



多功能数据采集器

UC50x-LTE Cat. 1

用户手册



安全须知

- ❖ 为保护产品并确保安全操作，请遵守本使用手册。如果产品使用不当或者不按手册要求使用，本公司概不负责。
- ❖ 严禁改装本产品。
- ❖ 旋下接口防护帽时请勿用力将接口一起旋动。
- ❖ 请勿将产品安装在不符合工作温度、湿度等条件的环境中使用，远离冷源、热源和明火。
- ❖ 请勿将产品安装在强振动、强磁场环境下。
- ❖ 使用拨码开关时注意断电或将电池接线拆下，保证主板不带电。
- ❖ 请在产品关机情况下完成与其它终端设备的接线，户外使用请做好防水处理。
- ❖ 为了您的设备安全，请及时修改设备默认密码（123456）。

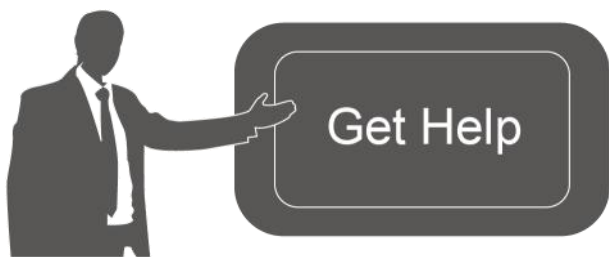
产品符合性声明

UC50x 系列符合 CE, FCC 和 RoHS 的基本要求和和其他相关规定。



版权所有© 2011-2023 星纵物联

保留所有权利。



如需帮助，请联系

星纵物联技术支持：

邮箱：contact@milesight.com

电话：0592-5023060

传真：0592-5023065

地址：厦门市集美区软件园三期 C09 栋

文档修订记录

日期	版本	描述
2024.3.21	V1.0	第一版

目录

一、产品简介	6
1.1 产品介绍	6
1.2 产品亮点	6
二、产品结构	7
2.1 包装清单	7
2.2 外部结构和接口说明	7
2.3 内部结构和接口说明	8
2.4 SIM 卡安装	10
2.5 产品尺寸 (mm)	10
三、硬件切换	10
四、产品配置	11
4.1 配置方式	11
4.1.1 NFC 配置	11
4.2 常用设置	12
4.3 蜂窝设置	13
4.3 应用模式设置	13
4.4 AI 设置	15
4.5 串口设置	17
4.5.1 RS485	17
4.5.2 RS232	23
4.6 GPIO 设置	25

4.7 规则引擎	27
4.8 SDI-12 设置	28
4.9 数据存储	32
4.9.1 数据重传	33
4.10 维护	33
4.10.1 升级	33
4.10.2 备份	34
4.10.3 重置	35
五、产品安装	35
5.1 壁挂式安装	35
5.2 抱杆式安装	36
六、通信协议	37
6.1 上行数据	37
6.1.1 基础数据包	38
6.1.2 阈值包	42
6.1.3 突变包	45
6.1.4 异常值包	48
6.1.5 回传、重传数据	49
6.2 控制命令下发	49
6.2.1 设置采集周期	49
6.2.2 设置上报周期	49
6.2.3 重启	49

6.2.4 设置时区	50
6.2.5 立即上报	50
6.2.6 清除历史记录	50
6.2.7 历史记录存储使能	50
6.2.8 历史记录重传使能	50
6.2.9 历史记录重传周期	50
6.2.10 回传一个时间点数据	51
6.2.11 回传一段时间内的历史数据	51
6.2.12 下发停止查询回传	51
6.2.13 请求时间同步	51
6.2.14 设置 GPIO1 的 DO 模式	51
6.2.15 设置 GPIO2 的 DO 模式	51
6.2.16 下发配置 MODBUS 通道	51

一、产品简介

1.1 产品介绍

星纵物联 UC50x 系列是一款专为传统传感器数据转换而设计的多功能数据采集器。UC50x 系列集成了 GPIO、串口、模拟输入、SDI-12 等多种类型的通信接口，能够对接多种类型的传统传感器。支持 MQTT/TCP/UDP 等多种协议对接主流物联网平台以及星纵云，轻松实现对现场设备的远程数据采集和控制。

UC50x 系列提供太阳能供电或大容量电池供电两种版本，UC50x 系列采用 IP67 防尘防水外壳和 M12 防水航空接头设计，可广泛应用于智慧农业、智慧工业等多种户外场景。

1.2 产品亮点

- 数据采集接口丰富：集成 GPIO、RS232、RS485、模拟量输入、SDI-12 等多种通信接口，支持对接多种传感器
- 对外供电接口丰富：设备支持给外部传感器供电 ($2 \times 5/9/12V + 2 \times 3.3V$)，适用不易取电的场景
- 防护等级高：防护等级高达 IP67，接口采用 M12 防水航空接头，适用各种恶劣环境
- 供电方式多样：功耗低，可选大容量电池供电或太阳能供电，同时支持 DC 供电或外接大功率太阳能板，满足不同场景供电需求
- 数据完整性：具备本地存储功能，可存储高达 10000 条传感器数据，且支持断网数据重传与数据回传功能，确保信息可追溯，避免数据丢失
- 规则引擎告警：通过添加本地规则引擎，实现数据突变告警和超阈值告警功能
- 电池高/低温保护：设备在高/低温环境下自行限制充电电流，保护电池免受损伤
- 简单易用：支持手机 NFC 快速配置
- 兼容性好：支持通过 MQTT/TCP/UDP 对接到第三方平台
- 管理一体化：支持对接 AWS 等主流物联网平台以及星纵云平台 (开发中)

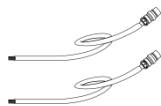
二、产品结构

2.1 包装清单

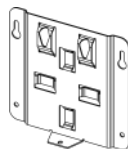
使用前请检查产品包装盒内是否包含以下物品。



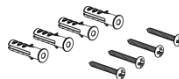
1 × UC50x 设备



2 × M12 航空插头数据线 (30cm)



1 × 安装板



壁挂套件



2 × 柱装抱箍



1 × 固定螺丝



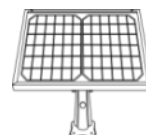
1 × 快速手册



1 × 合格证&保修卡



1 × SIM卡



1 × 太阳能板
(仅 UC501 可选)

1 × SIM 卡
(蜂窝版本,
已装入卡槽)



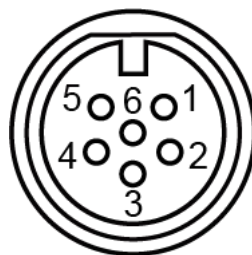
如果上述物品存在损坏或遗失的情况，请及时联系您的代理或销售代表。

2.2 外部结构和接口说明

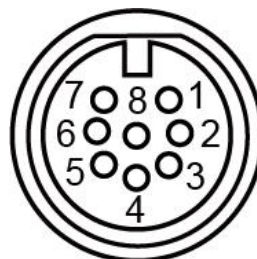


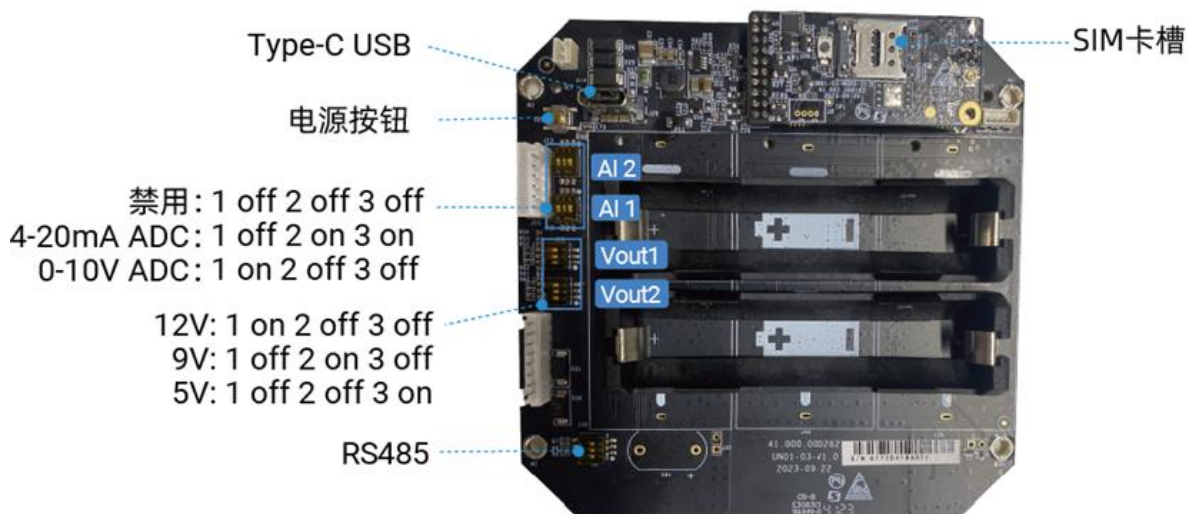
数据接口 1:

引脚	描述	颜色
1	5V/9V/12V OUT (可切换)	橘色
2	3.3V OUT	红色
3	GND	黑色
4	模拟量输入 1	黄色
5	模拟量输入 2	绿色
6	5-24V DC	白色

**数据接口 2:**

引脚	描述	颜色	
1	5V/9V/12V (可切换)	橘色	
2	3.3V	红色	
3	GND	黑色	
4	GPIO1	蓝色	
5	GPIO2	棕色	
6	RS232/RS485 (可切换)	绿色	
7		黄色	
8	SDI-12	紫色	
引脚	RS232	RS485	
6	TXD	A	
7	RXD	B	

**2.3 内部结构和接口说明**



拨码开关:

拨码类型	拨码设置
电源输出	12V: 1 on 2 off 3 off 9V: 1 off 2 on 3 off 5V: 1 off 2 off 3 on
模拟量输入	不启用: 1 off 2 off 3 off 4-20mA ADC: 1 off 2 on 3 on 0-10V ADC: 1 on 2 off 3 off
RS485	A 和 B 之间增加 120 欧电阻: 1 on 2 off 3 off A 上拉一个 1k 电阻: 1 off 2 on 3 off B 下拉一个 1k 电阻: 1 off 2 off 3 on

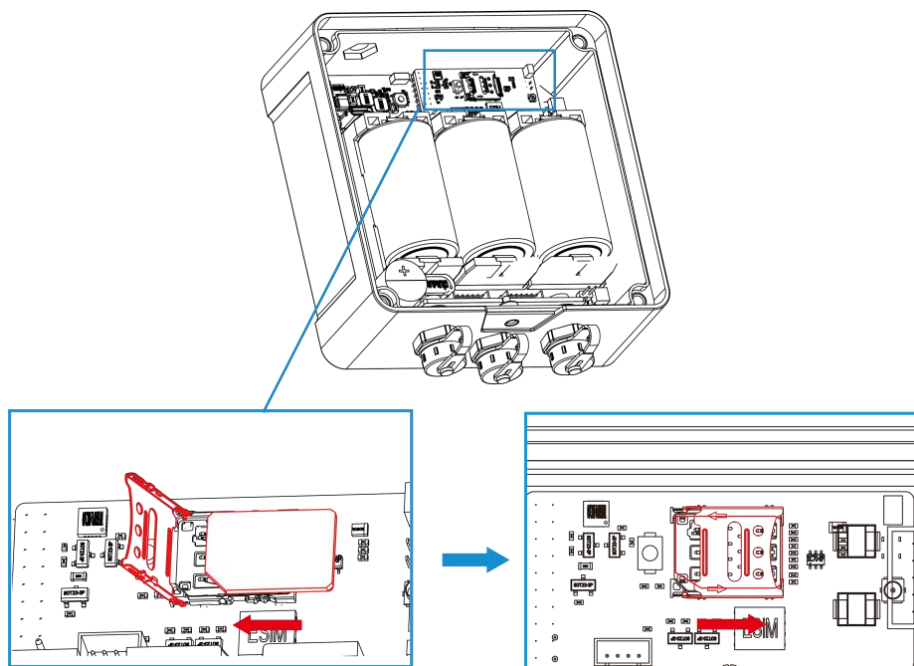
注意:

- (1) 电源输出默认为 12V，模拟量输入默认为 4-20mA，RS485 电阻默认全禁用。
- (2) 接口 1 的电源输出用于给模拟量输入设备供电，接口 2 的电源输出用于给 RS485/SDI-12 终端设备供电。
- (3) 切换拨码开关时请确保设备处于关机或断电状态。

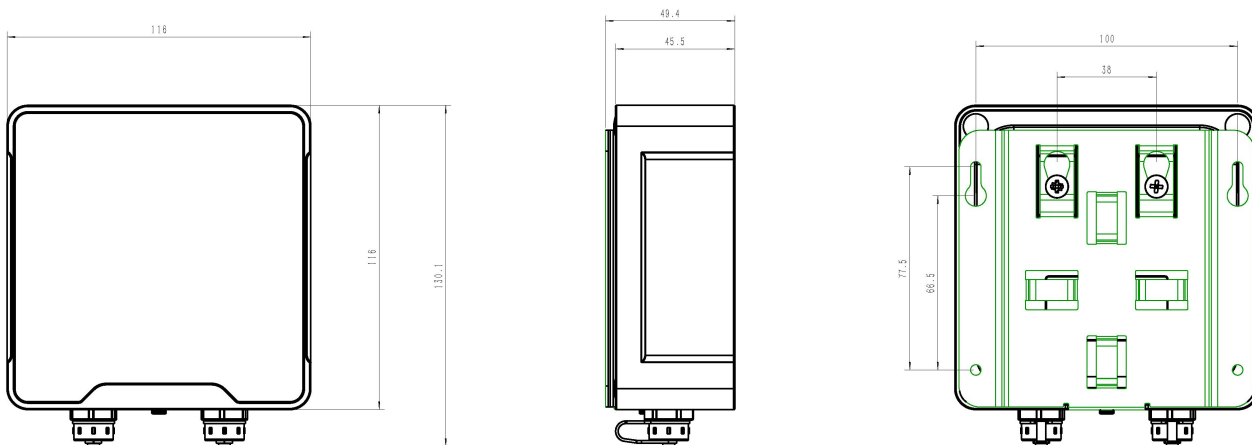
电源按钮:

功能	操作和指示灯状态
开机	长按电源按钮超过 3 秒。LED: 灭 → 亮
关机	长按电源按钮超过 3 秒。LED: 亮 → 灭
恢复出厂设置	长按电源按钮超过 10 秒。LED: 快速闪烁
确认开关状态	快速按一下电源按钮，如果亮则说明设备开启。

2.4 SIM 卡安装



2.5 产品尺寸 (mm)

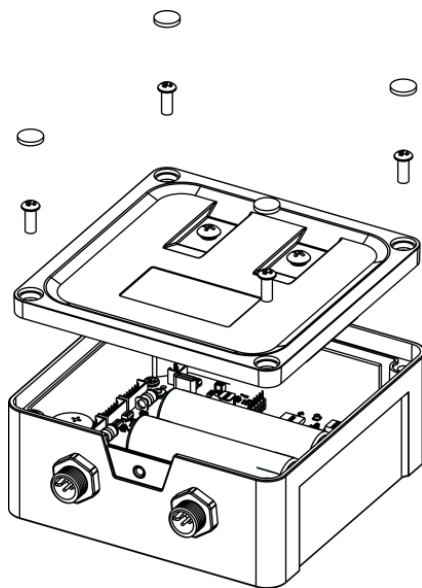


三、硬件切换

可根据需求切换硬件接口的工作模式。切换步骤如下：

1. 拆下设备背面的 4 个螺帽和螺丝，取下防水外壳。
2. 参考 2.3 的内部接口说明拨动拨码开关，切换所需的模拟量输入模式和电压输出模式。
3. 完成配置后盖上外壳，拧紧螺丝。

注意：硬件切换时请保证设备关机。



四、产品配置

4.1 配置方式

UC50x 系列支持 NFC 配置。配置前请确认已完成所有的硬件配置以及与采集终端的接线。

4.1.1 NFC 配置

配置准备：

- 手机（支持 NFC）

配置步骤：

1. 开启手机 NFC 功能后打开 Milesight ToolBox App；
2. 将手机的 NFC 区域紧贴在产品正面的 NFC 感应区几秒不动，即可获取产品的基本信息；
3. 在 App 上设置后紧贴产品的 NFC 感应区即可完成配置。第一次使用手机为设备配置时需要输入密码进行验证，默认密码：123456。



注意：

- (1) 不同安卓手机的 NFC 区域不同，大致位于背部摄像头周围，具体请查询手机说明书或咨询相关客服。
- (2) NFC 读写失败后，请将手机暂时远离设备再贴到设备上尝试。

4.2 常用设置

打开 ToolBox App 的“设置->常用设置”菜单设置设备的数据上报周期。

参数	说明
上报间隔	设备上报数据的时间间隔，默认值为 360 秒，可配置 1-64800 秒。 注意： RS232 数据上报不遵循此设置。
采集间隔	默认禁用（未启规则引擎前，采集间隔不生效），设置阈值告警规则后自动启用，采集间隔 \leq 上报间隔。
累计次数	填入累计次数后，将按照所设置的周期上报时间进行存储。 例如：周期上报时间为 360min，累计次数为 2 次。则需要将 2 个 360min 的周期包内容存下，到 720min 时将两个周期包内容一起上报。
工作模式	可选：低功耗模式或低延时模式。 低功耗模式： 默认的工作模式。设备在发送上行数据后会关闭蜂窝模块以节省电力。只有当设备发送上行数据时，才能接收下行命令。

	<p>低延时模式：设备会定期打开接收窗口以接收下行命令。打开接收窗口的间隔由移动通信运营商决定。该模式将消耗更多的电力并缩短电池寿命。</p> <p>注意：UC502 是电池设备，若不用下行功能，推荐优先使用低功耗模式。</p>
数据存储	是否启用本地数据存储功能，参考 4.5.1 章节。
数据重传	是否启用数据重传功能，启用后设备在断网后会记录断网时间点，待设备重新联网后重传断网时间点与联网时间点之间的存储数据包。参考 4.5.2 章节。
修改密码	修改设备登录密码。

4.3 蜂窝设置

打开 ToolBox App 的“设置->“蜂窝设置”菜单设置 sim 卡。

参数	说明
接入点	输入由本地互联网运营商提供的网络拨号连接的接入点。 默认为空。
认证方式	默认 None。 可选：NONE，PAP 或 CHAP。 选择 None 时隐藏用户名，密码。 选择 PAP 或 CHAP ，用户名&密码：输入由本地互联网服务运营商提供的网络拨号连接的用户名&密码。
PIN 码	输入用于解锁 SIM 卡的 PIN 代码，4-8 位数字。

4.3 应用模式设置

状态
设置
维护

应用模式设置 ^

应用模式
MQTT

保活间隔(s)
300

服务器地址

端口
1883

客户端ID
6772D41157200014

用户凭证

TLS

MQTT主题

上行主题 ①

下行主题

串口透传上行主题 ①

串口透传下行主题

SDI-12透传上行主题 ①

SDI-12透传下行主题

参数	说明
应用模式	可选：MQTT、AWS、TCP 或 UDP 模式。
保活间隔 (s)	连接成功后，设备将按照设置的保活间隔发送保活包，以保证与服务器的连接状态。默认 600s，可更改为 10~65535s。——仅在 低延时模式 下有此配置。
服务器地址	输入选择应用模式的服务器地址。 注意： TCP 和 UDP 可以填写两组服务器地址，但设备耗电量将上升。

	端口	填写 MQTT 服务器端口以接收数据。
	客户端 ID	默认为设备 SN，可支持用户更改。
用户凭证	用户名	连接到 MQTT 服务器的用户名。
	密码	连接到 MQTT 服务器密码。
TLS	根文件	导入 TLS 自签名证书相关的根文件证书。
	客户端证书	导入自签名证书相关的客户端证书文件。
	客户端密钥	导入 TLS 自签名证书相关的客户端密钥文件。
MQTT	上行	支持自定义上行主题以接收周期性数据。 默认：uc/[sn]/uplink 用于上行数据包。 注意： 上行主题不能包含 “+” 和 “#” 。
	下行	支持自定义下行主题以接收周期性数据。 默认：uc/[sn]/downlink 用于下行数据包。
	串口透传上下行	当启用 RS232 或 RS485 透明模式时，可以自定义上行或下行主题。 默认：uc/[sn]/uplink/passthrough/serial 用于串口上行透传。 uc/[sn]/downlink/passthrough/serial 用于串口下行透传。 注意： 上行主题不能包含 “+” 和 “#” 。
	SID-12 透传上下行	当启用 SDI-12 透明模式时，可以自定义上行或下行主题。 默认：uc/[sn]/uplink/passthrough/SDI-12 用于 SID-12 上行透传。 uc/[sn]/uplink/passthrough/SDI-12 用于 SID-12 下行透传。 注意： 上行主题不能包含 “+” 和 “#” 。
	QoS	<p>QoS 0 - 仅一次</p> <p>这是最快的传输方法，只需要 1 条消息。它也是最不可靠的传输模式。</p> <p>QoS 1 - 至少一次</p> <p>这个级别保证消息至少会被传递一次，但可能会被传递多次。</p> <p>QoS 2 - 确保一次</p> <p>QoS 2 是 MQTT 中最高级别的服务。该级别确保每条消息只被预期的接收者接收一次。QoS 2 是最安全和最慢的服务质量级别。</p>

4.4 AI 设置

采集配置：

1. 打开 ToolBox App 的 “设置->AI 设置” 菜单启用模拟量输入接口。
2. 根据终端设备类型选择 UC50x 设备的模拟量输入类型。**注意：**模拟量输入默认为 4-20mA。若需要使用 0-10V 采集，软硬件需要同步切换，硬件切换请参考[三、硬件切换](#)。

3. 如使用 ToolBox App, 请先点击“采集”后将手机紧贴设备完成数据采集; 然后点击“读取”将手机紧贴设备获取数据。

The screenshot shows the configuration interface for two analog inputs in the ToolBox App. Both inputs are enabled (indicated by green toggle switches). The configuration for each input is as follows:

- 模拟量输入1 (Analog Input 1):**
 - 输入信号类型 (Input Signal Type): 4-20mA
 - Osh: 20
 - Osl: 4
 - 单位 (Unit): mA
 - 状态 (Status): [Empty field]
 - 读取 (Read) button: [Blue button]
- 模拟量输入2 (Analog Input 2):**
 - 输入信号类型 (Input Signal Type): 4-20mA
 - Osh: 20
 - Osl: 4
 - 单位 (Unit): mA
 - 状态 (Status): [Empty field]
 - 读取 (Read) button: [Blue button]

参数	说明
启用	启用/禁用模拟量输入接口。
Osh/Osl	模拟量比例换算, 设备将对采集值进行换算后显示, 周期上报数据依旧为原始数据。Osh 需大于 Osl, 输入范围: -65535~65535。 Osh: 采集到最大值代表的数值。 Osl: 采集到最小值代表的数值。
单位	4-20mA 电流输入: 默认 mA, 可自定义。 0-10V 电压输入: 默认 V, 可自定义。
读取	点击“读取”获取当前模拟量输入数值。

供电配置:

启用“接口 1 (Pin1) 5/9/12V 输出”或“接口 1 (Pin2) 3.3V 输出”，并配置供电时间。**注意：**测试时建议先使用外部电源为终端设备供电。



参数	说明
接口 1 (Pin1) 5/9/12V 输出	默认为关闭，可手动开启，开启后接口 1 (Pin1)将对外供电（供电电压取决于硬件拨码配置）。
采集数据前对设备供电 时间/s	用于采集数据前对设备进行供电，默认为 1s，可手动更改，可更改范围为 0-600s。
接口 1 (Pin2) 3.3V 输出	默认为关闭，可手动开启，开启后接口 2 (Pin2)将对外进行 3.3V 供电。
供电方式	支持“持续供电”或“可配置供电时间”。当选择“可配置供电时间”时，可配置 0-600 秒。
供电电流	不实际决定对外供电电流，仅用于电池寿命计算。默认为 0.00mA，可手动更改，可更改范围为 0-100mA。

4.5 串口设置

4.5.1 RS485

使用 UC50x 设备的 RS485 前，请确认终端设备支持 Modbus RTU 标准协议，然后将设备接到 UC50x 数据接口 2 的串口。如需 UC50x 给终端设备供电，请将终端电源线接到接口 2 的供电输出接口。

注意：测试时建议先使用外部电源为 RS485 终端设备供电。

配置步骤：

1. 打开 ToolBox App 的“设置->串口设置”菜单，启用串口并设置为 RS485，配置串口的基本参数。串

口基本参数必须和终端设备的串口参数相同。

☰
UC502-L0ACN
☰

状态
设置
维护

串口设置

RS485(Modbus Master)
RS232

接口2 (Pin 1) 5/9/12V 输出

供电电流 / mA ①

0

采集数据前对设备供电时间 / s

1

接口2 (Pin 2) 3.3V 输出

供电方式

持续供电
▼

供电电流 / mA ①

0

写入

参数	说明
接口 2 (Pin1) 5/9/12V 输出	默认为关闭，可手动开启，开启后接口 1 (Pin1)将对外供电（供电电压取决于硬件拨码配置）。
供电电流/mA	不实际决定对外供电电流，仅用于电池寿命计算。默认为 0.00mA，可手动更改，可更改范围为 0-100mA。
采集数据前对设备供电 时间/s	用于采集数据前对设备进行供电，默认为 1s，可手动更改，可更改范围为 0-600s。
接口 2 (Pin1) 3.3V 输出	默认为关闭，可手动开启，开启后接口 2 (Pin2)将对外进行 3.3V 供电。

供电方式	支持“持续供电”或“可配置供电时间”。当选择“可配置供电时间”时，可配置 0-600 秒。
供电电流	不实际决定对外供电电流，仅用于电池寿命计算。默认为 0.00mA，可手动更改，可更改范围为 0-100mA。

状态
设置
维护

通讯配置

波特率
9600

数据位
8

停止位
1

奇偶位
None

命令执行时间间隔(ms)
50

最大响应时间(ms)
500

最大重试次数
3

RS485主动透传

协议
TCP

保活间隔(s)
300

重连间隔 (s)
10

重连次数
1

一组

二组

通道设置配置 全部采集

+ 继续添加 0/16

写入

参数	说明
波特率	默认为 9600。 可选：1200/2400/4800/9600/14400/19200/38400/57600/115200。
数据位	默认位 8 bits，可手动更改，可更改为 7 bits（仅对透传生效）。

停止位	默认为 1 bits，可手动更改，可更改为 2 bits。
奇偶位	默认无，可手动更改，可更改为奇校验、偶校验。
命令执行时间间隔 (ms)	每个 Modbus 通道指令的执行间隔。默认为 50ms，可手动更改，可更改范围为 10-1000ms。
最大响应时间	发送 Modbus 指令后等待 Modbus Slave 设备回复的最大时间。默认为 500ms，可手动更改，可更改范围为 10-1000ms。
最大重试次数	Modbus Slave 设备读取数据失败后的最大重试次数。默认为 3 次，可手动更改，可更改范围为 0-5。

RS485 (Modbus) 主动透传模式

协议	开启主动透传模式后，可以选择透传协议，默认为 TCP，可下拉更改为 UDP、MQTT。
保活间隔	连接成功后，设备将按照设置的保活间隔发送保活包，以保证与服务器间的连接状态。默认为 300s，可更改为 10~65535s。——仅在 低延时模式 下有此配置。
重连间隔	配置重连服务器的时间间隔，连接失败后，会按照所配置的重连间隔及连次数重新请求连接。默认为 10s，可配置 10-60s。
重连次数	配置重连服务器的次数，连接失败后，会按照所配置的重连间隔及连次数重新请求连接。默认为 1 次，可配置 0~3 次。当配置 0 次时，即不进行重新连接。

TCP/UDP

服务器地址/端口	<p>1.请填入 TCP/UDP 服务器地址 (IP 地址/域名)。</p> <p>2.请填写 TCP/UDP 服务器端口。范围：1-65535。</p> <p>注意：“串口设置处”服务器地址/端口“不能与应用模式处的 TCP/UDP 模式下的”服务器地址/端口“完全一致。2 组”服务器地址/端口“不能两两完全一致。</p>
网络状态	<p>显示连接状态：①未驻网②已驻网但未连接平台③已驻网已连接平台。</p> <p>注意：仅 TCP 会显示连接状态。</p>

通道采集：

Modbus 通道：

UC50x 支持作为 Modbus RTU 客户端 (主机) 工作，从 RS485 设备轮询数据并将数据返回给服务器。

1.请点击“+”按钮添加 Modbus 通道，然后保存配置。

<添加通道

通道

Channel 1▼

名称

从设备ID

读取寄存器起始地址

读取数据量

1▼

类型

Holding Register(INT16)▼

字节顺序

AB▼

有符号

保存

RS485主动透传



通道设置配置
全部采集

通道ID: 1
编辑

通道名称: 1

读取寄存器起始地址: 0

值

采集

+ 继续添加 1/16

GPIO设置 v规则引擎 vSDI-12设置 >

写入

参数	说明
通道	最多可配置 1-16 个通道，配置时通道 ID 不可重复。
名称	自定义通道名称以识别每个 Modbus 通道。
从设备 ID	用于设置 Modbus Slave 的 ID。
读取寄存器起始地址	配置要读取的 Modbus Slave 的起始地址。
读取数据量	从站寄存器读取的数据数量，默认为 1 不可更改。
类型	设置 Modbus 数据类型，Modbus 标准协议内的数据类型均支持。具体如下图所示：

	<div style="border: 1px solid black; padding: 5px;"> <p>Coil</p> <p>Discrete</p> <p>Input Register(INT16)</p> <p>Holding Register(INT16)</p> <p>Holding Register (INT32)</p> <p>Holding Register (Float)</p> <p>Input Registers (INT32)</p> <p>Input Registers (Float)</p> <p>Input Register (INT 32 with upper 16 bits)</p> <p>Input Register (INT 32 with lower 16 bits)</p> <p>Holding Register (INT 32 with upper 16 bits)</p> <p>Holding Register (INT 32 with lower 16 bits)</p> </div>
字节顺序	<p>当 Modbus 数据类型为 Input Register 或 Holding Register 时，可配置合适的字节读取顺序来正确处理来自 Modbus Slave 的数据。</p> <p>①INT32/Float: ABCD, CDBA, BADC, DCBA, 默认 CDAB。</p> <p>②INT16: AB, BA, 默认 AB。</p>
有符号	<p>用于选定采集的数据是否为有符号的数据 (+或者-)。</p>
值	<p>按上述配置，到达采集间隔将进行采集，到达周期上报时间/符合阈值条件后将进行上报。</p>
全部采集	<p>采集全部通道的数值。</p>
采集/读取	<p>采集/读取对应通道里面的数值。</p>

4.5.2 RS232

1. 打开 ToolBox App 的“设置->串口设置”菜单启用串口并设置为 RS232，配置串口的基本参数。串口基本参数必须和终端设备的串口参数相同。
2. 完成配置后，当终端设备向 UC50x 设备发送数据时，设备会将数据直接发送到网络服务器。

状态	设置	维护
通讯配置		
波特率		
9600		
数据位		
8		
停止位		
1		
奇偶位		
None		
协议		
MQTT		
串口分帧长度 (byte)		
256		
串口分帧间隔 (ms)		
100		
GPIO设置		
规则引擎		
SDI-12设置		
写入		
设备	模板	

参数	说明
波特率	默认为 9600。 可选：1200/2400/4800/9600/14400/19200/38400/57600/115200。
数据位	默认位 8 bits，可手动更改，可更改为 7 bits。
停止位	默认为 1 bits，可手动更改，可更改为 2 bits。
奇偶位	默认无，可手动更改，可更改为奇校验、偶校验。
协议	默认为 TCP，可下拉更改为 UDP、MQTT。
TCP/UDP	
串口分帧长度	串口根据所设长度将数据帧分包传输。默认为 256byte，可配置 1-1000。
串口分帧间隔	数据发送间隔。默认为 100ms，可配置 10~65535。

保活间隔	连接成功后，设备将按照设置的保活间隔发送保活包，以保证与服务器间的连接状态。默认为 300s，可更改为 10~65535s。 注意： 仅在 低延时模式 下有此配置
重连间隔	配置重连服务器的时间间隔，连接失败后，会按照所配置的重连间隔&重连次数重新请求连接。默认为 5s，可配置 1-60s。
重连次数	配置重连服务器的次数，连接失败后，会按照所配置的重连间隔&重连次数重新请求连接。默认为 1 次，可配置 0~3 次。当配置 0 次时，即不进行重新连接。
服务器地址/端口	1.请填入 TCP/UDP 服务器地址（IP 地址/域名）。 2.请填写 TCP/UDP 服务器端口。范围：1-65535。 注意： “串口设置处”服务器地址/端口“不能与应用模式处的 TCP/UDP 模式下的”服务器地址/端口“完全一致。2 组”服务器地址/端口“不能两两完全一致。
网络状态	显示连接状态：①未驻网②已驻网但未连接平台③已驻网已连接平台。 注意： 仅 TCP 会显示连接状态。

4.6 GPIO 设置

用于对接开关量、脉冲计数量设备，例如连接水表利用脉冲计数原理。使用 UC50x 设备的 GPIO 前，请将终端设备接到 UC50x 数据接口 2 的 GPIO 接口。

配置步骤：

1.打开 ToolBox App 的“设置->GPIO 设置”菜单，启用 GPIO 接口。

2.选择接口类型。

- **数字量输入：**检测设备的高低电平。
- **数字量输出：**发送电压信号触发设备。
- **计数器：**脉冲计数。

数字量输入：

数字输入可以用于检测设备的高低电平状态。

状态	设置	维护
GPIO1 🔴		
接口类型		
数字量输入1 ▾		
数字输入 ⓘ		
拉低 ▾		
状态	-	读取
GPIO2 🟢		
接口类型		
数字量输入2 ▾		
数字输入 ⓘ		
拉低 ▾		
状态	-	读取

参数	说明
数字量输入	数字输入的初始状态。 拉低：触发上升沿。 拉高：触发下降沿。
读取	点击获取数字输入的当前状态。

注意：（1）重启不会影响计数。（2）脉冲值可以通过 ToolBox 或下行命令手动清除，或在计算达到最大值 4000000000 时自动清除。

数字量输出：

数字输出将发送电压信号来控制设备。

GPIO设置 ^	
GPIO1 🟢	
接口类型	
数字量输出1 ▾	
状态	
-	读取
	切换

参数	说明
读取	点击获取数字输出的当前状态。

切换

点击以切换数字输出状态，以检查 UC50x 是否可以触发设备。

计数:

GPIO设置 ^

GPIO1

接口类型

计数器

数字输入 ⓘ

拉低

数字滤波 ⓘ

重启设备时计数不清零

计数值 -

刷新 停止 清零

修改计数值

确认

参数	说明
数字输入	计数器的初始状态。 拉低：检测到上升沿时增加。拉高：检测到下降沿时增加 1。（浮空状态）
数字滤波	支持设置滤波速率。 开启后，默认将 < 250us 速率的脉冲滤去，不计入计数范围。支持手动更改滤波的波形速率。当 Toolbox 选择 us 为单位时，仅支持 250us；当选择 ms 为单位时，可配置 1~65535ms。
开始/停止	使设备开始/停止计数。
刷新	刷新当前结果。
清零	将计数值清零。
修改计数值	修改当前计数值。

4.7 规则引擎

UC50x 支持配置命令，以向服务器发送报警数据包。每个设备最多可以添加 16 个条规则引擎。

1.前往设备 > 设置 > 规则引擎以添加命令。



2.设置一个引擎条件，包括模拟输入值或 RS485 Modbus 通道值。当值匹配条件时，设备将发送一次报警数据包。只有当值恢复正常并再次触发条件时，才会发送新的报警。

注意：设备只会发送一次报警。只有当值恢复正常并再次满足条件时，它才会发送新的报警。



3.在设置完所有命令后，点击保存以保存设置。

4.8 SDI-12 设置

1..将 SDI-12 传感器连接到接口 2 上的 SDI-12 端口。如果 SDI-12 设备需要来自 UC50x 的电源，请将 SDI-12

设备的电源线连接到接口 2 上的电源输出。

2.进入设备 > 设置 > SDI-12 设置，并连接手机以读取当前设置。

<
SDI-12配置

SDI-12

接口2(Pin1) 5/9/12V输出

供电电流(mA)

0.00

采集数据前对设备供电时间(s)

1

通讯配置

波特率

9600

数据位(bits)

8

停止位

1

奇偶位

无

最大重试次数

3

SDI-12主动透传

参数	说明
接口 2 (Pin1) 5/9/12V 输出	默认为关闭，可手动开启，开启后接口 1 (Pin1)将对外供电（供电电压取决于硬件拨码配置）。
供电电流	不实际决定对外供电电流，仅用于电池寿命计算。默认为 0.00mA，可手动更改，可更改范围为 0-100mA。
采集数据前对设备供电时间/s	用于采集数据前对设备进行供电，默认为 1s，可手动更改，可更改范围为 0-600s。
参数	说明
波特率	默认为 9600。

	可选：1200/2400/4800/9600/14400/19200/38400/57600/115200。
数据位	默认位 7 bits，可手动更改，可更改为 8 bits。
停止位	默认为 1 bits，可手动更改，可更改为 2 bits。
奇偶位	默认为偶校验，可手动更改，可更改为无、奇校验。
最大重试次数	默认为 3 次，可手动更改，可更改范围为 0-5。
SDI-12 主动透传	启用此模式，服务器可以向 SDI-12 设备发送 SDI-12 命令，并且设备只能根据服务器的命令做出反应。这仅在使用低延迟模式时有效。

通道配置

通道ID

1

* 名称

地址

6 写入

SDI-12指令 ①

aM!

⊖ aD0!

⊕ 继续添加

值

-

保存

通道设置配置
全部采集

通道ID: 1
编辑

通道名称: 2456

寄存器起始地址: 123456778888...

值

采集

通道ID: 1
编辑

通道名称: 2456

寄存器起始地址: 123456778888...

值

采集

+ 继续添加 2/16

通道设置配置

通道 ID	最多可配置 1-16 个通道，配置时通道 ID 不可重复。
名称	可以自定义每个通道的名称以便于识别它们。
寄存器起始地址	输入寄存器起始地址。
写入	修改 SDI-12 传感器地址。
SDI-12 指令	填写 SDI-12 指令，默认输入如下 2 个指令，可根据传感器需求添加/修改指令。 最多可添加 16 条指令。 aM! : 测量传感器数值 aD0! : 返回传感器测量的数值 其中 a 为传感器地址。
全部采集	采集全部通道的数值。
采集/读取	采集/读取对应通道里面的数值。
值	显示已采集的数值。

4.9 数据存储

UC50x 系列设备最多支持本地存储 10000 条数据记录，并通过 ToolBox 应用程序导出数据。即使设备未连接到网络，它也会根据报告间隔记录数据。

注意：不同情况，数据的存储条数不同。

1. 检查设备时间：

通过 Toolbox App 软件为设备同步准确的时间，该时间可以自动从移动运营商同步，也可以通过 Toolbox 应用程序手动进行同步。



2. 启用数据存储功能：

打开 Toolbox App 的“设置->常用设置”菜单，启用数据存储功能。启用后，设备将存储所有采集数据。



3. 本地数据导出与清除：

- Toolbox App：打开“维护”菜单，点击“导出”，选择导出数据时间段后将手机贴到设备的 NFC 区域完成数据导出；点击“数据清除”按钮，清除历史存储数据。
- 如有必要，点击清除以清楚设备内的所有存储数据。

注意：IOS 版本由于交互时间限制，无法导出存储数据。



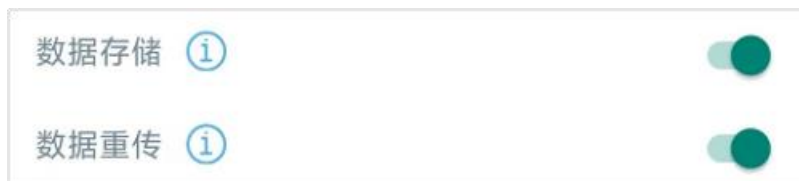
4.9.1 数据重传

UC50x 支持断网数据重传功能，当设备与网关失联，会主动记录断网时间点，待设备联网后重新传输断网时间点与联网时间点之间丢失的数据包，避免设备由于断网或丢包导致传感器数据丢失，保证数据完整性。

注意：选择 UDP 时无重传和回传功能，只有数据存储，选择 MQTT、AWS、TCP、IoT Cloud 时有数据存储、重传和回传功能。

配置步骤：

打开 ToolBox App 的“设置->常用设置”菜单，启用**数据存储**功能与**数据重传**功能。



4.10 维护

4.10.1 升级

ToolBox App

步骤 1：将固件下载到手机端；

步骤 2：打开 ToolBox App “维护” 菜单，点击“浏览”导入固件，开始升级。



4.10.2 备份

UC50x 系列支持备份设备配置并导入到其它设备中，可用于快速批量配置。备份导入仅适用于型号完全相同的设备。

ToolBox App

步骤 1: 打开 ToolBox App 的“模板”菜单，将当前配置保存为新的模板到手机上；

步骤 2: 选择已保存的模板，点击“写入”后将手机贴到设备的 NFC 区域写入配置。

注意：在“模板”页面选择对应的模板条目，向左划动选择编辑模板名称或删除模板。点击对应的模板条目即可查看和编辑具体的模板内容。



4.10.3 重置

可选择如下方法重置设备：

硬件重置：拆下外壳，长按主板上的电源按钮超过 10 秒直到 LED 灯闪烁。

ToolBox App：打开“维护”菜单，点击“重置”后将手机贴到设备的 NFC 区域写入配置。



五、产品安装

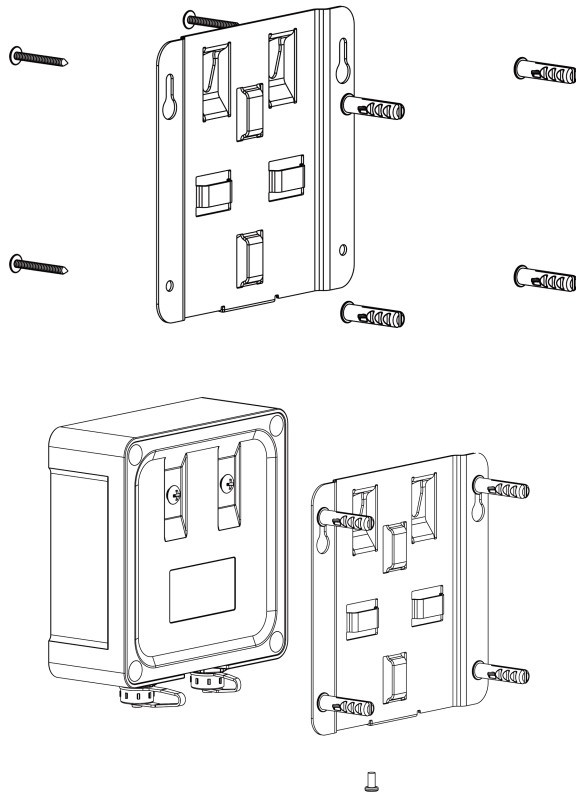
户外安装注意事项：

- (1) 注意 UC50x 与终端及电源接线的防水处理；
- (2) 安装在高处的设备，需要做好相应的防雷接地。

5.1 壁挂式安装

安装配件：壁挂安装板，安装螺钉，膨胀螺栓，壁挂螺钉和其它辅助工具。

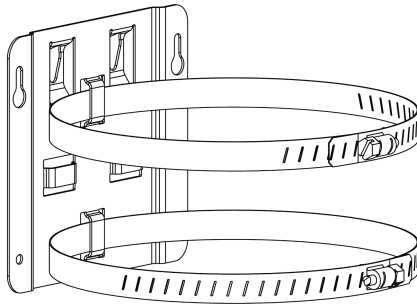
1. 将 4 个膨胀螺栓打到墙上，然后将壁挂螺丝穿过安装板钉入膨胀螺栓内。
2. 将设备通过背后的螺丝挂到安装支架上，再用 1 颗固定螺丝将设备底部和安装支架固定在一起。

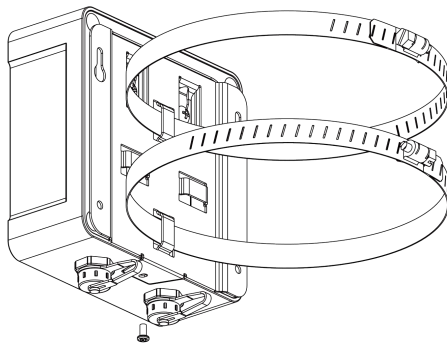


5.2 抱杆式安装

安装配件：平面安装板，安装螺钉，抱箍和其他辅助工具。

1. 逆时针拧开抱箍的锁固，将抱箍拉直并穿进安装板上的环上，然后将抱箍绕到目标杆上。用螺丝刀顺时针拧紧抱箍上的锁固。
2. 将设备通过背后的螺丝挂到安装支架上，再用 1 颗固定螺丝将设备底部和安装支架固定在一起。





六、通信协议

6.1 上行数据

固定标志	ID	负载长度	FLAG1	帧计数	协议版本
02	0001	2 字节	00	0000	01
软件版本	硬件版本	SN		IMEI	
4 字节	4 字节	16 字节		15 字节	
IMSI		ICCID		ICCID	信号
15 字节		20 字节		20 字节	1 字节
数据长度		数据		
2 字节		N 字节		

示例	
02 0001 0060 00 0008 00 30313031 30313030 36373732443431323335313830303133 383637313037303638373335343031 3436303038383333337363034323739 3839383630383133313032333830393630323739 11 000f 07ef 7bbe ee65 0300 0004 0000 0175 62	
类型	说明

固定标志	02
ID	0001
负载长度	00 60=96 bytes
FLAG	00
帧计数	0008
协议版本	00
软件版本	30 31 30 31 => 0101=V1.1
硬件版本	30 31 30 30 => 0100=V1.0
SN	36 37 37 32 44 34 31 32 33 35 31 38 30 30 31 33 =>6772D41235180013
IMEI	38 36 37 31 30 37 30 36 38 37 33 35 34 30 31=>867107068735401
IMSI	34 36 30 30 38 38 33 33 37 36 30 34 32 37 39 =>460088337604279
ICCID	38 39 38 36 30 38 31 33 31 30 32 33 38 30 39 36 30 32 37 39 => 89860813102380960279
网络信号	11=>17 asu
数据长度	00 0f=>15 Bytes
数据	详情参见下文

设备上/下行数据均基于十六进制格式。数据处理方式低位在前，高位在后。上/下行指令基本格式：

通道号 1	类型 1	数据 1	通道号 2	类型 2	数据 2	...
1 字节	1 字节	N 字节	1 字节	1 字节	M 字节	...

注意：（1）设备重新启动时，帧计数器将被清零。（2）有关 Milesight IoT 解码器示例，请在以下网址找到相关文件：<https://github.com/Milesight-IoT/SensorDecoders>

6.1.1 基础数据包

通道	类型	数据长度	数据内容
0x01	0x75	01	电池电量 <ul style="list-style-type: none"> ● 数据定义： ◇ battery(1B) ● 参数说明： ◇ battery: 电池电量, uint8

			<ul style="list-style-type: none"> ● 示例: ◇ 0175 10 ● 示例解析: ◇ 当前电量为 16%
0x03	0x00	01	GPIO 1(作为 DI 上报) <ul style="list-style-type: none"> ● 数据定义: ◇ gpio(1B) ● 参数说明: ◇ gpio: DI 输入类型, uint8 ■ 0: 低电平 ■ 1: 高电平
0x04	0x00	01	GPIO 2(作为 DI 上报) <ul style="list-style-type: none"> ● 数据定义: ◇ gpio(1B) ● 参数说明: ◇ gpio: DI 输入类型, uint8 ■ 0: 低电平 ■ 1: 高电平
0x03	0xC8	04	GPIO 1(作为 counter 上报) <ul style="list-style-type: none"> ● 数据定义: ◇ counter(4B) ● 参数说明: ◇ counter: 计数值, uint32
0x04	0xC8	04	GPIO 2(作为 counter 上报) <ul style="list-style-type: none"> ● 数据定义: ◇ counter(4B) ● 参数说明: ◇ counter: 计数值, uint32
0x05	0xF1	n	AI 1 上报 <ul style="list-style-type: none"> ● 数据定义: ◇ mode(1B) + data(nB) ● mode 说明: ◇ mode[bit0]: AI 的模式

			<ul style="list-style-type: none"> ■ 0: 4-20mA ■ 1: 0-10V; ◇ mode[bit1]: ■ 0: 只含有当前值 ■ 1: 包含当前值、最小值、最大值、平均值 ■ 注: 报警时只有当前值 ◇ mode[bit2]: ■ 0: 表示未压缩 float32 ■ 1: 表示压缩为 float16 ■ 注: 0 表示未缩表示后续数据使用的数据类型为 float(预留使用), 这里默认为 1 ● data 说明: ◇ value_current: 当前值(float16) ◇ value_min: 最小值(float16) ◇ value_max: 最大值(float16) ◇ value_avg: 平均值(float16)
0x06	0xF1	n	<p>AI 2 上报</p> <ul style="list-style-type: none"> ● 数据定义: ◇ mode(1B) + data(nB) ● mode 说明: ◇ mode[bit0]: AI 的模式 ■ 0: 4-20mA ■ 1: 0-10V ◇ mode[bit1]: ■ 0: 只含有当前值 ■ 1: 包含当前值、最小值、最大值、平均值 ■ 注: 报警时只有当前值 ◇ mode[bit2]: ■ 0: 表示未压缩 float32 ■ 1: 表示压缩为 float16 ■ 注: 0, 表示未缩表示后续数据使用

			的数据类型为 float(预留使用), 这里默认为 1 ● data 说明: ◇ value_current: 当前值(float16) ◇ value_min: 最小值(float16) ◇ value_max: 最大值(float16) ◇ value_avg: 平均值(float16)
0x07	0xef	04	上报时间戳, 使用 uint32_t 读取 ● 数据定义: ◇ timestamp(4B) ● 参数说明: ◇ timestamp 时间戳 单位: 秒
0x08	0xf2	n	SDI 数据上报 ● 数据定义: ◇ sdi_chn(1B)+mode(1B) + data(nB) ● 数据说明: ◇ mode 表示 sdi 的字节数, 为 0 时走的是异常值包 ● 参数说明: ◇ sdi_chn: SDI-12 通道, 0~15 ◇ data: SDI-12 通道数据, ASCII 码字符
0x09	0xf3	N	modbus 数据上报 ● 数据定义: ◇ modbus_chn(1B) + mode(1B)+ data(NB) ● 数据说明: ◇ mode[bit7]: 符号位 ■ 0: 无符号 ■ 1: 有符号 ◇ mode[bit6-bit4]: reserved ◇ mode[bit3-bit0]: 寄存器的类型

			<ul style="list-style-type: none"> ■ 0: 1 字节线圈; ■ 1: 1 字节离散; ■ 2: 2 字节输入整型; ■ 3: 2 字节保持整型; ■ 4: 4 字节保持整型; ■ 5: 4 字节保持浮点; ■ 6: 4 字节输入整型; ■ 7: 4 字节输入浮点; ● 参数说明: ◇ modbus_chn: modbus 通道, 0~15 ◇ data: (value), 长度由 modbus 类型定, ■ 寄存器的类型: 0, 1, data 为 bool 型 ■ 寄存器的类型: 2, 3, data 为 uint16/int16, 由符位决定 ■ 寄存器的类型: 4, 6, data 为 uint32/int32, 由符位决定 ■ 寄存器的类型: 5, 7, data 为 float
--	--	--	---

6.1.2 阈值包

通道	类型	数据长度	数据内容
0x85	0xF1	n	<p>AI 1 阈值包上报</p> <ul style="list-style-type: none"> ● 数据定义: ◇ mode(1B) + data(nB) + alarm_state(1B) ● 数据说明: ◇ mode[bit0]: AI 的模式 ■ 0: 4-20mA ■ 1: 0-10V; ◇ mode[bit1]:

			<ul style="list-style-type: none"> ■ 0: 只含有当前值 ■ 1: 包含当前值、最小值、最大值、平均值 ■ 注: 报警时只有当前值 ◇ mode[bit2]: <ul style="list-style-type: none"> ■ 0: 表示未压缩 float32 ■ 1: 表示压缩为 float16 ■ 注: 0 表示未缩表示后续数据使用的数据类型为 float32(预留使用), 这里默认为 1 ● 参数说明: <ul style="list-style-type: none"> ◇ value_current: 当前值(float16) ◇ value_min: 最小值(float16) ◇ value_max: 最大值(float16) ◇ value_avg: 平均值(float16) ◇ alarm_state:报警状态(uint8_t) <ul style="list-style-type: none"> ■ 0: 解除阈值报警 ■ 1: 阈值报警 <p>注: 目前需求只有阈值报警, 解除阈值为预留值</p>
0x86	0xF1	n	<p>AI 2 阈值包上报</p> <ul style="list-style-type: none"> ● 数据定义: <ul style="list-style-type: none"> ◇ mode(1B) + data(nB) + alarm_state(1B) ● 数据说明: <ul style="list-style-type: none"> ◇ mode[bit0]: AI 的模式 <ul style="list-style-type: none"> ■ 0: 4-20mA ■ 1: 0-10V ◇ mode[bit1]: <ul style="list-style-type: none"> ■ 0: 只含有当前值 ■ 1: 包含当前值、最小值、最大值、平均值 ■ 注: 报警时只有当前值 ◇ mode[bit2]:

			<ul style="list-style-type: none"> ■ 0: 表示未压缩 float32 ■ 1: 表示压缩为 float16 ■ 注: 0, 表示未缩表示后续数据使用的数据类型为 float32(预留使用), 这里默认为 1 ● 参数说明: <ul style="list-style-type: none"> ◇ value_current: 当前值(float16) ◇ value_min: 最小值(float16) ◇ value_max: 最大值(float16) ◇ value_avg: 平均值(float16) alarm_state:报警状态(uint8_t) <ul style="list-style-type: none"> ■ 0: 解除阈值报警 ■ 1: 阈值报警 注: 目前需求只有阈值报警, 解除阈值为预留值
0x89	0xf3	N	<p>modbus 数据上报</p> <ul style="list-style-type: none"> ● 数据定义: <ul style="list-style-type: none"> ◇ modbus_chn(1B) +mode(1B)+ data(NB)+ alarm_state(1B) ● 数据说明: <ul style="list-style-type: none"> ◇ mode[bit7]: 符号位 <ul style="list-style-type: none"> ■ 0: 无符号 ■ 1: 有符号 ◇ mode[bit6-bit4]:reserved ◇ mode[bit3-bit0]: 寄存器的类型 <ul style="list-style-type: none"> ■ 0: 1 字节线圈; ■ 1: 1 字节离散; ■ 2: 2 字节输入整型; ■ 3: 2 字节保持整型; ■ 4: 4 字节保持整型; ■ 5: 4 字节保持浮点; ■ 6: 4 字节输入整型; ■ 7: 4 字节输入浮点; ● 参数说明:

			<ul style="list-style-type: none"> ◇ modbus_chn: modbus 通道, 0 ~ 15 ◇ data: (value), 长度由 modbus 类型定, ■ 寄存器的类型: 0, 1, data 为 bool 型 ■ 寄存器的类型: 2, 3, data 为 uint16/int16, 由符位决定 ■ 寄存器的类型: 4, 6, data 为 uint32/int32, 由符位决定 ■ 寄存器的类型: 5, 7, data 为 float ◇ alarm_state:报警状态(uint8_t) ■ 0: 解除阈值报警 ■ 1: 阈值报警 <p>注: 目前需求只有阈值报警, 解除阈值为预留值</p>
--	--	--	--

6.1.3 突变包

通道	类型	数据长度	数据内容
0x95	0xF1	n	<p>AI 1 阈值包上报</p> <ul style="list-style-type: none"> ● 数据定义: <ul style="list-style-type: none"> ◇ mode(1B) + data(nB)+value_mutation(nB)+ alarm_state(1B) ● 数据说明: <ul style="list-style-type: none"> ◇ mode[bit0]: AI 的模式 ■ 0: 4-20mA ■ 1: 0-10V; ◇ mode[bit1]: <ul style="list-style-type: none"> ■ 0: 只含有当前值 ■ 1: 包含当前值、最小值、最大值、平均值 ■ 注: 报警时只有当前值 ◇ mode[bit2]: <ul style="list-style-type: none"> ■ 0: 表示未压缩 float32

			<ul style="list-style-type: none"> ■ 1: 表示压缩为 float16 ■ 注: 0 表示未缩表示后续数据使用的数据类型为 float(预留使用), 这里默认为 1 ● 参数说明: <ul style="list-style-type: none"> ◇ value_current: 当前值(float16) ◇ value_min: 最小值(float16) ◇ value_max: 最大值(float16) ◇ value_avg: 平均值(float16) ◇ value_mutation: 突变值(float16) ◇ alarm_state:报警状态(uint8_t) ■ 预留
0x96	0xF1	n	<p>AI 2 阈值包上报</p> <ul style="list-style-type: none"> ● 数据定义: <ul style="list-style-type: none"> ◇ mode(1B) + data(nB)+value_mutation(nB)+ alarm_state(1B) ● 数据说明: <ul style="list-style-type: none"> ◇ mode[bit0]: AI 的模式 <ul style="list-style-type: none"> ■ 0: 4-20mA ■ 1: 0-10V ◇ mode[bit1]: <ul style="list-style-type: none"> ■ 0: 只含有当前值, 报警时只有当前值 ■ 1: 包含当前值、最小值、最大值、平均值 ■ 注: 报警时只有当前值 ◇ mode[bit2]: <ul style="list-style-type: none"> ■ 0: 表示未压缩 float32 ■ 1: 表示压缩为 float16 ● 注: 0, 表示未缩表示后续数据使用的数据类型为 float(预留使用), 这里默认为 1 ● 参数说明: <ul style="list-style-type: none"> ◇ value_current: 当前值(float16)

			<ul style="list-style-type: none"> ◇ value_min: 最小值(float16) ◇ value_max: 最大值(float16) ◇ value_avg: 平均值(float16) ◇ value_mutation: 变化值(float16) ◇ alarm_state:报警状态(uint8_t) ■ 0: 预留
0x99	0xf3	N	<p>modbus 数据上报</p> <ul style="list-style-type: none"> ● 数据定义: <ul style="list-style-type: none"> ◇ modbus_chn(1B) +mode(1B)+ data(NB)+value_mutation(NB) + alarm_state(1B) ● 数据说明: <ul style="list-style-type: none"> ◇ mode[bit7]: 符号位 <ul style="list-style-type: none"> ■ 0: 无符号 ■ 1: 有符号 ◇ mode[bit6-bit4]:reserved ◇ mode[bit3-bit0]: 寄存器的类型 <ul style="list-style-type: none"> ■ 0: 1 字节线圈; ■ 1: 1 字节离散; ■ 2: 2 字节输入整型; ■ 3: 2 字节保持整型; ■ 4: 4 字节保持整型; ■ 5: 4 字节保持浮点; ■ 6: 4 字节输入整型; ■ 7: 4 字节输入浮点; ● 参数说明: <ul style="list-style-type: none"> ◇ modbus_chn: modbus 通道,0~15 ◇ data: (value), 长度由 modbus 类型定, <ul style="list-style-type: none"> ■ 寄存器的类型: 0, 1, data 为 bool 型 ■ 寄存器的类型: 2, 3, data 为 uint16/int16, 由符位决定

			<ul style="list-style-type: none"> ■ 寄存器的类型: 4, 6, data 为 uint32/int32, 由符位决定 ■ 寄存器的类型: 5, 7, data 为 float ◇ value_mutation : 变化值 ■ 格式同 data ◇ alarm_state: 报警状态(uint8_t) ■ 0: 预留
--	--	--	---

6.1.4 异常值包

通道	类型	数据长度	数据内容
0xb5	0xF1	n	AI 1 异常值包上报 <ul style="list-style-type: none"> ● 数据定义: <ul style="list-style-type: none"> ◇ er_type(1B) ● 参数说明: <ul style="list-style-type: none"> ◇ er_type: 异常值类型 ■ 0: 失败 ■ 1: 超量程
0xb6	0xF1	n	AI 2 异常值包上报 <ul style="list-style-type: none"> ● 数据定义: <ul style="list-style-type: none"> ◇ er_type(1B) ● 参数说明: <ul style="list-style-type: none"> ◇ er_type: 异常值类型 ■ 0: 失败 ■ 1: 超量程
0xb8	0xf2	n	SDI 数据上报 <ul style="list-style-type: none"> ● 数据定义: <ul style="list-style-type: none"> ◇ sdi_chn(1B) + er_type(1B) ● 参数说明: <ul style="list-style-type: none"> ◇ sdi_chn: SDI-12 通道, 0~15 ◇ er_type: 异常值类型 ■ 0: 失败 ■ 1: 超量程
0xb9	0xf3	N	modbus 数据上报 <ul style="list-style-type: none"> ● 数据定义:

			<ul style="list-style-type: none"> ◇ modbus_chn(1B) + er_type(1B) ● 参数说明: ◇ modbus_chn: modbus 通道, 0~15 ◇ er_type: 异常值类型 ■ 0: 失败 ■ 1: 超量程
--	--	--	---

6.1.5 回传、重传数据

通道	类型	数据长度	数据内容
0x20	0xef	M	基本参数, 回传功能 <ul style="list-style-type: none"> ● 数据定义: ◇ timestamp(4B) + 周期包内容 ● 参数说明: ◇ timestamp: 时间戳, uint32

6.2 控制命令下发

6.2.1 设置采集周期

通道	类型	数据长度	数据内容
0xff	0x02	2	采集周期, 单位: 秒

6.2.2 设置上报周期

通道	类型	数据长度	数据内容
0xff	0x03	2	上报周期, 单位: 分钟

例: b0 04 => 04 b0=1200 设置上报周期为 1200 秒。

例: FF 28 FF 设备立马进行周期上报。

6.2.3 重启

通道	类型	数据长度	数据内容
0xff	0x10	1	CTRL (1B) : 0xff

6.2.4 设置时区

通道	类型	数据长度	数据内容
0xff	0x17	2	时区 (2B), 注意: 有正负之分

6.2.5 立即上报

通道	类型	数据长度	数据内容
0xff	0x28	1	CTRL (1B) : 0xff

注: 收到该命令后, 设备立马进行周期上报。

6.2.6 清除历史记录

通道	类型	数据长度	数据内容
0xff	0x27	1	CTRL (1B) : 0x01

6.2.7 历史记录存储使能

通道	类型	数据长度	数据内容
0xff	0x68	1	0: 禁用, 1: 启用

6.2.8 历史记录重传使能

通道	类型	数据长度	数据内容
0xff	0x69	1	0: 禁用, 1: 启用

6.2.9 历史记录重传周期

通道	类型	数据长度	数据内容
0xff	0x6a	3	配置历史记录重传周期 <ul style="list-style-type: none"> ● 数据定义: ◇ type(1B) + period(2B) ● 数据说明: ◇ type: ■ 0: 配置重传周期

			<ul style="list-style-type: none"> ■ 1: 配置回传周期 ◇ period: 时间, 单位: 秒
--	--	--	--

6.2.10 回传一个时间点数据

通道	类型	数据长度	数据内容
0xfd	0x6b	4	时间戳

6.2.11 回传一段时间内的历史数据

通道	类型	数据长度	数据内容
0xfd	0x6c	8	起始时间+结束时间

6.2.12 下发停止查询回传

通道	类型	数据长度	数据内容
0xfd	0x6d	1	CTRL (1B) :0xff

6.2.13 请求时间同步

通道	类型	数据长度	数据内容
0xff	0x4a	1	CTRL (1B) :0x00

回复内容备注: fe 4a 00/01 最后一位数据位, 00 表示同步失败, 01 表示同步成功

6.2.14 设置 GPIO1 的 DO 模式

通道	类型	数据长度	data2	data3
0x03	3	0: 低电平 1: 高电平	0xff	0xff

6.2.15 设置 GPIO2 的 DO 模式

通道	类型	数据长度	data2	data3
0x04	3	0: 低电平 1: 高电平	0xff	0xff

6.2.16 下发配置 MODBUS 通道

通道	类型	数据长度	数据内容
----	----	------	------

0xff	0xEF	N	<ul style="list-style-type: none"> ● 数据定义: 创建 (oper==1) : (oper=1): oper(1B)+chan(1B)+slave id(1B)+address(2B)+type(1B)+quan_sign(1B) 删除(oper=0): oper(1B)+chan(1B) 设置通道名(oper=2) oper(1B)+chan(1B)+len(1B)+name(NB) ● 数据说明: ◇ oper: 0 删除; 1 创建; 2 修改通道名称 ◇ chan: 1-32 ◇ slaveid: 1-255 ◇ address: 0-65535 ◇ quan_sign bit7-bit5:reserved bit4:sign bit3-bit0:quantity, 寄存器数量, 只能为 1 ◇ type: 寄存器类型 MB_REG_COIL = 0, MB_REG_DIS = 1, MB_REG_INPUT = 2, MB_REG_HOLD_INT16 = 3, MB_REG_HOLD_INT32 = 4, MB_REG_HOLD_FLOAT = 5, MB_REG_INPUT_INT32 = 6, MB_REG_INPUT_FLOAT = 7, MB_REG_INPUT_INT32_AB = 8, MB_REG_INPUT_INT32_CD = 9, MB_REG_HOLD_INT32_AB = 10, MB_REG_HOLD_INT32_CD = 11,
------	------	---	--

			<p>MB_REG_INPUT_AB = 12, MB_REG_INPUT_BA = 13, MB_REG_INPUT_INT32_ABCD = 14, MB_REG_INPUT_INT32_BADC = 15, MB_REG_INPUT_INT32_CDAB = 16, MB_REG_INPUT_INT32_DCBA = 17, MB_REG_INPUT_FLOAT_ABCD = 18, MB_REG_INPUT_FLOAT_BADC = 19, MB_REG_INPUT_FLOAT_CDAB = 20, MB_REG_INPUT_FLOAT_DCBA = 21, MB_REG_HOLD_INT16_AB = 22, MB_REG_HOLD_INT16_BA = 23, MB_REG_HOLD_INT32_ABCD = 24, MB_REG_HOLD_INT32_BADC = 25, MB_REG_HOLD_INT32_CDAB = 26, MB_REG_HOLD_INT32_DCBA = 27, MB_REG_HOLD_FLOAT_ABCD = 28 , MB_REG_HOLD_FLOAT_BADC = 29 , MB_REG_HOLD_FLOAT_CDAB = 30 , MB_REG_HOLD_FLOAT_DCBA = 31 , MB_REG_INPUT_INT32_UPPER_AB = 32, MB_REG_INPUT_INT32_UPPER_BA = 33, MB_REG_INPUT_INT32_LOWER_AB = 34, MB_REG_INPUT_INT32_LOWER_BA = 35, MB_REG_HOLD_INT32_UPPER_AB = 36,</p>
--	--	--	---

			<p>MB_REG_HOLD_INT32_UPPER_BA = 37, MB_REG_HOLD_INT32_LOWER_AB = 38, MB_REG_HOLD_INT32_LOWER_BA = 39。</p>
--	--	--	---

示例	说明
ffef01010102000101	配置通道 1 开启, 名称为 1, ID 为 1, 寄存器地址为 1, 读取数量为 1, 类型为 MB_REG_DIS。
ffef01020406000F11	配置通道 2 开启, 名称为 2, ID 为 4, 寄存器地址为 6, 读取数量为 1, 类型为 MB_REG_INPUT_INT32_BADC, 有符号。
ffef0202063031323435	修改通道 1 的名称为 01245
ffef020206e4bda0e5a5bd	修改通道 2 的名称为 "你好" 注: 中文遵循 UTF8 编码
ffef0001	删除通道 1