产品目录

版本信息

版本	时间	修改
1.0	2023 年 11 月 22 日	初始版本：CC201、CC101 产品手册
1.1	2023 年 12 月 01 日	修改附录 1：添加输入寄存器 0000，读取固件版本
1.2	2023 年 12 月 05 日	附录 1：添加输入计数器、读模组 IMEI、序列号 SN; 读物联网卡 IMSI、ICCID
1.3	2023 年 12 月 20 日	小修改，文本格式整理
1.4	2023 年 12 月 26 日	新增参考文档一节。把 Modbus 寄存器改为单独文档
1.5	2024 年 03 月 16 日	增加：思可安配置工具的支持；增加：支持定制开发；支持固件重新下载
1.6	2024 年 07 月 23 日	增加：修改主动指令支持 Modbus JSON 格式数据上报
1.7	2025 年 03 月 20 日	小修改
1.8	2025 年 03 月 21 日	增加 CC101-B 模块
1.9	2025 年 06 月 13 日	增加配置工具使用方法
2.0	2025 年 08 月 03 日	4.2 节：增加主动指令出错时的 JSON 格式数据

参考文档

[参考文档 1]	《Modbus 寄存器-物联网模块》	下载

1. 产品简介

1.1 应用场景

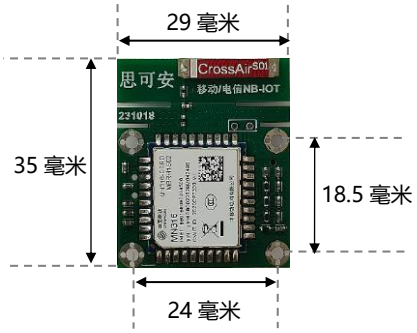
- 小尺寸、板载天线、NB-IOT 嵌入式物联网应用
- 设备状态上报、传感器数据上报、设备远程控制

1.2 型号选择

- CC201 模块: 串口 1：UART TTL，3.3 伏特；
  串口 2：RS485
- CC101-B 模块: 串口 1：UART TTL，3.3 伏特；

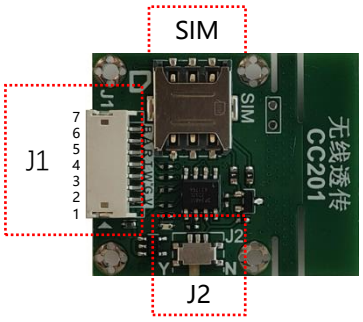
1.3 物理参数

- **封装与尺寸**
  - **板面尺寸**：35 x 29 (毫米)
  - **安装孔距**：24 x 18.5 (毫米)
  - **安装螺丝**：M2 螺丝/螺母

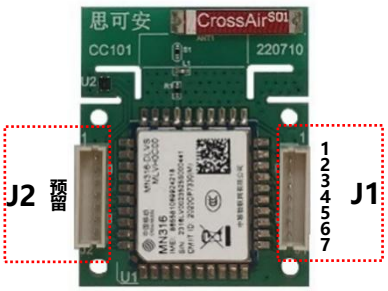


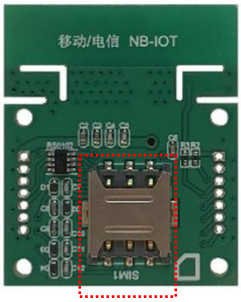
- **信号连接**
  - **CC201 信号连接**
    - SIM 卡座： Nano-SIM 物联网卡座；支持移动、电信 NB-IOT 物联网卡
    - J2 拨码开关： RS485 终端匹配电阻设置，Y 端：120Ω ； N 端：无
    - J1 接口信号线：7PIN，间距 1.5 毫米，从下到上：

脚号	编号	意义
1	V	电压输入，3.3~7.5 伏
2	G	接地
3	-	预留
4	T	串口 1：TXD 信号线
5	R	串口 1：RXD 信号线
6	A	串口 2：RS485，A 接线端
7	B	串口 2：RS485，B 接线端



➢ **CC101-B 信号连接**

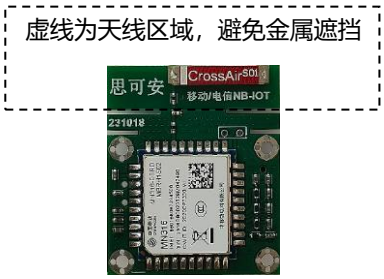




Nano SIM 卡座

J1 的脚号	编号	意义
1	V	电源输入，2.2 ~ 4.2 伏特
2	G	接地
3	-	预留
4	T	串口 1：TXD，3.3 伏特
5	R	串口 1：RXD，3.3 伏特
6	-	预留
7	-	预留

■ 板载天线



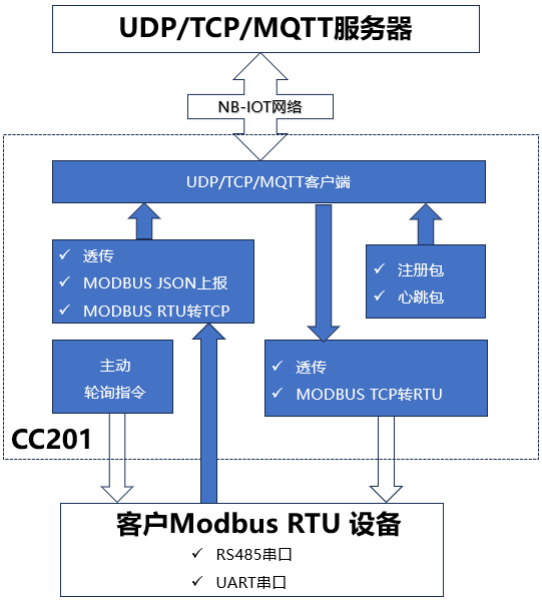
2. 产品功能

2.1 产品参数

产品型号	CC201: RS485 接口 CC101-B: UART TTL 串口
工作电压	CC201: 3.3 伏 ~ 7.5 伏，典型电压：3.6 伏 CC101-B: 2.2 伏 ~ 4.2 伏，典型电压：3.6 伏
工作温度	-25°~+75°
通信接口	CC201: 串口 1：UART TTL，3.3 伏，用作模块配置； 串口 2：RS485，过压/过流保护；防浪涌；带静电保护； CC101-B: 串口 1：UART TTL，3.3 伏；
硬件连线	■ J1 接口：7-PIN 底座，针距 1.5 毫米，参照信号连接一节

	■ SIM 卡座：移动/电信 Nano 物联网卡 ■ CC201：J2 开关：RS485 终端匹配电阻设置：Y 端：120Ω；N 端：无
天线	内置天线，无需外接天线
频段	B5/B8，移动/电信 NB-IOT
网络协议	UDP、TCP、MQTT
Cat NB2 速率	■ 126.0kbps，下行 ■ 158.5kbps，上行
软件功能	■ Modbus 寄存器：[参考文档 1] ■ 设备号：可配置；缺省为 8 位数字：93312190 ■ 注册包、心跳包机制：可自定义 ■ 数据传输：可实现外接设备和服务器双向透传 ■ 主动指令模式：支持最多 6 路独立主动指令 ■ Modbus JSON 格式上报 ■ Modbus RTU/TCP 互转
交货标准	■ 1 个：CC201 模块，或者 CC101-B 模块 ■ 1 条：7Pin 接口线，针间距 1.5 毫米
注意事项	如果输入电压不稳定，需要外加过压、过流保护；防进水
质保期	6 个月
使用寿命	8 年
使用环境	NB-IOT 网络信号
应用场景	小尺寸、NB-IOT 嵌入式物联网应用
资料下载	http://www.cyckn.com/download

2.2 功能框图



2.3 产品功能

- 串口参数
  - 串口的协议配置：
    - ✓ 工作模式：配置模式、透传模式、主动指令模式
    - ✓ 波特率：2400 ~ 115200
    - ✓ 校验位：无、奇、偶
    - ✓ 停止位：1 位、1.5 位、2 位
- 串口的工作模式
  - 配置模式：
    - ✓ 串口 1 支持该模式
    - ✓ 设备为 Modbus RTU 从机，通过计算机串口发送配置指令
    - ✓ 对串口 1 和串口 2 进行工作模式及参数配置
  - 透传模式：
    - ✓ CC101-B 的串口 1、CC201 的串口 2 支持该模式
    - ✓ 设备和服务器之间的双向传输数据
  - 主动指令模式：
    - ✓ CC101-B 的串口 1、CC201 的串口 2 支持该模式
    - ✓ 最多可独立设置 6 条指令，定时通过串口发送到外接串口设备
    - ✓ 打包从串口设备返回的数据，并传输到服务器
- 设备到服务器的数据格式
  - 透传模式：
    - ✓ 格式 1：透明传输设备到服务器的数据，缺省格式
    - ✓ 格式 2：接收设备 Modbus RTU 数据，转成 Modbus TCP 格式，并传输到服务器
  - 主动指令模式：
    - ✓ 格式 1：透明传输设备到服务器的数据，缺省格式
    - ✓ 格式 2：对于主动指令的响应数据以 JSON 格式上报
  - 设备到服务器的数据头可选以下数据：
    - □ 时间戳：UTC 格式，占 5 个字节，例如 1698662520 表示时间为 2023-10-30 18:42:00
    - □ 设备号：为 8 位 10 进制数，如 93312190
    - □ 固件版本：为两个字节，例如 C203：C2 表示模块 CC201；03 为固件版本
- 服务器到设备的数据格式
    - ✓ 格式 1：把从服务器接收到的数据，通过串口透明发送到外接串口设备
    - ✓ 格式 2：把服务器返回数据从 Modbus TCP 转换成 Modbus RTU，通过串口发送
    - ✓ 格式 3：把服务器返回数据从 Modbus JSON 转换成 Modbus RTU，通过串口发送
- 设备网络传输协议（三选一）：
  - TCP 协议(缺省)
  - UDP 协议

- MQTT 协议

- 注册包：开机时将设定的注册包发送到服务器
  - 注册包的内容可以设置
- 心跳包

  网络心跳包：定时发送心跳包到网络服务器：
  - 心跳包的上报周期可以设置，缺省为 15 分钟
  - 心跳包内容选择为自定义，最长为 30 字节
  - 服务器响应数据的处理可配置：
    - ✓ 不做处理(缺省)
    - ✓ 透明发送到串口
    - ✓ Modbus TCP 转为 Modbus RTU，并发送到串口
    - ✓ JSON 指令转为 Modbus RTU，并发送到串口
- Modbus 寄存器及指令

  本产品支持 Modbus 协议，参照文档 [参考文档 1]
- 本产品支持定制开发，如果有需要的客户，请联系客服。

3. 配置工具

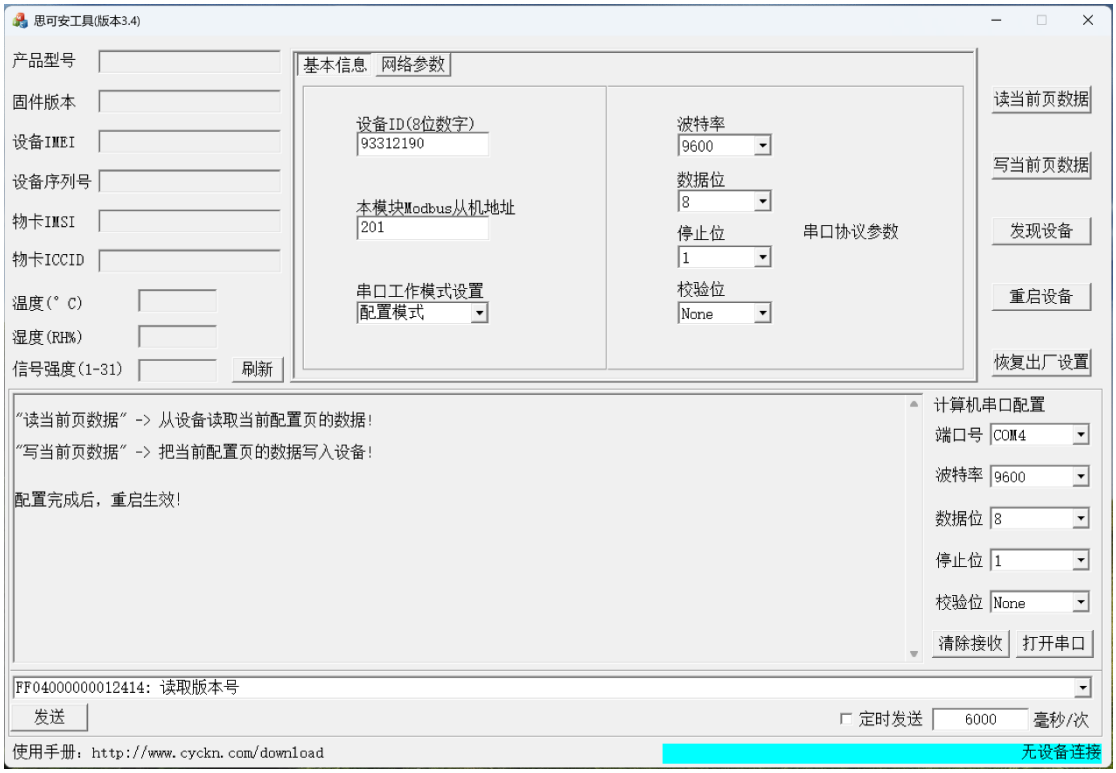
提供可视化的软件工具，可以在 http://www.cyckn.com/download 下载最新配置工具软件。

3.1 硬件连接

- CC201 模块与配置工具的硬件连接

- CC101-B 模块与配置工具的硬件连接

3.2 配置界面



■ ① 计算机串口的协议配置以及模块操作

- 选择串口参数：波特率、数据位数、停止位、校验位；缺省为 9600-8-1-N
- 打开/关闭串口：打开或关闭计算机串口
- 清除发送：删除发送区的数据

■ ② 设备配置操作

- 读当前页数据：从设备读取当前配置页的数据
- 写当前页数据：把当前配置页的数据写入设备
- 发现设备： 搜寻连接设备的型号
- 重启设备： 重启模块
- 恢复出厂设置

■ ③ 手动发送数据到模块

- 输入发送数据：可以从下拉框选择预设指令或手动输入数据指令，为 16 进制
- 定时发送功能：设置发送数据和发送周期，定时发送
- 发送：发送输入的数据

■ ④ 模块信息显示

- 产品型号：CC201 或 CC101
- 固件版本
- 设备 IMEI
- 设备序列号 SN
- 物卡 IMSI
- 物卡 ICCID
- 温度(℃)： 仅适应 CC101-C
- 湿度(RH%)： 仅适应 CC101-C

- 信号强度(1-31)，点击“刷新”按钮可读取最新数据

■ ⑤ 模块数据配置页面，根据不同的工作模式，显示对应配置的页面

- 基本信息：适合所有工作模式
- 网络参数：适合所有工作模式
- MQTT 参数：当网络传输协议为 MQTT 时，该配置页面显示

■ ⑥ 数据交互区

- 显示配置工具和模块的交互数据

3.3 基本信息配置

- 设备 ID：8 位 10 进制数，缺省为 93312190
- 本模块 Modbus 从机地址：从 1 到 247，为 10 进制数
- 串口协议参数设置
  - 波特率：可选 2400，4800，9600，19200，38400，57600，115200
  - 数据位：8
  - 停止位：1
  - 校验位：None

3.4 网络参数配置

- 网络传输协议：TCP，UDP，MQTT；选 MQTT 时，在 MQTT 参数页面设置参数
- 服务器 IP 地址及端口号
- 网络注册包设置：每次上电开机时发送。注册包内容设置：
  - 自定义：可以在网络注册内容中设置
  - 设备 ID：在基本信息页面设置，缺省为 93312190

- 网络心跳包周期：单位秒
- 网络心跳包响应处理：服务器收到心跳包后下发数据，模块的处理方式支持：
  - 不做处理：模块不做任何处理
  - 透明转发到串口：发送服务器下发数据到模块
  - Modbus TCP 转 RTU：服务器下发 Modbus TCP 包，模块转成 Modbus RTU 数据，并发送到模块
  - Modbus JSON 转 RTU：服务器下发 Modbus JSON 数据，模块转成 Modbus RTU 数据，并发送到模块。JSON 数据的配置在“JSON 设备”和“JSON 模板”配置页面设置。JSON 数据格式示例，请参考服务器 JSON 指令转 Modbus RTU。
- 网络心跳包内容：16 进制数据，最多 30 个字节

3.5 MQTT 参数配置

- 客户端 ID
- 用户名
- 用户密码
- 发布主题
- 订阅主题
- 保活时间（单位秒）
- 清理会话

3.6 透传模式参数配置

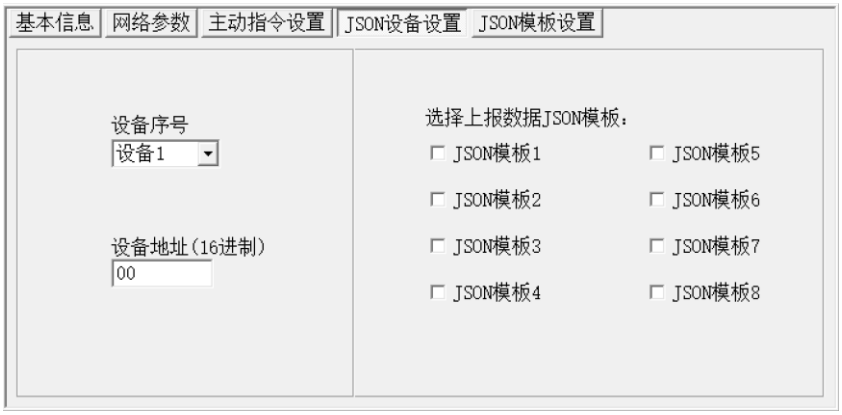
- 设备上报数据格式
  - 数据透传：把模块串口输入的数据透明传输到服务器
  - Modbus RTU 转 TCP：串口 Modbus RTU 数据转为 Modbus TCP 格式，并上报服务器
- 服务器下发数据处理
  - 不做处理：模块不做任何处理
  - 透明转发到串口：发送服务器下发数据到“被配置的串口”外接的设备
  - Modbus TCP 转 RTU：服务器下发 Modbus TCP 包，模块转成 Modbus RTU 数据，并发送“被配置的串口” 外接的设备
  - Modbus JSON 转 RTU：服务器下发 Modbus JSON 数据，模块转成 Modbus RTU 数据，并发送“被配置的串口”。JSON 数据的配置在“JSON 设备”和“JSON 模板”配置页面设置。JSON 数据格式示例，请参考服务器 JSON 指令转 Modbus RTU。
- 设备上报数据头添加：
  - 时间戳： 10 位 10 进制数字的 UTC 时间戳信息
  - 设备号： 8 为 10 进制数的设备号，在“基本信息”页面配置
  - 固件版本：2 字节固件版本

3.7 主动指令模式配置

- 指令序号：最多可设定 6 条主动指令，分别定时发送到外接设备
- 主动指令内容：16 进制
- 自动生成：对于 Modbus 指令，可通过选择不同参数，自动生成主动指令
  - 设备地址：16 进制，为 Modbus 地址，从 1 到 247。如果“上报数据格式”或“下发数据处理”为 JSON 数据，设备地址必须在“JSON 设备”和“JSON 模板”设置对应的 JSON 模板。
  - 功能码：01-读取线圈输出；02-读取状态输入；03-读保持寄存器；04-读输入寄存器
  - 寄存器地址：16 进制，该 Modbus 指令的操作起始地址
  - 寄存器数： 10 进制，该 Modbus 指令读取的寄存器数量
- 上报数据格式
  - 数据透传：透传外接设备返回数据到服务器
  - JSON 格式 1：参照上报数据 JSON 数据格式
  - JSON 格式 2：参照上报数据 JSON 数据格式
- 上报数据包含：可选择
  - 时间戳：10 位 10 进制数字的 UTC 时间戳信息
  - 设备号：8 为 10 进制数的设备号，在“基本信息”页面配置
  - 固件版本：2 字节固件版本

- 下发数据处理：
  - 不做处理：模块不做任何处理
  - 透明转发到串口：发送服务器下发数据到“被配置的串口”
  - Modbus TCP 转 RTU：服务器下发 Modbus TCP 包，模块转成 Modbus RTU 数据，并发送“被配置的串口”
  - Modbus JSON 转 RTU：服务器下发 Modbus JSON 数据，模块转成 Modbus RTU 数据，并发送“被配置的串口”。JSON 数据的配置在“JSON 设备”和“JSON 模板”配置页面设置。JSON 数据格式示例，参考服务器 JSON 指令转 Modbus RTU。
- 发送周期：单位秒，每个指令可独立设置，参照主动指令的周期和计时偏移设置
- 计时延时：单位秒，延时时间后开始计时第一个发送周期

3.8 JSON 设备配置



- 设备序号：最多可配置 8 个外接 Modbus 设备
- 设备地址：16 进制，取值 1 到 247
- 选择上报数据 JSON 模板：可选对应的 JSON 模板，在“JSON 模板设置”页面配置

3.9 JSON 模板配置

- 键值输入：最长 16 个字符
- 寄存器地址：对应的寄存器地址
- 数据顺序：ABCD 为字节顺序，A、B、C、D 表示 4 bits16 进制数，其表示的数值为：
  - ABCD：A 左移 24 位 + B 左移 16 位 + C 左移 8 位 + D 左移 0 位
  - BADC：A 左移 16 位 + B 左移 24 位 + C 左移 0 位 + D 左移 8 位
  - CDAB：A 左移 8 位 + B 左移 0 位 + C 左移 24 位 + D 左移 16 位

- DCBA：A 左移 0 位 + B 左移 8 位 + C 左移 16 位 + D 左移 24 位

➢ 数据类型：

- = 0: int16， 2 字节；
- = 1: uint16， 2 字节；
- = 2: int32， 4 字节；
- = 3: uint32， 4 字节；
- = 4: bool， 2 字节；
- = 5: float16，2 字节；
- = 6: float32，4 字节；

➢ 小数点数：当数据为 float16, float32 时，数据的小数点位数，从 0~3

➢ 结果比例 A 和结果偏移 B：Y=A*X+B，如果结果比例为 a，结果偏移为 b，设备返回结果为 X，则最终结果为 Y = a·X + b。

➢ 举例如下：数据为 float16，小数点 2 位，设备返回的寄存器数据为 0x1234，则 X = (0x1234/100) = 46.60；如果 a=1.5, b=2.86，最终结果 Y = 1.5·46.60 + 2.86 = 72.76

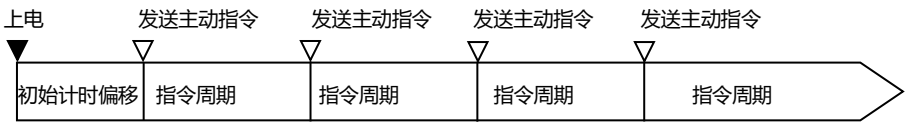
➢ 上报下限(TL)和上报上限(TH)：假设阈值上限为 ThresH，阈值下限为 ThresL，如果最终结果满足 Y <= ThresL 或 Y > ThresH，则上报；否则不上报。

4. 配置详解

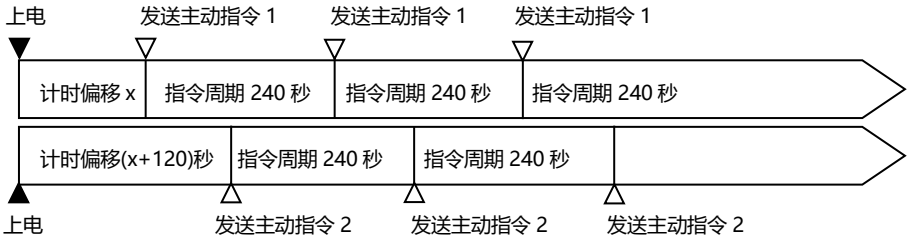
4.1 主动指令的周期和计时偏移设置

最多可设置 6 条主动指令，每条指令的指令周期和指令周期初始计时偏移都可单独设置。

■ **指令周期**以及**指令周期初始计时偏移**，单位秒



■ 举例：2 个主动指令等时轮询发送，间隔为 120 秒：
指令 1 周期为 240 秒，计时偏移为 x; 指令 2 周期为 240 秒，计时偏移为(x+120)：



4.2 主动指令的 JSON 格式上报数据

主动指令的上报数据格式设为“JSON 格式 1”或“JSON 格式 2”时的 JSON 数据格式：

■ **JSON 格式 1 示例:**

{"devid": 9331290,"fm": "C1.2","time": 1638865162, "a": 23.5,"b": 46.8}

- “devid”： 设备号，为数字，不加引号。在“主动指令模式”页面配置
- “fm”：固件版本号，可选，在“主动指令模式”页面配置
- “time”：4 字节时间戳，可选，在“主动指令模式”页面配置
- “a”，“b”：对应的外接设备返回数据，为数字，在“JSON 模板”中设置

■ **JSON 格式 2 示例**:

{"devid": 9331290,"fm": "C1.2","time": 1638865162,"data": {"a": 23.5,"b": 46.8}}

- “devid”： 设备号，为数字，不加引号，在“主动指令模式”页面配置
- “fm”：固件版本号，可选，在“主动指令模式”页面配置
- “time”：5 字节时间戳，可选，在“主动指令模式”页面配置
- “data”:{“a”，“b”}: 对应的外接设备返回数据，在“JSON 模板”中设置

■ **主动指令出错时的 JSON 格式数据示例:**

{"devid": 9331290,"fm": "C1.2","time": 1638865162,"err": xxx}

- “devid”： 设备号，为数字，不加引号，在“主动指令模式”页面配置
- “fm”：固件版本号，可选，在“主动指令模式”页面配置
- “time”：5 字节时间戳，可选，在“主动指令模式”页面配置
- “err”： xxx 为 3 位数字的错误码

err 代码	意义
401	数据格式错误
403	数据类型或设备地址设置错误
405	数据长度错误
409	功能码错误
411	没有配置对应键值
412	外接设备超时错误
其它值	未知错误

4.3 服务器 JSON 指令转 Modbus RTU

■ **服务器 JSON 指令格式**

```
{
    "type":   "coil",
    "devid":  9331290,
    "data": {
                "a": 1,
                "b": 0
            }
}
```

- "type" ： **必需项**

  寄存器类型

  = "coil": 对应功能码 01(读取线圈状态)、05(写单个线圈状态) 、15(写多个线圈状态)

  = "di": 对应功能码 02(读取离散输入状态)

  = "hold": 对应功能码 03(读取保持寄存器)、06(写单个保持寄存器)、16(写多个保持寄存器)

  = "input": 对应功能码 04(读取输入寄存器)

- "devid": **必需项**
  设备 ID 号, 8 位数字，不加引号。跟设备中设置的设备号一致，必须项
- "data": **必需项**
  寄存器对应的键值，以及设置值。键值跟模块中的设置保持一致。如果设置值为"?"，表示读取；否则为写入。譬如 "a":"?"表示读取"a"对应的寄存器值；"a":1 表示把 1(即打开开关)值写入寄存器。

■ **设备响应 JSON 数据**

- JSON 指令错误：{"devid":93312190,"type":"coil", "error":1}
  - “devid”：设备 ID 号, 8 位数字，不加引号。跟 JSON 指令中的设备号一致
  - “type”：寄存器类型，跟 JSON 指令中的寄存器类型一致
  - “error”：错误代码，0x01=JSON 指令数据格式错误
    0x02=JSON 指令中的设备号错误
    0x03=JSON 指令中的寄存器类型错误
    0x04=JSON 指令中的数据错误
- 执行成功：{"devid":93312190,"type":"coil", "data":{"a":"ok"}}
- 执行失败：{"devid":93312190,"type":"coil", "data":{"a":"nok"}}

■ **服务器 JSON 指令示例**

- **示例 1**：把线圈"a"对应的线圈写入 1，即打开开关

服务器下发 JSON 指令：

```
{
    "type":  "coil",
    "devid": 9331290,
    "data": {
              "a": 1
            }
}
```

模块响应数据：

```
{
    "type":  "coil",
    "devid": 9331290,
    "data": {
              "a": "OK"  // OK：成功；NOK：写入失败
            }
}
```

- **示例 2**：读取线圈"a"对应的状态

服务器下发 JSON 指令：

```
{
    "type":  "coil",
    "devid": 9331290,
    "data": {
              "a": "?"
            }
}
```

模块响应数据：

```
{
    "type":  "coil",
    "devid": "9331290",
    "data": {
              "a": 1
            }
}
```

➢ **示例 3**：读输入寄存器"temp"对应的温度值，"humi"对应的湿度值

服务器下发 JSON 指令：

```
{
    "type":  "input",
    "devid": 9331290,
    "data": {
              "temp": "?",
              "humi": "?"
            }
}
```

模块响应数据：

```
{
    "type":  "input",
    "devid": 9331290,
    "data": {
              "temp": 23.5,
              "humi": 53.8
            }
}
```

联系方式

客服电话： 13651106881 (同微信号)

淘宝店销售：思可安

微信公众号：思可安

公司网站： www.cyckn.com

产品资料： www.cyckn.com/download