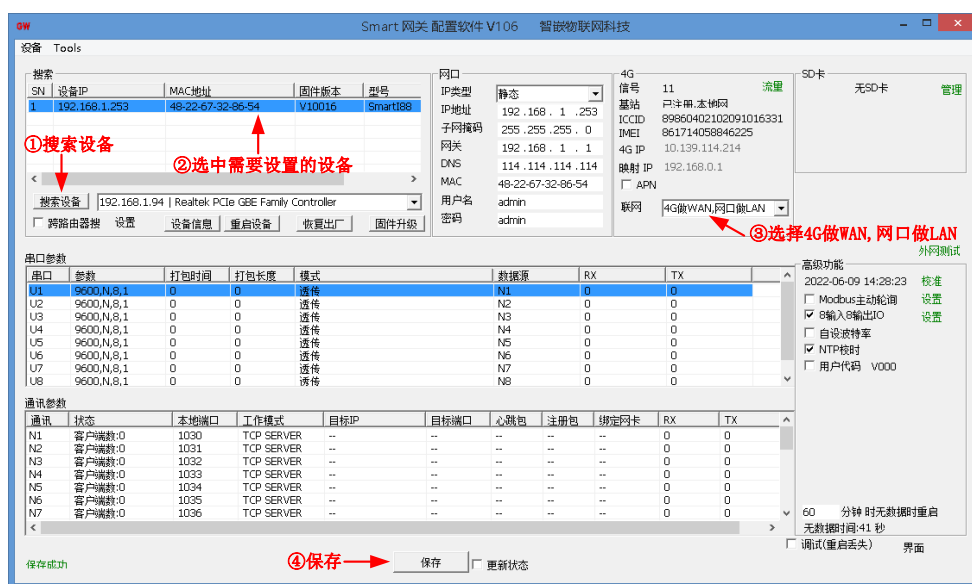


图 4.6 4G 做 WAN，网口做 LAN

2. 外网测试

用配置软件 ping 百度测试。

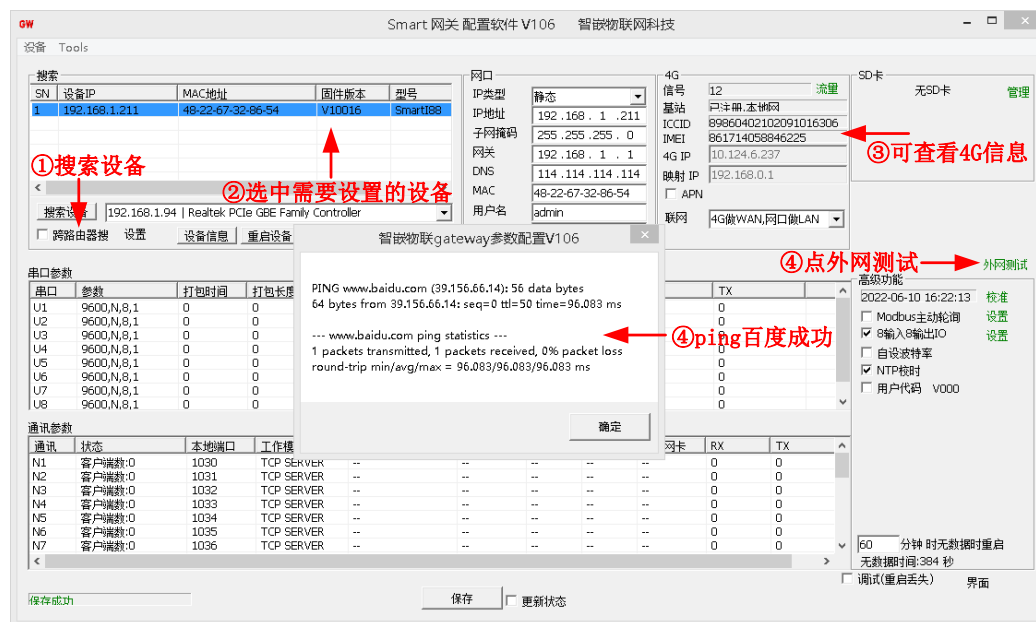


图 4.7 Ping 百度测试

- 把网关设备的网口与网络摄像头的网口用网线连接，然后按照网络摄像头厂家的说明书进行操作即可。

举例三：主动采集网口设备（如 PLC）数据，并将数据上云

实验效果：网关设备主动轮询 Modbus TCP 从站（Modbus slave 软件模拟），并将轮询到的数据，上传到智嵌的 MQTT 测试服务器上，并在 MQTXX 客户端软件上显示出来；并可

通过 MQTTX 客户端软件下发指令，写到 Modbus TCP 从站里。

硬件连接：用网线连接电脑和设备的网口，设备插上 SIM 卡，然后为设备上电。



图 4.8 硬件连接

实验步骤：

1. 打开 Modbus Slave 软件，用来模拟用户的网口设备。具体设置如图所示。

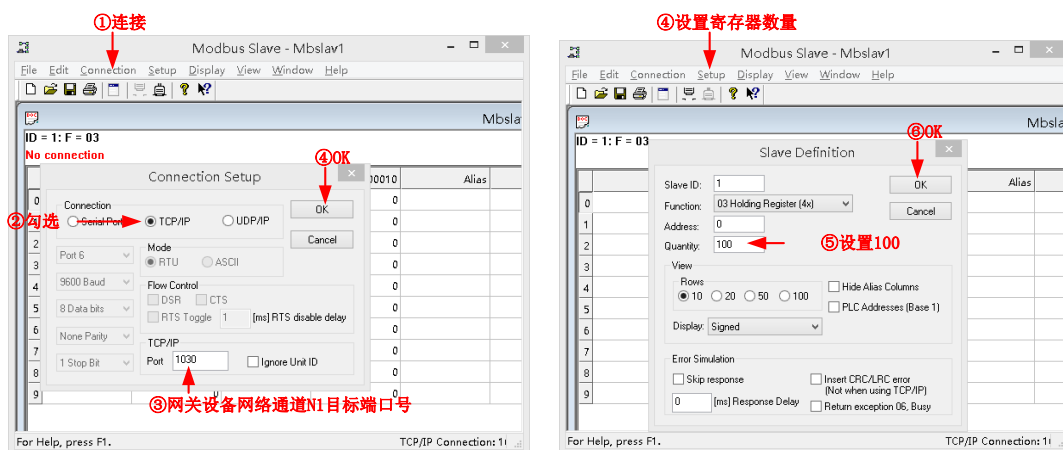


图 4.9 Modbus Slave 测试软件配置

2. 配置网关设备的联网方式以及主动轮询指令。根据 Modbus Slave 来设置主动轮询的 Modbus 指令。主动轮询 N1 网络通道，并将轮询的数据上传到 N2 网络通道上。具体配置如所示。

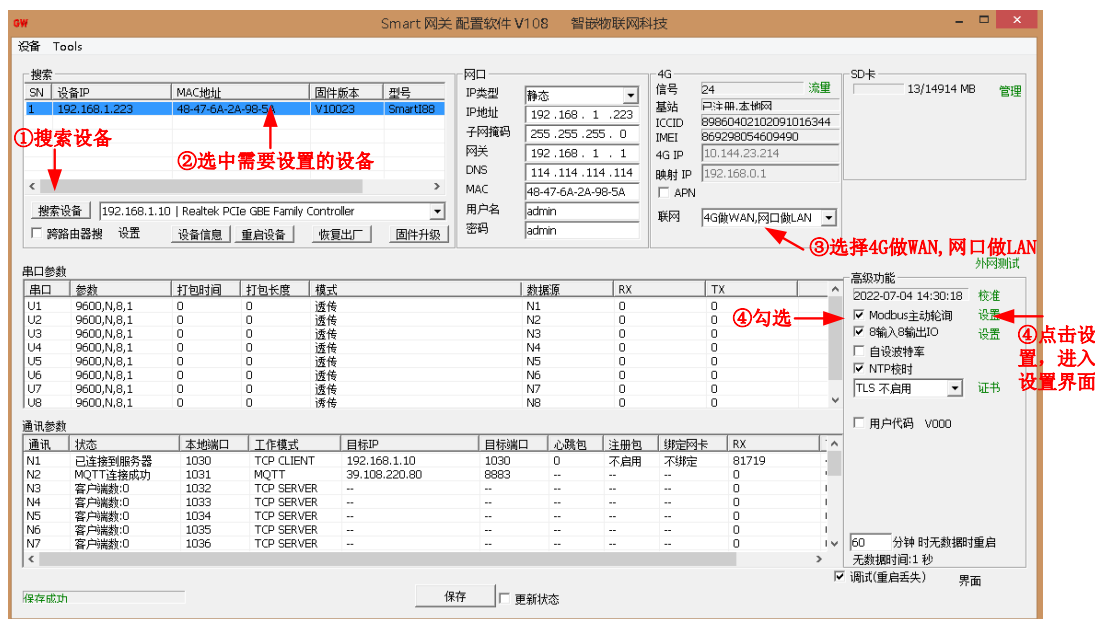


图 4.10 网关设备主动轮询配置 1

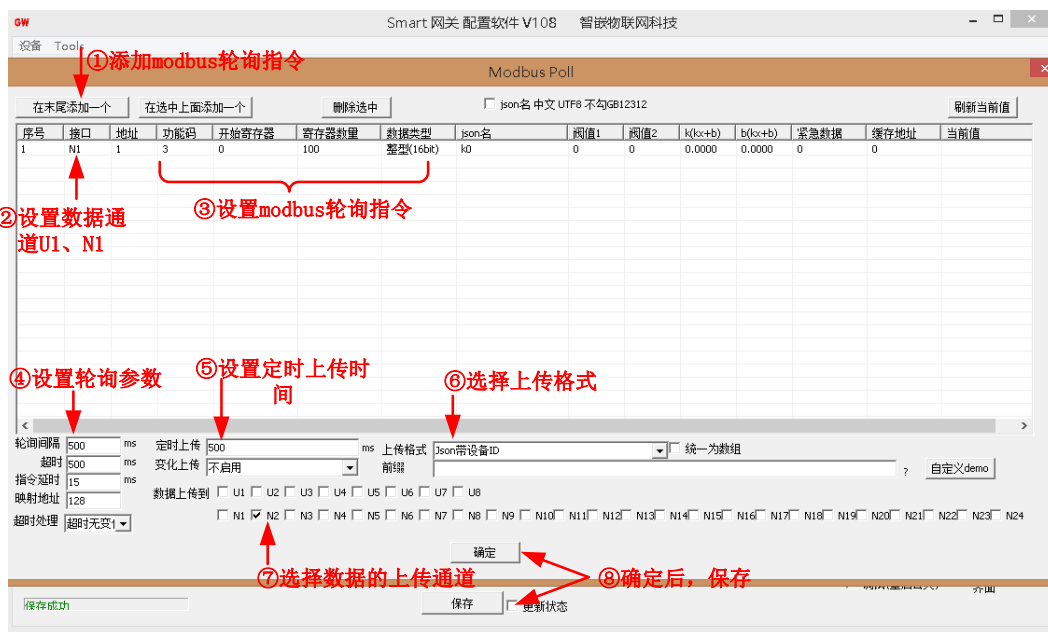


图 4.11 网关设备主动轮询配置 2

3. 设置网络通道 N1 的参数。网络通道 N1 连接 Modbus Slave。设置完成之后, 可在配置软件上查看连接状态。

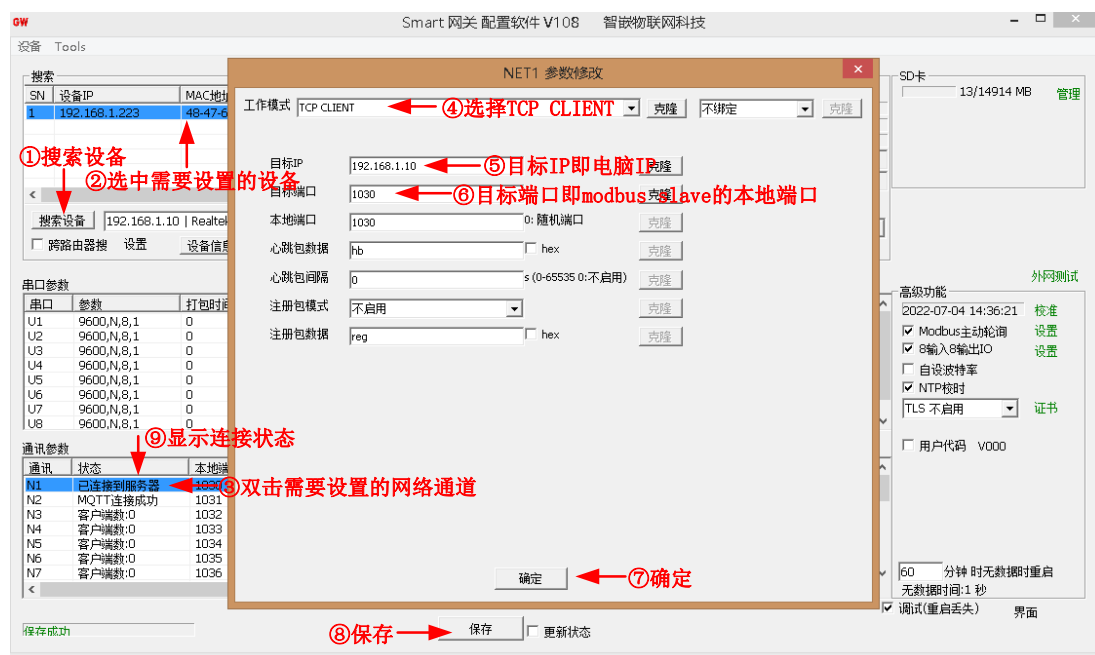


图 4.12 网络通道 N1 参数配置

4. 设置网络通道 N2 的参数。网关设备将轮询的 Modbus 数据，定时上传到网络通道 N2 上。

为方便用户测试设备的 MQTT 功能，智嵌物联创建了一个 MQTT 测试服务器，服务器 IP：39.108.220.80，端口号：8883。

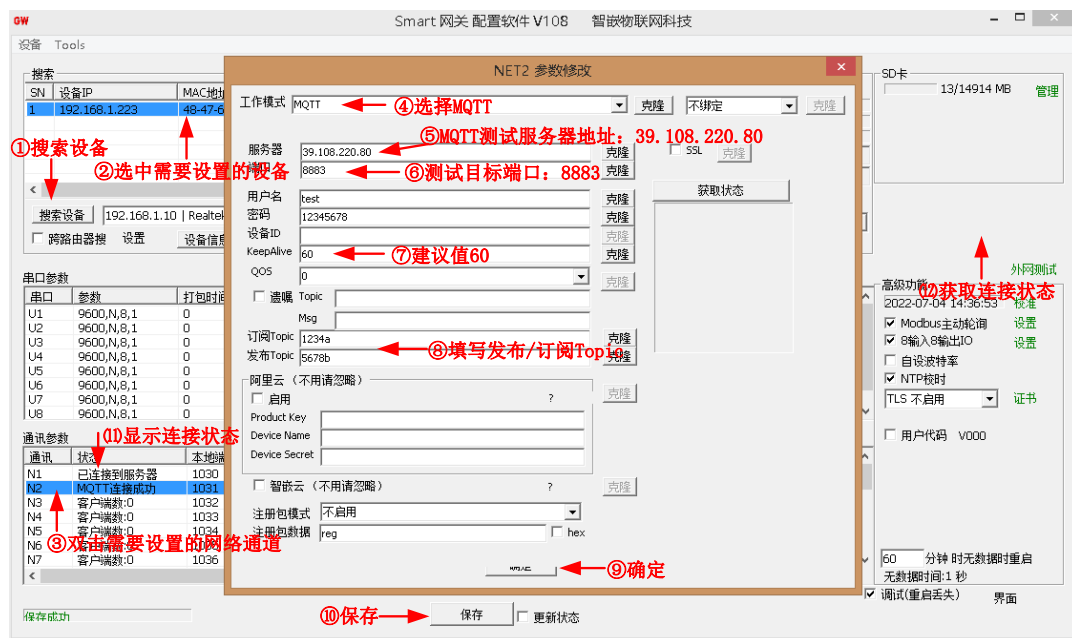


图 4.13 网络通道 N2 参数配置

5. 下载并打开 MQTTX 客户端软件

打开一个 MQTTX 客户端软件，连接到 MQTT 测试服务器，并创建发布/订阅 Topic。创

成功后即可接受到网关设备上传的轮询数据。

当上传格式选择 JSON 格式时，MQTTX 下发数据时要按照特定的下发格式来下发，具体详见 7.2.2 小节。本实验中的下发指令为：

```
{"mb":"000B0000000601060000000A","sn":1,"ack":0,"crc":0,"net":0}
```

当上传格式选择 Modbus TCP 时，MQTTX 下发数据时，直接按照标准的 Modbus TCP 协议下发即可如本实验中，下发指令可以为：

```
14D40000001B80100000000A14000100020003000000004000000000000000000000
```

当上传格式选择 Modbus RTU 时，MQTTX 下发数据时，直接按照标准的 Modbus RTU 协议下发即可如本实验中，下发指令可以为：

```
80100000000A14000100020003000400000000000000000000000000A31A
```

关于下发指令更详细的介绍，详见 Modbus 主动轮询章节。

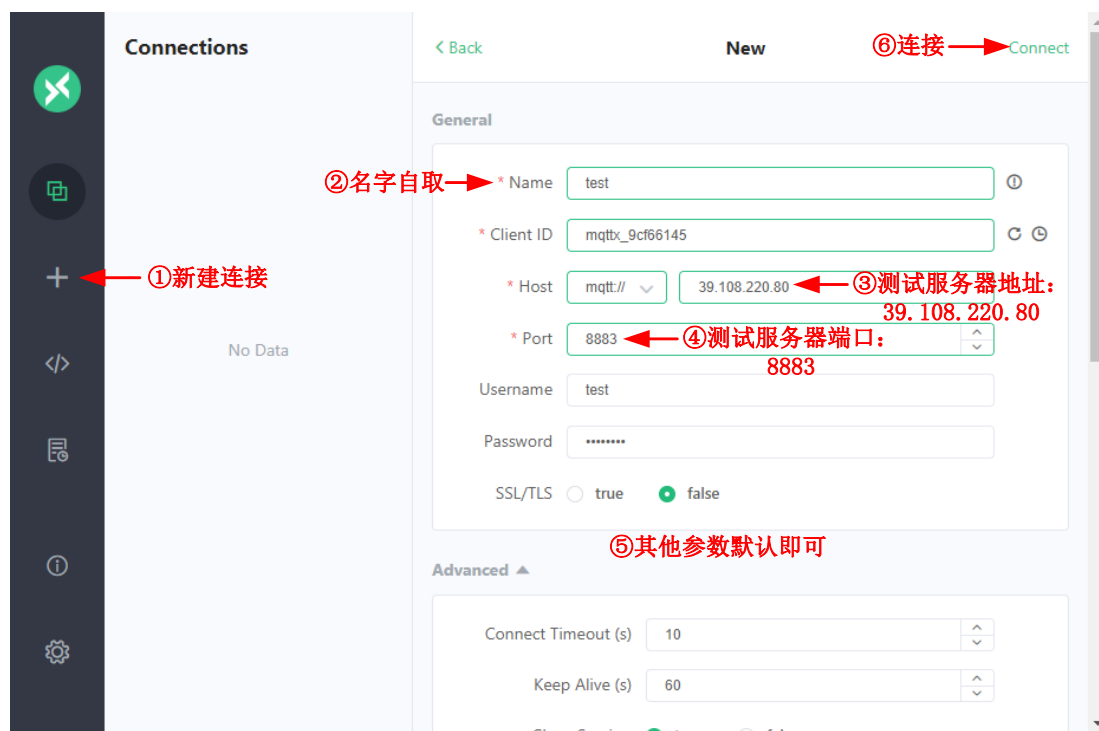


图 4.14 MQTTX 测试软件配置

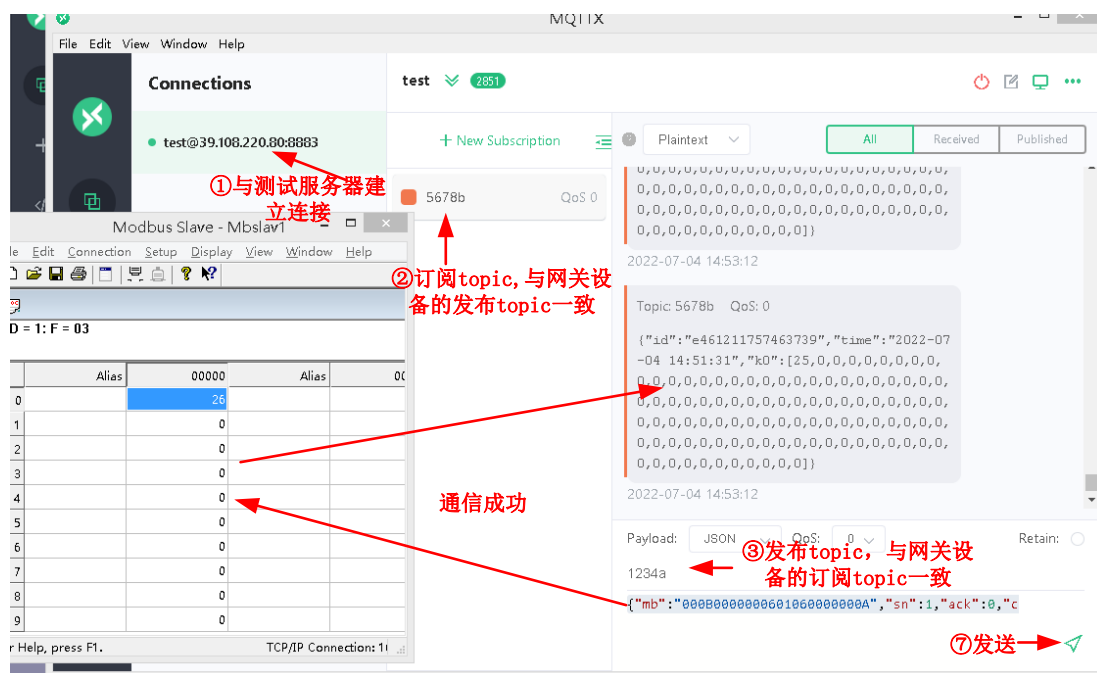


图 4.15 MQTTX 与 Modbus Slave 通信成功

举例四：通过 4G 读取串口设备的数据

实验效果：通过设备来实现 MQTTX 客户端软件与串口调试助手之间的双向数据传输

硬件连接：用 USB 转 RS485 连接电脑的 USB 口和设备的 RS485 接口；用网线连接电脑和设备的网口，设备插上 SIM 卡，然后为设备上电。



图 4.16 远程采集串口设备数据示意图

实验步骤：

1. 按照举例一中的步骤，先将电脑能连上互联网。
2. 将设备的网络通道 N1 设置为 MQTT

以设备的第一路网络通道 N1 为例，来说明 MQTT 模式的配置步骤，其他网络通道（N2~N24）的配置方法一样。

为方便用户测试设备的 MQTT 功能，智嵌物联创建了一个 MQTT 测试服务器，服务器 IP：39.108.220.80，端口号：8883。

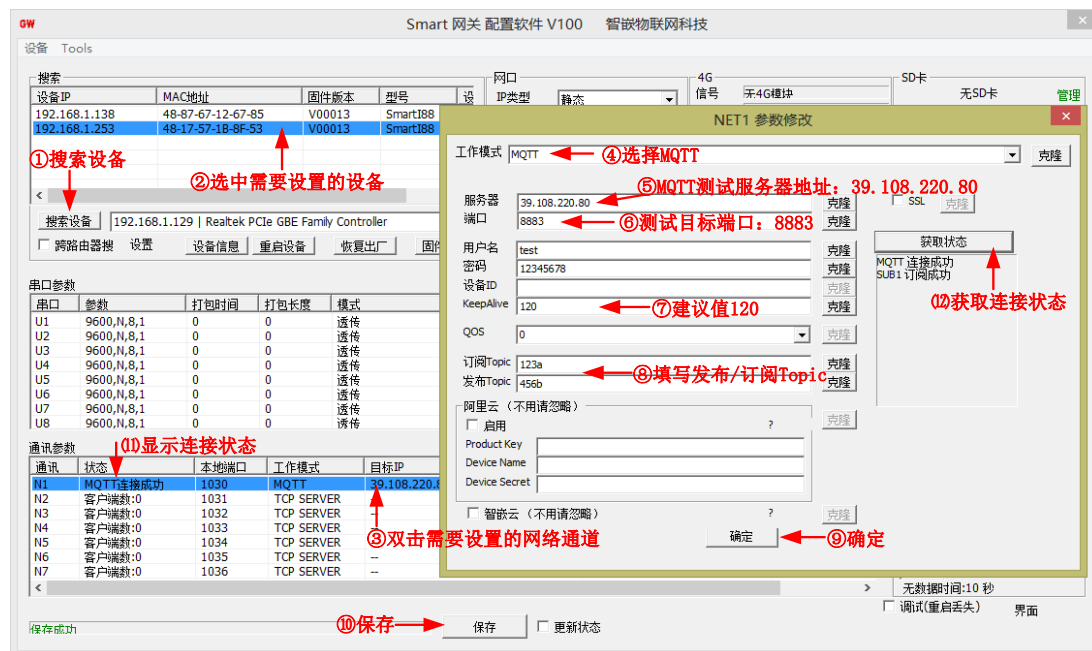


图 4.17 N1 通道配置为 MQTT 模式

3. 下载并打开 MQTXX 客户端软件；打开一个串口调试助手，模拟用户的串口设备。

打开一个 MQTXX 客户端软件，连接到 MQTT 测试服务器，并创建发布/订阅 Topic。打开一个串口调试助手，模拟接到网关设备 PORT1 上的串口设备。

MQTXX 软件下载地址：[点击下载](#)

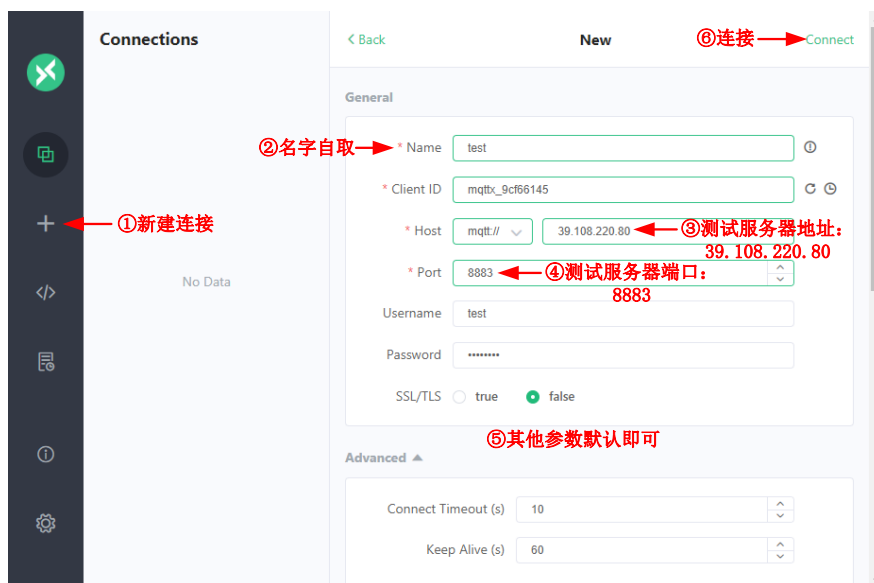


图 4.18 MQTTX 连接测试服务器

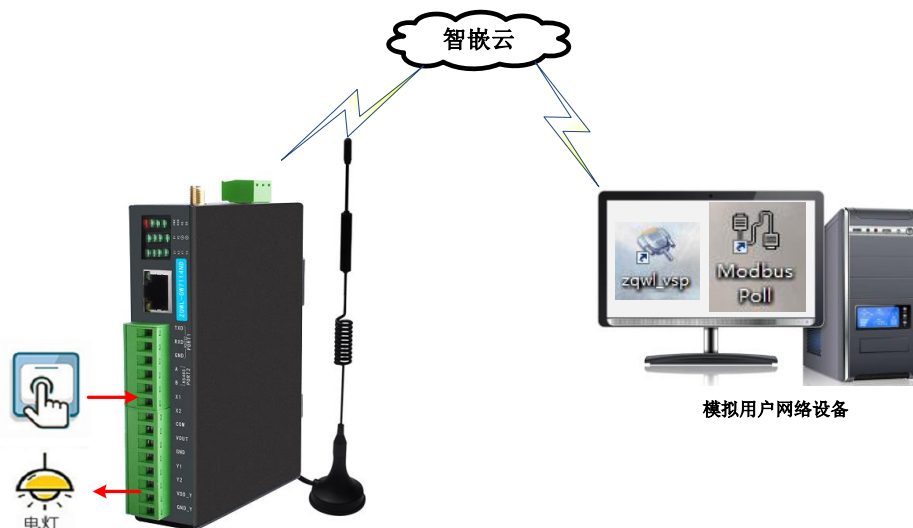


图 4.19 Topic 收发数据成功

举例五：通过 4G 采集并控制设备的 DI/DO

实验效果：Modbus poll 通过虚拟串口来远程采集设备的 DI 开关量状态，并远程控制设备的 DO 输出。DI 开关量接口可以接一个自复位开关、DO 输出接口可以接一个 12V 的指示灯来测试，具体接线方法详见：[点击下载](#)

硬件连接：用网线连接电脑和设备的网口（用来配置设备参数，配置完成之后可以不用接网线），设备插上 SIM 卡，然后为设备上电。



实验步骤：

1. 登录智嵌云平台账号

浏览器进入智嵌云管理平台，网址：www.zqwl.com，选择智嵌云控的新版本界面，如图 4.20 所示。进入登录界面，注册并登录智嵌云平台账号，若已注册，直接登录即可，如图 4.21 所示。



图 4.20 智嵌云管理平台



图 4.21 智嵌云登录界面

2. 添加设备

此步骤的目的是在用户的账号下添加该设备，此步完成后，系统会自动生成 SN 号，该 SN 号会在绑定设备时用到。

智嵌云平台是通过项目的方式来管理设备的，因此在用户添加设备之前要先创建一个项目及分组，如图 4.22 所示。



图 4.22 智嵌云平台创建项目步骤

在该项目分组下添加串口服务器设备，具体步骤如图 4.23 所示。



图 4.23 智嵌云平台添加设备步骤

设备添加完成后，在设备列表菜单下找到刚添加的设备，复制设备 SN 及通讯密码，后面备用。

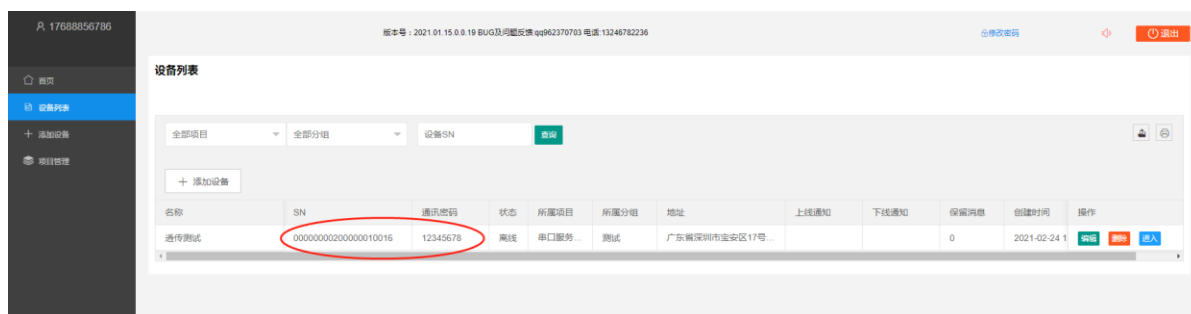


图 4.24 添加设备完成

① SN 号和设备密码在对设备配置时要用到。

3. 配置设备参数

此步骤的目的是将网关设备与智嵌云平台建立连接。

在“智嵌物联 gateway 参数配置软件”中双击 N1 网络通道,进入网络参数配置,如图 4.25 所示,选择 MQTT 配置框,如图 4.26 所示的 MQTT 配置框。

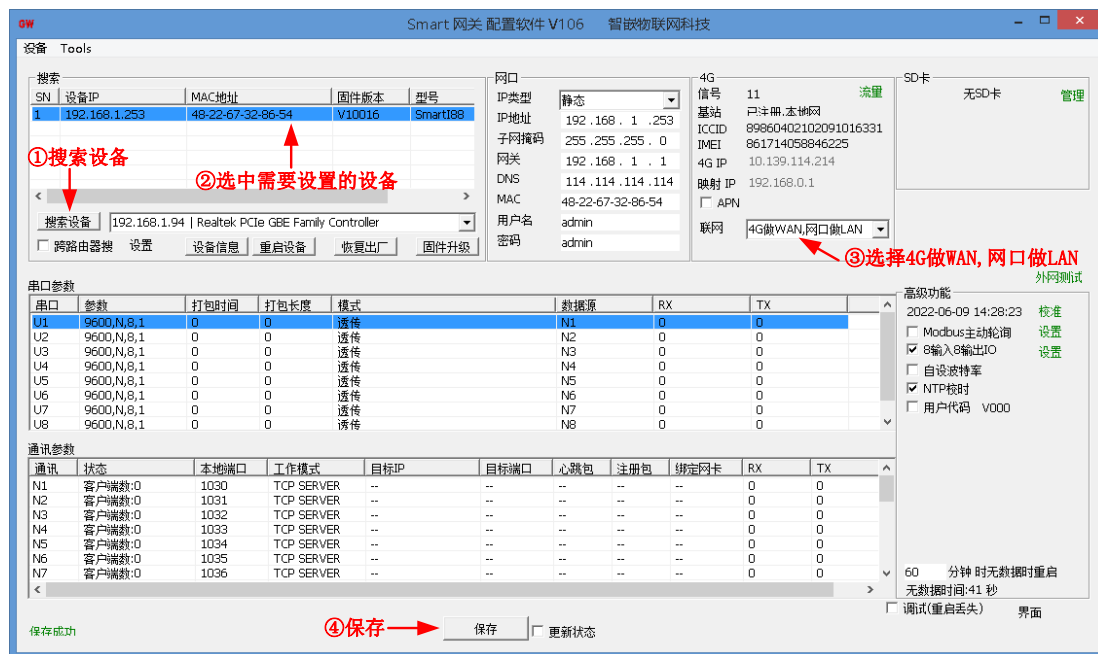


图 4.25 智嵌云参数配置 1

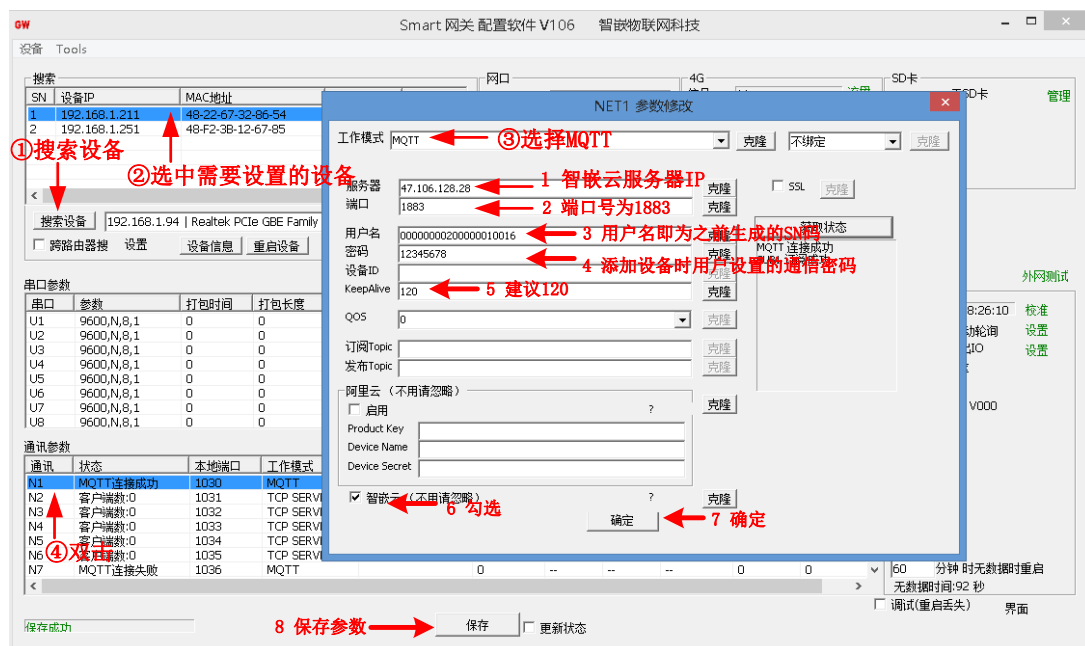


图 4.26 智嵌云参数配置 2

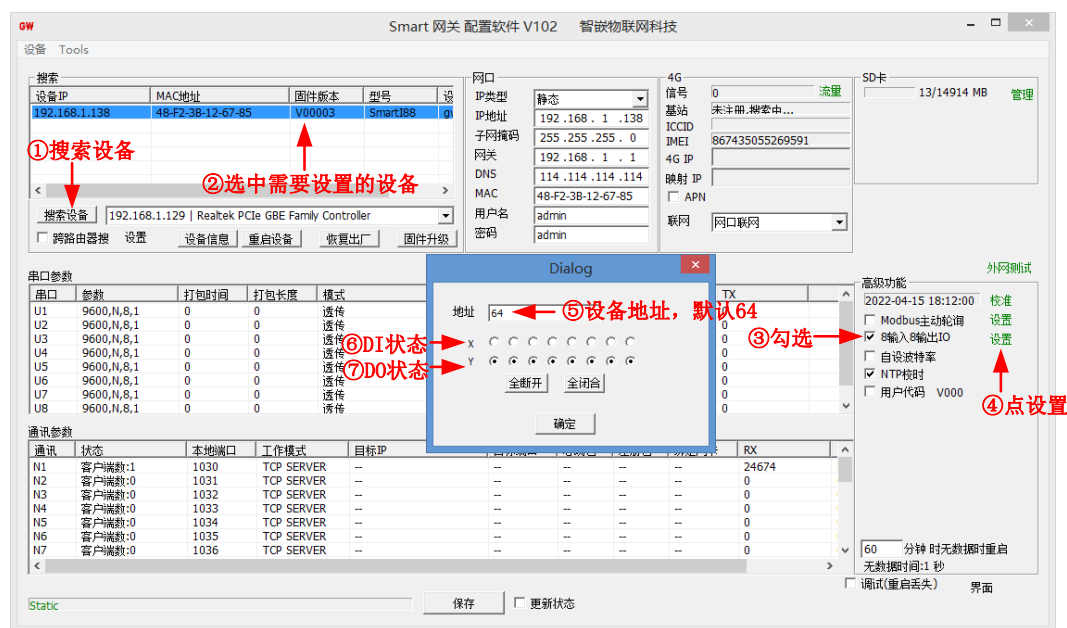


图 4.27 配置网关设备的地址

保存参数后，再获取参数，进入 MQTT 配置界面，点击【获取 MQTT 状态】，可以看到 MQTT 连接成功的信息。

4. 设备在线

以上步骤完成后，在智嵌云平台上可以看到这个设备已经在线。如图 4.28 所示。



图 4.28 设备在线

5. 安装虚拟串口软件

在智嵌物联官网下载虚拟串口软件“ZQWL-VCOM 虚拟串口软件”，按照提示安装并打开，界面如图 4.29 所示。



图 4.29 ZQWL-VCOM 界面

6. 创建虚拟串口并接入智嵌云

创建一个虚拟串口 COM4，并将 COM4 连接到智嵌云上，并将 COM4 与串口服务器设备进行绑定，具体步骤如图 4.30 所示。

注意：要将设备的 SN 号的第 10 位改为“1”。



图 4.30 虚拟串口接入智嵌云并绑定设备步骤



图 4.31 成功连接到智嵌云

7. Modbus Poll 软件读取 DI、控制 DO

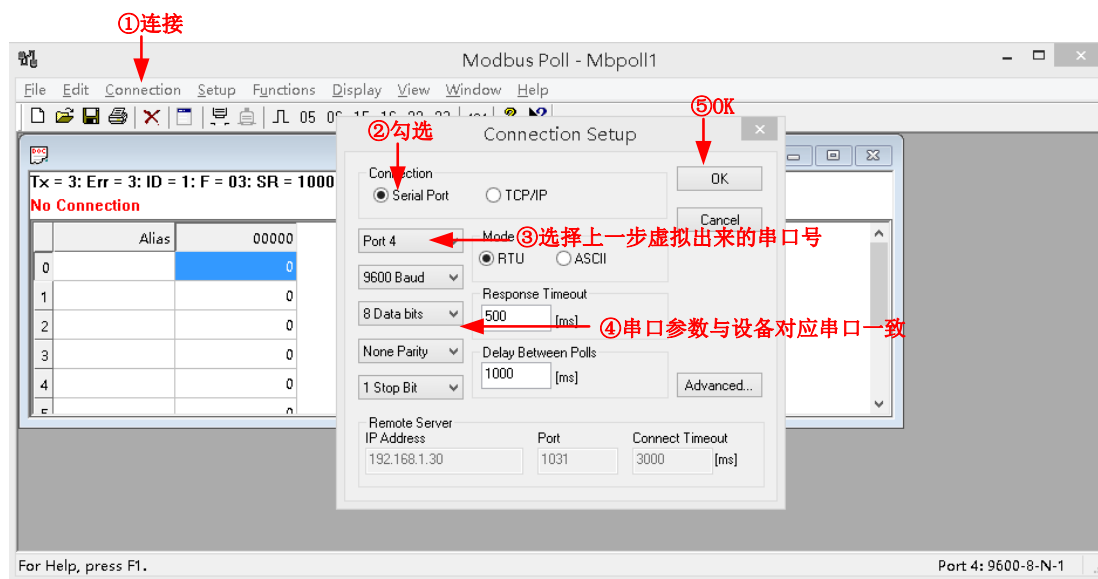


图 4.32 Modbus Poll 与设备建立 TCP 连接

读取 DI 输入状态

利用 02 功能码，读取设备的 8 路 DI 状态，具体步骤如所示。

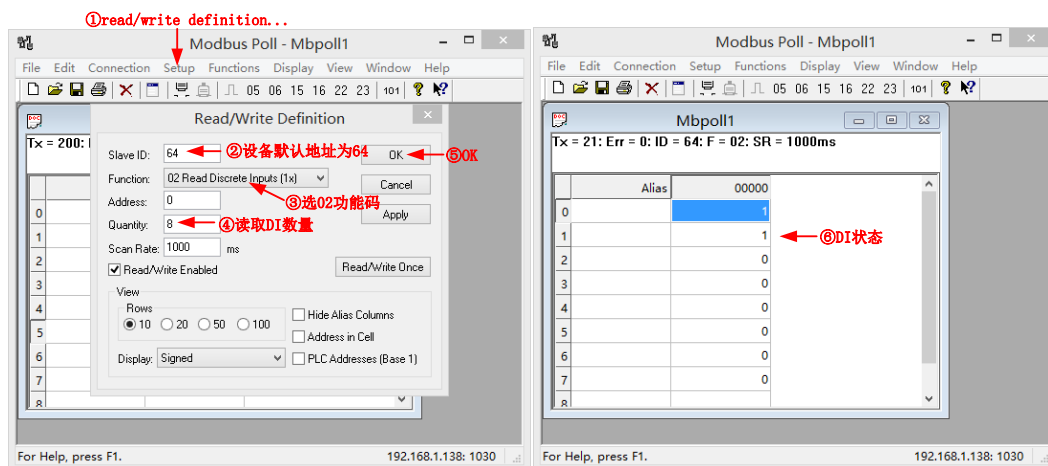


图 4.33 读取 8 路 DI 状态

8. 控制单路 DO 输出

利用 05 功能码，控制设备的某一路 DO 输出。具体步骤如图 6.4 所示。

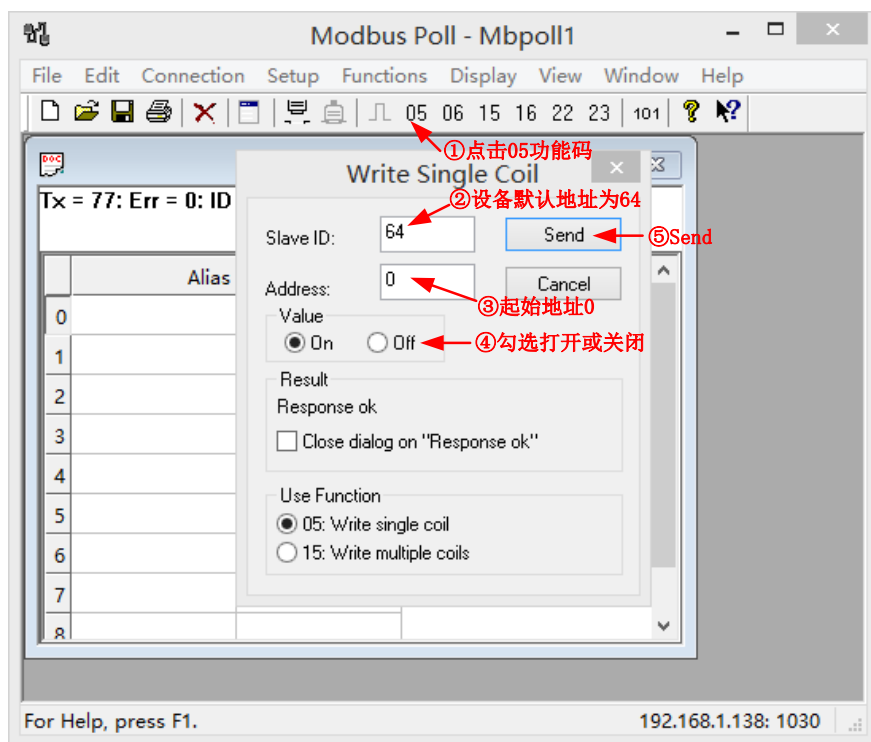


图 4.34 控制单路 DO 输出

9. 控制多路 DO 输出

利用 15 功能码，同时控制设备的多路 DO 输出。具体步骤如图 6.5 所示。

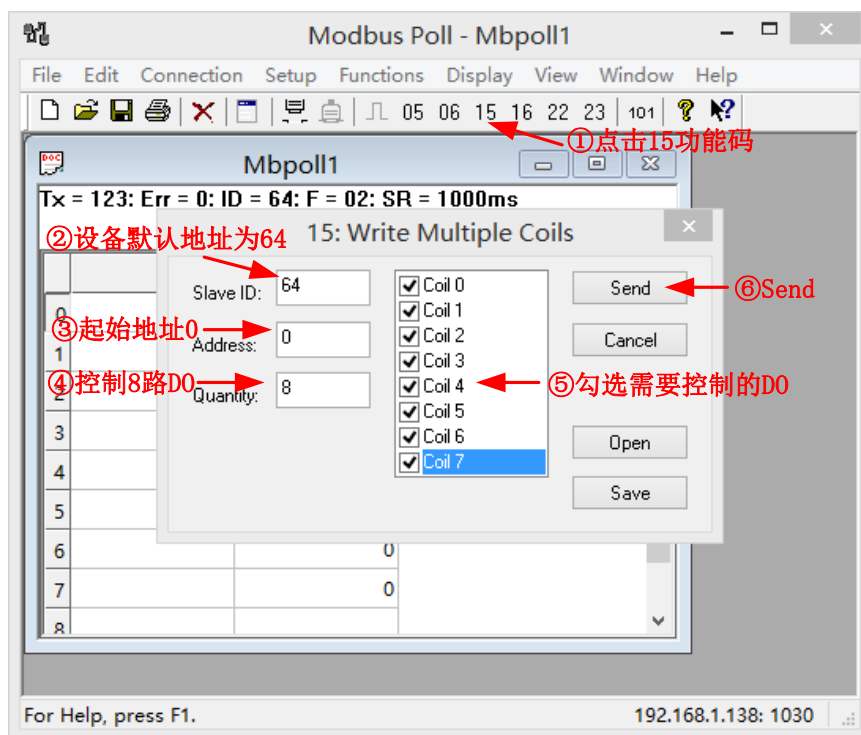


图 4.35 控制多路 DO 输出

5. 设备特色功能

5.1 支持 8 DI、8 路 DO

设备最大支持的开关量接口数量：8 个开关量输入（DI）、8 个开关量输出（DO），不同型号的设备支持的数量不同，具体可咨询客服。DI/DO 控制协议详见第 6 章。DI/DO 接线方法详见：《智嵌物联 IO 控制器设备接线说明》，下载地址：[点击下载](#)。

5.2 灵活设置设备的数据通道

串口通道：串口通道就是设备的物理通道：PORT1~PORT8，对应配置软件上的 U1~U8。

网络通道：设备支持 24 个网络通道 N1~N24，同一个设备 IP，24 个端口号。每个网络端口相互独立，可配置不同的参数。

数据通道：通过配置软件勾选对应的数据源，即可实现串口↔串口、串口↔网络的数据通道，每个数据通道之间相互独立，互不影响。

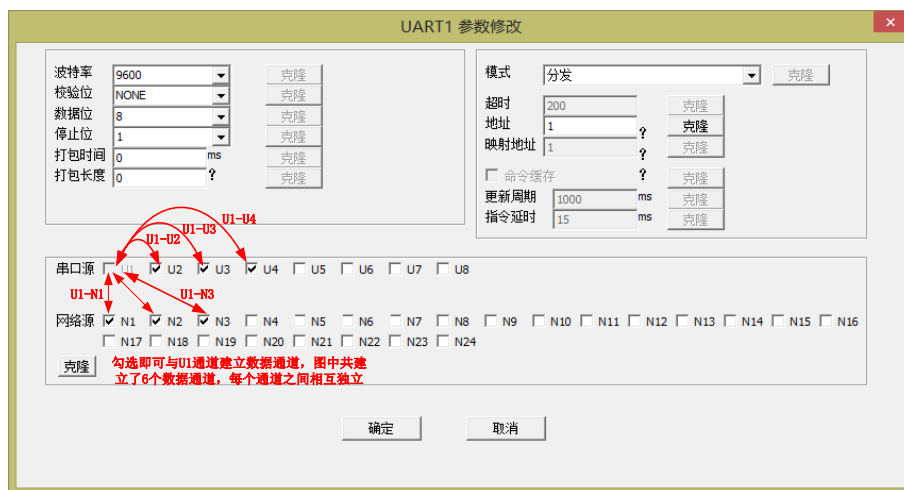


图 5.1 配置数据通道

5.3 可同时支持 24 个 socket、http、MQTT 连接

设备具有 24 个网络通道 N1~N24，每个网络通道可设置不同的网络工作模式，可将数据上传到不同的网络服务器，实现数据的备份等。

5.4 网络心跳包

在 TCP CLIENT、UDP CLIENT 模式下，用户可以根据需求设置心跳包数据和心跳包时间。当心跳包间隔设置为 0 或不勾选“启用心跳包”，心跳包功能不启用。

向服务器发送心跳包主要目的是为了保持连接稳定可靠，保证连接正常的同时还可以让服务器通过心跳包知道设备在线情况。用户可以选择让设备发送心跳包以实现特定的需求。

① 网络心跳包是在 TCP CLIENT、UDP CLIENT 模式下，一个心跳时间内没有数据向网络发送的时候才会发送，如果数据交互小于心跳时间，则不会发送心跳包。

5.5 网络注册包

在 TCP CLIENT、UDP CLIENT 模式下，用户可以根据需求来设置注册包的发送方式和注册包数据，也可以不使用注册包功能。

注册包可以作为设备获取服务器功能的识别码，也可以作为数据包头，方便服务器识别数据来源。

设备支持三种注册包发送方式，如图 5.2 所示。

图 5.2 注册包发送方式

注册包发送方式	说明
与服务器建立连接时，向服务器发送一次	连接服务器成功后，发送注册包到服务器，并且只发送一次
向服务器发送的每个数据包前都加上	向服务器发送数据时，在数据前增加注册包后发送到服务器
同时支持以上两种	连接服务器成功后，发送注册包到服务器，同时在向服务器发送数据时，在数据前增加注册包后再发送到服务器端

每个网络通道均可配置注册包、心跳包。以网络通道 N3 为例，说明注册包/心跳包的配置步骤。

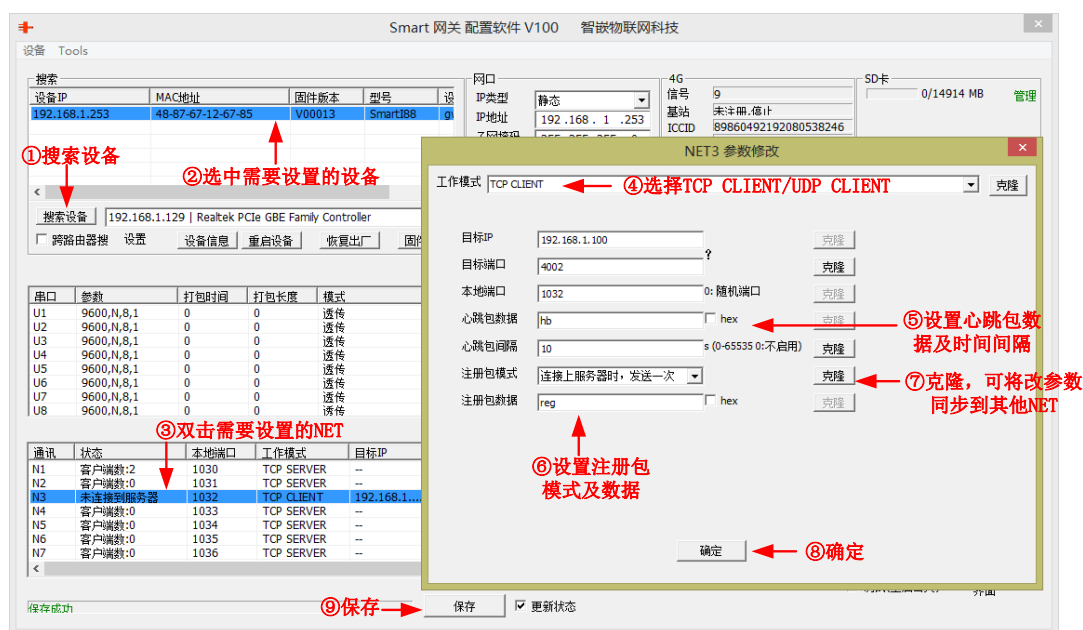


图 5.3 配置心跳包、注册包

5.6 网页配置

设备参数配置的另外一种方法是通过浏览器网页的方式来配置。

在浏览器中输入设备的 IP 地址，浏览器会弹出串口服务器参数配置的登录界面，输入用户名及密码（默认用户名：admin，默认密码：admin），点击【登录】按钮即可进入设备的参数配置界面。在网页配置界面，可以对设备的所有参数进行配置。

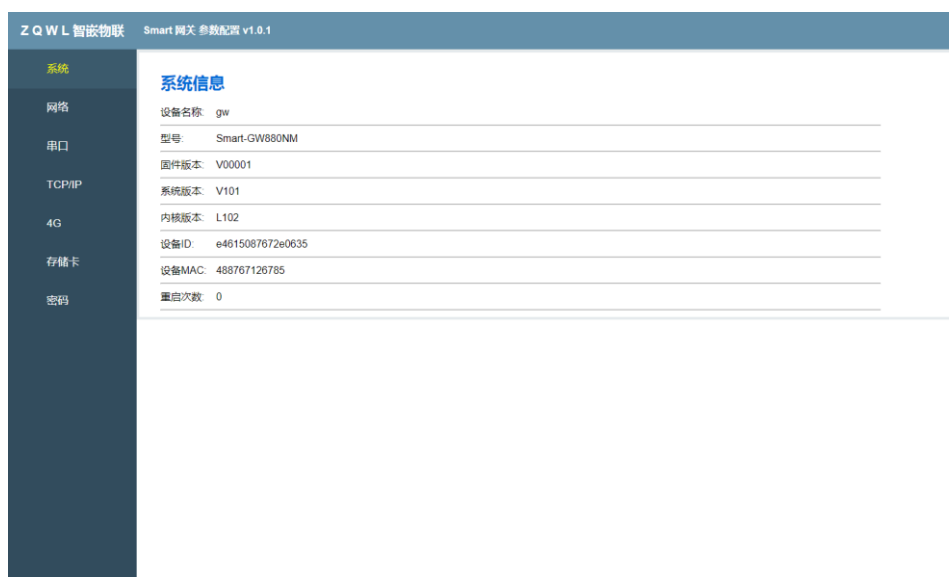


图 5.4 网页配置界面

① 设备的 IP 地址和用户电脑的 IP 地址必须在同一网段，否则不能弹出登录界面。

5.7 RTC 时钟

部分型号的设备支持 RTC 实时时钟。具体支持型号可联系客服。RTC 校准方法如图所示。

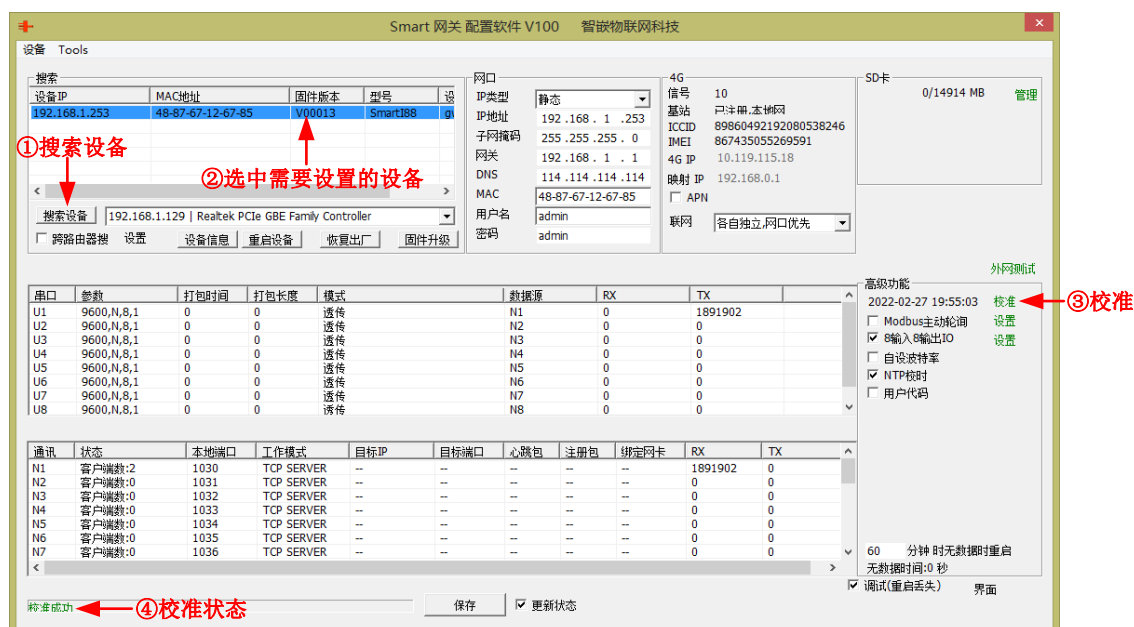


图 5.5 RTC 时钟校准

5.8 TF 卡保存数据

部分型号的设备支持插入 SD 卡，用以保存数据。具体支持插入 SD 卡的型号可联系客服。

5.9 GPS/北斗定位

部分型号支持 GPS/北斗定位，具体可联系客服确认。

设备内置高精度 GPS/BDS 定位模块，可实时精准定位设备所在的位置。通过用户服务器向设备发送定位信息的查询指令，设备会将 GPS 定位信息返回给服务器。

定位数据的查询指令为：{"cmd": "get", "data": ["gps"]}

设备会将经纬度信息返回给用户，返回指令为：{"cmd": "ret", "gps": [113.838871, 22.742772]}

用户可在 Google 地图上输入经纬度信息，查询定位位置，网址为：

<https://www.earthol.com/>



图 5.6 GPS 信息

5.10 支持 TLS 加密及证书加密

设备支持 TLS 证书加密，具体配置步骤如图 5.7 所示。

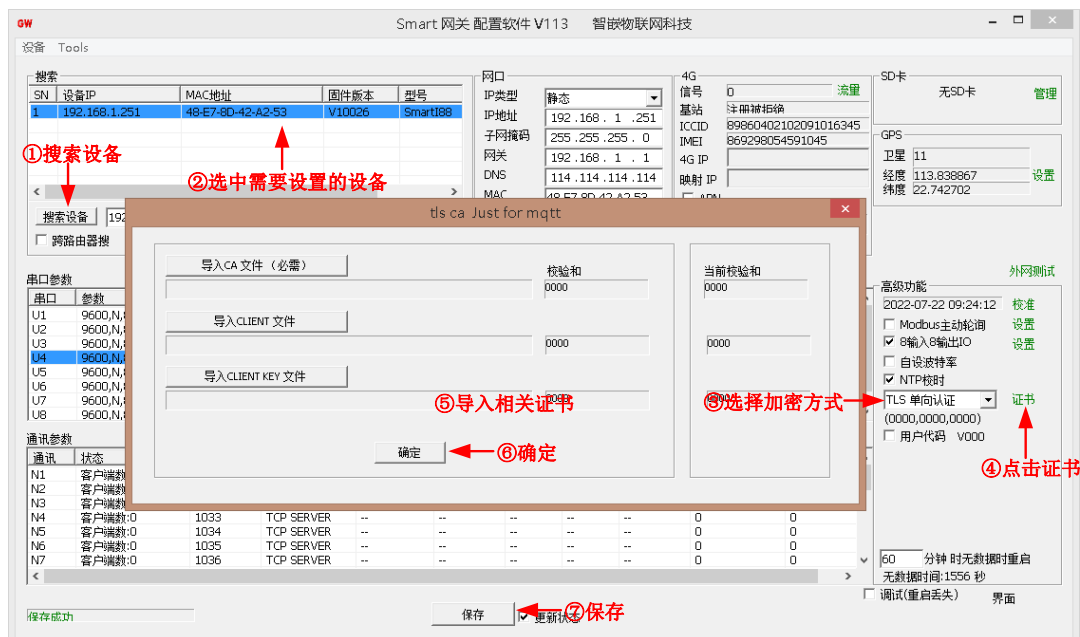


图 5.7 SSL 加密配置步骤

5.11 网络无数据设备自动重启

设备支持网络无数据时，设备自动重启。该功能主要是为了保证设备长期稳定的工作，当网络上在设置的时间内无任何数据时，设备会自动重启，从而避免异常情况对设备通信造成影响。配置方法如图所示。



图 5.8 无数据重启配置

5.12 自设波特率

在一些应用场合，为了传输数据的加密性，不少设备在数据传输的过程中会改变数据的波特率、校验位、数据长度等相关参数，此时，可启用设备的自设波特率功能，从而可实现现在数据传输过程中改变设备的串口参数的目的。

当需要改变串口参数时，通过网络向设备发送自设波特率的协议帧，设备收到协议帧后，会自动修改设备的串口参数。本次设置，本次生效，设备掉电则失效。

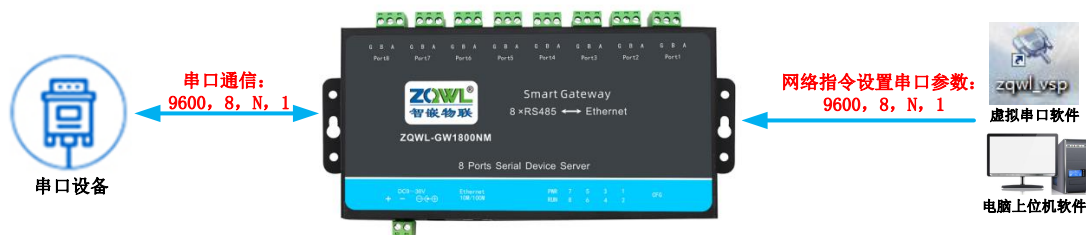


图 5.9 自设波特率原理框图

表 5.1 自设波特率协议

	帧头	波特率	数据位/停止位/校验位	校验和
字节数	3 byte	3 byte	1 byte	1 byte
说明	55 AA 55	波特率值换成 16 进制，高位在前。	具体详见 4.2	除去帧头的 4byte 之和，取低字节
举例 1: 9600,N,8,1	55 AA 55	00 25 80	03	A8
举例 2: 115200,N,8,1	55 AA 55	01 C2 00	03	C6

表 5.2 数据位/停止位/校验位含义

项目	无意义	校验位	停止位	数据位
8 bit 位	7:6	5:4:3	2	1:0
位含义	00	000: 无校验 001: ODD 奇校验 011: EVEN 偶校验	0: 1 个停止位 1: 2 个停止位	10: 7 个数据位 11: 8 个数据位
举例: N,8,1 (0000011, 即 0x03)	00	000	0	11

启用自设波特率的配置如图所示。

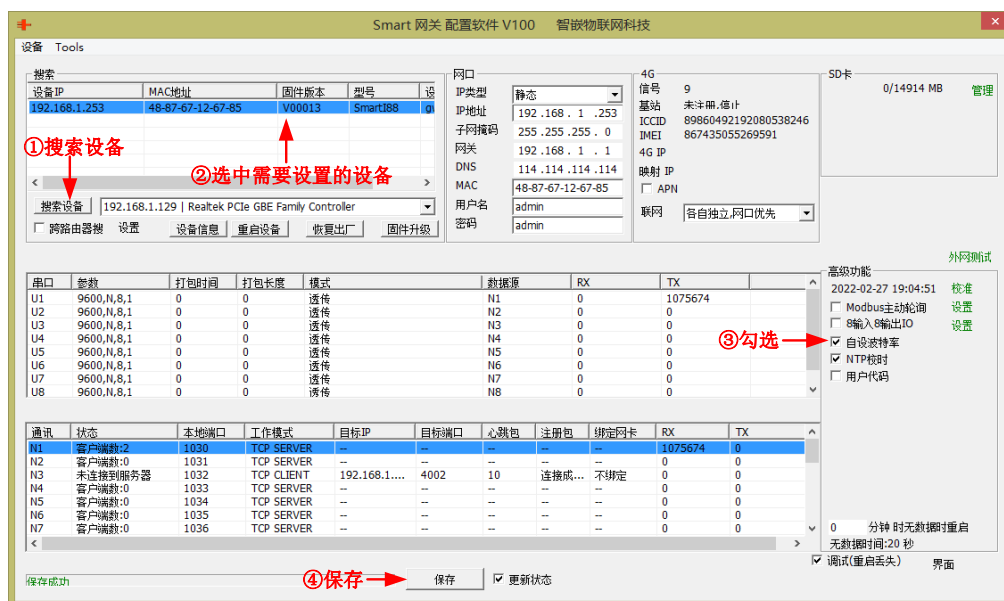


图 5.10 配置自设波特率

- ① 若用户使用智嵌物联的虚拟串口软件，可使用自设波特率功能，这样用户可以不用关心 U1~U2 的串口参数，虚拟串口软件会按照虚拟出的串口参数，给设备发送配置指令。

5.13 NTP 校时

设备支持 NTP 校时，设备首先必须要能连到外网，这样设备才会向 NTP 服务器请求时间。当设备只是在局域网内使用时，无法使用该功能，此时用户可选择支持 RTC 实时时钟功能的设备，从而实现获取时间的目的。

设备启用 NTP 校时功能的配置步骤如所示。



图 5.11 NTP 校时配置

5.14 跨路由搜索设备

设备支持跨路由搜索，即用户可以跨网段搜索到设备，并对设备进行参数配置及调试。
跨路由搜索设备的步骤如所示。

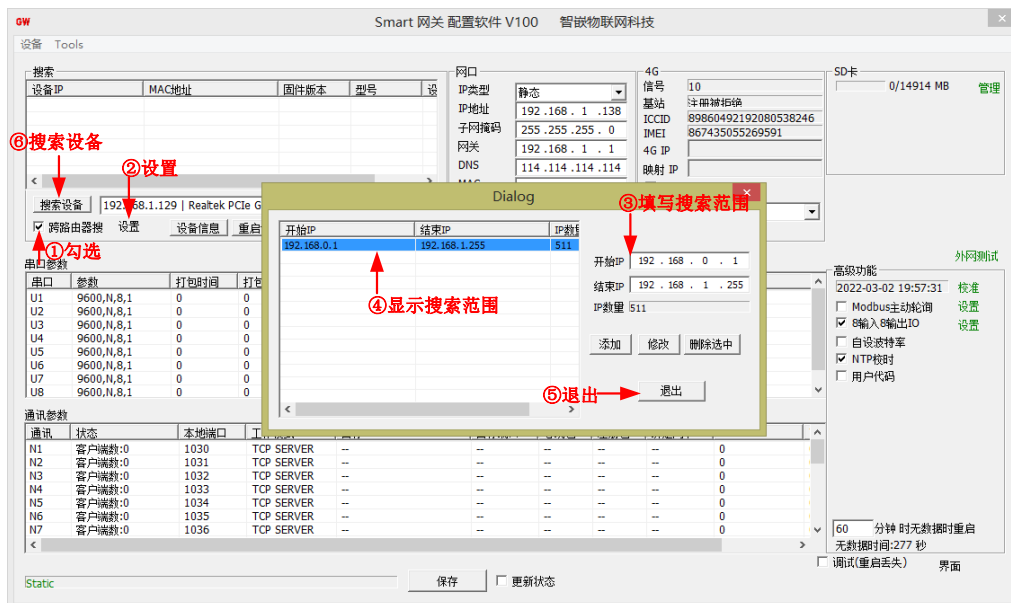


图 5.12 跨路由搜索配置

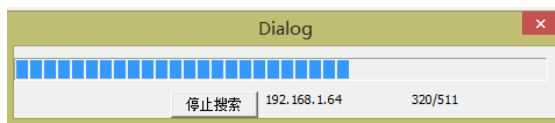


图 5.13 跨路由搜索进度条

5.15 支持配置参数导入、导出

用户可选择配置参数的导入、导出功能，当用户需要修改很多台设备的参数时，使用该功能事半功倍。

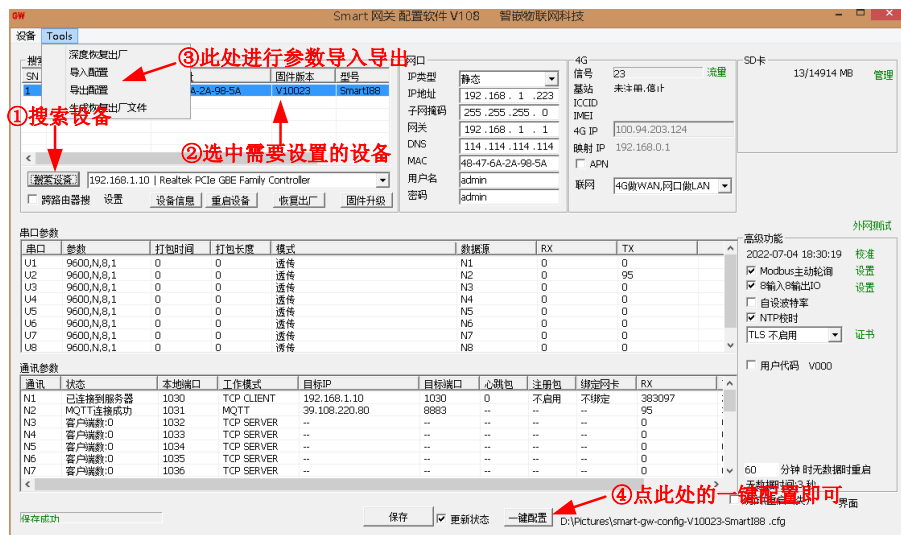


图 5.14 导入导出设备参数

5.16 调试日志

设备的参数配置软件，提供了丰富的调试信息，遇到问题，无需盲目调试，用户可根据设备调试信息，快速定位问题。通过调试信息，设备的联网状态、各个数据通道的数据、接收时间、超时信息等一览无余。

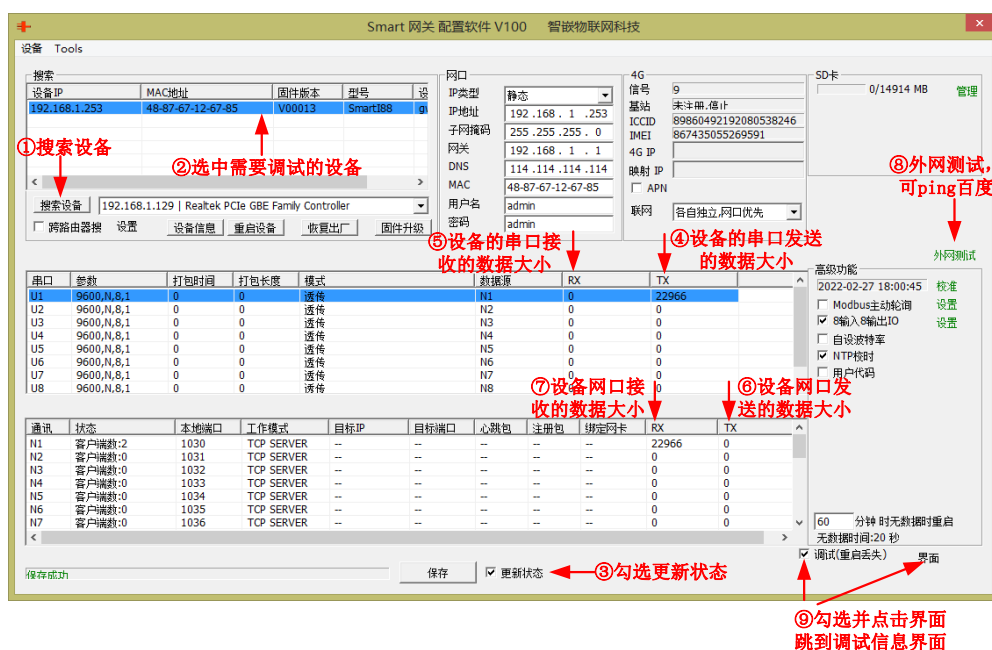


图 5.15 调试界面 1

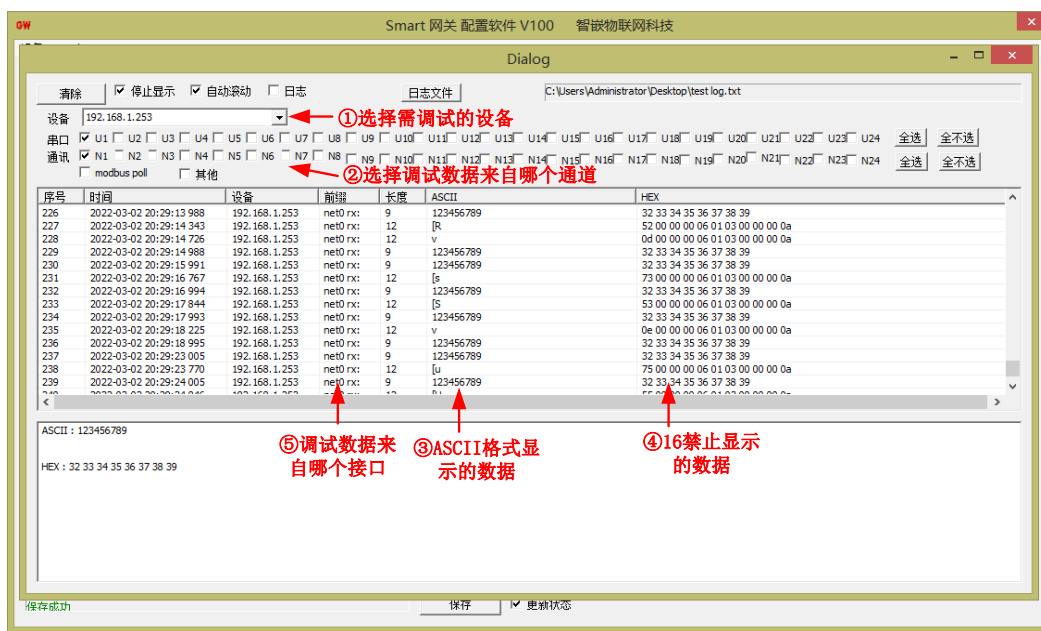


图 5.16 调试界面 2

5.17 二次开发

该设备支持用户二次开发，智嵌物联提供已经封装好的函数库，用户直接调用函数接口即可，无需关心内部复杂的联网机制。具体开发流程及方法详见智嵌物联 Smart Gateway 设备二次开发相关文档。

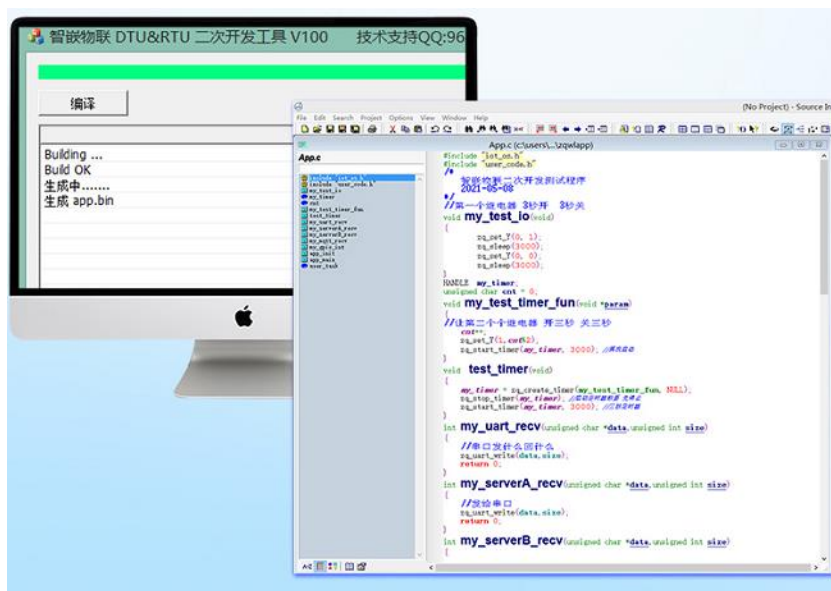


图 5.17 二次开发界面

❗ 若现有功能不能满足需求，联系我们，可支持个性化定制。

6.8 路 DI、8 路 DO 控制协议

设备支持 8 路 DI、8 路 DO，可通过标准的 Modbus TCP/RTU 协议来获取 DI/DO 状态及控制 DO 输出。DI、DO 接线说明详见：《智嵌物联 IO 控制器设备接线说明》，下载地址：[点击下载](#)。

ModBus RTU 协议是工业上常用的通讯协议，采用查询+响应的通讯模式：主站设备向指定地址的从站设备发送请求报文，从站设备收到指令后做出相应的动作，并向主站发送响应报文。ModBus RTU 协议的通讯格式如下：



6.1 地址域（Additional address）

Additional address 为从站地址，取值范围 0x00~0xff，设备的默认地址 0x40（即 64）。

当 Additional address 为 0xff 时，是广播地址，所有从机都能接收并处理，必要时要做回应。广播地址可以用于对控制板的编址以及获取控制板的地址。

6.2 功能码（Function code）

本设备支持的 ModBus RTU 功能码如表 6.1 所示。

表 6.1 ModBus RTU 功能码

功能码	含义	功能	属性
0x01	读多个线圈	Data: 2 字节起始地址+2 字节线圈个数，线圈个数不能超过 8	读
0x02	读离散量输入	Data: 2 字节起始地址+2 字节输入点个数，个数不能超过 8	读
0x05	写单个线圈	Data: 2 字节起始地址+2 字节线圈值	写
0x0f	写多个线圈	Data: 2 字节起始地址+2 字节线圈个数+1 字节个数+数值	写

1. 功能码：0x01

读 DO 继电器输出状态：

字节数	1	1	1	1	1	1	1	1
名称	设备地址	功能码 0x01	起始地址 高	起始地址 低	读取数量 高	读取数量 低	CRC 高	CRC 低
取值范围	0x01~0xff	0x01	0x00	0x00~0x07	0x00	0x00~0x08	CRC 计算值	

2. 功能码：0x02

读 DI 输入状态：

字节数	1	1	1	1	1	1	1	1
名称	设备地址	功能码 0x02	起始地址 高	起始地址 低	读取数量 高	读取数量 低	CRC 高	CRC 低
取值范围	0x01~0xff	0x02	0x00	0x00~0x07	0x00	0x00~0x08	CRC 计算值	

3. 功能码：0x05

写单路 DO 继电器输出：

字节数	1	1	1	1	1	1	1	1
名称	设备地址	功能码 0x05	起始地址 高	起始地址 低	写单路 DO	0x00	CRC 高	CRC 低
取值范围	0x01~0xff	0x05	0x00	0x00~0x07	FF:继电器闭合 00:继电器断开	0x00	CRC 计算值	

4. 功能码：0x0f

写多路 DO 继电器输出：

字节数	1	1	1	1	1	1	1	1	1	1
名称	设备地址	功能码 0x0f	起始地址 高	起始地址 低	写 DO 数量 高	写 DO 数量 低	字节 数量	写继电器	CRC 高	CRC 低
取值范围	0x01~0xff	0x0f	0x00	0x00~0x07	0x00	0x00~0x08	0x01	每个 bit 代表 1 路 DO	CRC 计算值	

6.3 8 路 DI/DO 控制举例

6.3.1 通过配置软件测试 DI/DO 功能

设备可以通过配置软件，简单测试 DI、DO 的功能是否正常。具体测试步骤如图 6.1 所示。

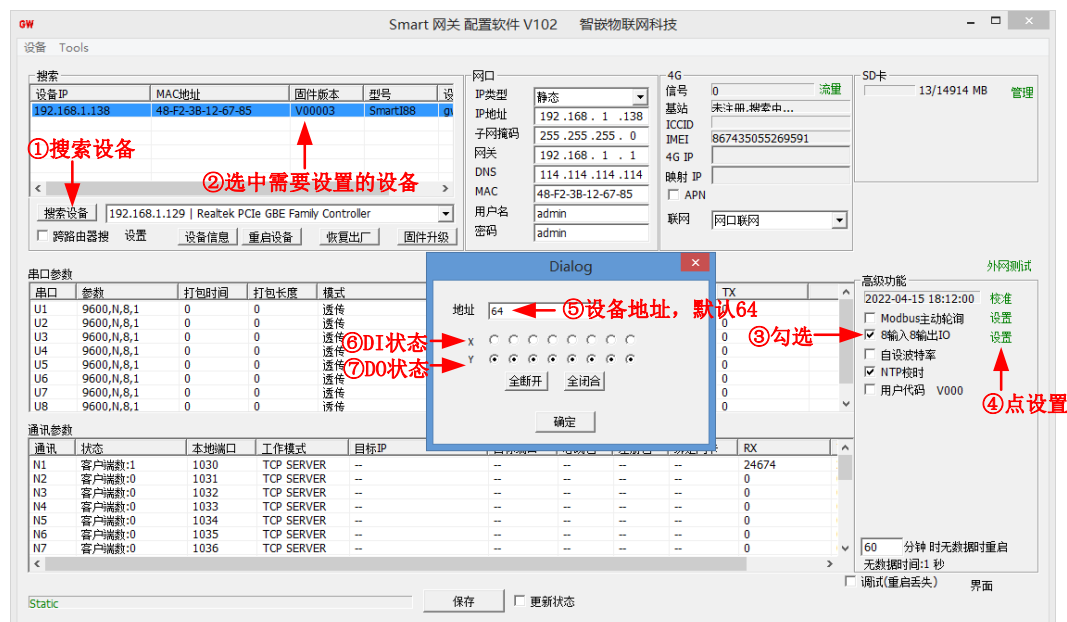


图 6.1 配置软件控制 DI/DO

6.3.2 通过 Modbus Poll 工具测试 DI/DO 功能

可将设备的 N1 通道配置为 TCP SERVER 模式，默认配置即可，Modbus Poll 软件主动与

设备建立连接，连接配置如图 6.2 所示。TCP 连接成功之后，就可以读取或控制设备的 DI/DO 状态了。

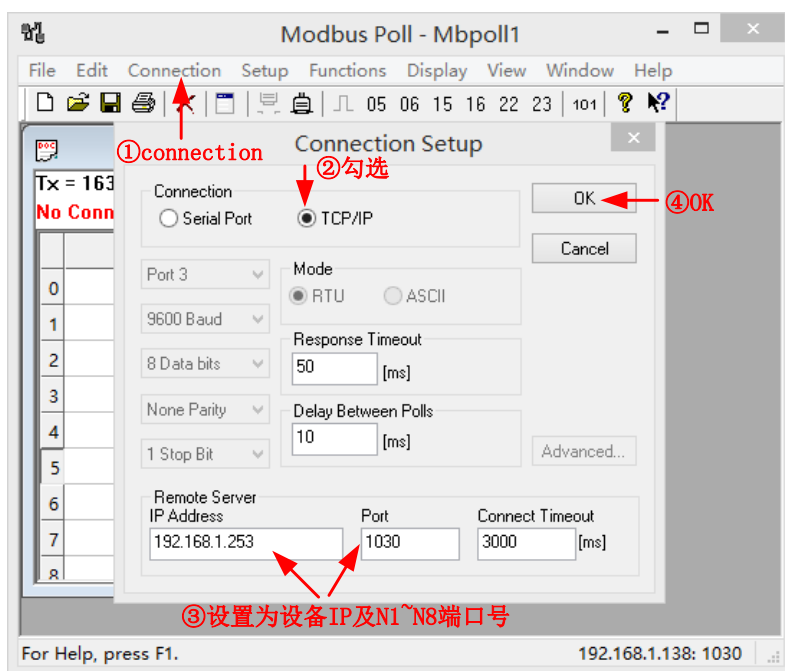


图 6.2 Modbus Poll 与设备建立 TCP 连接

1. 读取 DI 输入状态

利用 02 功能码，读取设备的 8 路 DI 状态，具体步骤如所示。

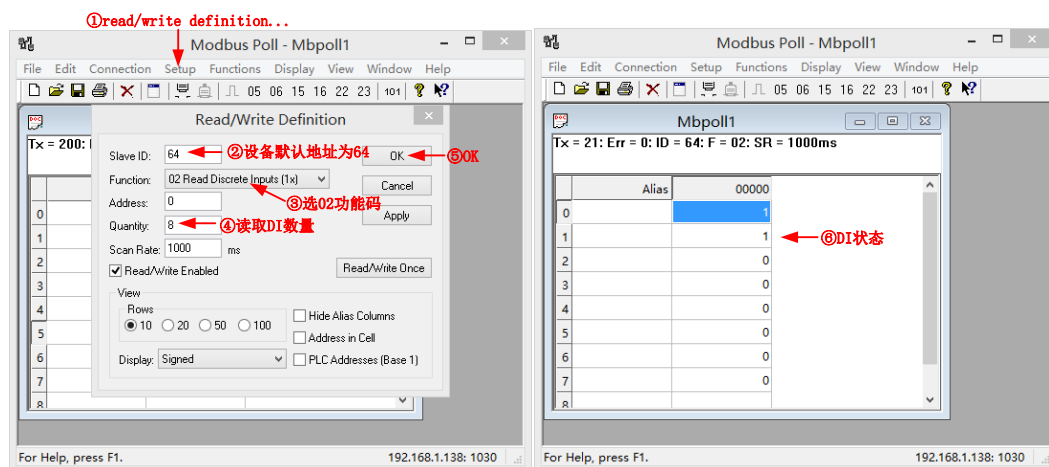


图 6.3 读取 8 路 DI 状态

2. 控制单路 DO 输出

利用 05 功能码，控制设备的某一路 DO 输出。具体步骤如图 6.4 所示。

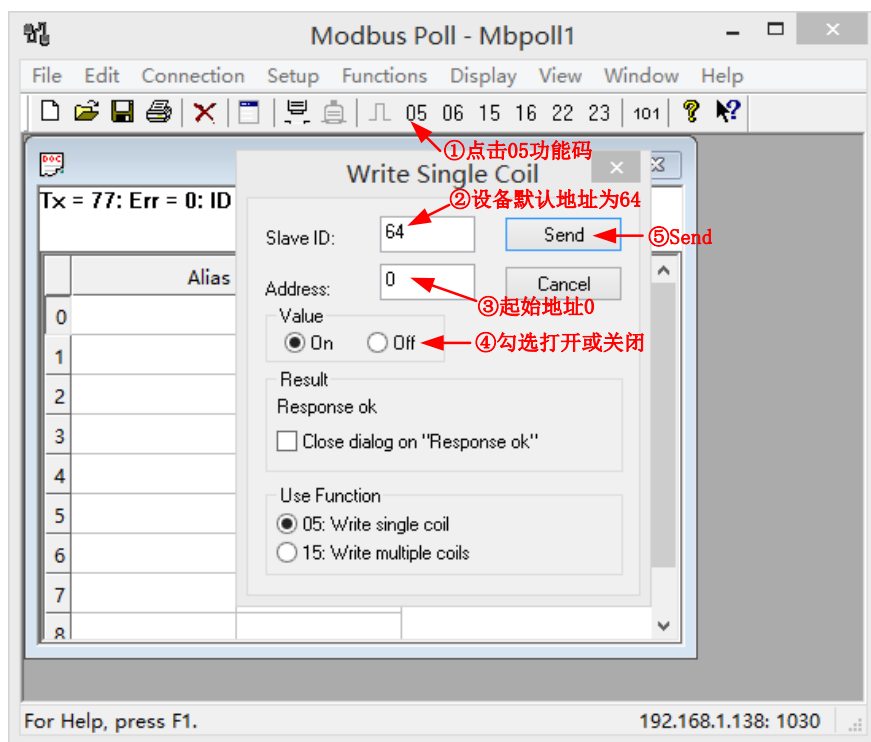


图 6.4 控制单路 DO 输出

3. 控制多路 DO 输出

利用 15 功能码，同时控制设备的多路 DO 输出。具体步骤如图 6.5 所示。

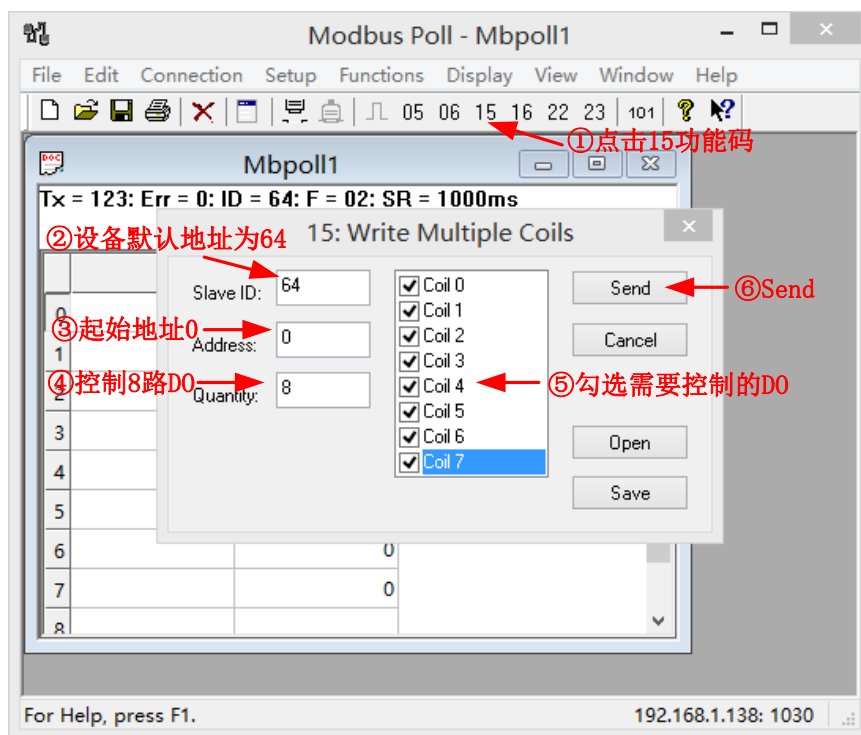


图 6.5 控制多路 DO 输出

用户如果需要查看对应的控制指令，可以点击 101，即可查看设备的控制指令及应答指

令，具体步骤如图 6.6 所示。

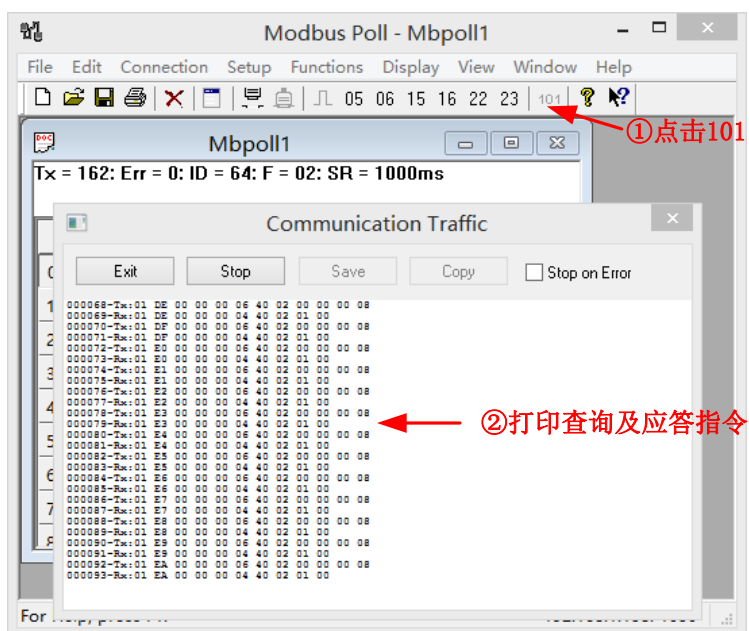


图 6.6 打印查询及应答指令

7. ModBus 主动轮询

7.1 Modbus 主动轮询原理

启用 Modbus 主动轮询功能后，用户服务器只需要接收数据，不需要再下发查询指令，可极大的缓解服务器的压力。

启用该功能后，设备会按照用户事先设置的 ModBus 指令轮询串口设备，设备会将不同指令返回的数据保存到缓存里。定时主动将缓存中的所有数据一次性上传到设置好的网络通道上。

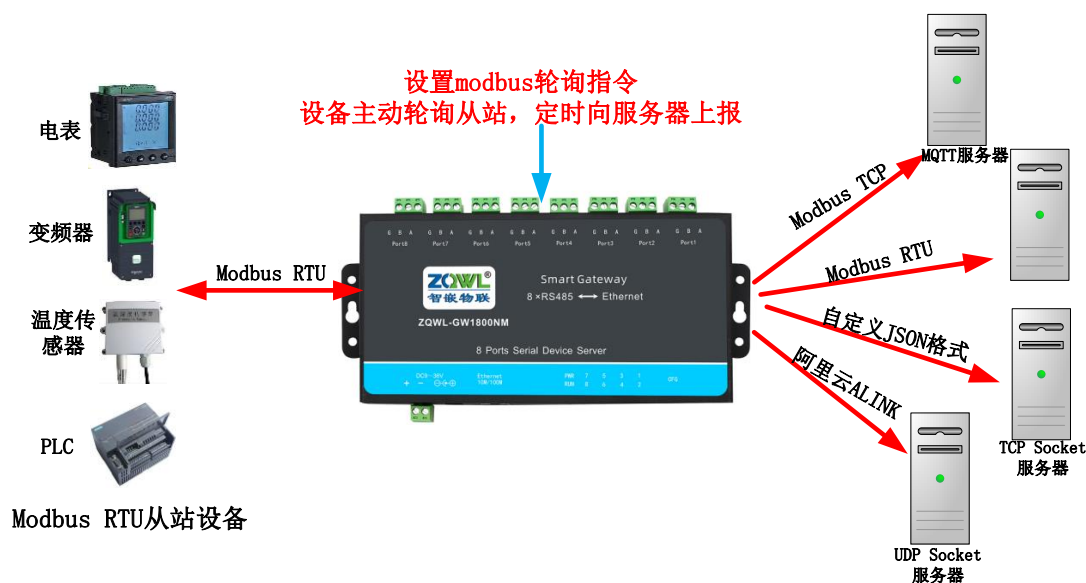


图 7.1 Modbus 主动轮询原理框图

设备 Modbus 主动轮询功能配置步骤如所示。



图 7.2 Modbus 主动轮询功能配置步骤 1

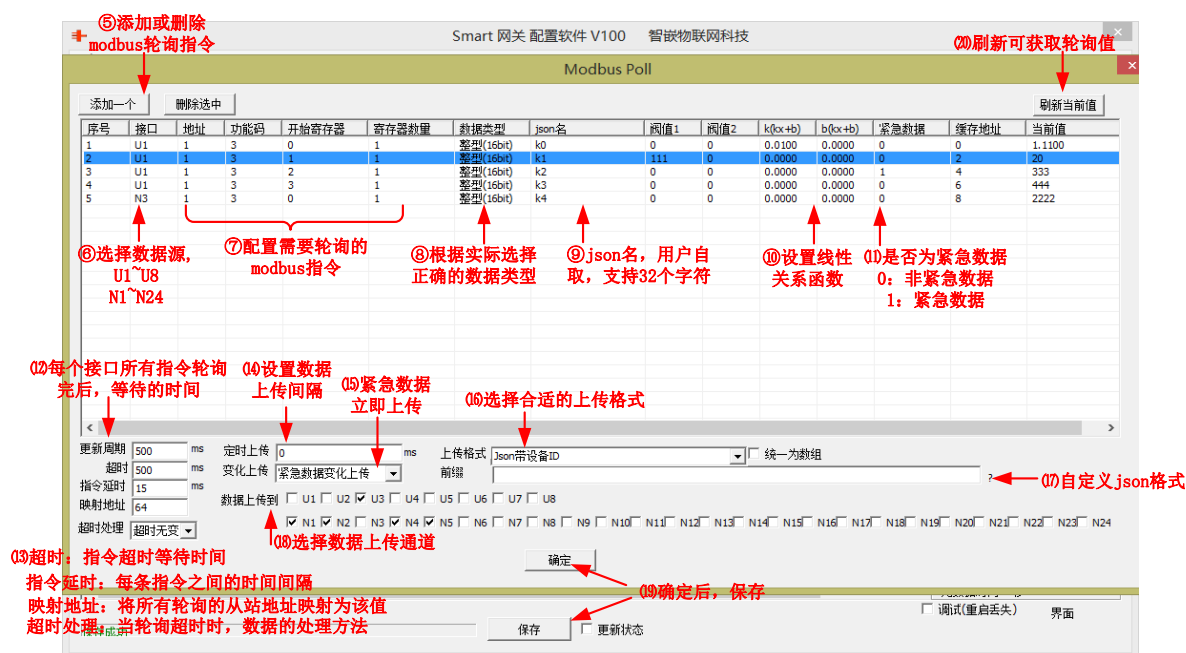


图 7.3 Modbus 主动轮询功能配置步骤 2

轮询接口：轮询接口可以选择设备的 8 个串口通道 PORT1~PORT2（U1~U8）、24 个网络通道（N1~N24），每个轮询接口之间相互独立。可以轮询 Modbus RTU 协议，也可以轮询 Modbus TCP 协议。

数据类型：支持 BYTE、整型（16bit）、整形（32bit 高在前）、整形（32bit 高在后）、浮点型（高在前）、浮点型（高在后）。用户根据实际情况选择合适的数据类型。

JSON 名：JSON 名支持自定义，最大支持 32 个字符。

kx+b：设备可对轮询的数据进行简单的线性计算。比如 k 设置为 0.01，b 设置为 0，则将数据缩小 100 倍后，上传到服务器。

紧急数据：对于非常重要且实时性要求比较高的寄存器数据，用户可将该数据设置为紧急数据，并将变化上传条件设置为紧急数据变化上传，则该数据一旦变化，就会立即上传，轮询间隔时间对此无效。设置 0，则该数据为非紧急数据，设置 1，则该数据为紧急数据。

轮询间隔：每个数据通道所有设置的指令，全部轮询完之后，等待该轮询间隔时间后，才会进行下一轮的指令轮询。轮询间隔时间设置的越小，数据的实时性也就越高。

超时时间：当轮询指令发出后，从站设备没有应答，网关设备需要等待该超时时间后，才会轮询下一条指令。

超时处理：当轮询指令发出后，没有收到从站设备的应答，此时上报的数据可以设置为上一次轮询的数据（超时无变化）、设置为 FF（超时清 FF）、设置为 00（超时清 00）。

指令延时：同一个数据通道中，每条轮询指令之间的时间间隔。不同的数据通道之间是独立的，可以并发轮询。

映射地址：将所有数据通道的轮询的从站地址，设置为该映射地址，默认 128。当上传格式选择 Modbus RTU 或 Modbus TCP 格式上传时，则会按映射地址上传。服务器下发设置指令，也是按照映射地址下发。

缓存地址：缓存地址可以理解为用户串口设备的 Modbus 寄存器地址的映射地址。当上传格式选择 Modbus RTU 或 Modbus TCP 格式上传时，该缓存地址即为上传的寄存器地址。

用户解析上传的数据时，可根据该缓存地址，来确定该数据来自那一条指令。

定时上传：数据上传服务器的时间间隔，单位毫秒，设置为 0，则不上传。如设置 5000ms，则设备会每隔 5 秒将数据上传到指定的服务器。

变化上传：若某个轮询数据对用户比较重要，可将该数据设置为紧急数据，并启用变化上传，这样网关设备会将本次轮询的数据与上一次的数据做比较，若有变化，则立即上传服务器，不需等待定时上传设置的时间间隔。

上传格式：数据上传格式支持 Modbus RTU、Modbus TCP、JSON 带设备 ID、JSON 自定义、阿里云 ALINK 等。

JSON 自定义格式说明：

"id":%id: 设备的唯一标识符，可在配置软件中的设备信息中查看，适用于基于设备识别的应用。

"imei":%imei: 设备内的 4G 模块的唯一识别码。

"iccid":%iccid : SIM 卡的唯一识别码，适用于基于 SIM 卡识别的应用。

"time":%t: 时间戳。

%d: 轮询的数据。

如设置为：{"id":%id,"imei":%imei,"iccid":%iccid,"time":%t,%d}

则上报的数据为：

{"id":e461211757372e32,"imei":,"iccid":,"time":2022-03-0317:13:15,"k0":0,"k1":0}

数据上传到：可将数据上传到多个网络通道或串口通道，通过配置软件勾选即可。

7.2 Modbus 主动轮询—数据上报/下发举例

7.2.1 ModBus 格式上传与下发

用 modbus Slave 模拟从站设备，网络调试助手和 Modbus Poll 模拟主站或服务器。

本次实验实现效果：网关设备主动轮询 modbus 从站数据（U1 通道和 N1 通道），并将数据上传到主站或服务器（N2 通道和 N3 通道）。同时主站或服务器也可向从站写数据。

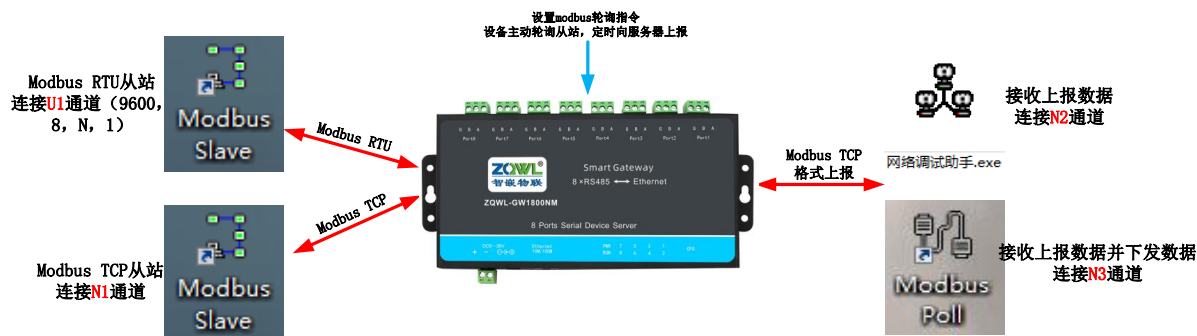


图 7.4 本次实验原理

1. 设置 Modbus 主动轮询指令

轮询数据通道选择 U1：向设备的第一路串口 PORT1 发送 Modbus RTU 轮询指令：

轮询数据通道选择 N1：向设备的网络通道 N1 发送 Modbus TCP 轮询指令；

映射地址：默认 128，将 U1、N1 通道上的从站地址全部映射为 128。

数据上传到：勾选 N2、N3，将轮询应答数据上传到网络通道 N2、N3。



图 7.5 主动轮询配置



图 7.6 轮询通道 U1、N1 并上传到 N2、N3 通道

2. 设置网络通道 N1 的参数

与网络通道 N1 相连接的设备作为 Modbus TCP 从站，因此网关设备的工作模式设置为 TCP CLIENT，目标 IP 及端口号即为从站设备的 IP 及本地端口号。

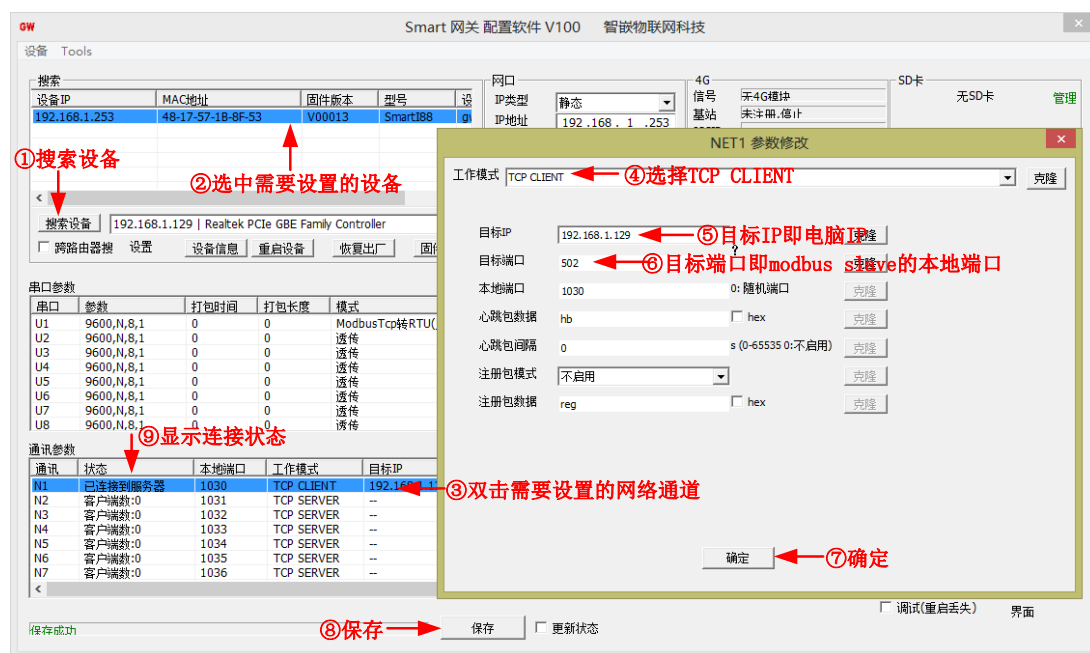


图 7.7 网络通道 N1 配置为 TCP CLIENT

3. 设置串口通道 U1 的参数

串口通道 U1 即为设备的第一路串口 PORT1，通过 USB 转 RS485 或 RS232 线连接电脑。U1 参数默认即可（9600,8,N,1）。

4. 设置网络通道 N2、N3 的参数

网络通道 N2、N3 作为轮询数据的上传通道，本次举例采用设备默认参数：

网络通道 N2：TCP SERVER 模式，IP 为 192.168.1.253，端口号为 1031

网络通道 N3：TCP SERVER 模式，IP 为 192.168.1.253，端口号为 1032

5. 打开两个 Modbus Slave 软件

打开两个 Modbus Slave 软件，分别模拟 Modbus RTU 从站（与串口通道 U1 通信）、Modbus TCP 从站（与网络通道 N1 通信）。

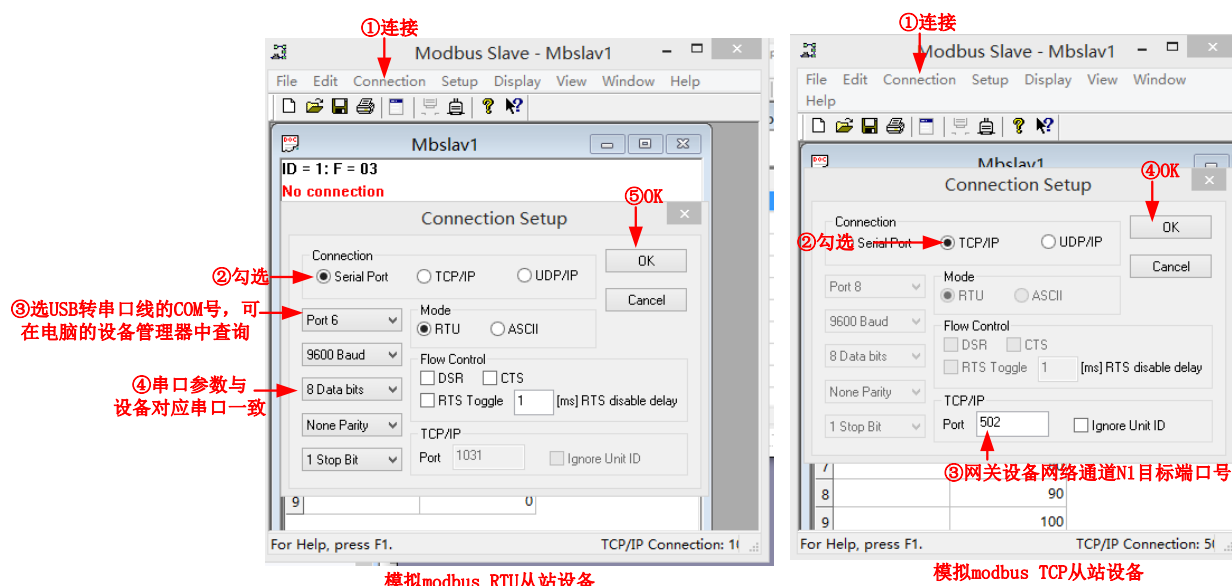


图 7.8 图 7.9 配置 Modbus Poll 和 Slave

6. 打开一个网络调试助手

打开一个网络调试助手，连接设备的网络通道 N2，在网络调试助手中就可以接收到主动轮询的数据。

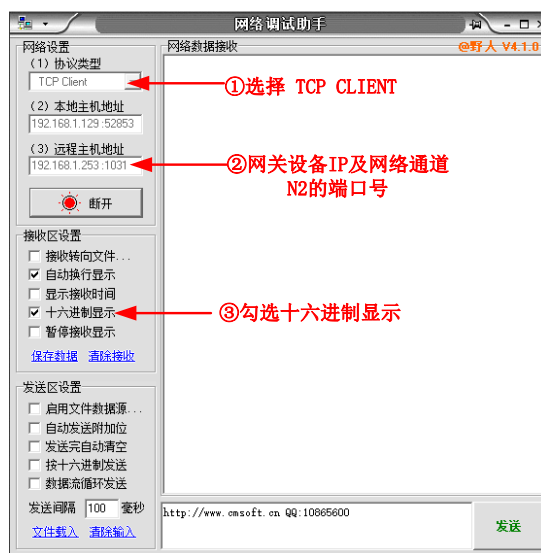


图 7.10 配置网络调试助手

7. 打开一个 Modbus Poll

打开一个 Modbus Poll，模拟 Modbus TCP 主站，连接设备的网络通道 N3（192.168.1.253:1032）。连接之后即可接收主动轮询的数据。

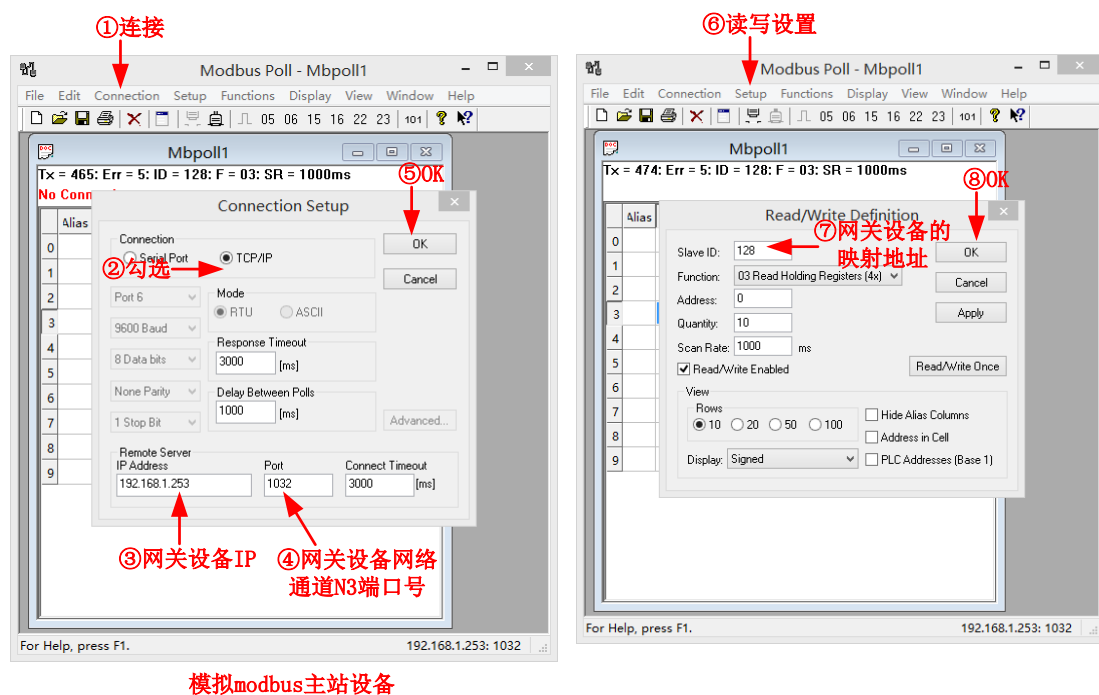


图 7.11 配置 Modbus Poll

8. 通信结果

通过以上配置之后，Modbus 主站设备就可以对从站进行读写操作了。

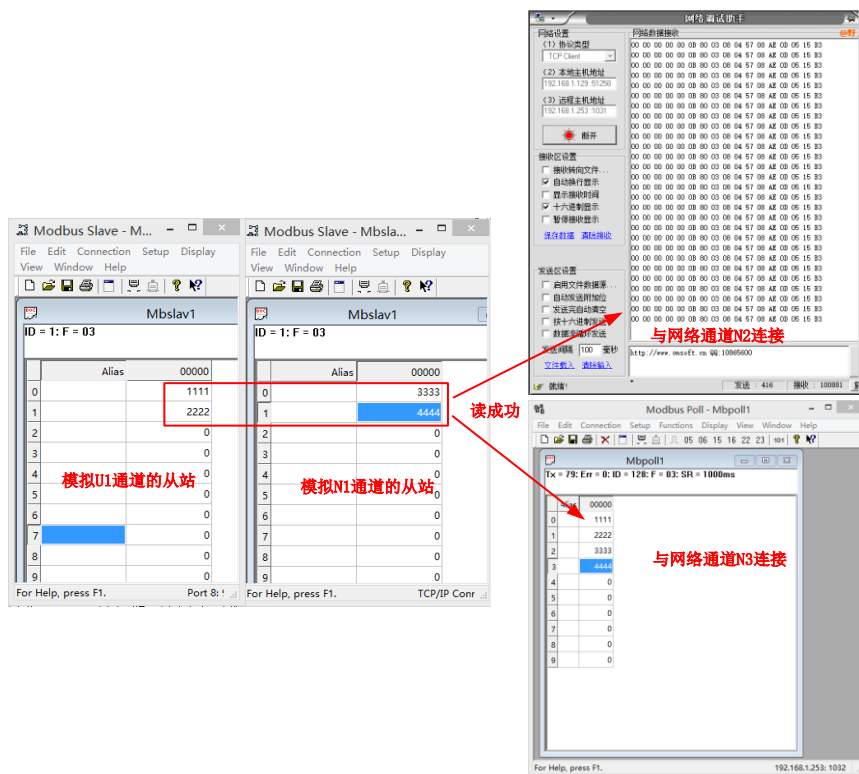


图 7.12 网络通道 N2、N3 接收到上报的数据

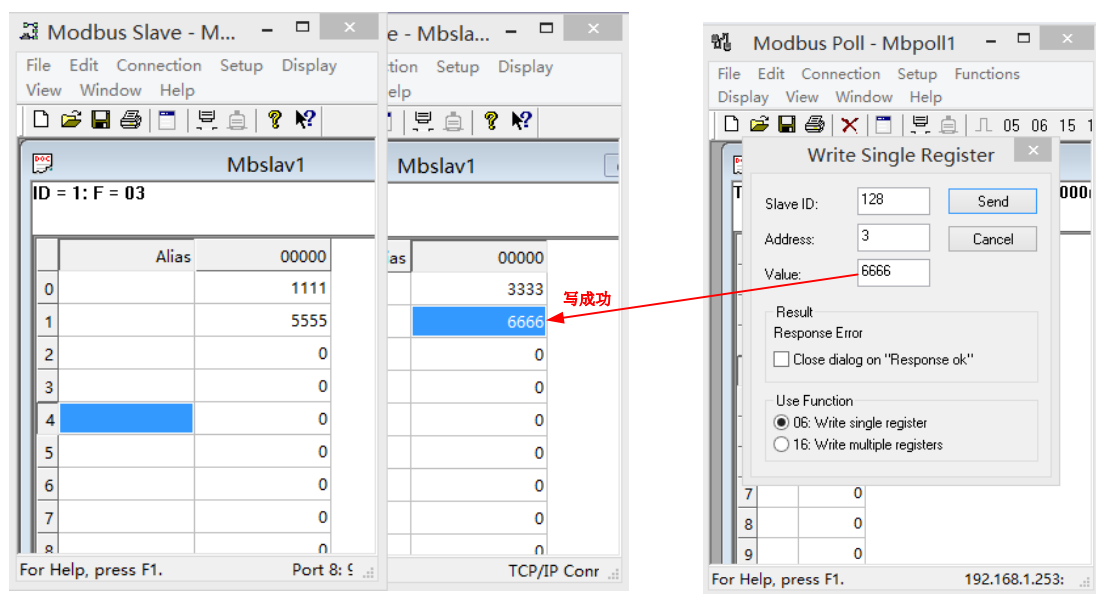


图 7.13 Modbus Poll 软件向从站写数据

7.2.2 JSON 格式上传与下发

1. 数据上传

在上一节的基础上，将设备 modbus 主动轮询的上报格式设置为 JSON 格式上传。网络调试助手不要 16 进制显示，这样 U1、N1 通道的 Modbus 从站数据就会转成 JSON 格式向网络通道 N2、N3 上发送。

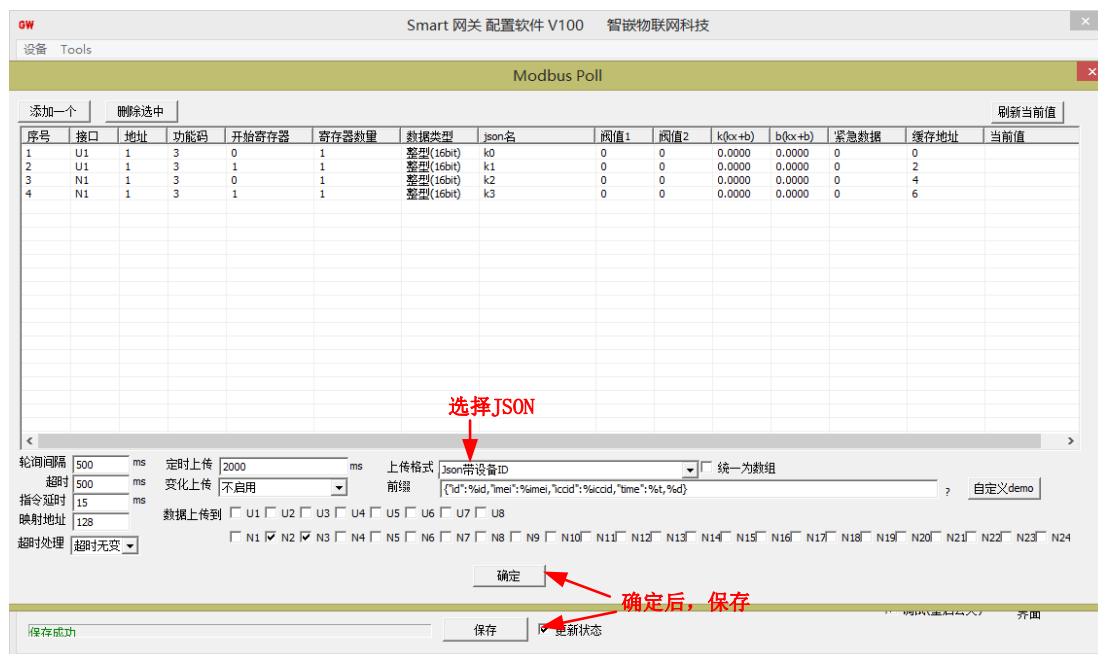


图 7.14 上传格式设置为 JSON 格式

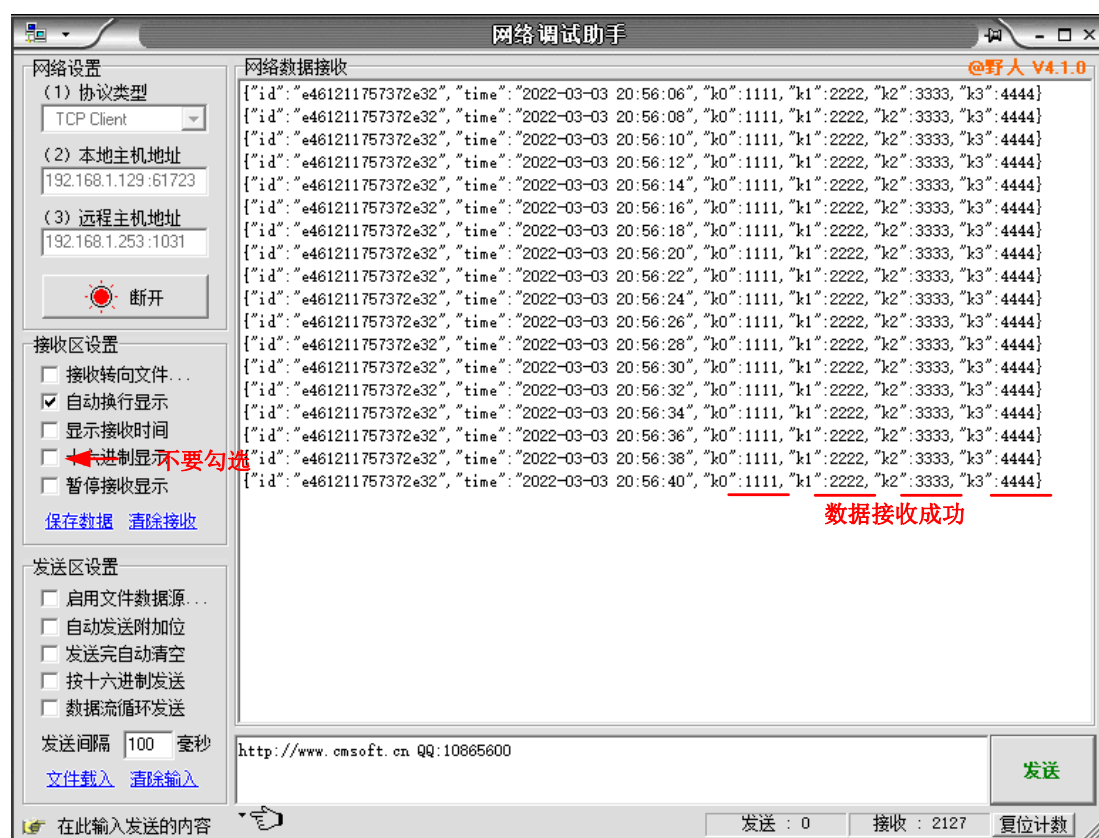


图 7.15 网络通道 N2 接收到上报的 JSON 数据

2. JSON 格式的数据下发协议

若用户需要对用户的串口设备进行写操作，如写线圈、写寄存器等，用户的服务器可按照网关设备的 JSON 格式的数据下发协议格式，向网关设备下发数据，网关设备收到服务器的下发数据后，会将解析后的数据发送到串口上；串口设备的应答数据，网关设备会转换成 JSON 格式上传给服务器。

JSON 格式的数据下发协议如下：

服务器下发格式：{"mb":"010600000008","sn":1,"ack":0,"crc":0,"uart":0}

网关设备应答数据格式：{"retmb":"010600000008880c","sn":1,"status":"ok"}

说明：

mb: 010600000008，是十六进制 ascii 格式，比如 ff 就是 255。网关设备收到服务器的下发数据后，会将此处的 ascii 格式的数据转换为对应的十六进制数据，然后转发到串口上。

sn: 数据包的顺序，整数。网关设备收到服务器的下发数据中的 sn 值是多少，网关设备应答给服务器的数据包中的 sn 值就是多少。用户可用该参数来识别网关设备的应答数据与服务器请求数据的对应关系。

ack: 是否需要网关设备应答数据，0：不将应答数据上传到服务器；1：将应答数据上传到服务器。非必须，若没有则按 ack 等于 1 处理。

crc: “mb”字段中的数据是否包含 crc 校验。1：mb 字段不包含 crc 校验值，网关设备会自动计算 crc 值；0：mb 字段包含 crc 校验值，网关设备则不再会计算 crc 值。非必须，

若没有则按 **crc** 等于 0 处理。

uart: 指定往哪一路串口（PORT）上发，0~23 分别代表第 1~24 路串口（PORT）。非必须，若没有则按 **uart** 等于 0 处理，即向第一路串口上下发。若想向网络通道上发数据，则可以用 **net** 替代 **uart** 即可，如：{"mb":"010300000001","sn":1,"ack":0,"crc":0,"net":0}。

retmb: 010300000001，是十六进制 **ascii** 格式，比如 **ff** 就是 255。网关设备收到串口设备的应答数据后，会将串口设备应答的十六进制格式的数据转换为对应的 **ascii** 格式的数据，然后转发到服务器上。

status: timeout: 超时；ok: 正常

8. 设备恢复出厂

设备恢复出厂后，所有的参数会重新设置到出厂时默认的参数。设备恢复出厂的方法有两种：通过配置软件恢复出厂、通过硬件恢复出厂。

图 8.1 设备默认参数

	项目	默认参数
所有路串口参数	波特率	9600bps
	数据位	8
	校验位	NONE
	停止位	1
	串口工作模式	透传
	数据通道	U1~U8 与 N1~N8 数据通道一一对应 N9~N24 默认不启用
网络参数	网络工作模式	TCP SERVER
	设备 IP	192.168.1.253
	端口号	N1~N8 端口号分别是 1030~1037 N9~N24 默认不启用
	网页登录用户名及密码	用户名：admin 密码：admin

1. 通过配置软件恢复出厂

使用“智嵌物联 gateway 参数配置软件”来恢复出厂。具体步骤如图 8.2 所示。

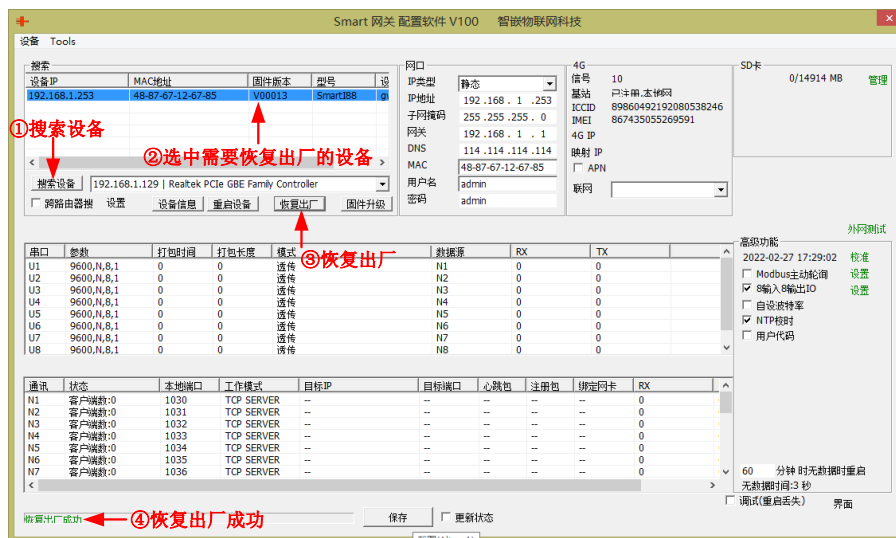


图 8.2 用配置软件恢复出厂步骤

2. 硬件恢复出厂

设备侧边上有用于复位及恢复出厂的按键，面板丝印“CFG”。

短按“CFG”按键，复位设备；

长按“CFG”按键，并保持 6 秒以上（NET 灯闪烁），实现恢复出厂设置；



图 8.3 设备恢复出厂设置

9. 固件升级

升级前，请先下载最新的固件，可在官网下载或直接向公司销售索要。

使用配置软件，通过设备的网口对本设备进行固件升级，具体步骤如图 9.1 所示。

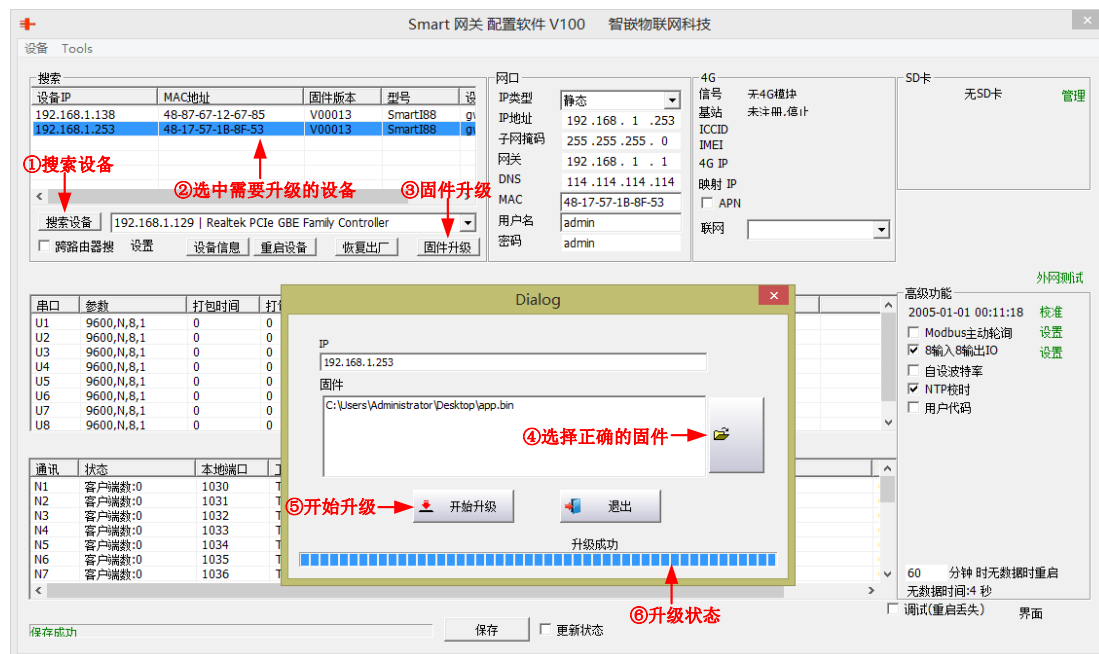


图 9.1 本地升级界面

① 固件升级前，请确保设备的 IP 地址与电脑 IP 地址在同一网段，且 IP 地址不冲突。

销售网络

智嵌物联，让连接更稳定！

企业愿景：成为国内物联网设备首选品牌！

企业使命：为客户利益而努力创新，为推动工业物联网发展而不懈奋斗！

产品理念：稳定！稳定！还是稳定！

服务理念：客户在哪里，我们就在哪里！

深圳总部

地址：广东省深圳市宝安区新桥街道新桥社区
新和大道 6-18 号 1203

网址：www.zhiqwl.com

电话：0755-23203231

北京办事处

地址：北京市房山城区德润街6号院8号楼3层
电话：18210365439

更多销售网络正在紧张筹备中……



淘宝店铺



京东店铺



微信公众号



公司官网