

Modbus/DeviceNet 网关

MD-21

产品手册



Sibotech

上海泗博自动化技术有限公司

2008 年 9 月

目录

目录	- 2 -
一、产品概述	- 4 -
1.1 产品简介	- 4 -
1.2 产品特点	- 4 -
1.3 产品用途	- 5 -
二、技术指标	- 5 -
2.1 通讯速率	- 5 -
2.2 DeviceNet拓扑结构	- 5 -
表 1—支线电缆总长度	- 5 -
2.3 DeviceNet工作方式	- 5 -
2.4 使用环境	- 5 -
2.5 EMC	- 6 -
2.6 电源	- 6 -
2.7 机械标准	- 6 -
三、使用指南	- 7 -
3.1 外观说明	- 7 -
表 2—模块状态指示灯 (MS)	- 7 -
表 3—网络状态指示灯 (NS)	- 7 -
3.2 Modbus接线	- 8 -
3.3 DeviceNet接线	- 8 -
3.4 手动设置DeviceNet地址	- 9 -
3.5 设置端口	- 9 -
3.5.1 配置超级终端	- 9 -
3.5.2 Modbus设置	- 12 -
3.5.3 设置Modbus扫描列表	- 14 -
3.5.4 DeviceNet设置	- 16 -
3.5.5 退出程序	- 17 -
3.6 终端电阻	- 17 -
3.7 网关模块参数设置	- 17 -
四、安装	- 18 -
4.1 产品清单	- 18 -
4.2 安装固定方法	- 19 -
五、DeviceNet I/O信息和参数说明	- 20 -
5.1 I/O配置	- 20 -
5.2 DeviceNet参数	- 20 -
六、运行维护及注意事项	- 25 -
七、附件介绍	- 26 -
7.1 通讯线	- 26 -
7.2 RS232/RS485 转换模块及连接线	- 26 -
7.2.1 RS232/RS485 转换模块	- 26 -
7.2.2 连接线L1	- 27 -
附 录 A：Modbus协议	- 28 -

附 录 B : DevieNet Profile文件	- 31 -
附 录 C : EDS文件	- 35 -

一、产品概述

1.1 产品简介



图 1 产品外观

DeviceNet 是一种简单的网络解决方案，在提供多供货商同零部件间的可互换性的同时，减少了配线和安装工业自动化设备的成本和时间。它的直接互连性不仅改善了设备间的通信，而且提供了相当重要的设备级诊断功能。DeviceNet 在工业控制底层，特别在工业自动化中得到大量应用。Modbus/RS485 是在变配电系统、工业自动化系统中广泛应用的通信总线，简单易用，使用广泛。MD21 型 Modbus /DeviceNet 接口的出现为用户选择不同总线解决方案提供了可能。

MD21 可分为定制型和标准型两种类型。

定制型通常为点对点，即一个 MD21 连接一个 Modbus 设备，模块内的 Firmware（固件程序）根据连接的模块实现了定制。定制型的模块优点为 (1) 使用简单，用户在工程应用时无需再编程; (2) 也不需要了解 DeviceNet 电子数据表格(EDS)的知识，因为 EDS 文件无需修改; (3) 而且用户可以指定将那些 Modbus 参数映射到 DeviceNet 参数，哪些映射到 DeviceNet I/O。 (4) 可以实现一个带 Modbus 接口的设备完全像自带 DeviceNet 一样。缺点是工程应用时缺乏灵活性，通常为点对点。

标准型是客户自己来定义如何将 Modbus 数据映射到 DeviceNet。优点是现场可修改，可以实现点对多点。用户需要做些网关的配置工作，另外需要了解一点 DeviceNet EDS 的知识。一般只能实现将 Modbus 参数映射到 DeviceNet I/O 数据。

1.2 产品特点

本模块通过 Modbus 通讯接口和带有 Modbus 通讯功能的设备进行数据交换，然后通过程序转换，把符合 DeviceNet 规范的数据通过 DeviceNet 网络进行传输。

本模块 Modbus 接口符合 MODICOM Modbus Protocol PI-MBUS-300 Rev.J。Modbus 接口为主站。

DeviceNet 通讯符合 GB/T18858.1,GB/T18858.3 及 DeviceNet Protocol Release2.0 中仅限组 2 预定义主

/从连接。DeviceNet 接口为从站。

1.3 产品用途

本产品主要用于Modbus设备到DeviceNet通讯转换, 设备Modbus规约必须符合附录A的规定。

二、技术指标

2.1 通讯速率

- DeviceNet 接口支持: 125kbit/s, 250kbit/s, 500kbit/s
- Modbus 接口缺省设置为 19200 Kbps, 8bit 数据位, 无奇偶校验位, 1bit 停止位。

2.2 DeviceNet 拓扑结构

- 干线(Trunk lines)

粗缆或者是细缆, 皆可用于构筑干线。粗缆和细缆混合使用时最大电缆距离用下列公式计算:

$$\begin{array}{ll} L_{thick} + 5 \times L_{thin} = 500m & 125kbit/s \\ L_{thick} + 2.5 \times L_{thin} = 250m & 250kbit/s \\ L_{thick} + L_{thin} = 100m & 500kbit/s \end{array}$$

这里 L_{thick} 是粗缆长度, L_{thin} 是细缆长度。

- 支线(Drop lines)

支线长度是从干线上的分接头到每个设备的收发器之间的距离, 应不超过 6m。支线电缆总的长度与波特率有关, 并且不能超过表 1 中规定的值。

表 1—支线电缆总长度

比 特 率	电 缆 长 度
125kbit/s	156m
250kbit/s	78m
500kbit/s	39m

2.3 DeviceNet 工作方式

本 DeviceNet 接口支持仅限组 2 预定义主/从连接。

2.4 使用环境

- 相对湿度: 5%至 95%的相对湿度(不凝固)
- 周围空气温度: -20℃--60℃, 且 24 小时的平均值不超过 35℃(特殊定货除外)

- 安装地点的海拔高度不超过 2000 米
- 污染等级为 3 级

2.5 EMC

- 静电放电 (ESD) 抗扰性
 - 对于非金属设备外壳用空气隙放电方法施加 $\pm 8\text{KV}$ 的测试电压。
 - 对金属设备外壳用空气隙放电方法施加 $\pm 4\text{KV}$ 的测试电压。
- 射频电磁场辐射抗扰性
 - 频率范围 80 MHz 至 1000MHz 强度为 10V/m 的调幅波。
- 电快速瞬态/脉冲群抗扰性
 - 5KHz 的 $\pm 1\text{KV}$ 最大测试电压施加在包含 CDI 通讯介质的电缆。
 - 5KHz 的 $\pm 2\text{KV}$ 最大测试电压施加在所有其它电缆和端口。
- 射频场感应的传导骚扰的抗扰性
 - 在 $150\text{KHz} \sim 80\text{MHz}$ 频率范围上 10V rms 调幅波。
- 发射
 - 按 GB4824, 组 1, A 级。
- 传导发射
 - 按 GB4824, 组 1, A 级。

2.6 电源

工作电源：直流 24V ($11\text{V} \sim 30\text{V}$)，标准工作电流 60mA ，最大为 80mA 。

2.7 机械标准

$100\text{ mm} \times 70\text{ mm} \times 25\text{ mm}$ [不包括导轨连接器]

三、使用指南

3.1 外观说明



指示灯显示说明如表 2 及表 3：

表 2—模块状态指示灯 (MS)

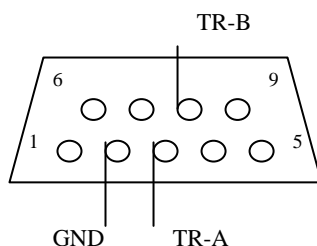
指 示 灯 状 态	含 义
关闭	未供电
绿色	工作正常
绿色闪烁	出错，不完全或不正确配置
红色闪烁	可恢复的故障，Modbus 通信故障
红	不可恢复的故障
红—绿闪烁	正进行自检

表 3—网络状态指示灯 (NS)

指 示 灯 状 态	含 义
关闭	未通过重复 MAC ID 检测或 DeviceNet 电路未供电
绿色闪烁	设备在线但没有已建立的连接
绿色	设备在线且已建立了连接
红色闪烁	一个或多个 I/O 连接已经超时
红	设备检测到一个错误，且不能进行链路通信

3.2 Modbus 接线

MD21 使用 DB9 母接头连接器:



Modbus 侧的接线规则:

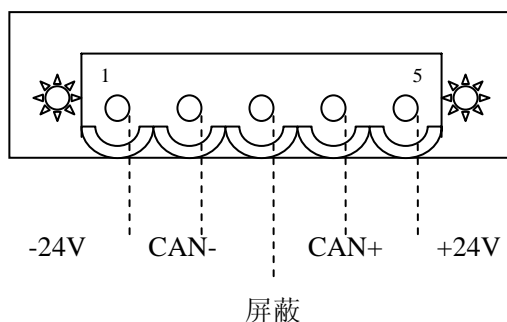
3 脚	RS485-A (Data +)
8 脚	RS485-B (Data -)
2 脚	RS485 GND

注意:

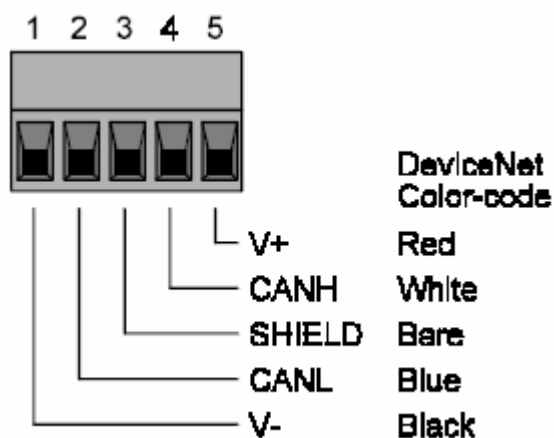
1. 一般设备只需连接 3, 8, 2 脚即可, 也可只连接 3、8 两个脚。
2. 由于配置口与 Modbus 复用 RS485 串口, 在配置工作状态下, 别的 Modbus 节点必须掉电或者与 MD21 的 RS485 口端口, 否则可能会影响配置。

3.3 DeviceNet 接线

MD21 使用开放式五针 DeviceNet 连接器:



开放五针连接器对应 DeviceNet 电缆的颜色如下, 用户也可参考模块上方的色标:



3.4 手动设置 DeviceNet 地址

- **启动:**
启动时可见闪烁的“dT”字样。
此时，立即按住外部按键超过 2.5s，可见闪烁的“AD”，和出厂设置的地址“63”，两者交替显示，则已进入 DeviceNet 地址设置模式。
- **设置:**
每次按一下按钮，则 DeviceNet 地址加 1，根据 LED 显示，调整到你想要的地址数字。
- **保存:**
按住按键超过 2.5s 保存设置，设备变为新地址并重新启动。此时，新地址存储已在非易失性存储器中。完成了 DeviceNet 地址设置的手动设置。

3.5 设置端口

3.5.1 配置超级终端

打开 Windows 自带程序超级终端 Hyper-Terminal（开始——所有程序——附件——通讯——超级终端）。波特率的设置为 19200bps，8 位数据位，无奇偶校验位，1 位停止位，无数据流控制。如下图所示。



图 2 超级终端设置

双击蓝色按键，使网关进入设置模式。



工作原理：MD21 的 Modbus 和 DeviceNet 之间的数据转换通过“映射”关系来建立。在 MD21 中有两块数据缓冲区，一块是 DEVICENET 网络输入缓冲区，另一块是 DEVICENET 网络输出缓冲区。Modbus 读取命令将读取的数据写入到网络输入缓冲区，供 DeviceNet 网络读取。Modbus 写寄存器类的命令从网络输出缓冲区取数据，通过写命令输出到相应的 Modbus 设备。



用户可以配置 48 条命令，每条命令可以一条 Modbus 命令读取一组连续的 Modbus 寄存器。

映射关系在配置模式下用户可自行组态。用户通过 RS232/485 接口可以将网关连接到 PC。通过 Windows 自带的超级终端软件可以配置网关。重新上电网关即进入配置模式。

备注：如果网关已处在正常的 DeviceNet 通讯情况下，你不能让它直接进入设置模式；快速双击按键，然后你将会看见模块状态指示灯和网络状态指示灯红绿交替闪烁。当红灯闪烁或者 DeviceNet 地址指示灯在闪烁“BR”时，及时地再次双击按键。

设置工具使用 19200-8-N-1 模式，即波特率 19200，8 位数据位，无奇偶校验位，1 位停止位。

成功进入设置模式，你将看见如下的开始菜单。



图 3: 主菜单

[以下叙述中，粗斜体是能够在 Hyper-Terminal 工具上直接看见的。]

3.5.2 Modbus 设置

开始设置 Modbus 的两个步骤:

- (1).设置 Modbus 参数
- (2).设置 Modbus 扫描指令列表

第三,第四个步骤用来显示你的 Modbus 通讯设置。

按“1”，你将会看见:



图 4: Modbus 通讯方式设置

MD21 的 Modbus 端口作为主站。Modbus 的地址通过(1). ***Change master station address.*** 被设置主站地址。项目(2). ***Change communication baudrate.*** 可以改变 Modbus 通讯波特率。你可以从这个列表中选择。

Please select communication baudrate.

Current baudrate is: 3

- (0) 1200
- (1) 2400
- (2) 9600
- (3) 19200
- (4) 28800
- (5) 57600
- (6) 115200

子菜单 3 用于设置通信时是否使用奇偶校验位

(3) ***Change parity mode.***

Please select parity mode.

Current parity mode is: 0

- (0) *NONE*
- (1) *ODD* （奇校验）
- (2) *EVEN* （偶校验）

子菜单 4 用于设置当前使用了多少个 Modbus 命令

(4) *Change used commands number.*

子菜单 5 用于改变多少时间等待 Modbus 响应超时

(5) *Change response timeout.*

子菜单 6 用于设置两条 Modbus 输入/输出命令的轮询时间间隔

(6) *Set delay between Polls.*

子菜单 7 使能/禁止 Modbus 输出命令

(7) *Enable/Disable Output Polling.*

3.5.3 设置 Modbus 扫描列表

在这个菜单中定义 Modbus 扫描命令。每条命令包括扫描哪个 Modbus 从站设备，使用哪个功能代码，读写哪个寄存器地址，以及映射到网关的哪个交换地址。如果你对 Modbus 不熟悉，请参照 Modbus 规格书。

设置 Modbus 扫描列表和映射列表，是网关设置的核心。

映射地址意思是 Modbus 的输入和输出与 DeviceNet 的输入和输出的对应关系。

在主菜单中，选择“(2). Configure Modbus scan command list.”你将见到：

Configure Modbus scan command list

Sibotech Automation Co.,Ltd 2004-2007

- (1) *Add a new command.***
- (2) *Delete a command.***
- (3) *Verify configuration conflict of memory map.***
- (4) *Return to main menu.***

Please enter your selection:

请输入你的选择：

总共可配置 48 条扫描命令。

举例来说，你可以定义一个扫描指令去读一个地址 3 的 Modbus 节点的 10 个寄存器，从 100 到 109，用以下方式：

选择“1”，增加一个新的命令列表。

Maxim configuration command number is 48.

New command No. is: 39

Please enter slave address (0-247):

选择“3”，然后“Enter”，就是读 Modbus 节点地址 3，你将会看见：

Please enter slave address (0-247): 50

Your enter is: 50

Please enter Function code

(Support FC is: 01;02;03;04;05;06;15;16):

选择“3”，意味着选择功能代码 03 的 Modbus 命令，3 号命令可以读入一组寄存器，需要指定寄存器组起始地址和读入寄存器的个数。

首先需要输入寄存器地址的开始地址，你将会看见

Please enter Starting Address: 0 假设是 0，确定，

然后是输入寄存器个数，每条命令限定读入 32 个寄存器

Please enter Number of Regs (1-32):

之后，你输入 10，然后是选择映射到 DeviceNet 输入的哪个字节块，

Please enter data mapping address(1-159):

Notes: Don't Map byte 0. It's for Modbus Status (1-159)

Please input Low byte of Number of Points:

选择合适的数字，这里我们输入“10”。

这样，数据映射到了 DeviceNet 的输入。Modbus 命令 03 是一个输入命令，所以数据将被映射到 DeviceNet 的输入。最大的 DeviceNet 的输入是 160。你可以选择的位置是 1~159。0 号输入用来表示 Modbus 的状态，如果哪一个 Modbus 扫描命令出错了，Modbus 状态字节就显示该命令的序号。如果正确，状态为 0。

注意：请不要选择“0”。这个数字有特殊的意义，它表示 Modbus 的状态。

最后是扫描周期，以 ms 毫秒为单位。

Please select scan cycle:

(0) 50ms (1) 200ms (2) 1000ms

意思是网关发送命令的周期。

你将会看见：

Create success!

Press any key to continue...

如果你要知道参数是否被正确设置，你可以在主菜单中选择

(4). Display Modbus scan command list.

回到主菜单，然后，

*Display Modbus scan command list**Sibotech Automation Co.,Ltd 2004-2007*

*(1) Display command list.**(2) Return main menu.**Please input your selection*

选择 1:

Please input your selection: 1

然后会按照顺序显示配置的命令

<i>No.</i>	<i>SlA</i>	<i>FC</i>	<i>StarA</i>	<i>No.P</i>	<i>Bytes</i>	<i>DA/SA</i>	<i>Bits</i>	<i>SC(ms)</i>
<i>1</i>	<i>11</i>	<i>3</i>	<i>0</i>	<i>12</i>	<i>1</i>		<i>200</i>	
<i>2</i>	<i>11</i>	<i>3</i>	<i>13</i>	<i>1</i>	<i>25</i>		<i>200</i>	
<i>3</i>	<i>11</i>	<i>3</i>	<i>15</i>	<i>3</i>	<i>27</i>		<i>200</i>	
<i>4</i>	<i>12</i>	<i>3</i>	<i>0</i>	<i>12</i>	<i>33</i>		<i>200</i>	
<i>5</i>	<i>12</i>	<i>3</i>	<i>13</i>	<i>1</i>	<i>57</i>		<i>200</i>	
<i>6</i>	<i>12</i>	<i>3</i>	<i>15</i>	<i>3</i>	<i>59</i>		<i>200</i>	
<i>7</i>	<i>13</i>	<i>3</i>	<i>0</i>	<i>6</i>	<i>66</i>		<i>200</i>	
<i>8</i>	<i>13</i>	<i>3</i>	<i>7</i>	<i>5</i>	<i>78</i>		<i>200</i>	

*Press 'Q' exit, press any key to next page.**Please enter your selection:**Press any key to continue...*

按任意键可以见到下一屏。

3.5.4 DeviceNet 设置

接下来两条被用来设置 DeviceNet 配置。

(5) 设置 DeviceNet 参数。

(6) 显示 DeviceNet 参数。

输入 5, 你将会看见

**** Configure DeviceNet Menu ****

*(1). Change adaptor vendor id to your id number if you want.**(2). Change INPUT bytes number of DeviceNet I/O connection.**(3). Change OUTPUT bytes number of DeviceNet I/O connection.**(4). Return main menu.*

(1). Change adaptor vendor id to your id number if you want.的意思是, 如果你想使用你自己的 DeviceNet 供应 ID, 你可以改变它。

(2) 和 (3) 的意思是, 你可以改变我们产品的 DeviceNet I/O 的输入和输出位数字。如你所知, 如果你只想输出四个字节, 网关会把 60 个字节填补到 64 个字节。这样对 DeviceNet 网络容量和 DeviceNet 控制器/扫描仪是没有效率的, 你可以重新设置它。但是请注意协调 Modbus-DeviceNet 数据映射。你的 DeviceNet 输入/输出数据的位数必须大于被使用的映射缓冲器的位数。

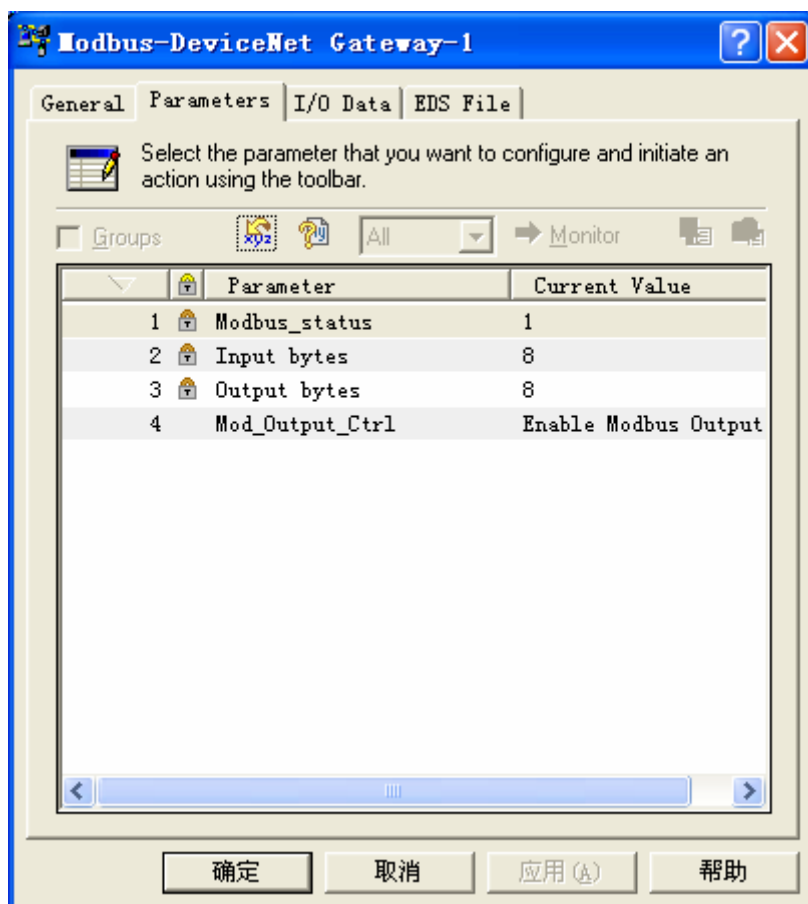
3.5.5 退出程序

在主菜单，输入“7”来退出程序。你需要重启网关去使所有的改变起作用。

3.6 终端电阻

DeviceNet 网络需要在网络的最远的两个端点处各接一个 120ohm 的终端电阻。Modbus 也需要终端电阻，在我们的设备 MD21 中，Modbus 网络端口已经有一个 RS485 终端电阻，用户只需再加一个 RS485 的终端电阻。

3.7 网关模块参数设置



Modbus_Status: Modbus 通信的状态，如果一直为零，那么通信正常，如果出现某个非零的数字，表示该序号的 Modbus 命令通信出现故障。

Input Bytes: DeviceNet I/O 连接输入字节数

Output Bytes: DeviceNet I/O 连接输出字节数，用户在配置 DeviceNet 主站扫描列表时，可以用上述两个参数来确定输入输出字节数，如果输入输出字节这里显示的数目与主站扫描列表里面的数目不匹配，将无法建立 DeviceNet IO 连接。

Mod_output_ctrl: Modbus 输出方式控制：

Continuous Output 连续输出

Disable Output 禁止输出

注意：如果为禁止输出，即使配置了 Modbus 输出命令，网关也不会发出 Modbus 输出命令。

为了保证输出的安全性，如果 PLC 没有有效的输出数据（如 PLC 在编程状态或者设备 DeviceNet 没有连接上）那么 Modbus 输出指令也不会发出。

通过 RsNetworx 或者本公司的 DNetStart 软件，用户可以手动修改这个参数。用户如使用 ControlLogix，可以通过 PLC 的 MSG 模块发送显式报文消息，自动修改这个参数，立即生效。

备注：

本产品 DeviceNet 端口支持波特率自适应。

Modbus 端口的波特率缺省为 19.2K，通信方式为无校验位、1 位停止位。特殊要求可在订货时说明。

如果 Modbus 通信出现问题，DeviceNet I/O 数据不能有效采集，则 I/O 扫描获得的数据为零。

四、安装

4.1 产品清单

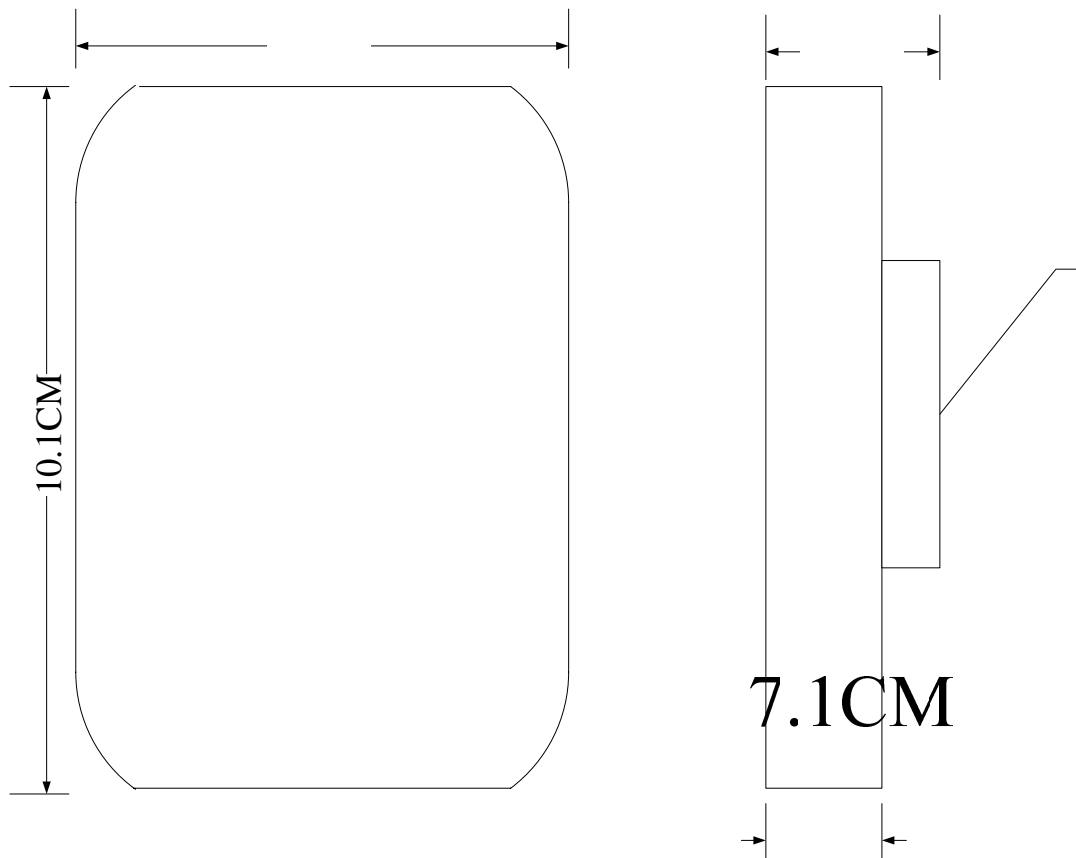
- | | |
|-------------------------|-----|
| ● Modbus/DeviceNet 接口模块 | 1 台 |
| ● 合格证 | 1 份 |
| ● 产品手册光盘 | 1 张 |

注意事项：

在打开产品包装时，请仔细检查接口模块是否有损坏，如有任何损坏请及时通知我公司或代理商，并请保留损坏的产品的外包装。如系我公司或代理商责任，我公司将及时予以更换。

4.2 安装固定方法

35mm 标准导轨安装。



五、DeviceNet I/O 信息和参数说明

5.1 I/O 配置

I/O 输出字节可配置为 8 个字节、16 个字节、32 个字节、64 个字节、96 个字节、128 个字节或 160 个字节。

I/O 输出

I/O 输出字节可配置为 8 个字节、16 个字节、32 个字节、64 个字节、96 个字节、112 个字节。

5.2 DeviceNet 参数

Modbus_Status: Modbus 通信的状态，如果一直为零，那么通信正常，如果出现某个非零的数字，表示该序号的 Modbus 命令通信出现故障。

Input Bytes: DeviceNet I/O 连接输入字节数

Output Bytes: DeviceNet I/O 连接输出字节数

以上两个参数必须与 RSNetWorx 等组态软件的 DeviceNet 主站扫描列表中的配置/输入输出字节一致，否则将无法连接。

Mod_output_ctrl: Modbus 输出命令使能/禁止，如果为禁止，即使配置了 Modbus 输出命令，网关也不会发出 Modbus 输出命令。

5.3 DeviceNet 网络配置说明

用户需要将光盘中的*.EDS 文件注册到 DeviceNet 组态软件，才能通过网络组态软件配置。

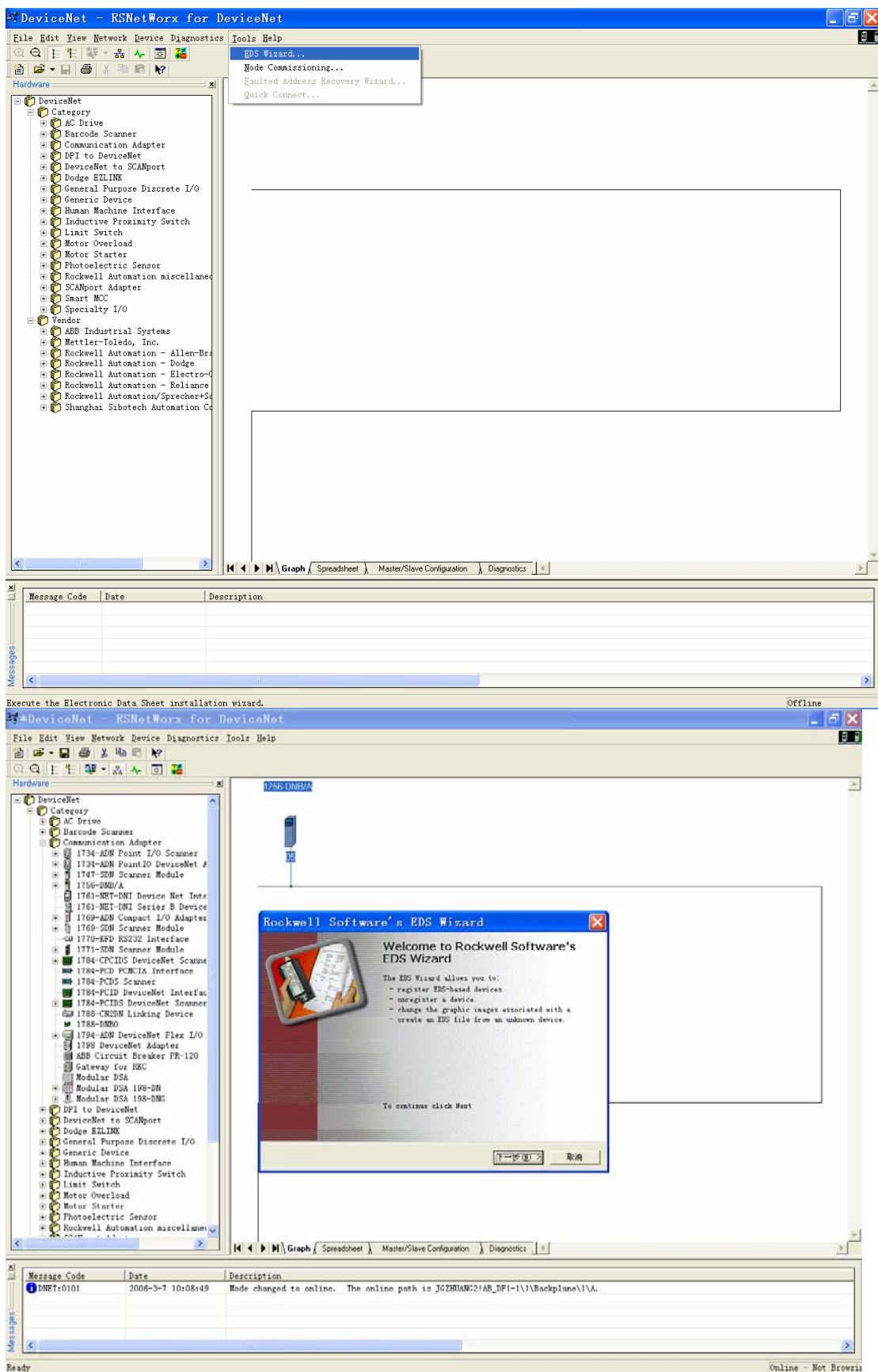
5.3.1. EDS 注册

EDS (Electronic Data Sheet) 电子数据表格是支持 DeviceNet 的设备的网络功能的全面描述。相当于 Windows 上设备的驱动程序。用户需要把 EDS 文件注册到 DeviceNet 网络组态软件，如 RsNetWorx 等，才可以通过网络组态软件进行进一步的配置。

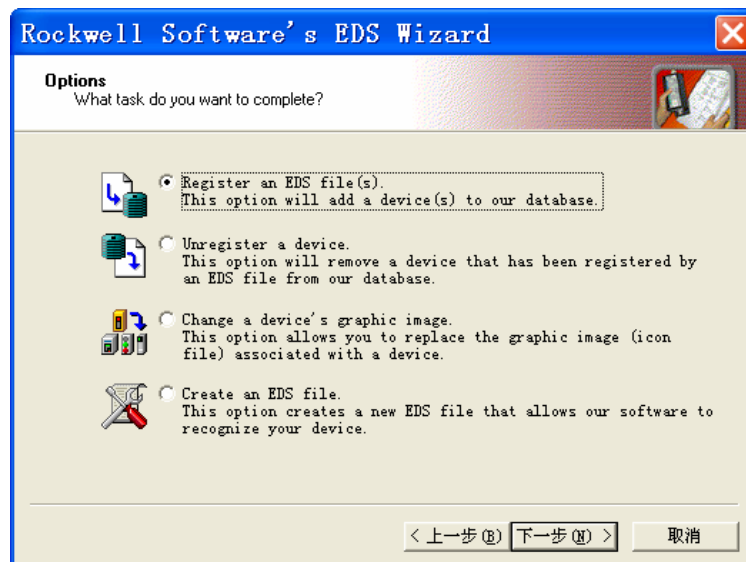
下面我们以常用的 Rockwell 公司的 RsNetWorx 为例（版本 4.12.0），说明如何注册，进一步的详细说明，请参考您所用的网络组态软件的说明书。

Step 1: 创建一个新的网络配置文件

Step 2: 选择 EDS 操作向导，在“Tool”菜单中，选择“EDS-Wizard”，您会看到：



Step 3: 选择下一步:



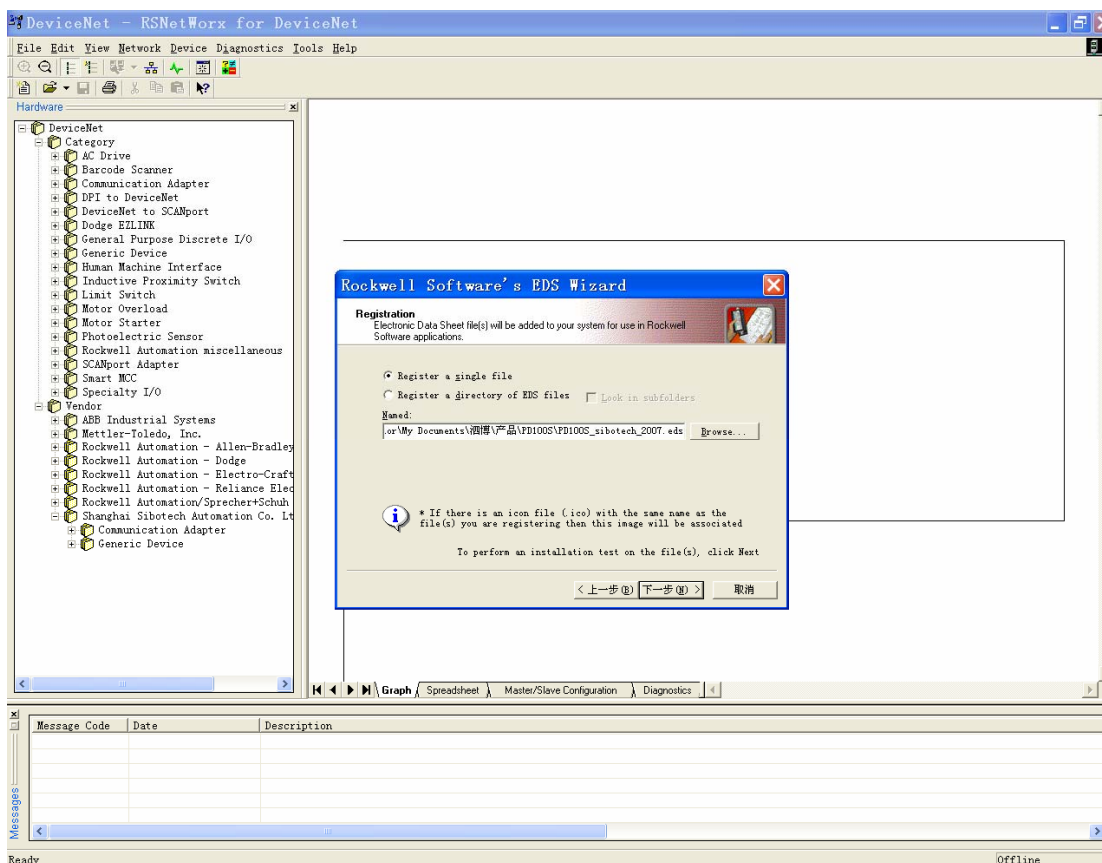
Step 4: 注册网关 MD-21

如上图所示，选择“Register an EDS file”，将显示下面界面：

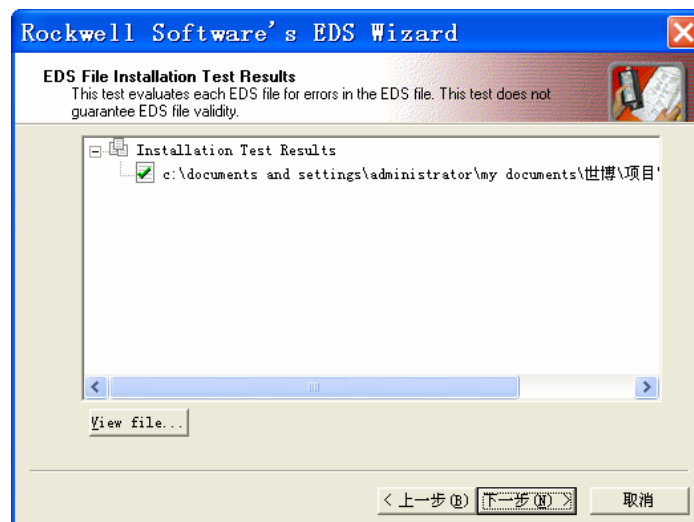
请注册我们提供的 md21_2007.EDS 文件，根据您存放 EDS 文件的位置，选中该文件。



Step 5: 确认注册所选择的文件；

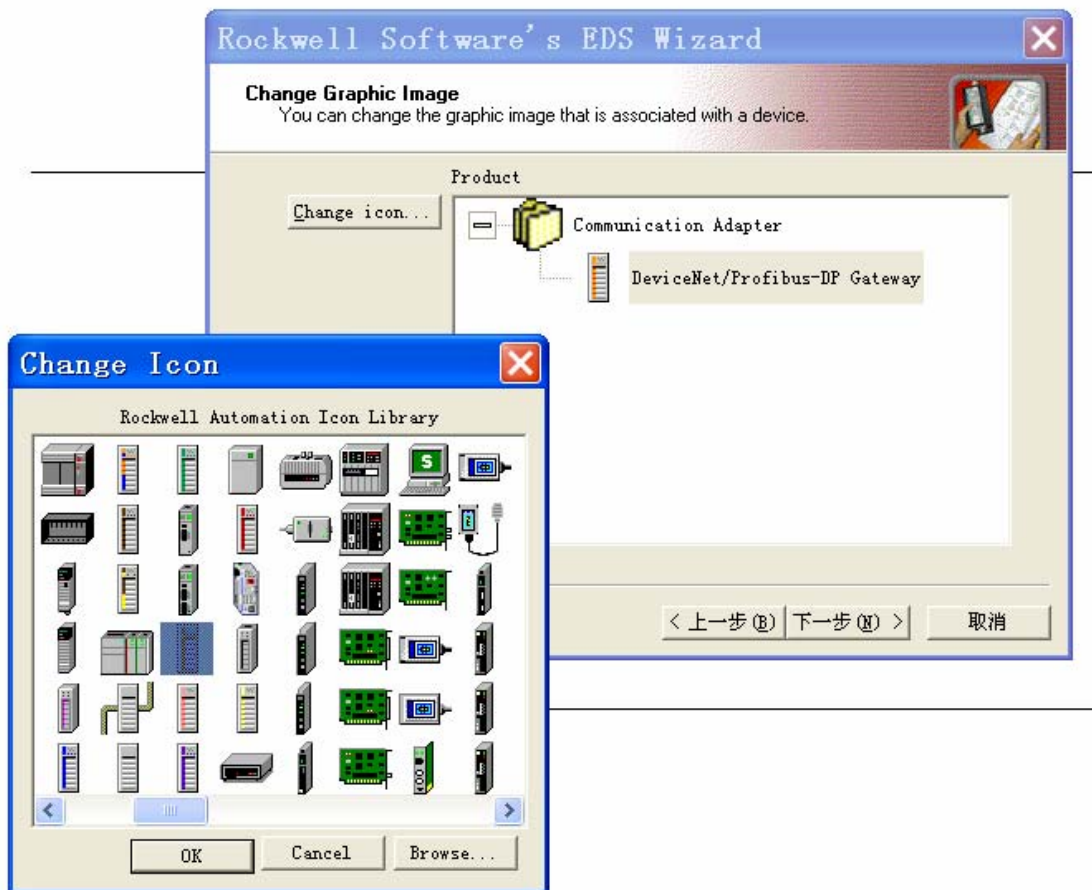


按下一步:

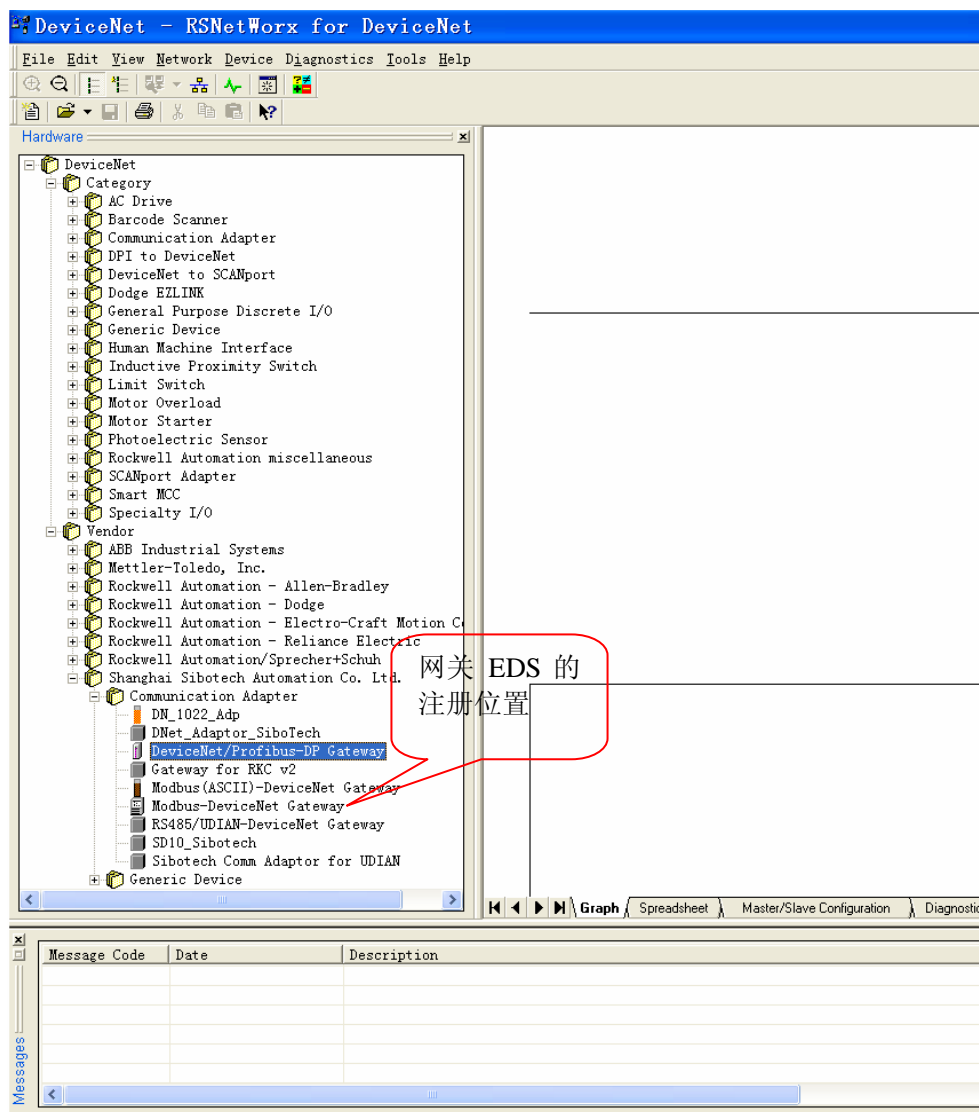


Step 6: 选择图标。

下面网络组态软件将提示您该设备在设备库中存放的类别，按缺省值确认，在这个过程中可以选择图标。



到这里，该设备已经成功地注册到了组态软件的设备库中的图示位置。



接下来，您将网关设备 MD20 接到 DeviceNet 网络上，按下 RsNetWorx 的“浏览”按钮，或者在菜单中选择“Network-Online”，您的网关设备将被系统扫描到并且正确的识别。

六、运行维护及注意事项

防止重压，以防面板损坏。

正常运行中可查看本产品的信息指示灯是否故障信息，发现问题及时分析处理。

七、附件介绍

7.1 通讯线

与 DeviceNet 干线连接需选用 3048A 五芯屏蔽线 (符合 GB/T18858.3 规定)。与 Modbus 设备接口需选用 A 类屏蔽线



A 类屏蔽线



3048A 五芯屏蔽线

7.2 RS232/RS485 转换模块及连接线

7.2.1 RS232/RS485 转换模块

UT-201 是一款商业级袖珍型接口转换器，能够将单端的 RS232 信号转换为平衡差分的 RS485 信号。具体内容可参照该产品包装内的说明书。



以下是该产品的端口介绍。

RS-232C 引脚分配:

DB9 Female (PIN)	RS-232C 接口信号
1	保护地
2	接收数据 SIN (RXD)
3	发送数据 SOUT (TXD)
4	数据终端准备 DTR
5	信号地 GND

6	数据装置准备 DSR
7	请求发送 RTS
8	清除发送 CTS
9	响铃指示 RI

RS-485 输出信号及接线端子引脚分配：

DB9 Male (PIN)	输出信号	RS-484 半双工接线
1	T/R+	RS-485 (A+)
2	T/R-	RS-485 (B-)
3	RXD+	空
4	RXD-	空
5	GND	地线
6	VCC	+5V 备用电源输入

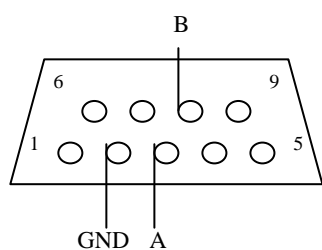
7.2.2 连接线 L1

L1 是连接 UT-201 与 MD21 的线。



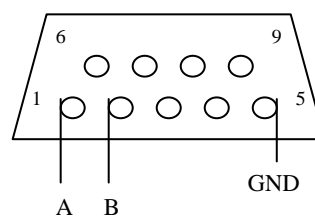
线的两端接线方法如下：

MD21 侧的阳端子



引脚 2 为 GND
引脚 3 为 A
引脚 8 为 B

UT-201 侧的阴端子



引脚 1 为 A
引脚 2 为 B
引脚 5 为 GND

附录 A: Modbus 协议

Modbus-RTU 协议:

说明：与本产品通讯的设备必须带有 Modbus 接口，同时设备 Modbus 协议必须符合下面的规定，本公司提供用户定制服务。

1. 协议概述

物理层：传输方式：RS485

通讯地址：3-126

通讯波特率：9600BPS，19200BPS

通讯介质：屏蔽双绞线

传输方式：主从半双工方式。

协议在一根通讯线上使用应答式连接（半双工），这意味着在一根单独的通讯线上信号沿着相反的两个方向传输。首先，主计算机的信号寻址到一台唯一的终端设备（从机），然后，在相反的方向上终端设备发出的应答信号传输给主机。

协议只允许在主计算机和终端设备之间，而不允许独立的设备之间的数据交换，这就不会在使它们初始化时占据通讯线路，而仅限于响应到达本机的查询信号。

一个数据帧格式：

1 位起始位，8 位数据，1 位停止位。

一个数据包格式

地址	功能码	数据	校验码
8-Bits	8-Bits	N x 8-Bits	16-Bits

协议详细定义了校验码、数据序列等，这些都是特定数据交换的必要内容。

当数据帧到达终端设备时，它通过一个简单的“口”进入寻址到的设备，该设备去掉数据帧的“信封”（数据头），读取数据，如果没有错误，就执行数据所请求的任务，然后，它将自己生成的数据加入到取得的“信封”中，把数据帧返回给发送者。返回的响应数据中包含了以下内容：终端从机地址(Address)、被执行了的命令(Function)、执行命令生成的被请求数据(Data)和一个校验码(Check)。发生任何错误都不会有成功的响应。

地址（Address）域

地址域在帧的开始部分，由 8 位（0~255）组成，这些位标明了用户指定的终端设备的地址，该设备将接收来自与之相连的主机数据。每个终端设备的地址必须是唯一的，仅仅被寻址到的终端会响应包含了该地址的查询。当终端发送回一个响应，响应中的从机地址数据便告诉了主机哪台终端正与之进行通信。

功能（Function）域

功能域代码告诉了被寻址到的终端执行何种功能。表 1-1 列出了所有的功能码、它们的意义及它们的初始功能。

表 1-1 功能码

代码	意义	行为
03	读数据	获得一个或多个寄存器的当前二进制值
06	预置单寄存器	放置一个特定的二进制值到一个单寄存器中
16	预置多寄存器	放置特定的二进制值到一系列多寄存器中

数据域

数据域包含了终端执行特定功能所需要的数据或者终端响应查询时采集到的数据。这些数据的内容可能是数值、参考地址或者极限值。例如：功能域码告诉终端读取一个寄存器，数据域则需要指明从哪个寄存器开始及读取多少个数据，内嵌的地址和数据依照类型和从机之间的不同能力而有

所不同。

错误校验域

该域允许主机和终端检查传输过程中的错误。有时，由于电噪声和其它干扰，一组数据在从一个设备传输到另一个设备时在线路上可能会发生一些改变，出错校验能够保证主机或者终端不去响应那些传输过程中发生了改变的数据，这就提高了系统的安全性和效率，出错校验使用了 16 位循环冗余的方法。

[注] 发送序列总是相同的 – 地址、功能码、数据和与方向相关的出错校验。

错误检测

循环冗余校验（CRC）域占用两个字节，包含了一个 16 位的二进制值。CRC 值由发送设备计算出来，然后附加到数据帧上，接收设备在接收数据时重新计算 CRC 值，然后与接收到的 CRC 域中的值进行比较，如果这两个值不相等，就发生了错误。

CRC 运算时，首先将一个 16 位的寄存器预置为全 1，然后连续把数据帧中的 8 位字节与该寄存器的当前值进行运算，仅仅每个字节的 8 个数据位参与生成 CRC，起始位和终止位以及可能使用的奇偶位都不影响 CRC。

在生成 CRC 时，每个 8 位字节与寄存器中的内容进行异或，然后将结果向低位移位，高位则用“0”补充，最低位（LSB）移出并检测，如果是 1，该寄存器就与一个预设的固定值进行一次异或运算，如果最低位为 0，不作任何处理。

上述处理重复进行，知道执行完了 8 次移位操作，当最后一位（第 8 位）移完以后，下一个 8 位字节与寄存器中的当前值进行异或运算，同样进行上述的另一个 8 次移位异或操作，当数据帧中的所有字节都作了处理，生成的最终值就是 CRC 值。

生成一个 CRC 的流程为：

预置一个 16 位寄存器为 0FFFFH（全 1），称之为 CRC 寄存器。

把数据帧中的第一个 8 位字节与 CRC 寄存器中的低字节进行异或运算，结果存回 CRC 寄存器。将 CRC 寄存器向右移一位，最高位填以 0，最低位移出并检测。

如果最低位为 0：重复第三步（下一次移位）。

如果最低位为 1：将 CRC 寄存器与一个预设的固定值（0A001H）进行异或运算。
重复第三步和第四步直到 8 次移位。这样处理完了一个完整的八位。
重复第 2 步到第 5 步来处理下一个八位，直到所有的字节处理结束。
最终 CRC 寄存器得值就是 CRC 的值。

2. 应用层功能详解

第一章已经简述了协议和数据帧，使用此软件的程序员可以使用下述的方法以便通过协议正确的建立他们的特定应用程序。

本章所述协议将尽可能的使用如图 2-1 所示的格式，（数字为 16 进制）。

地址	功能码	变量起始地址高字节	变量起始地址低字节	变量的个数高字节	变量的个数低字节	校验码低字节	校验码高字节
03H	03H	00H	01H	00H	03H	55H	E9H

图 2-1 协议例述

读数据（功能码 03）

查询

图 2-2 的例子是从 03 号从机读 3 个采集到的基本数据 U1,U2,U3, U1 的地址为 0001H, U2 的地址为 0002H, U3 的地址为 0003H,

地址	功能码	变量起始地址高字节	变量起始地址低字节	变量的个数高字节	变量的个数低字节	校验码低字节	校验码高字节
03H	03H	00H	01H	00H	03H	55H	E9H

图 2-2 读 Uca 和 Ia 的查询数据帧响应

响应包含从机地址、功能码、数据的数量和 CRC 错误校验。

图 2-3 的例子是读取 U1,U2,U3 的响应。

地址	功能码	变量的总字节数	变量值高字节	变量值低字节	变量值高字节	变量值低字节	变量值高字节	变量值低字节	校验码低字节	校验码高字节
03H	03H	06H	01H	7CH	01H	7DH	01H	7CH	F9H	9BH

图 2-3 读 U1,U2,U3 的响应数据帧

2. 2 预置多寄存器（功能码 10）

查询

功能码 10H 允许用户改变多个寄存器的内容，设备可从任何地址开始设置最多 16 个变量的值。

控制器是以动态扫描方式工作的，任何时刻都可以改变寄存器内容。

图 2-4 是修改 3 号从站设备的负载监控 1 和负载监控 2 的动作及延时时间的设定值，其中负载监控 1 的动作设定值地址为 2AH，延时时间的设定值为 2BH，负载监控 2 的动作设定值地址为 2CH，延时时间的设定值为 2DH。

地址	功能码	变量起始地址高字节	变量起始地址低字节	变量的个数高字节	变量的个数低字节	变量的总字节数	变量值高字节	变量值低字节	变量值高字节	变量值低字节	变量值高字节	变量值低字节	变量值高字节	变量值低字节	校验码低字节	校验码高字节
03H	10H	00H	2AH	00H	04H	08H	07H	D0H	00H	0AH	07H	0D0H	00H	0AH	25H	7CH

图示 2-4 修改负载监控 1 和负载监控 2 的动作值及延时时间的设定值
响应

地址	功能码	变量起始地址高字节	变量起始地址低字节	变量的个数高字节	变量的个数低字节	校验码低字节	校验码高字节
03	10H	00H	2AH	00H	04H	EBH	8DH

图示 2-5 修改负载监控 1 和负载监控 2 的动作值及延时时间的设定值的响应

2. 3 预置单寄存器（功能码 06）

查询

功能码 06 允许用户改变单个寄存器的内容，DAE 系统内部的任何单寄存器都可以使用此命令来改变其值。既然仪器是以动态扫描方式工作的，任何时刻都可以改变单寄存器内容。

下面的例子是请求 03 号从机修改过载动作设定值 Ir1，Ir1 地址是 002EH。

地址	功能码	变量起始地址高字节	变量起始地址低字节	变量值高字节	变量值低字节	校验码低字节	校验码高字节
03H	06H	00H	2EH	07H	0D0H	EBH	8DH

图示 2-6 修改过载动作设定值 Ir1

响应

对于预置单寄存器请求的正常响应是在寄存器值改变以后将接收到的数据传送回去。

地址	功能码	变量起始地址高字节	变量起始地址低字节	变量值高字节	变量值低字节	校验码低字节	校验码高字节
03H	06H	00H	2EH	07H	0D0H	EBH	8DH

图示 2-7 图示 2-6 修改过载动作设定值 Ir1

附录 B: Devicenet Profile 文件

DeviceNet 信息

设备类型为 12, 通信适配器。
产品代码 12。

DeviceNet 对象

Modbus/DeviceNet 接口支持以下对象:

类	对象
0x01	标识对象
0x02	报文路由对象
0x03	DeviceNet 对象
0x04	组合对象
0x05	连接对象
0xA2	应用对象

标识对象—类代码 0x01

类属性不支持

实例属性:

属性 ID	访问规则	名称	数据类型	属性值
1	读	生产商 ID	UINT	1016
2	读	设备类型	UINT	
3	读	产品代码	UINT	
4	读	版本 主要版本 次要版本	结构: UINT UINT	1 1
5	读	状态	WORD	0=没有分配 1=已分配
6	读	序列号	UDINT	唯一号码
7	读	产品名称: 字符串长度 ASCII 码串	结构: USINT STRING	
8	读	状态	USINT	0=不存在 1=设备自检 2=待机 3=可操作 4=重要可恢复故障 5=重要不可恢复故障

标识对象支持的公共服务:

服务代码	类	实例	服务名称
0X0E	不支持	支持	单个属性值(get_attribute_single)
0X05	不支持	支持	复位(reset)

报文路由对象—类代码 0X02

不支持任何实例属性

DeviceNet 对象—类代码 0X03

类属性

属性 ID	访问规则	名称	数据类型	属性值
1	读	版本	INT	

实例属性:

属性 ID	访问规则	名称	数据类型	属性值
1	读/设置	MAC ID	UINT	0—63
2	读	波特率	USINT	0=125K 1=250K 2=500K
5	读	分配信息: 分配连接选择 主站 MAC ID	结构: BYTE UINT	分配字节 地址=0—63 255=未分配
6	读	MAC ID 开关变化	BOOL	0=没有变化 1=从最近一次上电或复位以来已经改变

分配字节:

Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
N/a	应答抑制	循环	状态改变	多点轮询	位选通	轮询信息	显式信息

DeviceNet 对象支持的公共服务

服务代码	类	实例	服务名称
0X0E	支持	支持	读单个属性值(get_attribute_single)
0X10	不支持	支持	设置个属性值(set_attribute_single)
0X4B	不支持	支持	分配主从连接组 (allocate_master/slave_connection_set)
0X4C	不支持	支持	释放主从连接组 (release_master/slave_connection_set)

组合对象—类代码 0x04

不支持类属性

实例 120 (20byte)

属性 ID	访问规则	名称	数据类型	属性值
3	读	数据	Array of UINT	N/A

实例 130 (20byte)

属性 ID	访问规则	名称	数据类型	属性值
3	读	数据	Array of UINT	N/A

组合对象支持的公共服务:

服务代码	类	实例	服务名称
0X0E	不支持	支持	单个属性值(get_attribute_single)

连接对象—类代码 0x05

不支持类属性

连接对象实例 1 属性 (显示报文连接)

属性 ID	访问规则	名称	数据类型	属性值
1	读	状态	USINT	0=不存在 3=已建立
2	读	实例类型	USINT	0=显式报文

3	读	传输类触发	BYTE	0x83(服务器、类 3)
4	读	生产连接 ID	UINT	10xxxxx011(xxxxx=address)
5	读	消费连接 ID	UINT	10xxxxx100(xxxxx=address)
6	读	初始通信特征	BYTE	0x21 2=从站显式报文 (生产, 源) 1=主站显式请求 (消费, 目的)
7	读	生产连接数据大小	UINT	
8	读	消费连接数据大小	UINT	
9	读/设置	期望信息包速率	UINT	最小时间单位为 10ms (缺省 2500ms)
12	读/设置	看门狗定时器动作	USINT	1=自动删除 3=延迟删除
13	读	生产连接路径长度	UINT	0
14	读	生产连接路径		空
15	读	消费连接路径长度	UINT	0
16	读	消费连接路径		空

连接对象实例 2 属性(I/O 轮询连接)

属性 ID	访问规则	名称	数据类型	属性值
1	读	状态	USINT	0=不存在 1=正在配置 3=已建立 4=过时
2	读	实例类型	USINT	0=I/O 报文
3	读	传输类触发	BYTE	如果分配选择=轮询 或 如果分配选择=! 轮询 &&!应答抑制: 则为 0x82(服务器类 2) 如果分配选择=! 轮询 &&应答抑制: 则为 0x80(服务器类 0)
4	读	生产连接 ID	UINT	0111xxxxx(xxxxx=address)
5	读	消费连接 ID	UINT	10xxxxx101(xxxxx=address)
6	读	初始通信特征	BYTE	0x21 2=主站 POLL 响应 1=主站 POLL 消费
7	读	生产连接数据大小	UINT	256
8	读	消费连接数据大小	UINT	256
9	读/设置	期望信息包速率	UINT	最小时间单位为 10ms
12	读/设置	看门狗定时器动作	USINT	0=转变为超时 1=自动删除 2=自动复位
13	读	生产连接路径长度	UINT	7
14	读/设置	生产连接路径	EPATH	[63hex][hex string]
15	读	消费连接路径长度	UINT	7
16	读/设置	消费连接路径	EPATH	[63hex][hex string]

连接对象支持的公共服务:

服务代码	类	实例	服务名称
0x0E	不支持	支持	读单个属性(get_attribute_single)

0x10	不支持	支持	设置单个属性(set_attribute_single)
------	-----	----	------------------------------

应答处理对象- 类代码 0X2B

类属性不支持

应答处理对象支持下列实例属性：

属性 ID	访问规则	名称	数据类型	属性值
1	读/设置	应答定时器	UINT	16
2	读/设置	重发次数限制	SINT	1
3	读/设置	状态改变生产连接实例	UINT	4

应答处理对象支持的公共服务：

服务代码	类	实例	服务名称
0x0E	不支持	支持	读单个属性值 (get_attribute_single)
0x10	不支持	支持	设置单个属性值 (set_attribute_single)

附 录 C: EDS 文件

\$ DeviceNet Electronic Data Sheet for Modbus-DeviceNet Gateway
\$ Electronic Data Sheet MD21-2007
\$ Copyright (C) 2004-2007 Shanghai Sibotech Automation Co. Ltd.

[File]

DescText = "Modbus-DeviceNet Gateway";
CreateDate = 03-01-05;
CreateTime = 08:57:44;
ModDate = 01-10-07;
ModTime = 10:47:10;
Revision = 1.2;

[Device]

VendCode = 1016;
VendName = "Shanghai Sibotech Automation Co. Ltd.";
ProdType = 12;
ProdTypeStr = "Communication Adapter";
ProdCode = 12;
MajRev = 1;
MinRev = 2;
ProdName = "Modbus-DeviceNet Gateway";
Catalog = "MD21";

[IO_Info]

Default = 0x0001;

PollInfo = 0x0001, 7, 6;

Input1=
8, 0, 0x0001,
"Run-time measurements and State",
6, "20 04 24 74 30 03",
"8 bytes";

Input2=
16, 0, 0x0001,
"Network Input 2",
6, "20 04 24 75 30 03",
"16 Bytes";

Input3=
32, 0, 0x0001,
"Network input 3",
6, "20 04 24 76 30 03",
"32 Bytes";

Input4=
64, 0, 0x0001,
"Network input 4",
6, "20 04 24 77 30 03",
"64 Bytes";

```
Input5=
96, 0, 0x0001,
"Network Input 5",
6, "20 04 24 78 30 03",
"96 Bytes";
```

```
Input6=
128, 0, 0x0001,
"Network Input 6",
6, "20 04 24 79 30 03",
"128 Bytes";
```

```
Input7=
160, 0, 0x0001,
"Input7",
6, "20 04 24 7A 30 03",
"Input 7 160bytes";
```

```
Output1=
8, 0, 0x0001,
"Network Output 1",
6, "20 04 24 80 30 03",
"8 Bytes";
```

```
Output2=
16, 0, 0x0001,
"Network Output 2",
6, "20 04 24 81 30 03",
"16 Bytes";
```

```
Output3=
32, 0, 0x0001,
"Network Output 3",
6, "20 04 24 82 30 03",
"32 Bytes";
```

```
Output4=
64, 0, 0x0001,
"Network Output 4",
6, "20 04 24 83 30 03",
"64 Bytes";
```

```
Output5=
96, 0, 0x0001,
"Network Output 5",
6, "20 04 24 84 30 03",
"96 Bytes";
```

```
Output6=
112, 0, 0x0001,
"Network Output 6",
6, "20 04 24 85 30 03",
```

"112 Bytes";

[ParamClass]

MaxInst = 4;
Descriptor = 0x0001;
CfgAssembly = 0;

[Params]

Param1=
0,
6, "20 A2 24 01 30 64",
0x0030, 4, 1,
"Modbus_status",
"" ,
"Status of Modbus, OK or The time out error Command No.",
0, 1, 1,
1, 1, 1, 0,
0, 0, 0, 0,
0;

Param2=
0,
6, "20 A2 24 01 30 65",
0x0002, 8, 1,
"Input bytes",
"" ,
"Number of poll input connection bytes",
0, 255, 0,
1, 1, 1, 0,
0, 0, 0, 0,
0;

Param3=
0,
6, "20 A2 24 01 30 66",
0x0002, 8, 1,
"Output bytes",
"" ,
"Number of poll output connection bytes",
0, 255, 0,
1, 1, 1, 0,
0, 0, 0, 0,
0;

Param4=
0,
6, "20 A2 24 01 30 67",
0x0002, 8, 1,
"Mod_Output_Ctrl",
"" ,
"Control of Modbus Output Commands",
0, 1, 1,

```
1, 1, 1, 0,  
0, 0, 0, 0,  
0;
```

[EnumPar]

```
Param2=  
"8",  
"16",  
"32",  
"64",  
"96",  
"128",  
"160";
```

```
Param3=  
"8",  
"16",  
"32",  
"64",  
"96",  
"112";
```

```
Param4=  
"Disable Modbus Output",  
"Enable Modbus Output";
```

[Groups]