GCAN-8100

Modbus总线耦合器

用户手册

文档版本：V3.02 （2018/01/22）

修订历史

版本	日期	原因
V1.00	2015/10/16	创建文档
V2.01	2015/12/20	修正设备工作参数
V3.01	2017/11/22	添加部分参数
V3.02	2018/01/22	添加通信协议部分

目 录

1. 功能简介

1.1 功能概述

GCAN-8100 Modbus 总线耦合器可以用于连接 Modbus 总线系统与分布式总线端子模块，这些端子模块可以通过模块化的方式进行扩展。**一个完整的节点由一个总线耦合器、1-64 个任意数量的端子模块以及一个终端端子模块组成。**GCAN-8100 总线耦合器通过 GC-bus 扩展技术，最多可连接 64 个输入/输出端子模块。

GCAN-8100 Modbus 总线耦合器采用标准的 Modbus 总线协议，是一个标准的 Modbus 从站设备。**GCAN-8100 总线耦合器具有两种不同的款式，GCAN-8100-TCP 支持 Modbus TCP 通讯，GCAN-8100-RTU 支持 Modbus RTU 通讯。**除此之外，通过组态接口还可以对固件进行升级，以满足客户的定制化需求。

GCAN-8100 总线耦合器可连接所有的总线端子模块。就用户而言，模拟量输入/输出信号的处理方式与其它种类信号的处理方式没有任何区别。控制器过程映像区内的信息以字节阵列格式显示。根据不同型号，模拟量总线端子模块寄存器中包含温度范围、增益值和线性化的特性曲线。

GCAN-8100 总线耦合器支持自动组态，您无需在 PC 上设置参数。**GCAN-8100 总线耦合器采用 Modbus RTU 通信方式时，通信参数可通过 RS-485 接口进行配置；采用 Modbus TCP 通信方式时，通信参数可通过以太网接口进行配置。**

1.2 性能特点

- 支持标准Modbus协议，是一个标准的Modbus从站设备；
- 支持Modbus TCP、Modbus RTU通讯（**分别对应两种不同的型号**）；
- RS485 接口采用标准 2 线制；
- 串口波特率支持 600bps~115200bps 之间，可通过串口配置；
- 总线端子模块最大数量为 64 个；
- 组态方式为自动组态形式，可自由扩展；
- 以太网口支持静态或动态 IP 获取；
- 以太网口支持心跳和超时断开功能；
- 以太网口工作端口固定，目标 IP 和目标端口均可设定；
- 以太网口网络断开后自动恢复连接资源，可靠地建立 TCP 连接；
- 以太网口兼容 SOCKET 工作方式（TCP Server、TCP Client、UDP 等），上位机通讯软件编写遵从标准的 SOCKET 规则。
- 电源采用 24V DC（-15%/+20%）；
- 输入电流为 70mA+（总 GC-bus 电流），最大为 4500mA；
- 启动电流：约为 2.5 倍的持续电流；
- 电源触电：最大 24V DC/最大 10A；
- 电气隔离为 1000 Vrms （电源触点/总线耦合电源电压）；
- 工作温度范围：-40℃~+85℃；
- 标准 DIN 导轨安装方式，专为工业设计。

- 尺寸：长 100mm * 宽 69mm * 高 48mm。

1.3 典型应用

- 与分布式总线端子模块相连接，构成一个完整的控制节点；
- 执行数据采集并以 Modbus 协议进行数据传输。

2. 设备安装与使用

本章节将详细说明 GCAN-8100 Modbus 总线耦合器的安装方法、接线方法、指示灯的含义与接口的含义。

2.1 模块固定

GCAN-8100 Modbus 总线耦合器的安装方法如图 2.1 所示，您需要使用一字螺丝刀进行辅助安装。

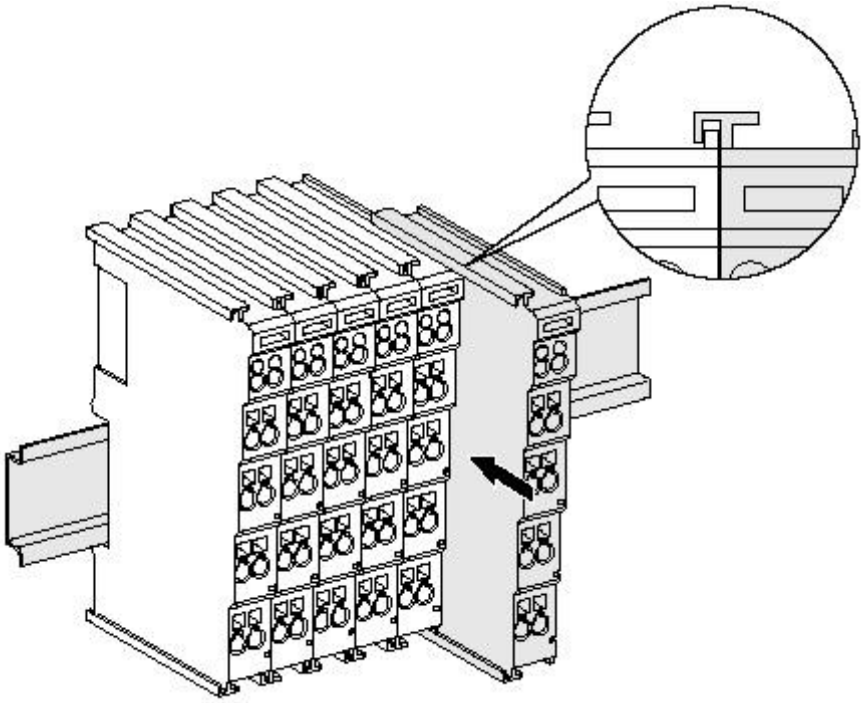

图 2.1 GCAN-8100 模块安装

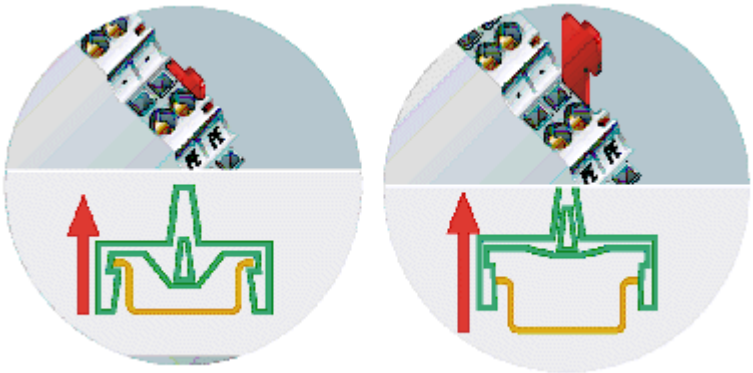

图 2.2 GCAN-8100 模块自锁机制

请按照图 2.1 所示，把 GCAN-8100 Modbus 总线耦合器安装在导轨上，直到锁扣卡死并发出“咔”的一声。GCAN-8100 Modbus 总线耦合器具有自锁机制，

可有效防止设备掉落。如图 2.2 所示，您可以通过拉出橙色的标签来释放自锁机制。

GCAN-8100 Modbus 总线耦合器最多可以连接 64 个分布式总线端子模块。插入总线端子模块时，一定要沿着凹槽，在已有模块的右侧顺次插入，直到锁扣卡死并发出“咔”的一声。在整个节点的最右端，您需要安装终端端子模块。该终端可以保障 GC-Bus 的数据传输与电力供应。

当您正确组装节点时，在端子模块之间不会存在明显的缝隙。如果模块之间未被正确组装，整个节点将不会正常运行。

2.2 接线方法

如图 2.3 所示，先使用一字螺丝刀插入方形孔中，顶住方形孔中的螺丝。之后将线缆插入圆形孔中。插好之后，拔出螺丝刀，线缆即可稳固地锁死在圆形孔中。

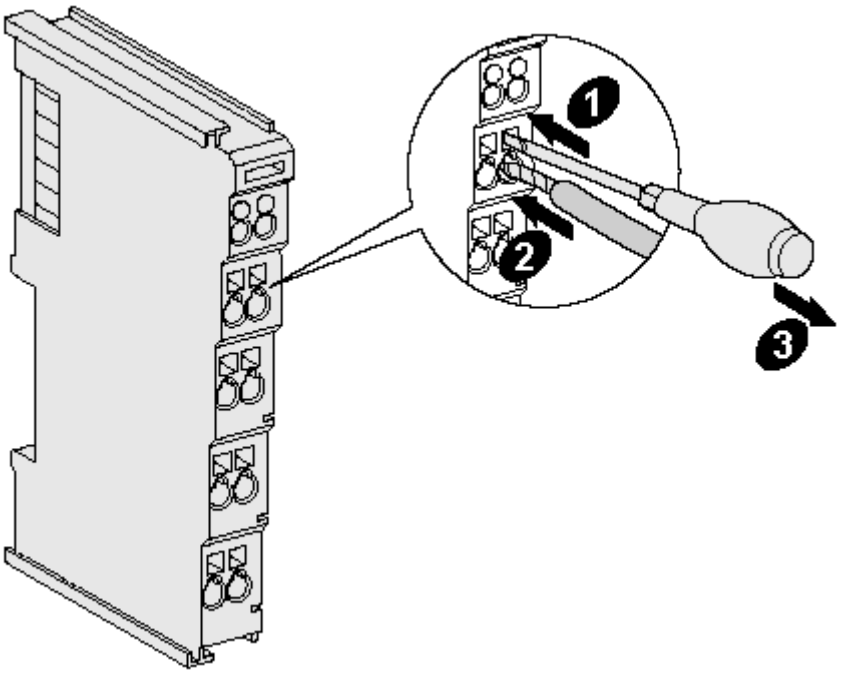

图 2.3 GCAN-8100 模块安装

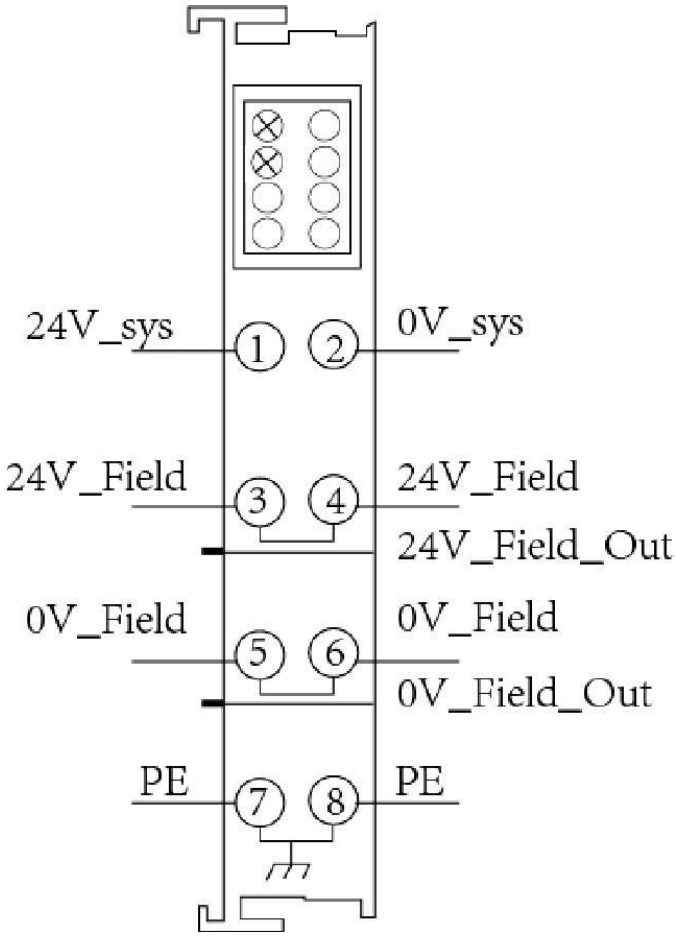

图 2.4 GCAN-8100 模块接线端子排

GCAN-8100 Modbus 总线耦合器的接线端子排如图 2.4 所示。GCAN-8100 Modbus 总线耦合器包含 8 个端子，各个端子对应的序号及其含义如表 2.1 所示。请注意，3 号端子与 4 号端子之间、5 号端子与 6 号端子之间、7 号端子与 8 号端子之间，在模块的内部是相连的。

端子	序号	含义
24V	1	电源24V输入
0V	2	电源GND
+	3	IO电源正
+	4	IO电源正
-	5	IO电源负
-	6	IO电源负
PE	7	屏蔽
PE	8	屏蔽

表2.1 GCAN-8100模块接线端子定义

2.3 系统状态指示灯

GCAN-8100 Modbus总线耦合器具有两组状态指示灯。左侧区域包含6个圆形状态指示灯，右侧区域包含2个小型电源指示灯。指示灯的具体指示功能见表2.2。指示灯处于不同状态下时，GCAN-8100模块状态如表2.3所示。

指示灯	颜色	指示状态
PWR	绿	电源指示
SYS	绿	系统指示
RUN	绿	运行指示
ERR	绿	错误指示
IO RUN	绿	内部总线运行指示
IO ERR	绿	内部总线错误指示
右侧1号位置	绿	电源指示
右侧3号位置	绿	内部总线电源指示

表2.2 GCAN-8100模块指示灯

指示灯	状态	指示状态
PWR	常亮	供电正常
	不亮	供电异常
SYS	闪烁	设备初始化通过，进入工作状态
	不亮	设备初始化失败
RUN	闪烁	设备运行正常
	不亮	设备运行停止
ERR	常亮	系统错误
	不亮	系统未出现错误
IO RUN	闪烁	内部总线运行正常
	不亮	内部总线停止
IO ERR	常亮	内部总线运行错误
	不亮	内部总线运行未出现错误
右侧1号位置	常亮	端子侧供电正常
	不亮	端子侧供电异常
右侧3号位置	常亮	端子内部总线供电正常
	不亮	端子内部总线供电异常

表2.3 GCAN-8100模块指示灯状态

3. 通信连接

3.1 串口连接

GCAN-8100-RTU 模块使用标准串口电平（RS485），因此该模块可以直接与带有 RS485 接口的设备进行连接。

3.2 与以太网连接

用户可以使用标准 5 类以上网线直接与 GCAN-8100-TCP 模块的 LAN 接口连接，并建立通信。

4. 通信举例

GCAN-8100 Modbus 总线耦合器执行 Modbus 通信协议，为 Modbus 从站设备。

当搭载 GC-1008 模块（8 路数字量输入）时，GCAN-8100 会将 DI 数据存放于 **Modbus 数字量输入寄存器**中，您可以通过 02 功能码进行读取。

当搭载 GC-2008 模块（8 路数字量输出）时，GCAN-8100 会将 DO 数据存放于 **Modbus 数字量输出寄存器**中，您可以通过 05、15 功能码进行写入，通过 01 功能码进行读取。

当搭载 GC-3804 模块（4 路 PT100 模拟量输入）时，GCAN-8100 会将 AI 数据存放与 **Modbus 模拟量输入寄存器**中，您可以通过 04 功能码进行读取。

本章将以 Modbus TCP 为例，使用网络调试助手进行 Modbus 协议数据的接收与发送。随货附带的 Modbus Poll 软件具有 Modbus 主站功能，可以您帮助调试 GCAN-8100 Modbus 从站，非常的方便实用。

您可以通过网络调试助手或 Modbus Poll 软件给 GCAN-8100 Modbus 总线耦合器发送控制指令。

如果我们使用一个 GCAN-8100 Modbus 从站，搭载两个 GC-1008 模块（8 路数字量输入）和两个 GC-2008 模块（8 路数字量输出），那么距离 GCAN-8100 模块最近的 GC-1008 模块为 1008-1，稍远的 GC-1008 为 1008-2。同理，距离 GCAN-8100 模块最近的 GC-2008 模块为 2008-1，稍远的 GC-2008 为 2008-2。

控制指令含义	控制指令及返回指令	功能码及含义
使能 2008-1 第一个通道	发送：00 00 00 00 00 06 01 **05** 00 00 FF 00 返回：00 00 00 00 00 06 01 **05** 00 00 FF 00	05 强置单线圈
使能 2008-1 第二个通道	发送：00 00 00 00 00 06 01 **05** 00 01 FF 00 返回：00 00 00 00 00 06 01 **05** 00 01 FF 00	05 强置单线圈
失能 2008-1 第一个通道	发送：00 00 00 00 00 06 01 **05** 00 00 00 00 返回：00 00 00 00 00 06 01 **05** 00 00 00 00	05 强置单线圈
读取 1008-1 所有通道状态	发送：00 00 00 00 00 06 01 **02** 00 00 00 08 返回：00 00 00 00 00 04 01 **02** 01 0C	02 读取输入状态
使能 2008-2 所有	发送：00 00 00 00 00 08 01 **0F** 00 08 00 08 01 **FF** 返回：00 00 00 00 00 08 01 **0F** 00 08 00 08	04 读取模拟量输入寄存器

表 4.1 Modbus TCP 实验测试指令

4.1 搭载 GC-1008 模块

数字量输入的状态由一个字节来表示，通道 8 在高位，通道 1 在低位。

例如，GCAN-8100 模块节点号设为 1。通道 8 和通道 4 状态为 1，其他状态均为 0，则 Modbus 一端显示的 DI 状态数据为 88。下表列举了两种常见的 DI 状态及其对应的状态数据。

DI 状态								
通道数	8	7	6	5	4	3	2	1
状态	1	0	0	0	1	0	0	0
Modbus 显示的数据	88							

DI 状态								
通道数	8	7	6	5	4	3	2	1
状态	0	1	0	1	1	0	1	0
Modbus 显示的数据	5A							

4.2 搭载 GC-2008 模块

数字量输出的状态由一个字节来表示，通道 8 在高位，通道 1 在低位。

例如，GCAN-8100 模块节点号设为 1。需设置通道 8 和通道 4 状态为 1，设置其他状态均为 0，则需要发送的 Modbus DO 状态数据为 88（15 功能码）。

DO 状态								
通道数	8	7	6	5	4	3	2	1
状态	1	0	0	0	1	0	0	0
Modbus 显示的数据	88							

DO 状态								
通道数	8	7	6	5	4	3	2	1
状态	0	1	0	1	1	0	1	0
Modbus 显示的数据	5A							

4.3 搭载 GC-3804 模块

每个通道的温度状态由两个字节来表示，四个通道共八个字节。

其中，代表温度状态的两个字节，第一个字节为低位，需将该字节的数据转换为十进制后乘以 0.1；第二个字节为高位，需将该字节的数据转换为十进制之后乘以 25.6。最后将两个数值加和，即为最终的温度值，单位为摄氏度。

例如，四个通道的温度分别为 25.6 度，25.5 度，20 度，30 度。模拟量输入数据为 0x00，0x01，0xFF，0x00，0xC8，0x00，0x2C，0x01。

GC-3804 温度与 CAN 数据对应关系		
Modbus 显示的数据	**低字节 C8**	**高字节 00**
系数	**200（0xC8）x0.1**	**0（0x00）x25.6**
温度值	**20℃**	

GC-3804 温度与 CAN 数据对应关系		
Modbus 显示的数据	**低字节 2C**	**高字节 01**
系数	**44（0x2C）x0.1**	**1（0x01）x25.6**
温度值	**30℃**	

4.4 同时搭载多组模块

若 GCAN-8100 同时搭载多组 GC-1008 模块，那么我们以它们距离 GCAN-8100 的远近，从近到远进行编号，离得最近的为 1 号。GCAN-8100 耦合器将按照下表进行**数字量输入寄存器首地址**的确认。例如，当 GCAN-8100 搭载 9 个 GC-1008 模块时，数字量输入寄存器地址依次为 01-09。

若 GCAN-8100 同时搭载多组 GC-2008 模块，那么我们以它们距离 GCAN-8100 的远近，从近到远进行编号，离得最近的为 1 号。GCAN-8100 耦合器将按照下表进行**数字量输出寄存器首地址**的确认。例如，当 GCAN-8100 搭载 9 个 GC-2008 模块时，数字量输出寄存器地址依次为 01-09。

若 GCAN-8100 同时搭载多组 GC-3804 模块，那么我们以它们距离 GCAN-8100 的远近，从近到远进行编号，离得最近的为 1 号。GCAN-8100 耦合器将按照下表进行**模拟量输入寄存器首地址**的确认。例如，当 GCAN-8100 搭载 3 个 GC-3804 模块时，模拟量输入寄存器地址依次为 01-04、05-08、09-12。

5. 技术规格

接口特点	
Modbus总线协议	Modbus TCP、Modbus RTU
总线端子模块数量	64个
现场总线的最大字节数	32字节输入和32字节输出
数字量I/O信号	256输入/输出
模拟量I/O信号	60输入/输出
组态方式	自动组态
总线接口	RJ45
电源	24V DC（-15%/+20%）
输入电流	70mA+（总GC-bus电流）/最大4500mA
启动电流	约2.5倍持续电流
建议保险丝容量	≤10A
GC-bus供电电流	500mA
电源触电	最大24V DC/最大10A
电气隔离	1000 Vrms （电源触点/总线耦合电源电压）
环境试验	
工作温度	-40℃~+85℃
工作湿度	95%RH，无凝露
EMC测试	EN 55024:2011-09 EN 55022:2011-12
抗振/抗冲击性能	EN 60068-2-6/EN 60068-2-27/29
抗电磁干扰/抗电磁辐射性能	EN 61000-6-2 /EN 61000-6-4
防护等级	IP 20
基本信息	
外形尺寸	100mm *69mm *44mm
重量	100g

6. GC 系列模块选型表

GCAN-8000 本身并不能执行完整的控制功能。一个完整的控制系统由一个总线模块控制器（GCAN-8000）、1-32 个任意数量的 GC 系列端子模块（GC-1008、GC-3804 等）以及一个终端端子模块组成。其中，GC 系列端子模块需在我司另行购买，终端端子模块随 GCAN-8000 附赠。

GC 系列可编程控制器扩展模块目前包括：数字量输入扩展模块、数字量输出扩展模块、模拟量输入扩展模块、模拟量输出扩展模块四大类，具体的选型表如下图所示。

I/O	型号	特性	信号	通道数
数字量输入	GC-1004	滤波 3.0ms	24V DC	4 通道
	GC-1008	漏型（NPN），滤波 3.0ms	24V DC	8 通道
	GC-1018	源型（PNP），滤波 3.0ms	24V DC	8 通道
	GC-1502	加/减 24V DC，100kHz	计数器	2 通道
数字量输出	GC-2008	源型（PNP），Imax=0.5A	24V DC	8 通道
	GC-2018	漏型（NPN），Imax=0.5A	24V DC	8 通道
	GC-2104	Imax=1.0A，固态	24V AC/DC	4 通道
	GC-2202	继电器，动合触点	220V AC	2 通道
	GC-2302	24V DC，0.1A	PWM	2 通道
	GC-2401		脉冲串	1 通道
模拟量输入	GC-3604	共地单端输入，16 位	0-10V	4 通道
	GC-3614	16 位	0-20mA	4 通道
	GC-3624	4x2 线制接线，16 位	±10V	4 通道
	GC-3644	4x2 线制接线，16 位	4-20mA	4 通道

	GC-3804	PT100，16 位	热电阻	4 通道
模拟量输出	GC-4602	12 位	0-10V	2 通道
	GC-4612	4x2 线制接线，12 位	0-20mA	2 通道
	GC-4622	12 位	±10V	2 通道
	GC-4642	4x2 线制接线，12 位	4-20mA	2 通道

附录 A：Modbus 协议简介

Modbus通信协议是由Modicon公司开发的应用在PLC或其他工业控制器上的一种通用语言。通过此协议，各控制器之间可以实现串行通信，Modbus通信协议定义了一个控制器能识别使用的消息结构，描述了主控制器访问从站设备的过程，例如规定从站怎样做出应答响应，检查和报告传输错误等。Modbus协议的通信方式为主从方式。主站首先向从站设备发送通信请求指令，从节点根据请求指令中的功能码向主站发回回答数据。网络中的每个从站设备都必须分配给一个唯一的地址，最多可达31个从站设备。通过多达24种总线命令实现主控制器与从站设备之间的信息交换。从站设备只执行发给自己的指令，对于其它从站地址开头的报文不作应答。这种一问一答的通信模式，大大提高了通信的正确率。因其具有操作简单、高效、通信可靠等优点，Modbus协议已成为一个国际通信标准，得到了国际上大多数工控产品生产厂家的支持。该通信协议已广泛应用于机械、水利、电力、环保等行业设备中。

Modbus TCP通信协议可供自动化设备的监控使用。常见的应用是开发基于该协议的网关，通过网关可以将PLC、I/O模块和其它总线连到以太网上。Modbus TCP是在不改变原有的Modbus协议基础上，只是将其作为应用层协议简单的移植到TCP/IP协议上。Modbus TCP协议每一个呼叫都要求一个应答。利用TCP/IP协议，通过网页的形式可以使用户界面更加友好。利用网络浏览器就可以查看企业网内部的设备运行情况。Schneider公司已经为Modbus注册了502端口，这样就可以将实时数据嵌入到网页中，通过在设备中嵌入Web服务器，就可以将Web浏览器作为设备的操作终端。但是Modbus协议本身存在一些缺陷，它不支持诸如基于对象的通信模型等一些正在被广泛采用的网络新技术，用户在使用的时候，不得不手工配置一些参数，比如信息数据类型、寄存器号等等。

A.1 Modbus RTU 协议数据格式

Modbus 协议有 ASCII(美国标准信息交换代码)和 RTU(远程终端单元)两种数据传输方式可由用户选择，但在一个 Modbus 网络上的所有设备都必须选择相同的传输模式和串口参数。其中 RTU 模式信息帧中的 8 位数据包括两个 4 位 16 进制字符，相对于 ASCII 模式表达相同的信息只需较少的位数，在相同的速率下较 ASCII 模式具有更大的数据流量。因此，在通常情况下较多使用 RTU 模式。GCAN-204 设备也采用 RTU 模式。

RTU 模式消息发送至少以 3.5 个字符间隔时间(如表 A.1 的 T1-T2-T3-T4)标志开始和结束，信息帧由地址域、功能域、数据域和 CRC 校验域构成，所有字符位由 16 进制 0-9、A-F 组成。整个消息帧必须作为一连续的流传输。如果在帧完成之前有超过 1.5 个字符时间的停顿时间，接受设备将刷新不完整的消息并假定下一个字节是一个新消息的地址域。同样的，如果一个新消息在小于 3.5 个字符时间内接着前个消息开始，接收的设备将认为它是前一消息的延续。这将导致一个错误，因为在最后的 CRC 域的值不可能是正确的。

起始位	设备地址	功能代码	数据	CRC 校验	结束符

T1-T2-T3-T4	8Bit	8Bit	N 个 8Bit	16Bit	T1-T2-T3-T4

表 A.1 RTU 消息帧格式

（1）地址域

指定报文的目的地址，包括 8bit。单个设备的地址范围是 1~247。主设备通过将要联络的从设备的地址放入消息中的地址域来选通从设备。当从设备发送回应消息时，它把自己的地址放入回应的地址域中，以便主设备知道是哪一个设备作出回应。地址 0 用作广播地址，以使所有的从设备都能认识。

（2）功能域

当消息从主设备发往从设备时，功能代码域将告之从设备需要执行哪些行为。例如去读取输入的开关状态，读一组寄存器的数据内容，读从设备的诊断状态，允许调入、记录、校验在从设备中的程序等。当从设备回应时，它使用功能代码域来指示是正常回应(无误)还是有某种错误发生(称作异议回应)。对正常回应，从设备仅回应相应的功能代码。主设备应用程序得到异议的回应后，典型的处理过程是重发消息，或者诊断发给从设备的消息并报告给操作员。

（3）数据域

数据域是由两个十六进制数集合构成的，范围 00~FF。从主设备发给从设备消息的数据域包含从机执行主机功能代码中所需的参数，如处理对象的寄存器地址，要处理项的数目，域中实际数据字节数。举例说明，如果主设备需要从设备读取一组保持寄存器（功能代码 03），数据域指定了起始寄存器以及要读的寄存器数量。如果主设备写一组从设备的寄存器(功能代码 16，即 10H)，数据域则指明了要写的起始寄存器以及要写的寄存器数量，数据域的数据字节数，要写入寄存器的数据。如果没有错误发生，从设备返回的数据域包含请求的数据。如果有错误发生，此域包含一异议代码，主设备应用程序可以用来判断采取下一步行动。在某种消息中数据域可以是不存在的(0 长度)。例如，主设备要求从设备回应通信事件记录(功能代码 0B H)，从设备不需任何附加的信息。

当传送一个 2 个字节的数据时，高字节(MSB)将被首先传送，然后传送低字节(LSB)。这与 DeviceNet 的传送方式刚好相反。

（4）CRC 校验域

CRC 域检测整个消息的内容，包括两个字节，包含一个 16 位的二进制值。它由传输设备计算后加入到消息中。接收设备将重新计算收到消息的 CRC，并与接收到的 CRC 域中的值进行比较。如果两值不同，则有误。CRC 添加到消息中时，低字节先加入，然后是高字节。

A.2 Modbus TCP 协议数据格式

TCP/IP 协议和以太网的链路层校验机制已可保证数据包传递的正确性，因此 Modbus TCP 报文中不再存在 CRC-16 或 LRC 校验域，但需要添加一个 Modbus 应用帧头(MBAP)。它可对 Modbus 的参数及功能进行解释。每个 TCP/IP 报文仅可含有一个 Modbus 帧。

在 Modbus TCP ADU 中，MBAP 头部占 7 个字节（含 4 个子域），及交易标识符 TI(Transaction Identifier)、协议标识符 PI(Protocol Identifier)，长度标识符 L(Length)(占用 2 字节，指明 Protocol Identifier 和 Data 域的总长度)和单元标识符 UI(Unit Identifier)组成。TI 占用 2 字节，用来标识 Modbus 帧的次序，PI 占用 2 字节，用于确认应用层协议。UI 占 1 字节，用于标识 Modbus 设备单元。功能码

占 1 字节，可分为位操作和 16 位字操作两类。功能码指出要进行的操作，如功能码 15 代表写多个位寄存器，功能码 06 表示对独立的 16 位字寄存器进行写操作。数据域最多可达 248 字节，其具体格式与功能码相关。当客户机发送请求数据时，数据域给出要操作的寄存器的起始地址（2 字节）和个数（l 字节）；当服务器发送应答数据时，数据域给出被操作的寄存器个数（1 字节）及各寄存器状态值。图 A.1 给出了 Modbus 与 Modbus TCP 数据帧格式比较。

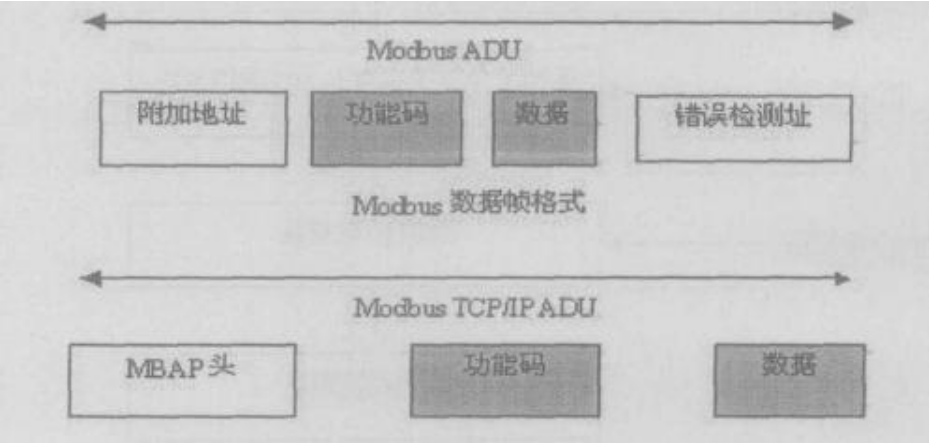

图 A.1 Modbus 与 Modbus TCP/IP 帧格式

Modbus TCP 的 ADU 数据单元规范如表 A.1 所示。

	描述	所占字节
MBAP 头	传输标识码高位 Hi	1
	传输标识码低位 Lo	1
	协议标识符	2
	长度标识符	2
	单元标识符	1
Modbus 请求	功能码	1
	开始地址	2
	寄存器数目	2

表 A.2 Modbus TCP 的 ADU 数据单元规范

在通过 Modbus TCP 传送数据之前，需要在客户机和服务器之间建立一个 TCP/IP 连接。服务器使用端口 502 作为 Modbus TCP 的连接端口。Modbus TCP 连接的建立通常由 TCP/IP Socket 接口的软件协议自动实现，因此对应用完全透明。一旦客户端和服务器之间的 TCP/IP 连接建立，同样的连接可以根据要求的方向用来传输任意数量的用户数据。客户端和服务器还可以同时建立多个TCP/IP 连接，最大的连接数量取决于 TCP/IP 接口的规范。

当某一设备发出请求，则其相应的设备要做出响应。响应的数据格式如表 A.2 所示。

字节	响应数据
Byte0、Byte1	传输标识码=0（响应时拷贝该数据）

Byte2、Byte3	协议标识符
Byte4	长度标识符高字节=0
Byte5	长度标识符低字节（标识其后有多少个字节）
Byte6	单元标识符（从设备地址）
Byte7	Modbus 功能码
Byte8	数据

表 A.3 Modbus TCP 响应数据格式

A.3 Modbus 常用功能码

在 Modbus 消息帧的功能码中较常使用的是 01、02、03、04、05、06 和 16 功能码，使用它们即可实现对从机的数字量和模拟量的读写操作。

Modbus 标准地址与各个功能码的对应关系如下所示。

Modbus 标准地址	数据	功能码
00001-0xxxx	DO	01、05、15
10001-1xxxx	DI	02
30001-3xxxx	AI	04
40001-4xxxx	保持寄存器	03、06、16

下面以在 RTU 传输模式下通讯为例，对这些功能码进行详细介绍。

功能码	名称	功能说明
01	读取线圈状态	取得一组线圈的当前状态(ON/OFF)
02	读取输入状态	取得一组开关输入的当前状态(ON/OFF)
03	读取保持寄存器	在一个或多个保持寄存器中取得当前的二进制值
04	读取输入寄存器	在一个或多个输入寄存器中取得当前的二进制值
05	强置单线圈	强置一个逻辑线圈的通断状态
06	预置单寄存器	把具体二进制值装入一个保持寄存器
07	读取异常状态	取得 8 个内部线圈的通断状态
08	回送诊断校验	把诊断校验报文送从机，通信诊断
16	预置多寄存器	把具体二进制值装入一串连续的保持寄存器
128~255	保留	用于异常应答

下面是 7 个 **Modbus RTU** 命令的主从机收发的数据包格式，其余的命令可参照其格式。

（1）功能码：01H

代码功能：读取线圈状态（DO）

说明：读取从机 DO 的 ON/OFF 状态，不支持广播。

查询：查询信息规定了要读的起始线圈地址和线圈量，线圈的起始地址为 0000H，1-16 个线圈的寻址地址分为 0000H-0015H。

主机发送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	01	读取线圈状态

线圈首地址	2 字节	00 00	线圈首址为 0000H
线圈数量	2 字节	00 08	连续读 8 个线圈
CRC	2 字节	3D CC	前 6 个字节的 CRC 校验码

响应：响应信息中的各线圈的状态与数据区的每一位的值相对应，即每个 DO 占用一位（1 = ON, 0 = OFF）。数据区从高位到低位依次为 DO7、DO6.....DO0。

从机回送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	01	读取线圈状态
数据字节数	1 字节	01	1 个字节
数据	1 字节	02	二进制为 0000 0010，DO1 为 ON
CRC	2 字节	D0 49	前 4 个字节的 CRC 校验码

（2）功能码：02H

代码功能：读取输入状态（DI）

说明：读取从机 DI 的 ON/OFF 状态，不支持广播。

查询：查询信息规定了要读的输入起始地址及输入信号的数量，输入寻址起始地址为 0000H，输入 1-16 所对应的地址分别为 0-15。

主机发送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	02	读取输入状态
输入首地址	2 字节	00 00	输入首址为 0000H
寄存器数量	2 字节	00 08	连续读 8 个输入口
CRC	2 字节	79 CC	前 6 个字节的 CRC 校验码

响应：响应信息中的各输入口的状态与数据区的每一位的值相对应，即每个 DI 占用一位（1 = ON, 0 = OFF）。数据区从高位到低位依次为 DI7、DI6.....DI0。

从机回送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	02	读取输入状态
数据字节数	1 字节	01	1 个字节
数据	1 字节	81	二进制为 1000 0001，DI7 与 DI0 为 ON
CRC	2 字节	61 E8	前 4 个字节的 CRC 校验码

（3）功能码：03H

代码功能：读取保持寄存器

说明：读从机保持寄存器的二进制数据，不支持广播。

查询：查询信息规定了要读的寄存器起始地址及寄存器的数量，寄存器寻址起始地址为 0000H，寄存器 1-16 所对应的地址分别为 0-15。

主机发送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	03	读取保持寄存器数据

寄存器首地址	2 字节	00 01	寄存器首址为 0001H
寄存器数量	2 字节	00 03	连续读 3 个寄存器
CRC	2 字节	54 0B	前 6 个字节的 CRC 校验码

响应：响应信息中的寄存器数据为二进制数据，每个寄存器分别对应 2 个字节，第一个字节为高位值数据，第二个字节为低位数据。

从机回送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	03	读取保持寄存器数据
数据字节数	1 字节	06	3 个寄存器占 6 个字节
数据 1	2 字节	02 0B	0001H 寄存器中的数据
数据 2	2 字节	00 00	0002H 寄存器中的数据
数据 3	2 字节	00 64	0003H 寄存器中的数据
CRC	2 字节	84 BD	前 9 个字节的 CRC 校验码

（4）功能码：04H

代码功能：读取输入寄存器（AI）

说明：读取从机输入寄存器(3X 类型)中的二进制数据，不支持广播。

查询：查询信息规定了要读的寄存器起始地址及寄存器的数量，寄存器寻址起始地址为 0000H，寄存器 1-16 所对应的地址分别为 0-15。

主机发送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	04	读取输入寄存器数据
寄存器首地址	2 字节	00 00	寄存器首址为 0000H
寄存器数量	2 字节	00 01	连续读 1 个寄存器
CRC	2 字节	31 CA	前 6 个字节的 CRC 校验码

响应：响应信息中的寄存器数据为二进制数据，每个寄存器分别对应 2 个字节，第一个字节为高位值数据，第二个字节为低位数据。

从机回送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	04	读取输入寄存器数据
数据字节数	1 字节	02	1 个寄存器占 2 个字节
数据 1	2 字节	0F FB	0000H 寄存器中的数据
CRC	2 字节	FD 43	前 5 个字节的 CRC 校验码

（5）功能码：05H

代码功能：强置单线圈（DO）

说明：强制单个线圈(DO，0X 类型)为 ON 或 OFF 状态，广播时，该功能可强制所有从机中同一类型的线圈均为 ON 或 OFF 状态。

查询：查询信息规定了需要强制线圈的地址及状态，线圈的起始地址为 0000H，寄存器 1-16 所对应的地址分别为 0-15。查询时，由查询数据区中的一个常量，规定被请求线圈的 ON/OFF 状态，FF00H 值请求线圈处于 ON 状态，

0000H 值请求线圈处于 OFF 状态，其它值对线圈无效，不起作用。

主机发送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	05	强置单线圈
线圈地址	2 字节	00 01	线圈地址为 0001H
线圈状态值	2 字节	FF 00	ON 状态
CRC	2 字节	DD FA	前 6 个字节的 CRC 校验码

响应：对这个命令请求的正常响应是在 DO 状态改变以后，原样传送接收到的数据。

从机回送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	05	强置单线圈
线圈地址	2 字节	00 01	线圈地址为 0001H
线圈状态值	2 字节	FF 00	ON 状态
CRC	2 字节	DD FA	前 6 个字节的 CRC 校验码

（6）功能码：06H

代码功能：预置单寄存器

说明：把一个值预置到一个保持寄存器（4X 类型）中，广播时，该功能把值预置到所有从机相同类型的寄存器中。该功能可越过控制器的内存保护。使寄存器中的预置值保持有效。只能由控制器的下一个逻辑信号来处理该预置值。若控制逻辑中无寄存器程序时，则寄存器中的值保持不变。

查询：查询信息规定了要预置寄存器的类型，寄存器寻址起始地址为 0000H，寄存器 1-16 所对应的地址分别为 0-15。

主机发送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	06	读寄存器数据
寄存器地址	2 字节	00 03	预置寄存器地址为 0003H
寄存器的值	2 字节	AB CD	将该值预置到寄存器中
CRC	2 字节	C7 6F	前 6 个字节的 CRC 校验码

响应：对这个命令请求的正常响应是在寄存器值状态改变以后，原样传送接收到的数据。

从机回送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	06	读寄存器数据
寄存器地址	2 字节	00 03	预置寄存器地址为 0003H
寄存器的值	2 字节	AB CD	将该值预置到寄存器中
CRC	2 字节	C7 6F	前 6 个字节的 CRC 校验码

（7）功能码：10H（十进制为 16）

代码功能：预置多个寄存器

说明：把数据按顺序预置到各(4x 类型)寄存器中，广播时该功能代码可把数据预置到全部从机中的相同类型的寄存器中。需要注意的是该功能代码可越过控制器的内存保护，在寄存器中的预置值一直保持有效，只能由控制器的下一个逻辑来处理寄存器的内容，控制逻辑中无该寄存器程序时，则寄存器中的值保持不变。

查询：信息中规定了要预置的寄存器类型，寄存器寻址的起始地址为 0。查询数据区中指定了寄存器的预置值，M84 和 484 型控制器使用 10 位二进制数据，2 个字节，剩余的高 6 位置 0。而其他类型的控制器使用一个 16 位二进制数据，每个寄存器 2 个字节。

主机发送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	10	预置多个寄存器
寄存器首地址	2 字节	10 20	写入寄存器首址为 1020H
寄存器数量	2 字节	00 03	连续 3 个寄存器
字节数	1 字节	06	3 个寄存器占 6 个字节
数据 1	2 字节	02 01	寄存器 1020H 中的数据
数据 2	2 字节	04 03	寄存器 1021H 中的数据
数据 3	2 字节	06 05	寄存器 1022H 中的数据
CRC	2 字节	BD 9B	前 13 个字节的 CRC 校验码

响应：正常响应返回从机地址、功能代码、起始地址和预置寄存器数。

从机回送	字节数	例（Hex）	注释
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	10	写寄存器数据
寄存器首地址	2 字节	10 20	写入寄存器首址为 1020H
寄存器数量	2 字节	00 03	连续 3 个寄存器
CRC	2 字节	85 02	前 6 个字节的 CRC 校验码

下面是 7 个 **Modbus TCP** 命令的主从机收发的数据包格式，其余的命令可参照其格式。本部分略去代码功能及说明，相关内容请参考 Modbus RTU 部分。

（1）功能码：01H

主机发送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	其后有 6 个字节
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	01	读取线圈状态
线圈首地址	2 字节	00 00	线圈首址为 0000H
线圈数量	2 字节	00 08	连续读 8 个线圈

从机回送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	

协议标识	2 字节	00 00	
数据长度	2 字节	00 04	其后有 4 个字节
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	01	读取线圈状态
数据字节数	1 字节	01	1 个字节
数据	1 字节	02	二进制为 0000 0010，DO1 为 ON

（2）功能码：02H

主机发送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	02	读取输入状态
输入首地址	2 字节	00 00	输入首址为 0000H
寄存器数量	2 字节	00 08	连续读 8 个输入口

从机回送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 04	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	02	读取输入状态
数据字节数	1 字节	01	1 个字节
数据	1 字节	81	二进制为 1000 0001，DI7 与 DI0 为 ON

（3）功能码：03H

主机发送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	03	读取保持寄存器数据
寄存器首地址	2 字节	00 01	寄存器首址为 0001H
寄存器数量	2 字节	00 03	连续读 3 个寄存器

从机回送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 09	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	03	读取保持寄存器数据

数据字节数	1 字节	06	3 个寄存器占 6 个字节
数据 1	2 字节	02 0B	0001H 寄存器中的数据
数据 2	2 字节	00 00	0002H 寄存器中的数据
数据 3	2 字节	00 64	0003H 寄存器中的数据

（4）功能码：04H

主机发送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	04	读取输入寄存器数据
寄存器首地址	2 字节	00 00	寄存器首址为 0000H
寄存器数量	2 字节	00 01	连续读 1 个寄存器

从机回送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 05	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	04	读取输入寄存器数据
数据字节数	1 字节	02	1 个寄存器占 2 个字节
数据 1	2 字节	0F FB	0000H 寄存器中的数据

（5）功能码：05H

主机发送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	05	强置单线圈
线圈地址	2 字节	00 01	线圈地址为 0001H
线圈状态值	2 字节	FF 00	ON 状态

从机回送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	05	强置单线圈
线圈地址	2 字节	00 01	线圈地址为 0001H
线圈状态值	2 字节	FF 00	ON 状态

（6）功能码：06H

主机发送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	06	读寄存器数据
寄存器地址	2 字节	00 03	预置寄存器地址为 0003H
寄存器的值	2 字节	AB CD	将该值预置到寄存器中

从机回送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	06	读寄存器数据
寄存器地址	2 字节	00 03	预置寄存器地址为 0003H
寄存器的值	2 字节	AB CD	将该值预置到寄存器中

（7）功能码：10H（十进制为 16）

主机发送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 0D	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	10	预置多个寄存器
寄存器首地址	2 字节	10 20	写入寄存器首址为 1020H
寄存器数量	2 字节	00 03	连续 3 个寄存器
字节数	1 字节	06	3 个寄存器占 6 个字节
数据 1	2 字节	02 01	寄存器 1020H 中的数据
数据 2	2 字节	04 03	寄存器 1021H 中的数据
数据 3	2 字节	06 05	寄存器 1022H 中的数据

从机回送	字节数	例（Hex）	注释
传输标识	2 字节	00 00	
协议标识	2 字节	00 00	
数据长度	2 字节	00 06	
从机地址	1 字节	01	与 01 号从机通信
功能码	1 字节	10	写寄存器数据
寄存器首地址	2 字节	10 20	写入寄存器首址为 1020H
寄存器数量	2 字节	00 03	连续 3 个寄存器

销售与服务

沈阳广成科技有限公司

地址：辽宁省沈阳市皇姑区崇山中路 42 号工业设计中心

邮编：110000

电话：024-31230060 全国服务电话：400-6655-220

网址：www.gcgd.net

全国销售与服务电话：400-6655-220

售后服务电话与微信号：13840170070