

L_mag 电磁流量计转换器 Modbus 通讯协议

V77

2015-7-16

目录

第一章 概述.....	3
1.1 技术参数.....	3
1.2 组网说明.....	3
1.3 数据帧格式.....	3
第二章 寄存器地址表.....	5
第三章 数据解析方法.....	6
3.1 瞬时流量.....	6
3.2 瞬时流速.....	7
3.3 正向累计量整数部分.....	8
3.4 流量单位.....	9
3.5 空管报警.....	10
第四章 应用举例.....	11
4.1 命令的发送与接收.....	11
4.2 接收到的数据的含义.....	12
4.3 接收到数据的解析.....	13
第五章 常见问题处理方法.....	16
5.1 测试软件使用方法.....	16
5.2 从站不响应.....	17
5.3 数据解析异常.....	17
附录.....	20
附录一 Modbus poll 通讯示例.....	20
附录二 modscan32 通讯示例.....	22
附录三 西门子 200 PLC 通讯示例.....	24
附录四 组态王通讯示例.....	27
附录五 力控 6.1 通讯示例.....	32
附录六 MCGS 通讯实例.....	36
附录七 VB 例程.....	41
附录八 C 语言例程.....	47

第一章 概述

L-mag 电磁流量计具有标准的 RS-485 通讯接口，采用 Modbus-RTU 标准协议，可以采集瞬时流量，瞬时流速，累积流量等参数。

1.1 技术参数

L-mag 电磁流量计的 Modbus 协议采用 04 号功能码读取数据，支持波特率 1200，2400，4800，9600，19200。

串口参数为：1 位起始位 8 位数据位 1 位停止位 N 无校验。

1.2 组网说明

L-mag 电磁流量计标准 MODBUS 通讯网络是总线型网络结构，支持 1 到 99 个电磁流量计组网，标准通讯连接介质为屏蔽双绞线在网络最远的电磁流量计通常要在通讯线两端并联一个 120 欧姆的终端匹配电阻（如图 1-1 所示）。

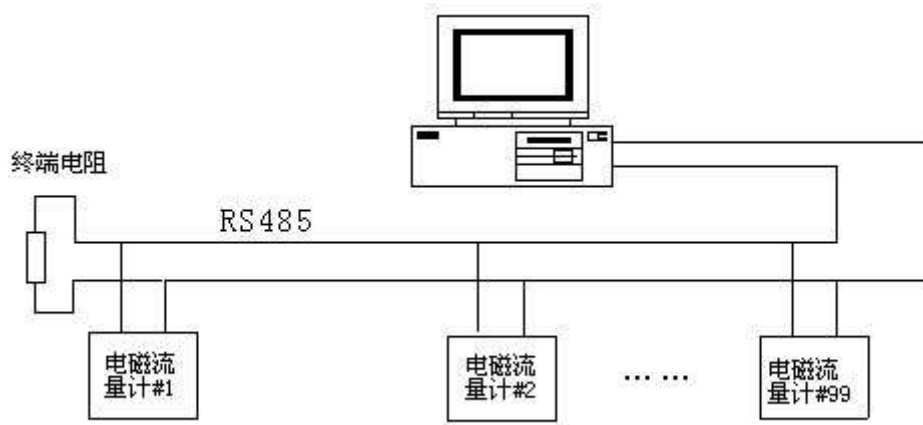


图 1-1 Modbus 通讯组网结构

1.3 数据帧格式

Modbus-RTU 格式（十六进制格式）为主从结构，即：主站先发送一帧数据，从站接收到后再给予应答。

主站命令帧结构（如图 1-2 所示）：

帧起始	设备地址	功能代码	寄存器地址	寄存器长度	CRC 校验	帧结束
T1-T2-T3-T4	8Bit	8Bit	16Bit	16Bit	16Bit	T1-T2-T3-T4

图 1-2 主站命令帧结构

主站命令帧说明：设备地址为流量计设置的通讯地址，功能代码为 Modbus 的功能码（这里通常为 04），寄存器地址为所要读取数据的寄存器地址（详见第二章），寄存器长度为所需要读的寄存器个数，CRC 校验可用相关软件计算。

从站响应帧结构（如图 1-3 所示）：

帧起始	设备地址	功能代码	数据长度	数据	CRC 校验	帧结束
T1-T2-T3-T4	8Bit	8Bit	8Bit	n 个 8Bit	16Bit	T1-T2-T3-T4

图 1-3 从站响应帧结构

从站响应帧说明：设备地址为流量计设置的通讯地址（即主站发送过来的地址），功能码也与主站发送的一致，数据长度为从站回复数据的个数。

第二章 寄存器地址表

L-mag 电磁流量计的 Modbus 通讯地址表（如表 1 所示），包括瞬时流量、累计流量等数据的寄存器地址，其中所提供的地址为寄存器地址，即有些 PLC、组态王等寄存器地址需要加 1（详见附录）。

表 2-1 L-mag 电磁流量计 Modbus 通讯地址表

Protocol Addresses (Decimal)	Protocol Addresses (HEX)	数据格式	寄存器定义
4112	0x1010	Float Inverse	瞬时流量浮点表示
4114	0x1012	Float Inverse	瞬时流速浮点表示
4116	0x1014	Float Inverse	流量百分比浮点表示（电池供电表保留）
4118	0x1016	Float Inverse	流体电导比浮点表示
4120	0x1018	Long Inverse	正向累积数值整数部分
4122	0x101A	Float Inverse	正向累积数值小数部分
4124	0x101C	Long Inverse	反向累积数值整数部分
4126	0x101E	Float Inverse	反向累积数值小数部分
4128	0x1020	Unsigned short	瞬时流量单位（表 3）
4129	0x1021	Unsigned short	累积总量单位(表 4/表 5)
4130	0x1022	Unsigned short	上限报警
4131	0x1023	Unsigned short	下限报警
4132	0x1024	Unsigned short	空管报警
4133	0x1025	Unsigned short	系统报警

第三章 数据解析方法

L-mag 电磁流量计的 Modbus 通讯从站响应的数据，大体分为 Float Inverse（瞬时流量）、Long Inverse（正向累计整数部分）、Unsigned short（流量单位）三种格式，具体解析方法如下。

3.1 瞬时流量

1 数据收发

主站发送命令：

设备地址	功能码	寄存器地址高位	寄存器地址高位	寄存器长度高位	寄存器长度低位	CRC 高位	CRC 低位
01	04	10	10	00	02	74	CE

主站接收到数据：

设备地址	功能码	数据长度	4 个字节浮点数 (瞬时流量)				CRC 高位	CRC 低位
01	04	04	C4	1C	60	00	2F	72

2 数据解析

瞬时流量数据为 Float Inverse 格式，采用 IEEE754 32 位浮点数格式，其结构如下：

0X1010 (34113)		0x1011 (34114)	
BYTE1	BYTE2	BYTE3	BYTE4
S EEEEEEE	E MMMMMMM	MMMMMMMM	MMMMMMMM

S—尾数的符号；1=负数,0 = 正数；

E—指数；与十进制数 127 的差值表示。

M—尾数；低 23 位，小数部分。

当 E 不全” 0” 时,且不全” 1 时浮点数与十进制数转换

公式：
$$V = (-1)^S 2^{(E-127)} (1 + M)$$

由上述公式可计算当前瞬时流量为：

浮点数 C4 1C 60 00

1100 0100 0001 1100 0110 0000 0000 0000

浮点数字节 1 浮点数字节 2 浮点数字节 3 浮点数字节 4

S=1: 尾数符号为 1 表示是负数。

E = 10001000: 指数为 136

M= 001 1100 0110 0000 0000 0000, 尾数为

$$V = (-1)^1 2^{(136-127)} \left(1 + \frac{1}{8} + \frac{1}{16} + \frac{1}{32} + \frac{1}{512} + \frac{1}{1024}\right)$$

$$= -625.5$$

故 C4 1C 60 00 代表的瞬时流量值为-625.5。

3.2 瞬时流速

1 数据收发

主站发送命令：

设备地址	功能码	寄存器地址高位	寄存器地址高位	寄存器长度高位	寄存器长度低位	CRC高位	CRC低位
01	04	10	12	00	02	D5	0E

主站接收到数据：

设备地址	功能码	数据长度	4 个字节浮点数 (瞬时流速)				CRC高位	CRC低位
01	04	04	C1	B0	80	00	A6	5F

2 数据解析

瞬时流速数据为 Float Inverse 格式，采用 IEEE754 32 位

浮点数格式。解析方法与解析瞬时流量一致。

浮点数为： C1 B0 80 00
 1100 0001 1011 0000 1111 1000 0000 0000

S = 1
E = 10000011
M = 011 0000 1111 1000 0000 0000

$$V = (-1)^1 2^{(131 - 127)} (1 + \frac{1}{4} + \frac{1}{8} + \frac{1}{256})$$

= - 22.0625

故 C1 B0 80 00 代表的瞬时流速值为-22.0625。

注：所有 Float Inverse 格式的数据都可参考瞬时流量、瞬时流速的方法解析，即流量百分比浮点表示、流体电导比浮点表示、正向累积数值小数部分、反向累积数值小数部分都可参考瞬时流量解析，以下不做过多解释。

3.3 正向累计量整数部分

1 数据收发

主站发送命令：

设备地址	功能码	寄存器地址高位	寄存器地址高位	寄存器长度高位	寄存器长度低位	CRC高位	CRC低位
01	04	10	18	00	02	F5	0C

主站接收到数据：

设备地址	功能码	数据长度	4 个字节浮点数 (正向累积量整数部分)				CRC高位	CRC低位
01	04	04	01	23	45	67	78	C8

2 数据解析

正向累积量整数部分数据为 Long Inverse 格式，可直接计算进行解析。

$$0 \times 16^7 + 1 \times 16^6 + 2 \times 16^5 + 3 \times 16^4 + 4 \times 16^3 + 5 \times 16^2 + 6 \times 16^1 + 7 \times 16^0 = 19088743$$

故 01 23 45 67 代表的正向累计流量整数部分值为 19088743。

在加上小数部分，即可计算出正向累积量。

注：所有 Long Inverse 格式的数据都可参考正向累积量整数部分的方法解析，即流反向累计量整数部分可参考正向累积量整数部分解析，以下不做过多解释。

3.4 流量单位

1 数据收发

主站发送命令：

设备地址	功能码	寄存器地址高位	寄存器地址高位	寄存器长度高位	寄存器长度低位	CRC高位	CRC低位
01	04	10	20	00	01	34	C0

主站接收到数据：

设备地址	功能码	数据长度	2 个字节整型 (瞬时流量单位)		CRC高位	CRC低位
01	04	02	00	05	79	33

2 数据解析

正向累积量整数部分数据为 Unsigned short 格式，需查表得知接收到数据的含义。

瞬时流量单位表如下：

表 3-1 瞬时流量单位表

代码	瞬时单位	代码	瞬时单位	代码	瞬时单位	代码	瞬时单位
0	L/S	3	M3/S	6	T/S	9	GPS
1	L/M	4	M3/M	7	T/M	10	GPM
2	L/H	5	M3/H	8	T/H	11	GPH

接收到的数据为 00 05，查表可知，当前瞬时流量为 m3/h。

累计流量的单位与瞬时流量单位解析方法一致，这里不做过多解释，累计流量单位表如下：

表 3-2 B 型及 511 型电磁流量计转换器累计流量单位表

代码	0	1	2	3
累积单位	L	M3	T	USG

表 3-3 C 型电磁流量计转换器累计流量单位表

代码	0	1	2	3	4	5
累积单位	L	L	L	M3	M3	M3
代码	6	7	8	9	10	11
累积单位	T	T	T	USG	USG	USG

3.5 空管报警

1 数据收发

主站发送命令：

设备地址	功能码	寄存器地址高位	寄存器地址高位	寄存器长度高位	寄存器长度低位	CRC 高位	CRC 低位
01	04	10	24	00	01	75	01

主站接收到数据：

设备地址	功能码	数据长度	2 个字节整型 (报警)		CRC 高位	CRC 低位
01	04	02	00	01	78	F0

2 数据解析

空管报警数据为 Unsigned short 格式，其中 1 为报警，0 为不报警。如例所示，接收到数据位 00 01 故此时流量计空管报警。

第四章 应用举例

此章节为应用举例部分，需要使用一些工具软件，软件可从网上自行下载。

需要用到的软件有：串口调试助手、16 to 10 (IEEE754)、程序员用计算器。

4.1 命令的发送与接收

打开串口调试助手，调整波特率 9600、数据位 8、校验位 N、停止位 1、设置 COM 口，并且选择好校验方式 CRC16 ModbusRTU。再将流量计的通讯地址设为 1、波特率设为 9600，连接好串口后即可进行测试。

在发送区输入 01 04 10 10 00 16 74 C1（后两位 CRC 为软件自动生成），即可在接受区收到流量计相应的数据（如图 4-1 所示）。



图 4-1 串口调试助手通讯图

4.2 接收到的数据的含义

如上图所示，接受到的流量计响应的数据为：

01 04 2C C3 36 D9 9A C0 CE F1 AA 42 81 51 EC 42 64 00 00
00 00 00 4C 3E 17 8D 50 00 00 00 28 3D 71 A9 FC 00 05 00
01 00 00 00 00 00 00 00 00 00 C7 D2

根据流量计的寄存器表，可知接收到的流量计响应数据的含义为
(如图 4-2 所示)：

01				流量通讯地址
04				流量计功能码
2C				响应数据长度
C3	36	D9	9A	瞬时流量
C0	CE	F1	AA	瞬时流速
42	81	51	EC	流量百分比
42	64	00	00	流体电导比
00	00	00	4C	正向累积量整数部分
3E	17	8D	50	正向累积量小数部分
00	00	00	28	反向累积量整数部分
3D	71	A9	FC	反向累积量小数部分
00	05			瞬时流量单位
00	01			累计流量单位
00	00			上限报警
00	00			下限报警
00	00			空管报警
00	00			系统报警
C7	D2			CRC校验位

图 4-2 接收到数据的含义图

4.3 接收到数据的解析

根据接受到的数据，可以利用工具软件解析。

1. 瞬时量的解析

瞬时量可利用 16 to 10 (IEEE754) 软件进行解析，如图 4-3 所示：



图 4-3 瞬时量的解析

综上，瞬时流量为-185.85，瞬时流速为-6.46。

2. 百分比的解析

百分比也可利用 16 to 10 (IEEE754) 软件进行解析，如图 4-4 所示：

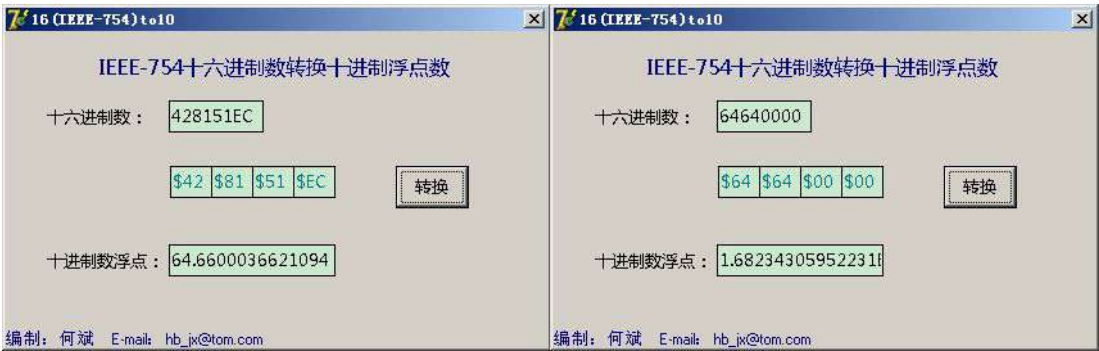


图 4-3 百分比的解析

综上，流量百分比为 64.66，流体电导比为 1.68。

3. 累积量的解析

累积量整数部分可利用程序员用计算器进行解析，如图

4-5 所示：

原数据：



转化为：



原数据：



转化为：



图 4-5 累积量整数部分的解析

累积量小数部分可利用 16 to 10 (IEEE754) 软件进行解析，如图 4-6 所示：

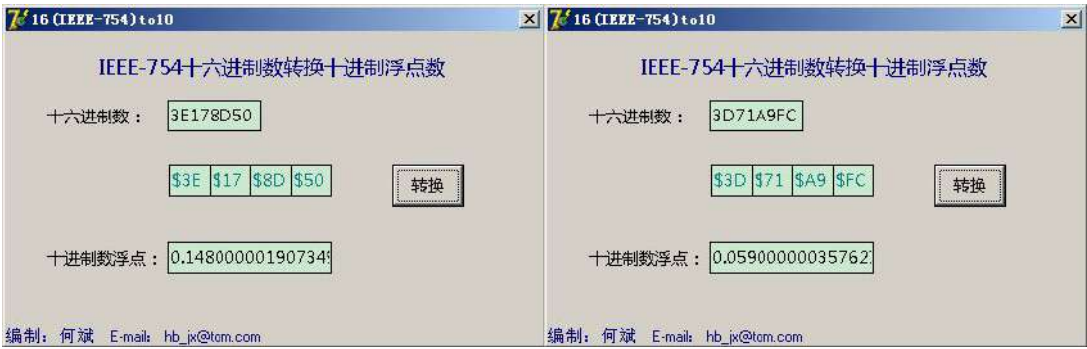


图 4-6 累积量小数部分的解析

综上，流量百分比为 76.148，流体电导比为 40.059。

4. 流量单位的解析

流量单位可直接查表解析。

接收到的数据，瞬时流量单位为 00 05，累计流量单位为 00 01。

根据表 3-1、3-2、3-3 可知，瞬时流量单位为 m^3/h ，累计流量单位为 m^3 （如果是 C 型表为 L）。

5. 报警的解析

报警数据可以根据 1 为报警，0 为不报警来解析。

接受到的是数据，上、下限报警和空管、系统报警的数据都为 00 00，故流量计正常运行，无报警状态。

第五章 常见问题处理方法

常见问题有两种情况。一为主站发送数据，从站不予响应。一种为从站相应的数据解析异常。出现通讯问题，建议先使用我公司的测试软件进行测试。

5.1 测试软件使用方法

1. 设置流量计波特率为 9600，通讯地址为 1。
2. 用 USB（或 232）转 485 口连接计算机和流量计（连接成功后可以在我的电脑→属性中找到 COM 口号，如图 5-1 所示）。



图 5-1 COM 口提示


3. 双击  图标打开软件，调整波特率为 9600，通讯地址为 1，COM 与图 5-1 相同。设置完成后单击“开始通讯”。
- 通讯成功画面如图 5-2 所示。



图 5-2 测试软件通讯成功图

5.2 从站不响应

如果主站发送数据，从站不予响应。则考虑参数设置和物理接线问题。

1. 判断流量计是否有通讯功能：

查询流量计型号，并检查是否具有通讯板。

2. 判断通讯线是否连接正确

将通讯线调换再试一次。

3. 判断波特率是否正确：

要求上位机波特率和流量计所设置的波特率一致。

4. 判断通讯地址是否正确：

要求上位机通讯地址和流量计通讯地址设置一致。

5. 判断 COM 口是否有异常：

在我的电脑→属性中查询该串口是否可用。

6. 查看通讯板：

打开仪表上盖查看通讯板上的通讯灯是否闪烁如图（5-4所示），其中 TXD 灯为发送数据，RXD 灯为接收数据。

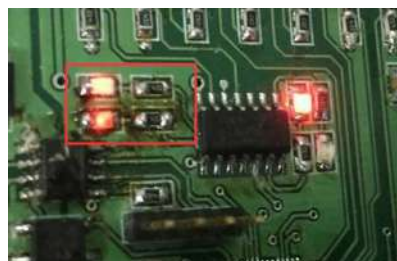


图 5-4 通讯灯闪烁示例图

5.3 数据解析异常

用户可以使用串口调试助手来检测通讯过程，具体步骤如

下：

1. 将主站设备与流量计正确连接，再用 USB（或 232）转 485 口并联至系统中，如图 5-5 所示。

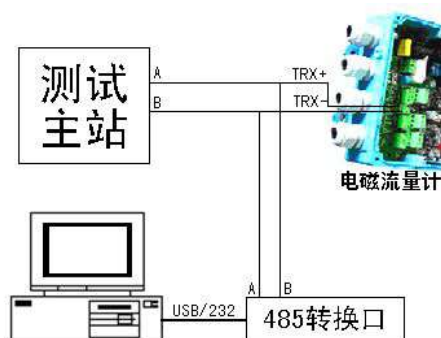


图 5-5 串口调试助手连接示例图

2. 打开串口调试助手，设置波特率 9600、通讯地址 1、起始位 1、数据位 8、停止位 1、无校验。
3. 单击“打开串口”并用主站开始发送数据与流量计通讯。串口调试上即可显示通讯过程如图 5-6 所示。



图 5-6 串口调试助手侦听示例图

4. 根据接受到的数据，查找主站发送的内容和流量计相应的内容（如图 5-7 所示）。



图 5-7 串口调试助手数据分析示例图


红色注释的为主站发送的数据，紫色和黄色为从站响应的数据。紫色部分为从站按照协议格式要求所响应的数据（解析时可以不考虑），黄色部分为从站响应的数据部分，用于解析时使用。

根据侦听的数据进行分析，分析主站发送的通讯地址、功能码、寄存器地址、寄存器长度、CRC 校验码是否正确（详细方法可以参加第四章）。

附录

附录一 Modbus poll 通讯示例

流量计通讯地址设置为 1，波特率设置为 9600。

双击  打开 Modbus Poll 软件，如图 F1-1 所示。

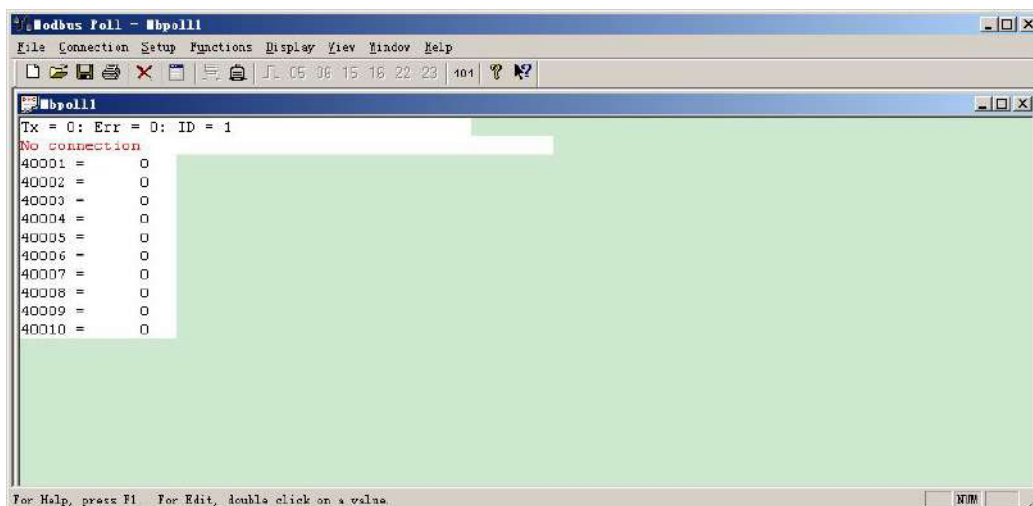


图 F1-1

单击 Setup→Poll Definition，设置采集命令包括设备地址 1、MODBUS 功能码 04、寄存器地址 4113、寄存器长度 22、采集间隔 1000 如图 F1-2 所示。

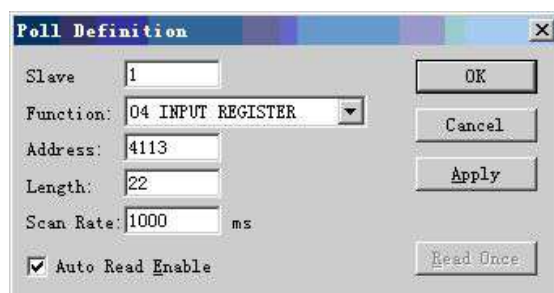


图 F1-2

单击 Connection→Connection，设置流量计串口格式：1 位起始位 8 位数据位 1 位停止位，无校验如图 F1-3 所示。

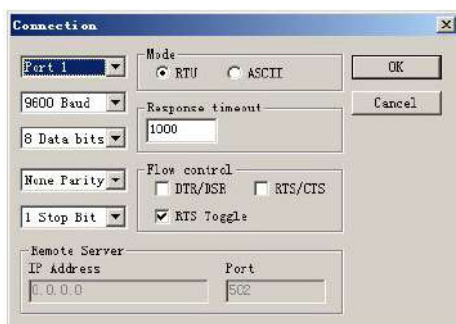


图 F1-3

单击 OK 后，即可进行通讯。

通讯成功后，可单击 Display 根据表 2-1 选择数据格式（如图 F1-4 所示），通讯成功界面如图 F1-5 所示。

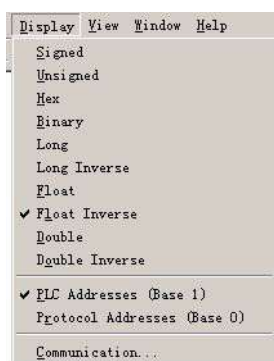


图 F1-4

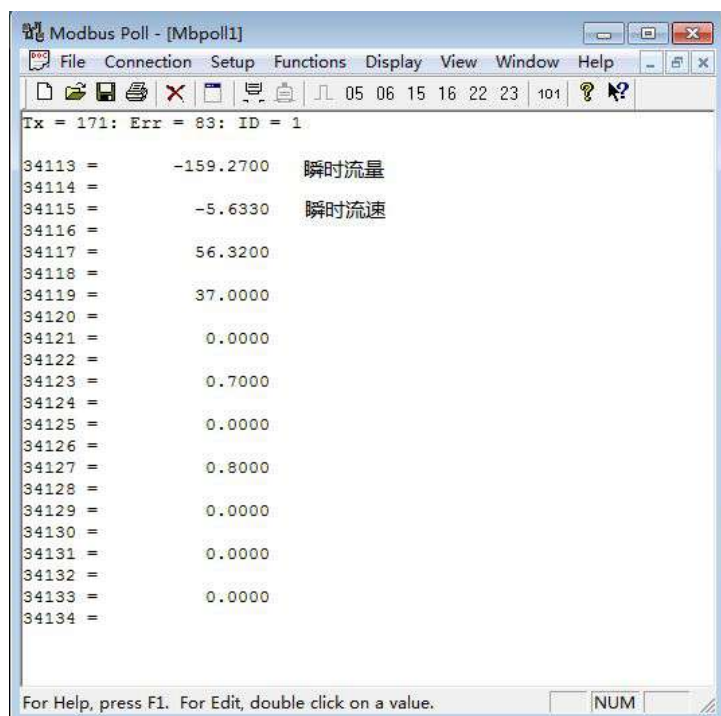


图 F1-5

附录二 modscan32 通讯示例

流量计通讯地址设置为 1，波特率设置为 9600。

单击 Setup→Poll Definition，设置采集命令包括设备地址 1、MODBUS 功能码 04、寄存器地址 4113、寄存器长度 22、采集间隔 1000 如图 F2-1 所示。

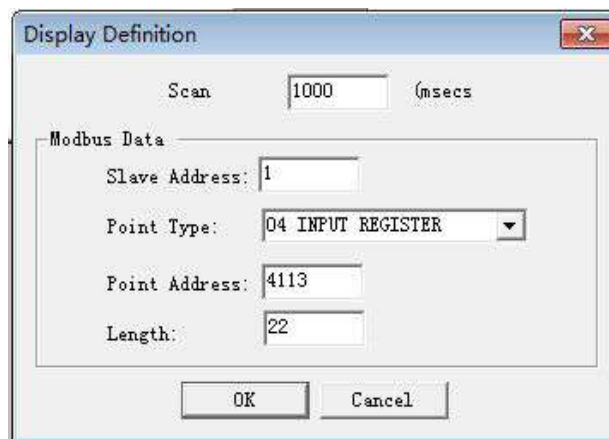


图 F2-1

单击 Connection→Connection Details，设置流量计串口格式：1 位起始位 8 位数据位 1 位停止位，无校验如图 F2-2 所示。

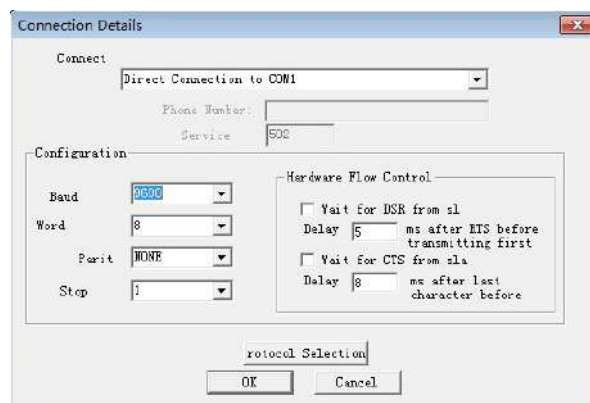


图 F2-2

单击 OK 后，即可进行通讯。

通讯成功后，可单击 Display 根据表 2-1 选择数据格式（如图 F2-3 所示），通讯成功界面如图 F2-4 所示。

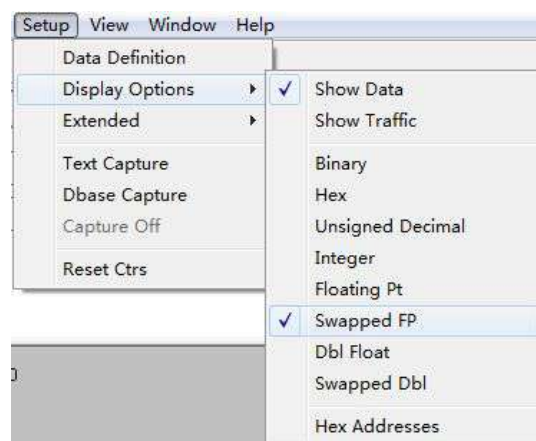


图 F2-3



图 F2-4

附录三 西门子 200 PLC 通讯示例

B 系列电磁转换器 V77 版本的通讯协议，支持标准 MODBUS 协议，可以和支持 MODBUS 的 PLC 进行通讯来读取数据。下面以西门子的 200 系列 PLC 来举例说明通讯的设置方法：（软件为 micro win V4.0）

第一步：找到 PLC 支持 MODBUS 通讯的库。如图 F3-1 所示：
如果在系统上没有找到图 1 中的库，请上西门子网址下载并安装即可。

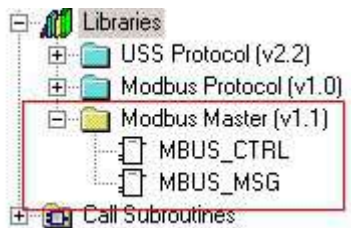


图 F3-1 MODBUS 应用库

第二步：使用 MBUS_CTRL 初始化 PLC 的 MODBUS 功能。如图 F3-2 所示：

其中 Baud: 9600 代表流量计的默认波特率为 9600。
Parity: 0 代表流量计使用无校验。

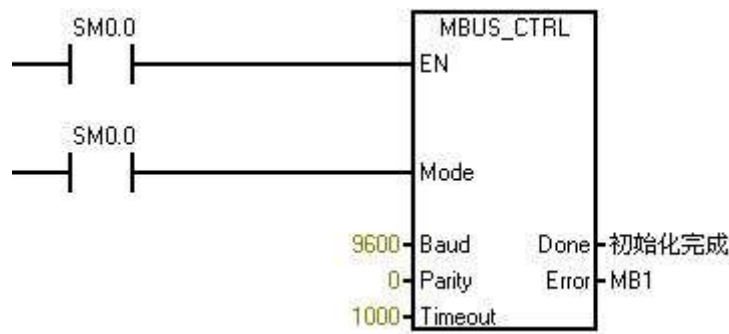


图 F3-2 初始化 MODBUS

第三步：使用 MBUS_MSG 读取流量计的数据。如图 F3-3 所示：
Slave: 1 代表流量计的通讯地址为 1。

Addr: 34113 代表读取瞬时流量的通讯起始地址，地址的详细说明可以参见表 F3-1。

Count: 2 代表读取瞬时流量的长度为 2，详细说明可以参见表 F3-1。

DataPtr: &VB1000 代表瞬时流量传递到的 PLC 内部地址空间。

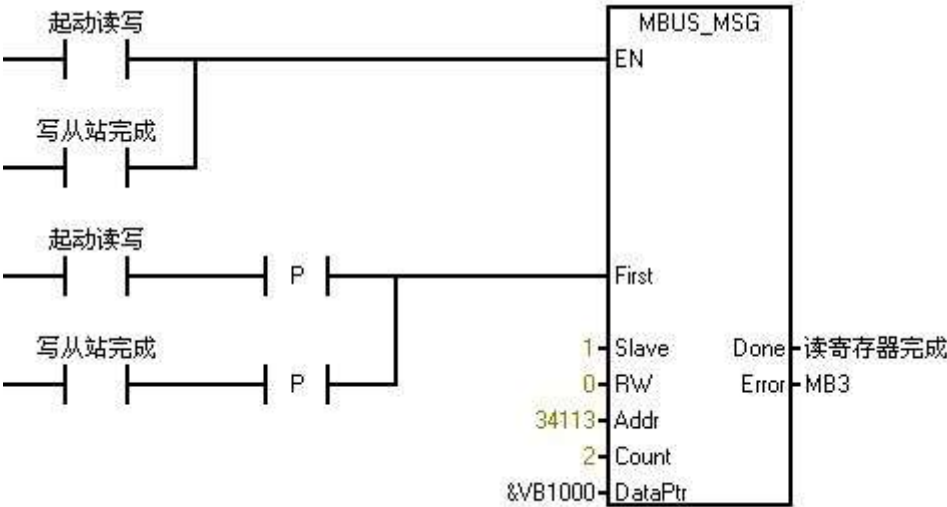


图 F3-3 读瞬时流量

第四步：查看流量计的数据。如图 F3-4 所示：

VD10000 和图 F3-3 的 VB1000 对应，表示瞬时流量传递到的 PLC 内部地址空间。

显示格式选择 Floating Point 表示浮点数。

其它数据的设置可以参看表 F3-1。

12	VW1002	Signed		
13	VW1000	Signed		
14	VD1008	Floating Point		
15	VD1004	Floating Point		
16	VD10012	Floating Point		
17	VD1000	Floating Point		
18	VB3000	Unsigned		
19	VB1008	Unsigned		
20	VB1007	Unsigned		

图 F3-4 查看数据

表 F3-1 西门子 200PLC 变量对应表

Addr	Count	数据格式	寄存器定义
34113	2	Floating Point	瞬时流量浮点表示
34115	2	Floating Point	瞬时流速浮点表示
34117	2	Floating Point	流量百分比浮点表示
34119	2	Floating Point	流体电导比浮点表示
34121	2	Unsigned	正向累积数值整数部分
34123	2	Floating Point	正向累积数值小数部分
34125	2	Unsigned	反向累积数值整数部分
34127	2	Floating Point	反向累积数值小数部分
34129	1	Unsigned	瞬时流量单位
34130	1	Unsigned	累积总量单位
34131	1	Unsigned	上限报警
34132	1	Unsigned	下限报警
34133	1	Unsigned	空管报警
34134	1	Unsigned	系统报警

附录四 组态王通讯示例

第一步：

创建组态王工程, 点击新建弹出如下界面，输入工程路径及工程名称。



图 F4-1

第二步：打开新建的工程，选择设备栏在 COM 口下新建标准 modbus 设备。

组态王设备列表中找到-PLC-莫迪康-modbus（RTU）（L-mag 电磁流量计借助莫迪康 PLCmodbus（RTU）驱动）。



图 F4-2

按照电磁流量计中的地址设置设备地址。下图以地址 1 为例：



图 F4-3

第三步：双击设备中的 COM 设置串口参数



图 F4-4

L-mag 电磁流量计串口参数：波特率与电磁流量计中设置相同、1 位起始位、 8 位数据位、 1 位停止位、无校验。下图以波特率 9600 为例：



图 F4-5

第四步：点击数据词典添加 L-mag 数据变量

根据组态王驱动说明莫迪康-modbus（RTU）变量名称、寄存器地址和数据格式见下表：

表 F4-1

变量名	寄存器值	数据格式	采用频率	读写属性
瞬时流量	34113	Float	500	只读
瞬时流速	34115	Float	500	只读
流量百分比	34117	Float	500	只读
流体电导比	34119	Float	500	只读
正向累积值整数部分	34121	Long	500	只读
正向累积值小数部分	34123	Float	500	只读
反向累积值整数部分	34125	Long	500	只读
反向累积值小数部分	34127	Float	500	只读
数据转换寄存器	SwapL0	Byte	0	只写



图 F4-6

注意：因电磁流量计数据存储格式的原因，在组态王添加变量时必须添加数据转换寄存器，否则通讯数据显示不正常。

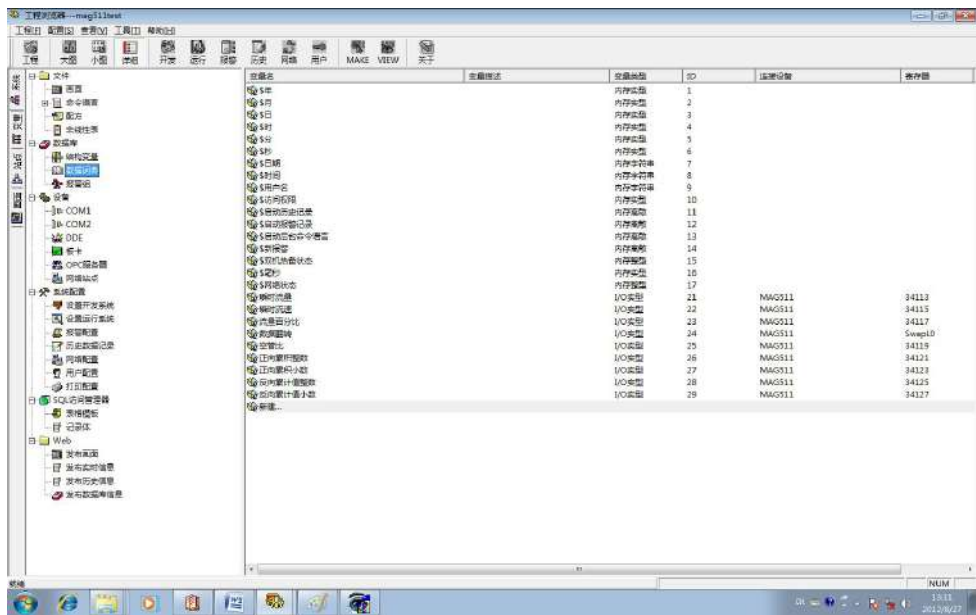


图 F4-7

数据寄存器对应的功能码

功能码用十六进制数表示。

寄存器	读的功能码	写的功能码	说明
0	0x01	0x06	逻辑块图
1	0x02		输入寄存器
3	0x04		输入寄存器
4	0x03	0x06	保持寄存器
7	0x14	0x15	配置寄存器 (General Reference)
8	0x04		输入寄存器
9	0x03	0x10	保持寄存器
PMC		0x0F	强制多线圈状态

图 F4-8

第五步：创建窗口界面并建立数据链接。



图 F4-9

第六步：保存工程并运行工程

瞬时流量	-00116.42999
瞬时流速	-04.118
流量百分比	041.17
流体电导比	00009
正向累积值整数部分	0145570342
正向累积值小数部分	0.000
反向累积值整数部分	0488902442
反向累积值小数部分	0.000

图 F4-10

附录五 力控 6.1 通讯示例

说明使用方法

第一步：

创建一个工程输入工程名称及工程路径。

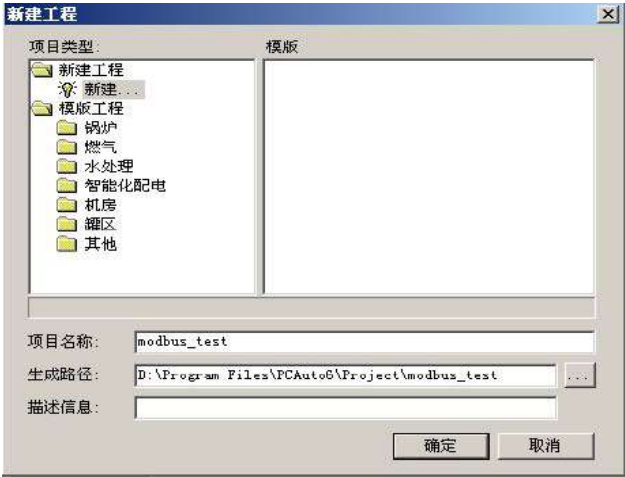


图 F5-1

第二步：添加设备

IO 口设备组态选择 IO 设备-modbus-标准 modbus-modbus（RTU 串口）



图 F5-2

点击高级选项选择串口并设置串口参数（9600，8 为数据位、1 位停止位、无校验）



图 F5-3

设置数据显示格式



图 F5-4

第三步：数据库组态

修改：区域0 - 模拟I/O点 - [ss11]

基本参数

报警参数

数据连接

历史参数

点名 (NAME):

ss11

点说明 (DESC):

瞬时流量

单元 (UNIT):

0

测量初值 (PV):

0.00000

小数位 (FORMAT):

5

工程单位 (EU):

量程下限 (EULO):

-100000.000

裸数据下限 (PVRAWLO):

0.000

量程上限 (EUHI):

100000.000

裸数据上限 (PVRAWHI):

4095.000

数据转换

☐ 开平方 (SQRTFL)

☐ 分段线性化 (LINEFL)

分段线性化

+

☐ 量纲变换 (SCALEFL)

☐ 统计 (STATIS)

☐ 滤波 (ROCFL)

滤波限值 (ROC):

0.000

确定

取消

应用 (A)

图 F5-5

设置数据格式及地址偏移

基本参数

报警参数

数据连接

历史参数

参数

连接类型

连接项

☐ DESC

☒ PV I/O设备 mag511: ARF

☐ EU

☐ LL

☐ LO

☐ HI

☐ HH

☐ SP

☒ I/O设备

☐ 网络数据库

☐ 内部

连接I/O设备

设备:

mag511

组态界面

内存区:

04号功能码 (AR输入寄存器)

偏置

4113

10进制

数据格式:

32位IEEE浮点数

☐ 可读可写

☒ 只可读

☐ 只可写

提示: 寄存器地址304113 偏置 4113

确定

取消

图 F5-6

数据举例

	NAME [点名]	DESC [说明]	%IOLINK [I/O连接]	%HIS [历史参数]
1	ss11	瞬时流量	PV=mag511:ARF4113	
2	ss1s	瞬时流速	PV=mag511:ARF4115	
3	11bfb	流量百分比	PV=mag511:ARF4117	
4	1tddb	流体电导比	PV=mag511:ARF4119	
5	zxljzrsbf	正向累积值整数部分	PV=mag511:ARL4121	
6	zxljxrsbf	正向累积值小数部分	PV=mag511:ARF4123	
7	fxljzrsbf	反向累积值整数部分	PV=mag511:ARL4125	
8	fxljxrsbf	反向累积值小数部分	PV=mag511:ARF4127	

图 F5-7

第四步：

创建窗口并连接变量

瞬时流量	#####.#####
瞬时流速	##.###
流量百分比	###.##
流体电导比	#####
正向流量累积值整数部分	#####
正向流量累积值小数部分	#.###
反向流量累积值整数部分	#####
反向流量累积值小数部分	#.###

图 F5-8

第五步：

运行工程

瞬时流量	-116.51999
瞬时流速	-4.121
流量百分比	41.20
流体电导比	8
正向流量累积值整数部分	145570342
正向流量累积值小数部分	0.000
反向流量累积值整数部分	488903076
反向流量累积值小数部分	0.000

图 F5-9

附录六 MCGS 通讯实例

说明使用方法

第一步：

创建一个工程，出现如下界面，选择设备窗口，双击。

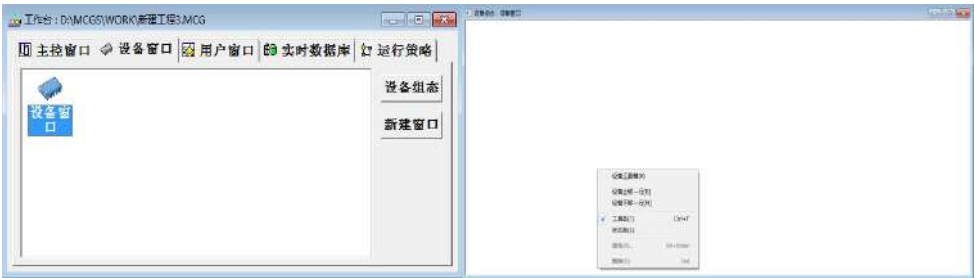


图 F6-1

选择设备工具箱，点击设备管理，把通用串口父设备和标准MODBUSRTU 设备添加到工程。

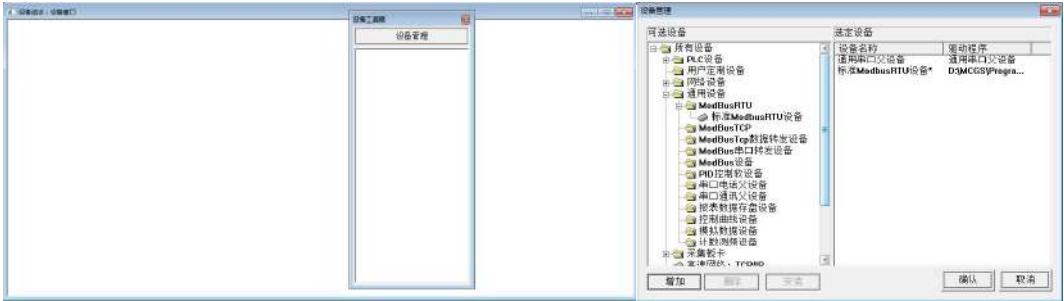


图 F6-2

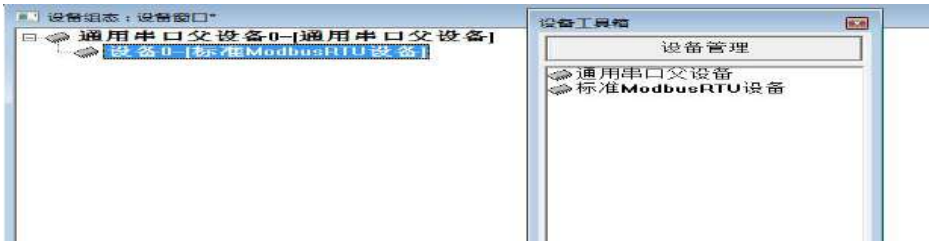


图 F6-3

选择通用串口父设备 0 属性和设备 0 属性，进行如下设置。



图 F6-4

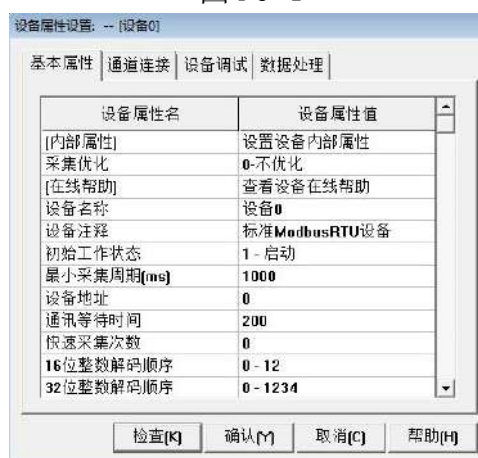


图 F6-5

设备地址为 1, 32 位浮点数解码顺序 0-1234, 校验方式 0-LH[低字节, 高字节]。选择设置内部属性。



图 F6-6



图 F6-7

点击添加通道，出现如下界面。

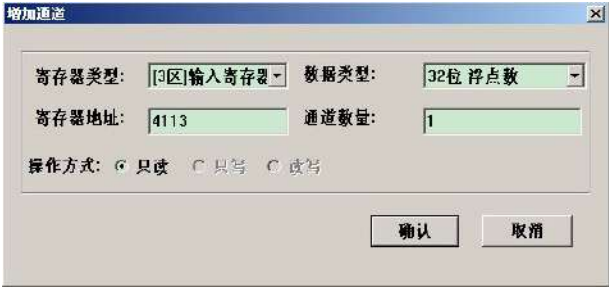


图 F6-8

表 F6-1

寄存器地址	数据类型	通道数量	寄存器定义
4113	32 位浮点数	1	瞬时流量
4115	32 位浮点数	1	瞬时流速
4117	32 位浮点数	1	流量百分比
4119	32 位浮点数	1	流量电导比
4121	32 位无符号二进制	1	正向累积整数
4123	32 位浮点数	1	正向累积小数
4125	32 位无符号二进制	1	反向累积整数
4127	32 位浮点数	1	反向累积小数



图 F6-9

单击快速链接变量



图 F6-10

如下图链接变量

索引	连接变量	通道名称	通
0000		通讯状态	
0001	瞬时流量00	只读3DF4113	
0002	瞬时流速00	只读3DF4115	
0003	流体百分比00	只读3DF4117	
0004	流体电导比00	只读3DF4119	
0005	正向累计整数00	只读3DUB4121	
0006	正向累计小数00	只读3DF4123	
0007	反向累计整数00	只读3DUB4125	
0008	反向累计小数00	只读3DF4127	

图 F6-11

选择设备调试，开始通讯（通讯成功如下图）:

索引	连接变量	通道名称	通道处理	调试数据	采集周期
0000		通讯状态		0	1
0001	瞬时流量00	只读3DF4113		318.6	1
0002	瞬时流速00	只读3DF4115		11.2	1
0003	流体百分比00	只读3DF4117		112.6	1
0004	流体电导比00	只读3DF4119		3.0	1
0005	正向累计整数00	只读3DUB4121		67.0	1
0006	正向累计小数00	只读3DF4123		0.9	1
0007	反向累计整数00	只读3DUB4125		37.0	1
0008	反向累计小数00	只读3DF4127		0.8	1

图 F6-12

整体画面：

索引	连接变量	通道名称	通道处理	调试数据	采集周期	信息备注
0000		通讯状态		0	1	
0001	瞬时流量00	只读3DF4113		318.6	1	
0002	瞬时流速00	只读3DF4115		11.2	1	
0003	流体百分比00	只读3DF4117		112.6	1	
0004	流体电导比00	只读3DF4119		3.0	1	
0005	正向累计整数00	只读3DUB4121		67.0	1	
0006	正向累计小数00	只读3DF4123		0.9	1	
0007	反向累计整数00	只读3DUB4125		37.0	1	
0008	反向累计小数00	只读3DF4127		0.8	1	

图 F6-13

附录七 VB 例程

1、主站发送：

```
Private Sub Timer1_Timer()  
Dim gg As Variant  
If RecieveCounterBak = RecieveCounter Then  
Form2.Label1 = "呼叫无应答"  
TempString = Form2.MSComm1.Input  
End If  
RecieveCounterBak = RecieveCounter  
baud = Form2.Combo3.ListIndex + 1  
If baud = 0 Then  
baud = 1  
End If  
ch = ch + 1  
If ch = 9 Then  
ch = 1  
End If  
address = Combo2.ListIndex + 1  
If address = 0 Then  
address = 1  
End If  
If Mcommaddr > 1 Then  
Mcommaddr = 0  
End If  
receiveflag = 1  
command(0) = address  
command(1) = 4  
command(2) = 0  
command(3) = Mcommaddr * 4 + 10  
command(4) = 0  
command(5) = 5  
Call sendcrc  
command(6) = secrcl  
command(7) = secrchi  
gg = command  
Form2.MSComm1.Settings = baud_list(baud) & "n,8,1"  
Form2.MSComm1.Output = gg  
TempString = Form2.MSComm1.Input  
Mcommaddr = Mcommaddr + 1  
End Sub
```

2、主站接收

```
Private Sub MSComm1_OnComm()  
Dim i As Integer  
Select Case MSComm1.CommEvent  
    Case comEvReceive  
        temp = Form2.MSComm1.Input  
        RecieveCounter = RecieveCounter + 1  
        If RecieveCounter >= 1000 Then  
            RecieveCounter = 0  
        End If  
        If commtest = 0 Then "....."  
        Select Case receiveflag  
        Case 0  
            If temp(0) <> address Then  
                MsgBox ("收到数据地址错误")  
                erre = erre + 1  
            End If  
            If temp(1) <> &H4 Then  
                MsgBox ("收到数据命令错误")  
                erre = erre + 1  
            End If  
            If temp(2) <> &HA Then  
                MsgBox ("收到数据长度错误")  
                erre = erre + 1  
            End If  
            Call receivecrc(12)  
            erre = erre + 1  
            If temp(13) <> recrclo Then  
                erre = erre + 1  
                MsgBox ("CRC 校验出错")  
                Exit Sub  
            End If  
            If temp(14) <> recrchi Then  
                MsgBox ("CRC 校验出错")  
                erre = erre + 1  
                Exit Sub  
            End If  
        Case 1  
            If temp(0) <> address Then  
                MsgBox ("收到数据地址错误")  
                erre = erre + 1  
            End If  
            If temp(1) <> &H4 Then
```

```

MsgBox ("收到数据命令错误")
    erre = erre + 1
End If
If temp(2) <> &H8 Then
MsgBox ("收到数据长度错误")
    erre = erre + 1
End If
Call receivecrc(10)
If temp(11) <> recrclo Then
MsgBox ("CRC 校验出错")
    erre = erre + 1
Exit Sub
End If
If temp(12) <> recrchi Then
MsgBox ("CRC 校验出错")
    erre = erre + 1
Exit Sub
End If
End Select
Call datasum
End If
If commtest = 1 Then "''''''''''
If temp(1) <> &H3 Then "''''''''
MsgBox ("收到数据命令错误") "''''''
End If "''''''''''''''''''
Call receivecrc(6) "''''''''''
If temp(7) <> recrclo Then "''''''''
MsgBox ("CRC 校验出错") "''''''''
Exit Sub "''''''''
End If "''''''''
If temp(8) <> recrchi Then "''''''''
MsgBox ("CRC 校验出错") "''''''''
Exit Sub "''''''''
End If "''''''''
Form2.Label1 = "通讯测试" "''''''''
Form2.Label2 = temp(3) * 256# * 256# * 256# + temp(4) * 256# * 256# + temp(5) * 256#
+ temp(6) "''''''
End If "''''''''
Case Else
End Select
End Sub

```

3. 数据解析

```
Sub datasum()  
Dim X(4) As Double  
Dim Y As String  
Dim Z, m, n As Integer  
Form2.Label1 = tongxun  
Select Case receiveflag  
Case 0  
    Select Case Mcommaddr - 1  
        Case 0  
            'Form2.Text2 = ""  
            Form2.Text3 = ""  
            Form2.Text4 = ""  
            Form2.Text5 = ""  
            Form2.Text8 = ""  
            X(0) = temp(3) * 256# + temp(4)  
            If (temp(5) And &H80) \ &H80 = 1 Then  
                Y = "-"  
            Else: Y = "+"  
            End If  
            Z = 5 - ((temp(5) And &H70) \ 16)  
            m = temp(5) And &H7  
            X(1) = (temp(6) * 256# + temp(7)) / 10 ^ 3  
            X(2) = (temp(8) * 256# + temp(9)) / 10 ^ 2  
            X(3) = temp(10) * 256# + temp(11)  
            .....  
            'Form2.Text2 = Y & X(0) / (10 ^ Z) & unitchar4(m)  
            Form2.Text3 = X(1) & "m / s"  
            Form2.Text4 = X(2) & "%"  
            Form2.Text5 = X(3) & "%"  
            Form2.Text8 = ""  
            If (temp(12) And &H80) = &H80 Then  
                Form2.Text8 = Form2.Text8 & "Low"  
            End If  
            If (temp(12) And &H40) = &H40 Then  
                Form2.Text8 = Form2.Text8 & "High"  
            End If  
            If (temp(12) And &H20) = &H20 Then  
                Form2.Text8 = Form2.Text8 & "Mtsnsr"  
            End If  
            If (temp(12) And &H10) = &H10 Then  
                Form2.Text8 = Form2.Text8 & "Cut"  
            End If
```

```

If (temp(12) And &H8) = &H8 Then
Form2.Text8 = Form2.Text8 & "Rev"
End If
If (temp(12) And &H4) = &H4 Then
Form2.Text8 = Form2.Text8 & "Fwd"
End If
If (temp(12) And &H2) = &H2 Then
Form2.Text8 = Form2.Text8 & "Coil"
End If
If (temp(12) And &H1) = &H1 Then
Form2.Text8 = Form2.Text8 & "Enable"
End If
Case 1
X(0) = (temp(3) * 256# * 256# * 256# + temp(4) * 256# * 256# + temp(5) * 256# + temp(6))
X(1) = (temp(7) * 256# * 256# * 256# + temp(8) * 256# * 256# + temp(9) * 256# + temp(10))
Form2.Text6 = X(0) & unichar3(temp(11) \ 4) & unichar2(temp(11) Mod 4)
Form2.Text7 = X(1) & unichar3(temp(11) \ 4) & unichar2(temp(11) Mod 4)
Case 2
Form2.Text2 = ""
X(0) = temp(3) * 65536# * 256# + temp(4) * 65536# + temp(5) * 256# + temp(6)
If (temp(7) And &H80) \ &H80 = 1 Then
Y = "-"
Else: Y = "+"
End If
Z = 5 - ((temp(7) And &H70) \ 16)
m = temp(7) And &H7
Form2.Text2 = Y & X(0) / (10 ^ Z) & unichar4(m)
End Select
Case 1
Select Case Mcommaddr - 1
Case 0
X(0) = temp(3) * 256# + temp(4)
If (temp(5) And &H80) \ &H80 = 1 Then
Y = "-"
Else: Y = "+"
End If
Z = 5 - ((temp(5) And &H70) \ 16)
m = temp(5) And &H7
Form3.Label9 = Y & X(0) / (10 ^ Z) & unichar1(m)
X(1) = (temp(6) * 256# * 256# * 256# + temp(7) * 256# * 256# + temp(8) * 256# + temp(9)) /
unitint(temp(10) Mod 4)
Form3.Label10 = X(1) & unichar2(temp(10) \ 4)
Case 1
X(0) = temp(3) * 256# + temp(4)

```

```

If (temp(5) And &H80) \ &H80 = 1 Then
Y = "-"
Else: Y = "+"
End If
Z = 5 - ((temp(5) And &H70) \ 16)
m = temp(5) And &H7
Form3.Label9 = Y & X(0) / (10 ^ Z) & unitchar1(m)
X(1) = (temp(6) * 256# * 256# * 256# + temp(7) * 256# * 256# + temp(8) * 256# + temp(9)) /
unitint(temp(10) Mod 4)
Form3.Label11 = X(1) & unitchar2(temp(10) \ 4)
End Select
End Select
End Sub

```

附录八 C 语言例程

1. 主站发送

```
void RS485_Send(unsigned char Device_addr,unsigned int Res_addr,unsigned char Len)
{
    unsigned int crc;
    unsigned char i;
    RS485_Sendbuf[0]=Device_addr;
    RS485_Sendbuf[1]=0x04;
    RS485_Sendbuf[2]=Res_addr/256;
    RS485_Sendbuf[3]=Res_addr%256;
    RS485_Sendbuf[4]=Len/256;
    RS485_Sendbuf[5]=Len%256;
    crc=CRC16(RS485_Sendbuf,6);
    RS485_Sendbuf[6]=crc/256;
    RS485_Sendbuf[7]=crc%256;
    UART_Transmit;
    delay_ms(1);
    for(i=0;i<8;i++)
    {
        UCA0TXBUF=RS485_Sendbuf[i];
        delay_ms(1);
        while((UCA0IFG&0x02)==0x00);
        UCA0IFG=0;
    }
    UART_Recieve;
}
```

2、主站接收

```
char Modbus_Recieve(char l)
{
    unsigned char i=0;
    while(1)
    {
        timer_TA1_init();
        while((UCA0IFG&0x01)==0x00);
        TA1CTL = TASSEL_1 | MC_0 | TACLR;           //停定时器 TA1
        UCA0IFG=0;
        RS485_Recievebuf[i]=UCA0RXBUF;
        i++;
        if(i==l)
            break;
    }
}
```

```
}
```

3. 数据解析

```
void V77_Instant(void)
{
    signed long ddd,eee;
    float temp;
    WriteMenu(0,0,0,8,0);
    WriteMenu(4,0,0,9,0);
    union   ieee754_to_float
    {
        unsigned char IEE[4];
        float FLOA;
    };
    union   ieee754_to_float Change;
    Change.IEE[3] = RS485_Recievebuf[3]; ////////////////瞬时流量
    Change.IEE[2] = RS485_Recievebuf[4];
    Change.IEE[1] = RS485_Recievebuf[5];
    Change.IEE[0] = RS485_Recievebuf[6];
    temp = Change.FLOA;
    ddd = (signed long)(temp*100000);
    disp_511_Number(2,0,ddd,10,5); //显示瞬时流量
    switch(RS485_Recievebuf[36])    //显示瞬时流量单位
    {
        case 0x00:WriteMenu(2,96,0,13,0);break;
        case 0x01:WriteMenu(2,96,0,14,0);break;
        case 0x02:WriteMenu(2,96,0,15,0);break;
        case 0x03:WriteMenu(2,96,0,16,0);break;
        case 0x04:WriteMenu(2,96,0,17,0);break;
        case 0x05:WriteMenu(2,96,0,18,0);break;
        case 0x06:WriteMenu(2,96,0,19,0);break;
        case 0x07:WriteMenu(2,96,0,20,0);break;
        case 0x08:WriteMenu(2,96,0,21,0);break;
        default:break;
    }
    Change.IEE[3] = RS485_Recievebuf[7]; ////////////////瞬时流速
    Change.IEE[2] = RS485_Recievebuf[8];
    Change.IEE[1] = RS485_Recievebuf[9];
    Change.IEE[0] = RS485_Recievebuf[10];
    temp = Change.FLOA;
    eee = (signed long)(temp*100000);
    disp_511_Number(6,0,eee,10,5); //显示瞬时流速
    WriteMenu(6,96,0,22,0);    //显示瞬时流速单位
}
```



```

void V77_Accumulate(void)
{
    unsigned long ddd,eee,fff,ggg;
    float temp;
    WriteMenu(0,0,0,11,0);
    WriteMenu(4,0,0,12,0);
    ddd=(unsigned long)RS485_Recievebuf[19]*16777216+(unsigned
long)RS485_Recievebuf[20]*65536+(unsigned long)RS485_Recievebuf[21]*256+(unsigned
long)RS485_Recievebuf[22];
    delay_ms(1);
    eee=(unsigned long)RS485_Recievebuf[27]*16777216+(unsigned
long)RS485_Recievebuf[28]*65536+(unsigned long)RS485_Recievebuf[29]*256+(unsigned
long)RS485_Recievebuf[30];
    union ieee754_to_float
    {
        unsigned char IEE[4];
        float FLOA;
    };
    union ieee754_to_float Change;
    Change.IEE[3] = RS485_Recievebuf[23]; ///////////////正向累积量小数部分
    Change.IEE[2] = RS485_Recievebuf[24];
    Change.IEE[1] = RS485_Recievebuf[25];
    Change.IEE[0] = RS485_Recievebuf[26];
    temp = Change.FLOA;
    fff = (signed long)(temp*1000);
    disp_511_Number(2,0,ddd,9,0);disp_511_Number(2,80,fff,3,0);
    Change.IEE[3] = RS485_Recievebuf[31]; ///////////////反向累积量小数部分
    Change.IEE[2] = RS485_Recievebuf[32];
    Change.IEE[1] = RS485_Recievebuf[33];
    Change.IEE[0] = RS485_Recievebuf[34];
    temp = Change.FLOA;
    ggg = (signed long)(temp*1000);
    disp_511_Number(6,0,eee,9,0);disp_511_Number(6,80,ggg,3,0);
    switch(RS485_Recievebuf[38])
    {
        case 0x00:WriteMenu(2,112,0,23,0);WriteMenu(6,112,0,23,0);break;
        case 0x01:WriteMenu(2,112,0,24,0);WriteMenu(6,112,0,24,0);break;
        case 0x02:WriteMenu(2,112,0,25,0);WriteMenu(6,112,0,25,0);break;
        case 0x03:WriteMenu(2,112,0,23,0);WriteMenu(6,112,0,23,0);break;
        default:break;
    }
}

```