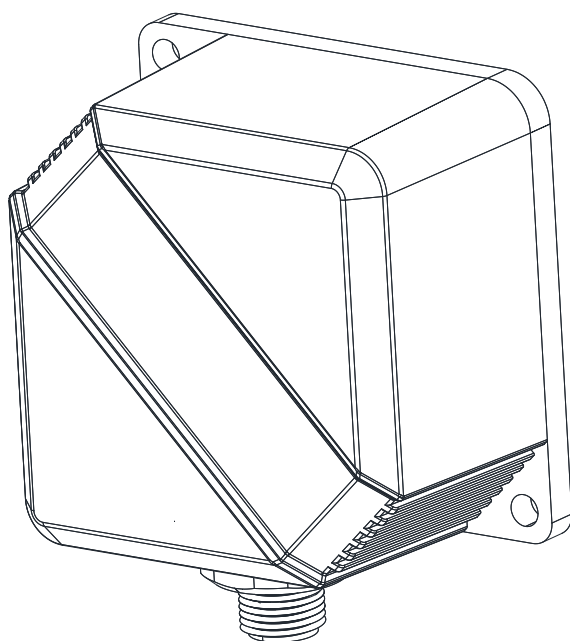


Modbus-RTU 总线 RFID 产品使用手册





前言

- 感谢您购买了华茂欧特产品。
- 本手册主要介绍 RFID 系列 Modbus RTU 协议高频读写器产品的参数及使用。
- 在使用产品前，需详细阅读本手册，在充分理解其功能和性能的前提下完成系统构建，发挥其优越性能。

使用须知

- 使用产品需具备一定电气知识的专业人员才可以对产品进行接线等其他操作。
- 对产品进行操作，需遵守手册进行。
- 将产品组合使用时，请确认规格是否可以组合。

手册获取

- 登录华茂欧特官网（www.wellauto.cn）→服务与支持→资料下载，查找所需产品资料并进行下载。
- 通过华茂欧特知识库对所需资料进行下载。

联系方式

- 技术与服务热线：400-900-8687
- 传真：0755-27673307 0755-26078683
- Email: market@wellauto.cn
- 网址: www.wellauto.cn
- 地址：深圳市宝安区航城街道奋达西乡科技创新园 C 栋 4 楼

安全注意事项（使用前请务必阅读）

- 本章对所需关注的安全注意事项进行说明，为了您的人身安全以及避免财产损失，请在熟悉了所有关于设备的指示、安全信息，以及注意事项后使用。
 - 即使是[注意]中所标注的事项，根据状况的不同也可能导致重大事故的发生。
 - 在产品使用过程中易引发的问题在安全事项中有标注，未进行标注的事项，请遵守基本的电气操作进行。
 - 在使用产品过程中，如果未以制造商指定的方式使用设备，可能有损设备所提供的保护。

在安全注意事项章节中使用[提示]、[注意]来注明：

 提示：	对操作的描述进行必要的补充或说明
 注意：	错误使用时，可能会产生危险，导致轻微身体伤害或设备损坏。

产品的收货



注意：

- ① 开箱前请检查外包装是否完整，是否有破损、浸湿、变形等情况。
- ② 请按照顺序依次打开包装，切勿暴力拆包。
- ③ 请检查产品表面是否有碰伤，腐蚀等情况。
- ④ 根据装箱清单仔细查看产品是否与购买的型号一致及附件数量、资料是否齐全。

产品的安装



注意：

- ① 安装前请仔细阅读产品使用说明书及安全注意事项。
- ② 请勿在下列场所使用产品：有灰尘、油烟、导电性尘埃、腐蚀性气体、可燃性气体的场所；暴露于高温、结露、风雨的场合；有振动、冲击的场合。电击、火灾、误操作也会导致产品损坏和恶化。
- ③ 安装时需做好防护，否则可能引发触电的危险。
- ④ 进行螺丝孔的加工时，需将产品遮盖，防止粉末、电线碎屑掉等异物入产品内导致产品故障，相关作业结束后，需将遮盖物撤掉，以免影响产品散热。
- ⑤ 在使用扩展模块时需确认线缆连接紧密、接触良好，否则会导致通讯不良，影响使用。

内部资料，请勿外传

产品内容如有变动，恕不另行通知



产品的接线

 提示:

- ① 接线端子电缆仅适用于铜芯电缆。
- ② 请根据手册接线图进行接线，若接错电源可能会导致产品故障。
- ③ 使用电线连接端子时，请一定要拧紧，且不可将导电部分触碰到其他电线或端子，有可能会使产品损坏。
- ④ 接线时，应在确认接口类型的基础上进行操作，如果连接到不相同的接口上或配线错误，可能导致模块、外部设备故障。

 注意:

- ① 在对产品进行接线操作前，需将外部电源断开，否则会有触电的危险。
- ② 进行产品接线时需经过电气设备培训、有充分电气知识的专业人员进行操作。
- ③ 线缆端子应做好绝缘，确保线缆安装到端子台后，线缆之间的绝缘距离不会减少。否则会导致 触电或者设备损坏。
- ④ 进行模块配线时，应确认产品的额定电压及信号排列后再进行操作，连接与额定值不同的电源或配线错误，会导致故障或火灾。

产品的运行及维护

 注意:

- ① 对产品上电后，请勿触碰端子，会有触电危险。
- ② 请勿对带电的产品进行接线、拆线等操作，会有触电危险。
- ③ 请勿私自拆卸、组装、更改本产品，有可能导致产品加速老化或直接损坏



目 录

1. 产品简介	1 -
1.1. 产品型号表	1 -
2. 产品参数	2 -
3. 产品说明	3 -
3.1. 模块示意图	3 -
3.2. 模块接线说明	4 -
3.2.1. AU3 IHR-3004-COM 接线说明	4 -
3.3. M12 接线端子说明	5 -
3.3.1. AU3 IHR-3004-COM 接线端子定义	5 -
3.4. 模块指示灯说明	5 -
3.5. 通讯协议	6 -
3.5.1. 默认波特率/校验方式	6 -
3.5.2. 默认站号 ID	6 -
3.5.3. 地址	6 -
3.5.4. 功能码	6 -
3.5.5. 数据区	6 -
3.5.6. CRC 校验码	7 -
4. 使用说明	8 -
4.1. 通讯使用说明	8 -
4.1.1. 通讯连接	8 -
4.1.2. 硬件配置	8 -
4.1.3. 建立连接	9 -
4.2. 读写标签操作说明及示例	11 -
4.2.1. 功能码 0x03 读标签操作	11 -
4.2.2. 功能码 0x06/0x10 写标签操作	19 -
4.3. 读写器的配置寄存器详细说明和示例	23 -
4.3.1. 站号 ID 寄存器(0x0000)	23 -
4.3.2. 波特率寄存器(0x0001)	25 -
4.3.3. 校验位寄存器(0x0002)	28 -
4.3.4. 射频信号开启/关闭寄存器(0x0003)	30 -
4.3.5. 射频功率输出寄存器 (0x0004)	32 -
4.3.6. 标签状态寄存器 (0x0008)	34 -



4.3.7. 错误代码寄存器 (0x000A).....- 35 -

4.3.8. 标签用户数据 Word 高低交换寄存器(0x000B).....- 36 -

4.3.9. 标签格式化寄存器(0x000C).....- 38 -

4.3.10. 读写器复位重启寄存器(0x000F).....- 39 -

4.3.11. 读版本号寄存器(0x0012).....- 40 -

4.3.12. 修改站号、波特率、校验位保存使能(0x001D).....- 41 -

附表一 读写器寄存器地址分配表.....- 42 -

附表二 读写器寄存器和西门子 PLC Modubs 功能块映射关系.....- 46 -

附表三 错误代码表.....- 47 -

附表四 CRC 算法.....- 48 -



手册版本	说明
V1.0	初始版本
V1.1	删除 AU3 IHR-3004-232/AU3 IHR-3004-485 模块说明，新增 AU3 IHR-3004-COM 模块说明，更新模块接线图。



1. 产品简介

AU3 IHR 30xx-COM 系列读写器是基于射频识别技术的高频 RFID 标签读写器，采用 Modbus RTU 通信协议、支持 RS485/232 通讯接口，1.5KBytes/S 的高速标签读写速度，具有丰富的诊断与 LED 功能，创新与实用的 RFID 技术能很好的应用于制造与物流业。

1.1. 产品型号表

订货号	产品规格
AU3 IHR-3004-COM	Modbus RTU、ASCII 自定义总线、RS232/RS485 接口，最大垂直检测距离 100mm。



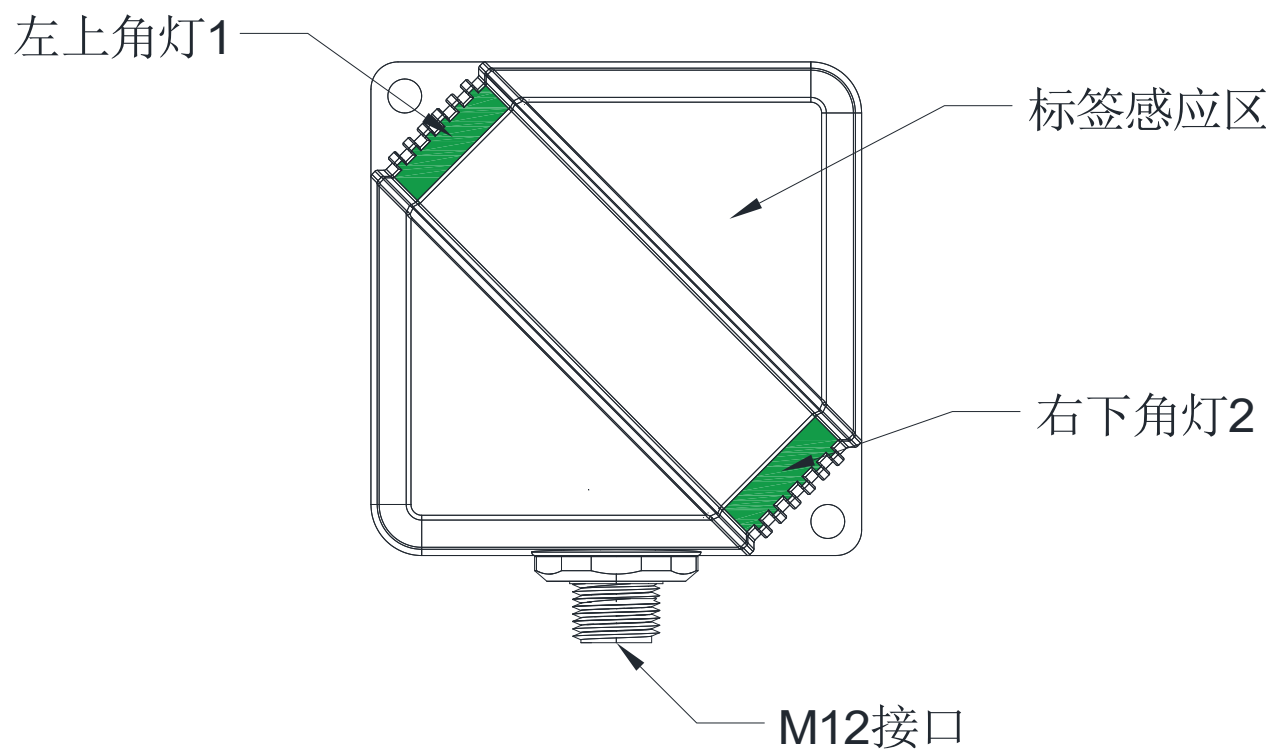
2. 产品参数

产品型号	AU3 IHR-3004-COM
技术规格	
供电电压	18~36V DC
工作频率	13.56MHz
工作时最高功耗	1.2W
最大垂直检测距离	100mm
天线	内置集成
通信接口	RS232/RS485
标签读写无线传输率 (最高)	1500Bytes/S
允许读或写的最大标签 容量 (FRAM)	8000Bytes
通讯协议	Modbus RTU、ASCII 自定义
支持标签类型	I-CODE2、I-CODE SLI, 等符合 ISO15693 通讯协议电子标签
空气协议	ISO15693、NFC、ISO14443A/B
可操作标签芯片	支持 TI、NXP、ST、INFINEON、FUJITSU、Philips 等
电气连接器设计 (供电和通讯口)	M12, 8Pin
显示指示	绿、蓝、红三色 LED 指示灯
电源保护	内置 ESD 与浪涌保护电路
EMC/EMI	符合 EN61131-2/EN50082-2
ESD 性能	空气放电±15KV,接触放电±8KV
材料	PC+ABS+工业胶
IP 等级	IP67
抗冲击性符合	50g
抗振动性符合	20g
工作时允许的环境温度	-20 ... +70 °C
存储时允许的环境温度	-25 ... +85 °C
重量(g)	130g
安装方式	M5 螺栓*4
跌落测试	1.2 米自由跌落
尺寸 (长×宽×高)	58×58×36 (mm)



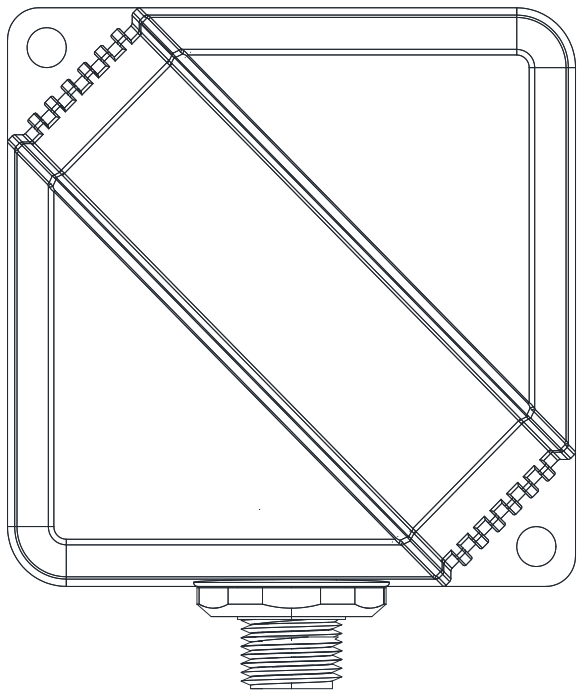
3. 产品说明

3.1. 模块示意图

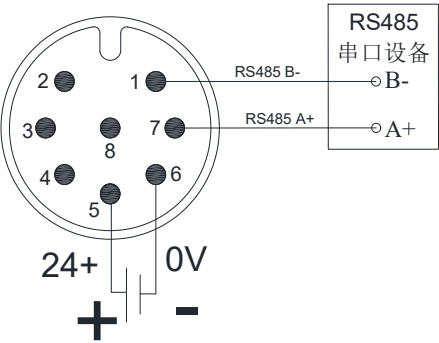


3.2. 模块接线说明

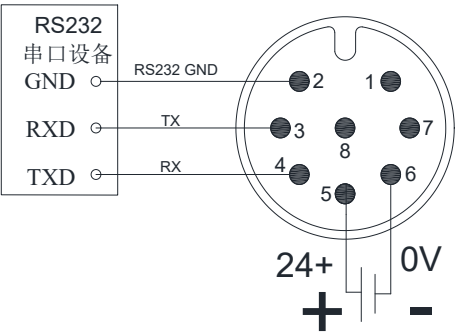
3.2.1. AU3 IHR-3004-COM 接线说明



RS485接线



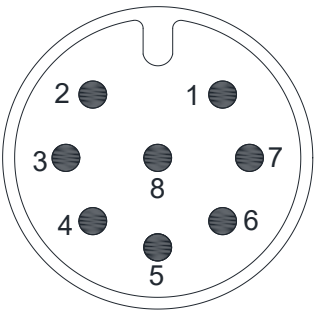
RS232接线



PIN8脚为悬空

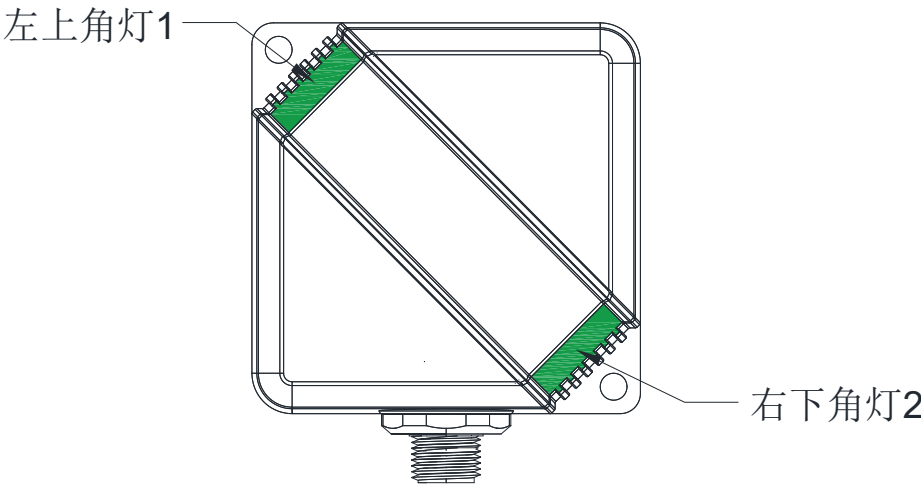
3.3. M12 接线端子说明

3.3.1. AU3 IHR-3004-COM 接线端子定义

端子示意图	编号	说明
	1	RS485 B-
	2	RS232 GND
	3	TX
	4	RX
	5	供电电源 24V DC
	6	供电电源 GND
	7	RS485 A+
	8	NC

注：NC 表示为悬空脚

3.4. 模块指示灯说明

指示灯示意图	
	
指示灯	说明
绿灯点亮	未检测到标签卡
蓝灯点亮	检测到标签卡
红灯闪烁	读写器出现错误



3.5. 通讯协议

3.5.1. 默认波特率/校验方式

- 115200
- 无校验

3.5.2. 默认站号 ID

- 2

在 Modbus RTU 通讯模式下，主站需要通过时间来判断 RTU 帧的开始和结束。主站监视总线上的通讯数据，如果发现总线有超过 4.5 个字符时间（时间绝对值随字符传送波特率的变化而变化）的空闲，则认为一个帧已经结束（如果存在），同时认为下一个字符为下一帧的开始。

完整的 RTU 帧应该是下面的格式（不论是主站发起还是从站应答）：

空闲	地址	功能码	数据	CRC	空闲
---	1 字节	1 字节	N 字节	2 字节	---

3.5.3. 地址

Modbus 协议的站地址由一个字节组成，站地址用来指示哪个从设备来应答主站的通讯报文。在总线上，每个从设备必须指定一个唯一的站地址，只有当通讯报文中地址与该从设备地址相同时，该从设备才能应答主站的通讯报文。从设备应答的通讯报文也必须包含该地址，以告知主站，这个通讯报文是哪个从设备应答的。广播报文的地址是零，所有的从站可以根据广播报文进行相应的动作，但是一般不能应答该广播报文。

3.5.4. 功能码

功能码指示从设备执行相应的操作，从设备回复相应的数据。做为从站，使用了部分常用的 Modbus RTU 通讯功能码：

- 1、0x03 读保持寄存器（Read holding register）
- 2、0x06 写单个寄存器（Preset single register）
- 3、0x10 写多个寄存器（Preset multiple registers）

3.5.5. 数据区

数据区是主站需要发送给从站的数据，或者是从站需要返回给主站的数据。数据的具体含义由功能码来定义。有些功能码不包含数据区，数据区大小可以为 0x00。



3.5.6. CRC 校验码

CRC 校验码让接收数据方来检查通讯的传输过程中是否有错误发生。

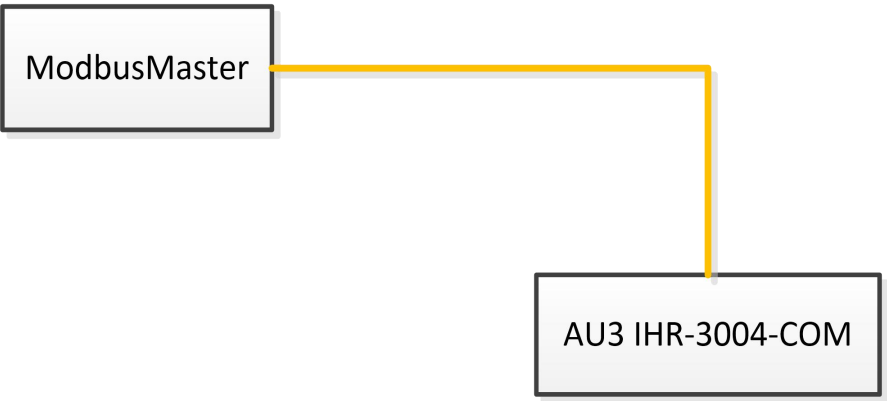
在 Modbus RTU 中，RTU 模式必须采用 CRC16 校验码。在单片机中实现一般有两种方法，查表法或者运算法，实现方法可以查阅相关的资料。

4. 使用说明

4.1. 通讯使用说明

4.1.1. 通讯连接

通讯连接示意图，如下图所示：



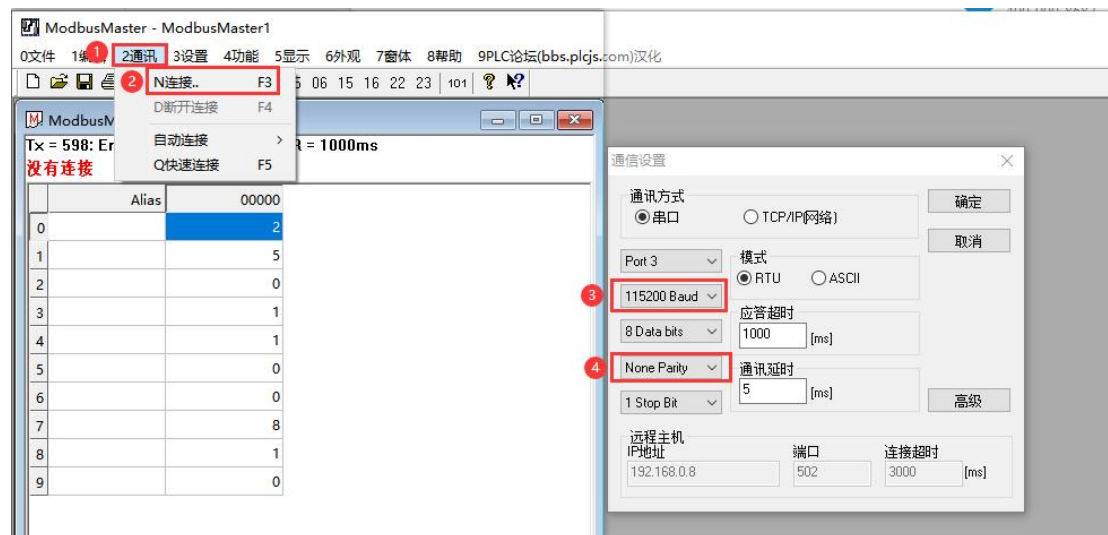
4.1.2. 硬件配置

硬件配置如下表所示：

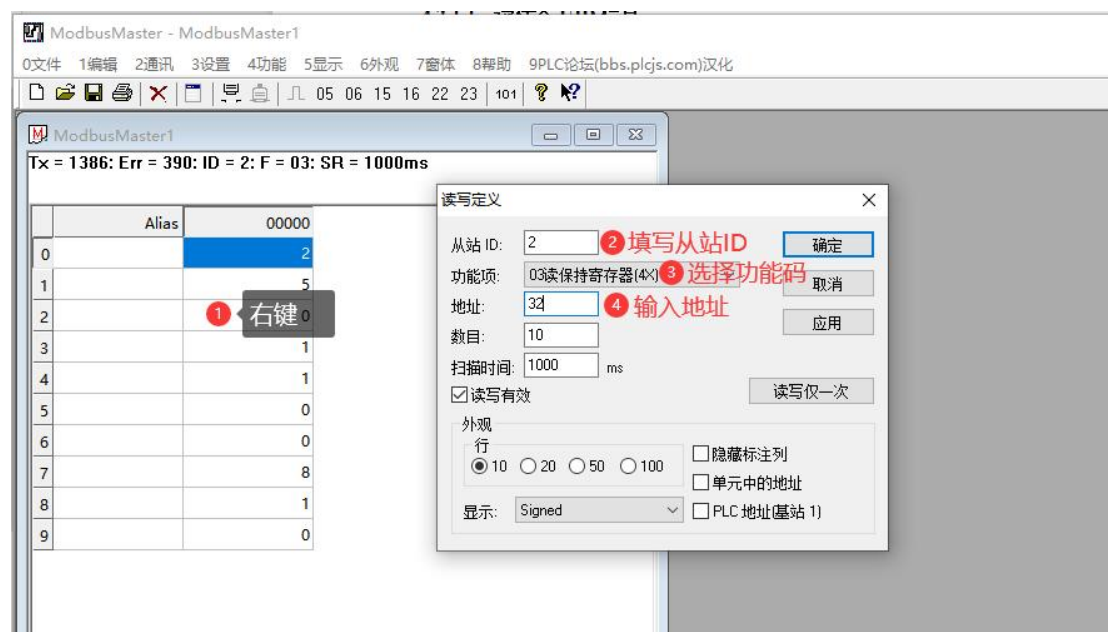
硬件	数量	备注
编程电脑	1 台	ModbusMaster
AU3 IHR-3004-COM	1 个	读写器
通讯网线	1 条	
电源	1 个	

4.1.3. 建立连接

(1) 打开软件，通过串口方式进行连接，主站波特率及校验方式需与从站读写器的一致，否则无法通讯。



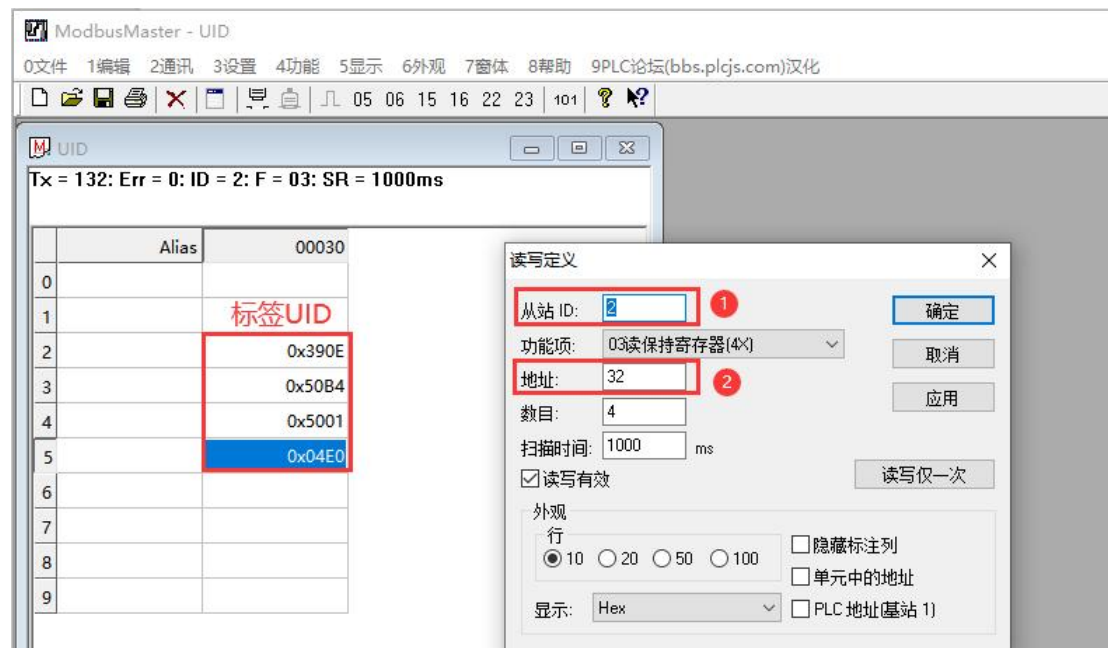
(2) 设置地址读写器默认从站 ID 为 2



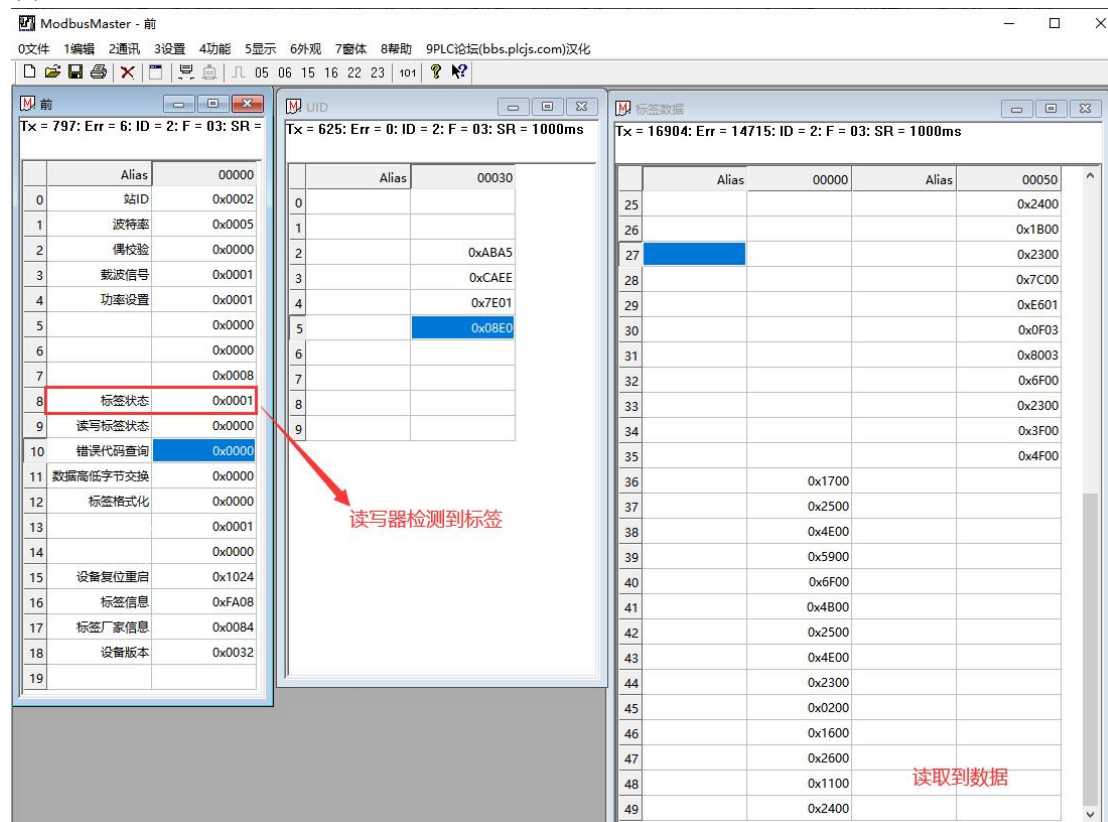
4.1.3.1. 数据监控

可参考附录一，读写器寄存器地址分配表，写入相关地址对标签进行数据的查看及读写

- (1) 读取标签 UID 信息，在读写定义中写入读取 UID 信息的地址，可读取标签的 UID 信息



- (2) 读取标签数据





4.2. 读写标签操作说明及示例

4.2.1. 功能码 0x03 读标签操作

4.2.1.1. 读标签 UID 信息

标签 UID 信息在读写器的寄存器
0x0020(32),0x0021(33),0x0022(34),0x0023(35) 4 个 (Word) 寄存器中。

备注:

- (1) 读写器接收到主机命令后，在有标签且是亮蓝灯时蓝灯会快闪一下。
- (2) 读标签 UID 信息只能从地址 0x0020 开始，连续读 4 个 (Word) 才能读到一串完整的 8 个 Byte 的标签 UID。
- (3) 无标签时返回的是 8 个 Byte 0x00，且读写器会闪红灯报错。

范例：以广播方式读取标签 UID 信息。

→主机发送：00 03 00 20 00 04 44 12

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址 (高字节)	00
开始地址 (低字节)	20
数量 (高字节)	00
数量 (低字节)	04
CRC校验 (高字节)	44
CRC校验 (低字节)	12

← 读写器应答 (有标签返回 UID)： 02 03 08 BC AF 1D 0B 08 01 04 E0
38 1C

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数	08
数据 (寄存器0x0020 高字节)	BC



数据（寄存器0x0020 低字节）	AF
数据（寄存器0x0021 高字节）	1D
数据（寄存器0x0021 低字节）	0B
数据（寄存器0x0022 高字节）	08
数据（寄存器0x0022 低字节）	01
数据（寄存器0x0023 高字节）	04
数据（寄存器0x0023 低字节）	E0
CRC校验（高字节）	38
CRC校验（低字节）	1C

← 读写器应答（无标签返回 8 个 Byte 0x00）：02 03 08 00 00 00 00 00 00
00 00 9A 93

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数	08
数据（寄存器0x0020 高字节）	00
数据（寄存器0x0020 低字节）	00
数据（寄存器0x0021 高字节）	00
数据（寄存器0x0021 低字节）	00
数据（寄存器0x0022 高字节）	00
数据（寄存器0x0022 低字节）	00
数据（寄存器0x0023 高字节）	00
数据（寄存器0x0023 低字节）	00
CRC校验（高字节）	9A
CRC校验（低字节）	93

备注 1：读 UID 的 Word 长度大于 4 将会以默认的 4 读取。

备注 2：读写器收到读标签 UID 命令后，感应区范围内无卡，读写器将闪一次红灯。

4.2.1.2. 读标签用户区数据

标签用户区数据保存在读写器的寄存器 0x0024(36)开始的 N 个寄存器中。0x0024 的地址就是对应标签的起始地址 0x0000。（详细见附表：读写器寄存器地址分配表）。

范例：以广播方式读取寄存器 0x0024 开始的 8 个 Word 用户数据（1 个 Word=2 个 Byte）。也就是读标签 0x0000 开始的 16 个 Byte 用户数据。

→主机发送：00 03 00 24 00 08 05 D6

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	24
数量（高字节）	00
数量（低字节）	08
CRC校验（高字节）	05
CRC校验（低字节）	D6

← 读写器应答（有标签）：02 03 10 99 99 99 99 99 99 99 99 99 99 99 99 99 99 99 9F 18

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数（Byte）	10
数据（寄存器0x0024 高字节）	99
数据（寄存器0x0024 低字节）	99
数据（寄存器0x0025 高字节）	99
数据（寄存器0x0025 低字节）	99
数据（寄存器0x0026 高字节）	99
数据（寄存器0x0026 低字节）	99
数据（寄存器0x0027 高字节）	99



数据（寄存器0x0027 低字节）	99
数据（寄存器0x0028 高字节）	99
数据（寄存器0x0028 低字节）	99
数据（寄存器0x0029 高字节）	99
数据（寄存器0x0029 低字节）	99
数据（寄存器0x002A 高字节）	99
数据（寄存器0x002A 低字节）	99
数据（寄存器0x002B 高字节）	99
数据（寄存器0x002B 低字节）	99
CRC校验（高字节）	9F
CRC校验（低字节）	18

← 读写器应答（无标签）：02 83 07 00 01 04 45

应答	
段名	例子 (HEX)
站号ID	02
错误功能码	83
字节数（本帧总字节数）	07
数据（错误代码高字节）	00
数据（错误代码低字节）	01
CRC校验（高字节）	04
CRC校验（低字节）	45

备注：读写器收到读标签的用户数据命令后，感应区范围内无卡，读写器将闪一次红灯和返回错误代码，具体见附表三，错误代码表。

4.2.1.3. 查询上次读写标签的状态(寄存器 0x0009)

- (1) 每次进行读写标签操作时，读写状态均保存在0x0009寄存器中，可通过查询0x0009寄存器内容获取上一次的读写标签操作状态。
- (2) 包括读标签，写标签，格式化标签。相关和标签操作等。
- (3) 上电默认寄存器 0x0009 的值是 0x0000。
- (4) 返回状态值是上一次主机发送执行读写标签操作的执行情况，该寄存器只读。执行下一条命令之后将更新状态值。
- (5) 写此寄存器的值=0x0000，将内容置 0x0000，写其他值无效。

范例：以广播方式读出标签读写状态寄存器0x0009的信息。

→主站发送：00 03 00 09 00 01 55 D9

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	09
数量（高字节）	00
数量（低字节）	01
CRC校验（高字节）	55
CRC校验（低字节）	D9

← 读写器正确响应：02 03 02 00 00 FC 44

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数	02
数据（高字节） 固定值00	00
数据（低字节） 状态值	00
CRC校验（高字节）	FC
CRC校验（低字节）	44



- 读写器应答（上一次读写标签操作无误）：02 03 02 00 00 FC 44。
- 读写器应答（上一次读标签操作出错）：02 03 02 00 01 3D 84。
- 读写器应答（上一次写标签操作出错）：02 03 02 00 02 7D 85。

备注：当读操作中包含读写状态寄存器时，返回的数值中，读写状态寄存器状态均为上一次读写操作（非读写操作查询命令操作）。

4.2.1.4. 读标签厂商信息寄存器(0x0010,0x0011)

- (1) 标签信息在寄存器0x0010(16), 0x0011(17) 2个寄存器中。
- (2) 0x0010(16)的高字节是块大小, 低字节是每块多少个字节。
- (3) 0x0011(17)的高字节无效, 低字节是标签厂商信息。
- (4) 此寄存器只读。

范例：以广播方式读出标签信息（从0x0010开始，读两个Word寄存器）。

→主站发送：00 03 00 10 00 02 C4 1F

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	10
数量（高字节）	00
数量（低字节）	02
CRC校验（高字节）	C4
CRC校验（低字节）	1F

← 读写器正确响应：02 03 04 FF 20 00 A0 F8 95（有标签）

02 03 04 00 00 00 00 C9 33（无标签）

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数 (Byte)	04
数据 (0x0010 高字节)	FF
数据 (0x0010 低字节)	20
数据 (0x0011 高字节)	00
数据 (0x0011 低字节)	A0
CRC校验（高字节）	F8
CRC校验（低字节）	95



备注 1: 寄存器 0x0010=0xFF20 0xFF=255（10 进制） 表示有 255 个块，
0x20=32（10 进制）表示每块是 32 个字节,255*32=8160（标签的容量）。

备注 2: 寄存器 0x0011=0x00A0 是厂商信息，具体见标签手册。



4.2.2. 功能码 0x06/0x10 写标签操作

4.2.2.1. 功能码(0x06)写单个数据到标签

读写器的寄存器 0x0024 对应标签的地址是 0x0000 到标签的最大容量（每个标签容量不一样），都可以用 0x06 命令进行写单个寄存器的操作。

范例：以广播方式写寄存器0x0024的数据内容为0x1122。

→主站发送：00 06 00 24 11 22 44 59

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	24
写入数据（高字节）	11
写入数据（低字节）	22
CRC校验（高字节）	44
CRC校验（低字节）	59

← 读写器正确响应（写入数据到标签成功）：02 06 00 24 11 22 45 BB

应答	
段名	例子 (HEX)
站号ID	02
功能码	06
写入地址（高字节）	00
写入地址（低字节）	24
写入数据（高字节）	11
写入数据（低字节）	22
CRC校验（高字节）	45
CRC校验（低字节）	BB



←读写器应答（无标签写入失败）：02 86 07 00 01 04 89（待PLC验证否能解释）

应答	
段名	例子 (HEX)
站号ID	02
错误功能码	86
字节数（本帧总字节数）	07
数据（错误代码高字节）	00
数据（错误代码低字节）	01
CRC校验（高字节）	04
CRC校验（低字节）	89

备注：读写器收到写标签的用户数据命令后，感应区范围内无卡，读写器将闪一次红灯和返回错误代码，具体见附表三，错误代码表。

4.2.2.2. 功能码（0x10）写多个数据到标签

标签用户区数据保存在0x0024(36)开始的寄存器。

范例：将数据12 34 56 78写入标签0x0024, 0x0025 两个（Word）寄存器中。

→主机发送：00 10 00 24 00 02 04 12 34 56 78 8F 8C

发送	
段名	例子 (HEX)
站号ID	00
功能码	10
起始地址（高字节）	00
起始地址（低字节）	24
寄存器数量（高字节）（Word）	00
寄存器数量（低字节）（Word）	02
字节数（Byte）	04
数据（寄存器0x0024 高字节）	12
数据（寄存器0x0024 低字节）	34
数据（寄存器0x0025 高字节）	56
数据（寄存器0x0025 低字节）	78
CRC校验（高字节）	8F
CRC校验（低字节）	8C

← 读写器应答（写数据到标签成功）：02 10 00 24 00 02 01 F0

应答	
段名	例子 (HEX)
站号ID	02
功能码	10
起始地址（高字节）	00
起始地址（低字节）	24
寄存器数量（高字节）	00
寄存器数量（低字节）	02
CRC校验（高字节）	01
CRC校验（低字节）	F0



←读写器应答（写数据到标签失败）：02 90 07 00 01 00 C1

应答	
段名	例子 (HEX)
站号ID	02
错误码（0x80+功能码）	90
字节数（应答返回的全部字节）	07
数据（自定义错误代码）	00
数据（自定义错误代码）	01
CRC校验（高字节）	00
CRC校验（低字节）	C1

备注：读写器收到写标签的用户数据命令后，感应区范围内无卡，读写器将闪一次红灯和返回错误代码，具体见附表三，错误代码表。

4.3. 读写器的配置寄存器详细说明和示例

读写器的配置寄存器在0x0000~0x001F(31)的 32个(Word)地址中（附表一读写器寄存器地址分配）。

4.3.1. 站号 ID 寄存器(0x0000)

- (1) 站号ID在寄存器0x0000中。
- (2) 站号ID=0x00或0xFF为广播地址。
- (3) 默认出厂设备站号ID是2，寄存器的值是0x0002。
- (4) 由寄存器0x001D使能保存生效。

4.3.1.1. 读站号 ID 寄存器

范例：以广播方式读出读写器的站号ID

→主站发送：00 03 00 00 00 01 85 DB

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	00
数量（高字节）	00
数量（低字节）	01
CRC校验（高字节）	85
CRC校验（低字节）	DB

← 读写器正确响应：02 03 02 00 02 7D 85

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数 (Byte)	02
数据（高字节）	00
数据（低字节）站号ID	02
CRC校验（高字节）	7D
CRC校验（低字节）	85



4.3.1.2. 设置站号 ID 寄存器

范例：以广播方式设置站号ID为2。

→主站发送：00 06 00 00 00 02 09 DA

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	00
写入数据（高字节）	00
写入数据（低字节）	02
CRC校验（高字节）	09
CRC校验（低字节）	DA

← 读写器正确响应：06 06 00 00 00 02 09 BC

应答	
段名	例子 (HEX)
站号ID	06
功能码	06
写入地址（高字节）	00
写入地址（低字节）	00
写入数据（高字节）	00
写入数据（低字节）	02
CRC校验（高字节）	09
CRC校验（低字节）	BC

备注：设置新地址后读写器先按原地址返回响应值，而后新地址自动生效。

本范例的读写器原地址是0x06，要写入的新地址是0x02。



4.3.2. 波特率寄存器(0x0001)

- (1) 读写器的波特率在寄存器0x0001中。
- (2) 默认的出厂值是 115200, 寄存器的值是0x0005。
- (3) 波特率的范围是4800,9600,19200,38400,57600,115200,共6种。
- (4) 由寄存器0x001D使能保存生效。

波特率	4800	9600	19200	38400	57600	115200
对应寄存器值	0x0000	0x0001	0x0002	0x0003	0x0004	0x0005

写入到寄存器0x0001的值大于5都是默认为5。



4.3.2.1. 读波特率寄存器

范例：以广播方式读出波特率的对应数据值。

→主站发送：00 03 00 01 00 01 D4 1B

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	01
数量（高字节）	00
数量（低字节）	01
CRC校验（高字节）	D4
CRC校验（低字节）	1B

← 读写器正确响应：02 03 02 00 05 3C 47

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数（Byte）	02
数据（高字节）	00
数据（低字节）	05
CRC校验（高字节）	3C
CRC校验（低字节）	47

4.3.2.2. 设置波特率寄存器

范例：以广播方式设置读写器的波特率是19200，写入对应寄存器的值是0x0002。

→主站发送：00 06 00 01 00 02 58 1A

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	01
写入数据（高字节）	00
写入数据（低字节）	02
CRC校验（高字节）	58
CRC校验（低字节）	1A

← 读写器正确响应：02 06 00 01 00 02 59 F8

应答	
段名	例子 (HEX)
站号ID	02
功能码	06
写入地址（高字节）	00
写入地址（低字节）	01
写入数据（高字节）	00
写入数据（低字节）	02
CRC校验（高字节）	59
CRC校验（低字节）	F8

备注：设置新波特率后设备先按原波特率返回响应值，而后新波特率自动生效。发送主机需要同步更新波特率，否则不能正常通信。

4.3.3. 校验位寄存器(0x0002)

- (1) 读写器的偶校验在寄存器0x0002中。
- (2) 默认的出厂值是 无校验, 寄存器的值是0x0000。
- (3) 校验方式目前分为无校验和偶校验二种。
- (4) 由寄存器0x001D使能保存生效。

校验方式	无校验	偶校验
对应寄存器值	0x0000	0x0002

写入到寄存器0x0002的值非0x0000或0x0002都是默认为无校验方式。

4.3.3.1. 读校验位寄存器

范例：以广播方式读出校验方式对应数据值。

→主站发送：00 03 00 02 00 01 24 1B

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址 (高字节)	00
开始地址 (低字节)	02
数量 (高字节)	00
数量 (低字节)	01
CRC校验 (高字节)	24
CRC校验 (低字节)	1B

← 读写器正确响应：02 03 02 00 00 FC 44

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数 (Byte)	02
数据 (高字节)	00
数据 (低字节)	00
CRC校验 (高字节)	FC
CRC校验 (低字节)	44



4.3.3.2. 设置校验位寄存器

范例：以广播方式设置读写器的校验方式为无校验，写入对应寄存器的值是0x0000。

→主站发送：00 06 00 02 00 00 29 DB

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	02
写入数据（高字节）	00
写入数据（低字节）	00
CRC校验（高字节）	29
CRC校验（低字节）	DB

← 读写器正确响应：02 06 00 02 00 00 28 39

应答	
段名	例子 (HEX)
站号ID	02
功能码	06
写入地址（高字节）	00
写入地址（低字节）	02
写入数据（高字节）	00
写入数据（低字节）	00
CRC校验（高字节）	28
CRC校验（低字节）	39

备注：设置新校验方式后设备先按原校验方式返回响应值，而后新校验位自动生效。发送主机需要同步更新校验方式，否则不能正常通信。

4.3.4. 射频信号开启/关闭寄存器(0x0003)

- (1) 射频信号开启/关闭在寄存器0x0003中。
- (2) 上电默认是射频信号开启，寄存器的值是0x0001。
- (3) 对应值

射频信号开启/关闭	开启	关闭
对应寄存器值	0x0001	0x0000
读写器灯的状态	绿灯（无标签时）	闪绿灯

写入到寄存器0x0003的值非0x0000或0x0001都是默认为0x0001。

4.3.4.1. 读射频信号开启/关闭状态寄存器

范例：以广播方式读出射频信号状态对应数据值。

→主站发送：00 03 00 03 00 01 75 DB

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	03
数量（高字节）	00
数量（低字节）	01
CRC校验（高字节）	75
CRC校验（低字节）	DB

← 读写器正确响应：02 03 02 00 01 3D 84

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数 (Byte)	02
数据（高字节）	00
数据（低字节）	01
CRC校验（高字节）	3D
CRC校验（低字节）	84



4.3.4.2. 设置射频信号开启/关闭状态寄存器

范例：以广播方式设置读写器的射频信号为开启状态，写入对应寄存器的值是0x0001。

→主站发送：00 06 00 03 00 01 B9 DB

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	03
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	B9
CRC校验（低字节）	DB

← 读写器正确响应：02 06 00 03 00 01 B8 39

应答	
段名	例子 (HEX)
站号ID	02
功能码	06
写入地址（高字节）	00
写入地址（低字节）	03
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	B8
CRC校验（低字节）	39

4.3.5. 射频功率输出寄存器（0x0004）

(1) 射频功率输出在寄存器0x0004中。

(2) 上电默认值是 100%射频功率输出，寄存器的值是0x0001。

射频功率输出	100%	75%
对应寄存器值	0x0001	0x0000

写入到寄存器0x0004的值非0x0000或0x0001都是默认为0x0001。

4.3.5.1. 读射频功率输出寄存器

范例：以广播方式读出射频功率输出对应数据值。

→主站发送：00 03 00 04 00 01 C4 1A

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	04
数量（高字节）	00
数量（低字节）	01
CRC校验（高字节）	C4
CRC校验（低字节）	1A

← 读写器正确响应：02 03 02 00 01 3D 84

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数 (Byte)	02
数据（高字节）	00
数据（低字节）	01
CRC校验（高字节）	3D
CRC校验（低字节）	84

4.3.5.2. 设置射频功率输出寄存器

范例：以广播方式设置读写器的射频功率输出为100%，写入对应寄存器的值是0x0001。

→主站发送：00 06 00 04 00 01 08 1A

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	04
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	08
CRC校验（低字节）	1A

← 读写器正确响应：02 06 00 04 00 01 09 F8

应答	
段名	例子 (HEX)
站号ID	02
功能码	06
写入地址（高字节）	00
写入地址（低字节）	04
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	09
CRC校验（低字节）	F8



4.3.6. 标签状态寄存器（0x0008）

- (1) 标签状态在寄存器0x0008中。
- (2) 上电默认0，寄存器的值是0x0000。
- (3) 寄存器只读。

4.3.6.1. 读标签状态寄存器

范例：以广播方式读出标签状态寄存器数据值。

→主站发送：00 03 00 08 00 01 04 19

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	08
数量（高字节）	00
数量（低字节）	01
CRC校验（高字节）	04
CRC校验（低字节）	19

← 读写器正确响应：02 03 02 00 01 3D 84（有标签）

02 03 02 00 00 FC 44（无标签）

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数（Byte）	02
数据（高字节）	00
数据（低字节）	01
CRC校验（高字节）	3D
CRC校验（低字节）	84



4.3.7. 错误代码寄存器 (0x000A)

- (1) 读写器的错误代码在寄存器0x000A(10)中。
- (2) 上电默认0，寄存器的值是0x0000。
- (3) 写此寄存器的值=0，将清除错误代码值，写其他值无效。
- (4) 详细见附件三：错误代码表

4.3.7.1. 读错误代码寄存器

范例：以广播方式读错误代码数据值。

→主站发送：00 03 00 0A 00 01 A5 D9

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址 (高字节)	00
开始地址 (低字节)	0A
数量 (高字节)	00
数量 (低字节)	01
CRC校验 (高字节)	A5
CRC校验 (低字节)	D9

← 读写器正确响应：02 03 02 00 00 FC 44

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数 (Byte)	02
数据 (高字节)	00
数据 (低字节)	00
CRC校验 (高字节)	FC
CRC校验 (低字节)	44



4.3.8. 标签用户数据 Word 高低交换寄存器(0x000B)

- (1) 标签用户数据Word高低交换寄存器0x000B(11)中。
- (2) 默认的出厂是标签用户数据Word高低不交换，寄存器的值是0x0000。
- (3) 只是针对用户区数据的Word 字节的高低交换，下例是实例对比

不交换：02 03 14 11 22 11 22 11 22 11 22 11 22 11 22 11 22 11 22 11 22 99 99 6A 67

交换：02 03 14 22 11 22 11 22 11 22 11 22 11 22 11 22 11 22 11 22 11 99 99 91 2B

4.3.8.1. 读标签用户数据 Word 高低交换寄存器

范例：以广播方式读标签用户数据Word高低交换寄存器。

→主站发送： 00 03 00 0B 00 01 F4 19

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	0B
数量（高字节）	00
数量（低字节）	01
CRC校验（高字节）	F4
CRC校验（低字节）	19

← 读写器正确响应： 02 03 02 00 00 FC 44

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数（Byte）	02
数据（高字节）	00
数据（低字节）	00
CRC校验（高字节）	FC
CRC校验（低字节）	44



4.3.8.2. 设置标签用户数据 Word 高低交换寄存器

范例：以广播方式设置标签用户数据Word高低交换 写入对应寄存器的值是0x0001。

→主站发送：00 06 00 0B 00 01 38 19

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	0B
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	38
CRC校验（低字节）	19

← 读写器正确响应：02 06 00 0B 00 01 39 FB

应答	
段名	例子 (HEX)
站号ID	02
功能码	06
写入地址（高字节）	00
写入地址（低字节）	0B
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	39
CRC校验（低字节）	FB



4.3.9. 标签格式化寄存器(0x000C)

- (1) 标签格式化在寄存器0x000C(12)中。
- (2) 放置好标签到感应区后才能发送标签格式化命令，格式化过程中读写器的绿灯在闪的过程中标签不能移开，否则标签不能完全被格式化。
- (3) 此寄存器只能写，

4.3.9.1. 标签格式化寄存器

范例：以广播方式格式化标签的值全为0x01， 写入对应寄存器的值是0x0001。

→主站发送：00 06 00 0C 00 01 89 D8

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	0C
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	89
CRC校验（低字节）	D8

← 读写器正确响应：02 06 00 0C 00 01 88 3A

应答	
段名	例子 (HEX)
站号ID	02
功能码	06
写入地址（高字节）	00
写入地址（低字节）	0C
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	88
CRC校验（低字节）	3A



4.3.10. 读写器复位重启寄存器(0x000F)

- (1) 读写器的复位重启在寄存器0x000F(15)。
- (2) 写入0x0001是设备复位重启。
- (3) 此寄存器只能写。

4.3.10.1. 设置读写器复位重启寄存器

范例：以广播方式启动读写器重新复位上电， 写入对应寄存器的值是0x0001。

→主站发送：00 06 00 0F 00 01 79 D8

发送	
段名	例子 (HEX)
站号ID	00
功能码	06
写入地址（高字节）	00
写入地址（低字节）	0F
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	79
CRC校验（低字节）	D8

← 读写器正确响应：02 06 00 0F 00 01 78 3A

应答	
段名	例子 (HEX)
站号ID	02
功能码	06
写入地址（高字节）	00
写入地址（低字节）	0F
写入数据（高字节）	00
写入数据（低字节）	01
CRC校验（高字节）	78
CRC校验（低字节）	3A

备注：读写器返回响应后立即重启。



4.3.11. 读版本号寄存器(0x0012)

- (1) 版本号在寄存器0x0012(18)中。
- (2) 此寄存器只读。

4.3.11.1. 读版本号

范例：以广播方式读读写器版本号

→主站发送：00 03 00 12 00 01 25 DE

发送	
段名	例子 (HEX)
站号ID	00
功能码	03
开始地址（高字节）	00
开始地址（低字节）	12
数量（高字节）	00
数量（低字节）	01
CRC校验（高字节）	25
CRC校验（低字节）	DE

← 读写器正确响应：02 03 02 00 10 FD 88

应答	
段名	例子 (HEX)
站号ID	02
功能码	03
字节数 (Byte)	02
数据（高字节）	00
数据（低字节）	10
CRC校验（高字节）	FD
CRC校验（低字节）	88

备注：数据 0x0010 代表版本号是V1.0。

**4.3.12. 修改站号、波特率、校验位保存使能(0x001D)**

- (1) 修改站号、波特率、校验位保存使能在寄存器0x001D(29)中。
- (2) 当修改站号、波特率、校验位后，将此寄存器后，将此值由 **0** 置 **1** 后，才能保存生效，站号、波特率、校验位保存后即时生效。
- (3) 此寄存器只能写。

附表一 读写器寄存器地址分配表

寄存器描述(每个寄存器占用 1 个 Word, 2 个字节 (Byte), 适用 Modbus RTU)

地址	寄存器含义	说明	可否读写	备注
0x0000(0)	站号 ID	0x00,0xFF 广播地址	R/W	掉电保存。 设置新地址后设备先按原地址返回响应值, 而后新地址自动生效。
0x0001(1)	波特率	0x0000: 4800 0x0001: 9600 0x0002: 19200 0x0003: 38400 0x0004: 57600 0x0005: 115200 (默认) 其他值: 按默认值	R/W	掉电保存。 设置新波特率后设备先按原波特率返回响应值, 而后新波特率自动生效。
0x0002(2)	偶校验	0x0000: 无校验, 1 个停止位 (默认) 0x0001: 保留 0x0002: 偶校验, 1 个停止位 其他值: 按默认值	R/W	掉电保存。 设置新校验后设备先按原校验返回响应值, 而后新校验自动生效。
0x0003(3)	载波信号	0x0000: 关闭载波输出, 0x0001: 开启载波输出 (默认) 其他值: 按默认值	R/W	掉电不保存,复位上电按默认值。 更改后立即生效, 关闭载波输出时绿灯会 0.5S 间隔一直闪。
0x0004(4)	功率设置	0x0000: 75% 0x0001: 100%(默认) 其他值: 按默认值	R/W	掉电不保存,复位上电按默认值。
0x0005(5)	保留	---	---	---
0x0006(6)	保留	---	---	---



0x0007(7)	保留	---	---	---
0x0008(8)	标签状态寄存器	0x0000: 感应区无标签 0x0001: 感应区有标签	R	掉电不保存,复位上电值 0x0000。 写入寄存器会报错(见错误故障代码) (感应区当前是否有标签)
0x0009(9)	读写标签状态寄存器	上一次的读写标签操作 0x0000: 无错误 0x0001: 读错误 0x0002: 写错误	R	掉电不保存,复位上电值 0x0000。 读写状态寄存器用于显示读写标签操作后的状态变化,该寄存器只读。执行下一条命令之后将更新状态值。 写入寄存器会报错(见错误故障代码)
0x000A(10)	错误代码查询	0x0000: 无错误 0x0001: 射频区域没有标签 0x0002: 超出标签的地址 0x0003: 读写过程中失败(或受干扰) 0x0004: 获取标签信息失败 0x0005: 超出单次读写数据的长度 0x0006: 寄存器不可以写	R/W	掉电不保存,复位上电值 0x0000。 W: 写入 0x0000 将使本寄存器的值值 0,执行下一条命令之后将更新状态值。
0x000B(11)	数据高低字节交换	0x0000: 不翻转(默认值) 0x0001: 翻转 其他按默认值	R/W	掉电保存。 默认高位在前,低位在后排序。比方某个寄存器是: 0x4567,读取时: 高位在前,0x45, 0x67。如果设置为翻转低位在前,那么就是 0x67,



				0x45。
0x000C(12)	标签格式化	只取低字节,高字节无效 例如: 写入 0x0800 数据全部格式化为 0x00 写入 0x0809 数据全部格式化为 0x09	W	掉电不保存,复位上电值 0x0000。 标签格式化时间需要 2 秒左右时间,读写器的绿灯会闪耀指示格式化中。(中途移开标签,格式化失败)。
0x000D(13)	保留	---	---	---
0x000E(14)	保留	---	---	---
0x000F(15)	设备复位重启	0x0001: 设备复位重启	W	掉电不保存。 复位上电值 0x0000。
0x0010(16)	标签信息	高字节是块大小,低字节是每块多少个字节.	R	掉电不保存。 复位上电值 0x0000。
0x0011(17)	标签信息	标签厂家信息	R	掉电不保存,复位上电值 0x0000。
0x0012(18)	设备版本	设备版本	R	掉电保存
0x0013(19)	保留	---	---	---
0x0014(20)	保留	---	---	---
0x0015(21)	保留	---	---	---
0x0016(22)	保留	---	---	---
0x0017(23)	保留	---	---	---
0x0018(24)	保留	---	---	---
0x0019(25)	保留	---	---	---
0x001A(26)	保留	---	---	---
0x001B(27)	保留	---	---	---
0x001C(28)	保留	---	---	---
0x001D(29)	修改站号、波特率、校验位保存使	0x0000: 不保存站号、波特率、校验位 0x0001: 保存站号、波特率、校验位 (上升沿保存)	W	不保存



	能			
0x001E(30)	保留	---	---	---
0x001F(31)	保留	---	---	---
0x0020(32)	标签 UID	UID 的 1,0 位	R	不保存
0x0021(33)	标签 UID	UID 的 3,2 位	R	不保存
0x0022(34)	标签 UID	UID 的 5,4 位	R	不保存
0x0023(35)	标签 UID	UID 的 7,6 位	R	不保存
0x0024(36)	标签 USER 区数据	Data1, Data0	R/W	不保存
0x0025(37)	标签 USER 区数据	Data3, Data2	R/W	不保存
.....		如此类推到卡片的最大	R/W	不保存



附表二 读写器寄存器和西门子 PLC Modbus 功能块映射关系

No.	读写器寄存器地址	PLC 地址映射	功能说明
1	0x0000	40001	设备地址
2	0x0001	40002	波特率
3	0x0002	40003	偶校验
4	0x0003	40004	载波信号
5	0x0004	40005	功率设置
6	0x0005	---	保留
7	0x0006	---	保留
8	0x0007	---	保留
9	0x0008	40009	标签状态寄存器
10	0x0009(9)	40010	读写状态寄存器
11	0x000A(10)	40011	故障状态代码查询
12	0x000B(11)	40012	高低字节交换
13	0x000C(12)	40013	标签格式化
14	0x000D(13)	---	保留
15	0x000E(14)	---	保留
16	0x000F(15)	40016	设备复位重启
17	0x0010(16)	40017	高字节是块大小，低字节是每块多少个字节
18	0x0011(17)	40018	标签厂商信息
19	0x0012(18)	40019	设备版本
20	0x0013(19)~0x001C(28)		保留
21	0x001D(29)	40030	修改站号、波特率、校验位保存使能
...		...	保留
33	0x0020(32)	40033	标签 UID
34	0x0021(33)	40034	标签 UID
35	0x0022(34)	40035	标签 UID
36	0x0023(35)	40036	标签 UID
37	0x0024(36)	40037	标签数据 地址 1,0
38	0x0025(37)	40038	标签数据 地址 3,2
39	0x0026(38)	40039	标签数据 地址 5,4
			



附表三 错误代码表

序号	错误代码	错误说明	指示灯显示
1	0x31	射频区域没有标签	读写器闪 1 下红灯
2	0x32	超出卡的地址	读写器闪 2 下红灯(间隔 0.5 秒)
3	0x33	读写过程中失败(或受干扰)	读写器闪 3 下红灯(间隔 0.5 秒)
4	0x34	获取标签信息失败	读写器闪 4 下红灯(间隔 0.5 秒)
5	0x35	超出单次读写数据的长度	读写器闪 5 下红灯(间隔 0.5 秒)
6	0x36	寄存器不可以写,格式不对, 数量长度, 或读取失败,CRC 校验失败。	读写器闪 6 下红灯(间隔 0.5 秒)
7	0x37	设备配置写入, 读出失败	读写器闪 7 下红灯(间隔 0.5 秒)



附表四 CRC 算法

```
unsigned int CRC16_modbus_YOK(unsigned char *buff1, unsigned char Len1)
{
    unsigned char i,n;

    unsigned int tmp =0xffff;

    unsigned int wcr = 0;

    unsigned char buff[250];

    unsigned char Len;

    Len=Len1;

    for(i=0;i<Len;i++){buff[i]=buff1[i];}

    for(n = 0; n < Len; n++) // 第一位 0xA0 用 CRC for(n = 0; n < Len; n++)
    { // 此处的 6 -- 要校验的位数为 6 个

        tmp = buff[n] ^ tmp;

        for(i = 0;i < 8;i++)

            { // 此处的 8 -- 指每一个 char 类型又 8bit, 每 bit 都要处理

                if(tmp & 0x01)

                {

                    tmp = tmp >> 1;

                    tmp = tmp ^ 0xA001;

                }

                else

                {

                    tmp = tmp >> 1;

                }

            }

    }

}
```



/*将 CRC 校验的高低位对换位置*/

```
wcrc = tmp >> 8;
```

```
wcrc = wcrc | (tmp << 8);
```

// 返回高位在前，低位在后，Modbus_rtu 是低位在前，高位在后，返回后处理

```
return wcrc;
```

```
}
```