

TR030 温度变送器

使用说明书

V0.20



警告

担保

壹衡的产品对发货之日起一年内由于材料和制造工艺的缺陷提供担保。在担保期限内，壹衡对证明却有缺陷的产品，可以选择修理或更换。

担保的限制

上述的担保不适用于由于购买方不适当的或不正确的维护、使用购买方提供的产品或接口、非授权的调节或误用、在产品规定的环境条件之外工作、以及在不适当的场合进行配置或维护所造成的产品缺陷。使用该产品设计和实现电路是购买方单独的责任。壹衡不担保购买方的电路故障或由于购买方电路造成壹衡产品的故障。此外壹衡不担保购买方电路损坏以及由于使用购买方提供的产品造成的任何缺陷。

有限责任

壹衡不对任何直接的、间接的、特别的、偶然的或连带的损坏承担责任，不管是根据合同、民事行为或任何法律理论。

注意

本文件所包含的资料可以不经通知而修改。壹衡不提供任何商业的或适应特定目的的隐含担保。壹衡不对本资料中包含的错误或与本资料的提供、内容、使用相关造成的偶然的或间接地损坏承担责任。没有壹衡事先的书面许可，本资料不得复制、转载或翻译成任何语言。

安全

不要装上任何代用零部件或对产品进行任何非授权的调节。把产品返回壹衡销售和服务部门进行维修可确保产品的安全性能。

目录

警告	1
第一章 产品简介	1
一、测量特性	1
二、系统特性	1
四、外形和安装	2
五、接口定义和接线方式	2
第二章 快速入门	4
一、前期准备工作	4
二、接线	4
三、连接 PC 软件	5
第三章 温度设置软件	6
一、软件的下载与安装	6
二、主界面和参数配置界面	6
三、用户校准界面	10
四、高级参数界面	13
第四章 串口通讯	15
一、概述	15
二、Modbus-RTU	16
三、主动上报模式	22
第五章 技术规格	23
一、AD 特性	23
二、测量特性	23
三、使用环境	23

第一章 产品简介

TR030 温度变送器的作用是将“电阻温度探测器”（RTD）输出的电阻信号转换为数字温度信号，通过 RS485 串口输出。具有 Modbus-RTU 从机和主动上报两种通讯协议，可以连接 PC、PLC、HMI、组态屏等设备。

支持 Pt100、Cu100、Cu50 这三类 RTD。温度测量范围为 $-200^{\circ}\text{C}\sim 859^{\circ}\text{C}$ ，分辨率为 0.01°C ，精度为 0.1°C （典型值）。实际温度测量范围和精度还取决于 RTD 的测量范围和精度。

三线制和两线制两种测量方式可选。三线制测量时，变送器会对 RTD 导线电阻进行测量补偿，消除导线电阻误差。支持开机、手动零点自校准功能，可以消除系统零点的温度和时间漂移，保证测量的长期准确性。

提供免费的《温度设置软件》PC 软件，可读取温度、总电阻、导线电阻（三线制测量时）和 RTD 电阻，设置 RTD 的类型和模式，配置通讯模式和通讯参数，还支持用户校准，来匹配特定 RTD，消除误差。

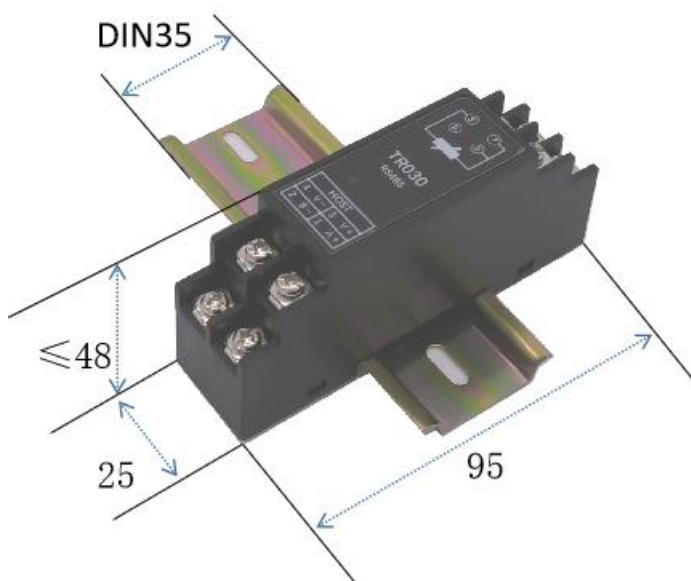
一、测量特性

- ◇ RTD 类型：Pt100、Cu100、Cu50 可选
- ◇ 测量方式：三线制、两线制可选
- ◇ 温度测量范围： $-200^{\circ}\text{C}\sim 859^{\circ}\text{C}$ ，分辨率： 0.01°C ，精度： 0.1°C （典型值）
- ◇ 总电阻测量范围： $0\ \Omega\sim 400\ \Omega$ ，分辨率： $1\text{m}\ \Omega$ ，精度： 0.02% （典型值）
- ◇ 导线电阻测量范围： $0\sim 40\ \Omega$ ，分辨率： $1\text{m}\ \Omega$ ，精度： 0.2% （典型值）
- ◇ 测量电流： $<0.55\text{mA}$
- ◇ 增益温度漂移： $15\text{ppm}/^{\circ}\text{C}$ （典型值）
- ◇ 采样速度： 10Hz 、 20Hz 、 80Hz 可选

二、系统特性

- ◇ 串行接口： RS485
- ◇ 通讯协议： Modbus-RTU 从机模式、主动上报模式可选
- ◇ 安装方式： 35mm 导轨安装
- ◇ 供电电源：直流 $6\sim 24\text{V}$ ， 10mA

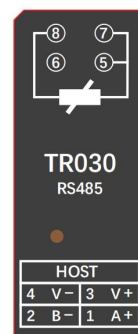
四、外形和安装



五、接口定义和接线方式

1. 接口定义

序号	符号	含义	接线方法
1	A+	RS485 的 A+	接主机 RS485 接口的 A+
2	B-	RS485 的 B-	接主机 RS485 接口的 B-
3	V+	电源+	接直流 6~24V 的电源+
4	V-	电源- RS485 的 GND	接直流 6~24V 的电源- 同时接主机 RS485 接口的 GND
5	T-	三线制 RTD 补偿导线接口	接三线制 RTD 补偿导线(注 1-1) 两线制 RTD 不连接
6	NC	内部没有连接	不连接
7	T-	RTD 电阻接口	接 RTD 电阻(注 1-2)
8	T+	RTD 电阻接口	接 RTD 电阻(注 1-3)



注 1-1: 三线制 RTD 三根导线，一般有两种颜色。其中两根颜色相同的导线，它们是导通的。可以任选其中的一根作为补偿导线接入 5 号端口，另一根接入 7 号端口。颜色不同的那根导线接入 8 号端口。

注 1-2: 两线制 RTD 两根导线，分别接入 7 号和 8 号端口即可。

注 1-3: 切勿将电源接入 5、7、8 号 RTD 测量端口，否则将立即损坏变送器，由此引起的损坏不在保修范围内！

2. RTD 接线示意图



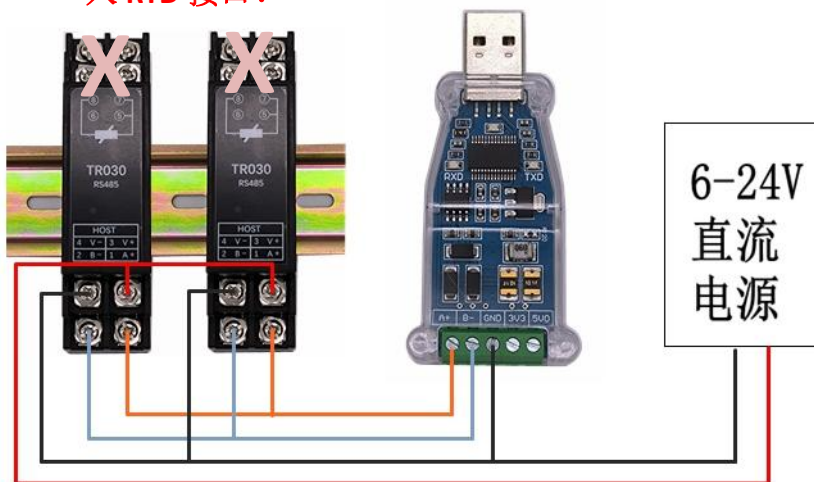
三线制接法

两线制接法

3. 通讯和电源接线示意图

切勿将电源接入 RTD 接口！

USB 转 RS485



切勿使用超过 24V 的直流电源，否则将立即损坏变送器，由此引起的损坏不在保修范围内！

必须将主机 RS485 的地（GND）与变送器的 V-相连接，否则可能会使通讯不可靠，甚至损坏变送器通讯接口，由此引起的损坏不在保修范围内！

第二章 快速入门

一、前期准备工作

1. 选择 RTD

变送器支持 Pt100、Cu100、Cu50 三类 RTD，其中 Pt100 精度最高，也最常用。建议使用三线制的 Pt100 温度传感器。

2. 准备直流电源

变送器需要直流 6~24V 电源供电，建议使用直流 12V 电源适配器。如使用 24V 电源适配器，需注意其空载电压不能大于 25V，否者容易损坏变送器。

3. 准备 RS485 通讯线

首次使用需要连接 PC 软件进行配置，变送器使用 RS485 进行通讯，而 PC 一般没有 RS485 接口，所以需要准备一根 USB 转 RS485 的通讯线。如果电脑有 RS232 接口的话，也可以使用 RS232 转 RS485 的通讯线。

二、接线

按第一章第五节所述方法，先将 RTD 和变送器连接好，然后将 RS485 通讯线的 RS485 端和变送器连接好，再把通讯线的另一端插入电脑的对应接口，最后接入直流电源，注意正负极性。

❖ **切勿将电源接入变送器 RTD 温度测量接口！**

❖ **否则将立即损坏变送器，由此引起的损坏不在保修范围内！**

三、连接 PC 软件

从网址 http://www.ehengio.com/file/tool/EH-TR03X_SET.zip 下载《温度设置软件》。解压缩后，双击 eh-tr030-set.exe 即可运行软件。

1. 软件操作方法简介

变送器出厂默认通讯参数为：通讯模式 Modbus-RTU，波特率 38400，校验方式为偶校验，通讯地址为 1。

软件使用的是 Modbus-RTU 协议，所以打开软件后，按默认通讯参数配置，选择对应的“端口号”，点击“打开”按钮，即可和变送器进行通讯。见下图：



打开后如下图显示：



从软件中可以看到当前的温度测量值和各种电阻值测量值，还可以设置各种参数。具体内容详见第三章。

第三章 温度设置软件

一、软件的下载与安装

从网址 http://www.ehengio.com/file/tool/EH-TR03X_SET.zip 下载《温度设置软件》。解压缩后，双击 eh-tr030-set.exe 即可运行软件。

二、主界面和参数配置界面



1. 通讯接口区



通讯接口区，主要用于来配置 PC 和变送器之间使用什么通讯参数进行通讯。

a. 端口号

软件和变送器进行通讯，首先需要确定 PC 和变送器通讯使用的是哪个端口号。本软件可选择的端口号和 PC 实际存在的端口号一致，如果使用的是 USB 转 RS485 通讯线的话，必须先在电脑上插上通讯线（如果需要安装驱动的话则先安装驱动），然后再打开软件才能看到这个端口号。通过这个特性，可以在用 USB 转 RS485 通讯线时快速的判断出使用的端口号。

b. 通讯参数

然后需要确定变送器的通讯参数，软件固定使用 Modbus-RTU 协议和变送器进行通讯，所以只要设置波特率、校验方式和通讯地址就可以了。

变送器出厂的默认参数为：通讯模式 Modbus-RTU，波特率 38400，校验方式为偶校验，通讯地址为 1。如果已知变送器的通讯参数，且变送器的通讯模式为 Modbus-RTU，那么配置好后点击“打开”按钮。

如果不知道变送器的通讯参数，或者变送器的通讯模式不是 Modbus-RTU，则必须要使用“自适应”功能来连接软件。

操作方法如下：勾选“自适应”选项框，点击“打开”按钮。然后给变送器断电 3 秒后重新上电。此时软件就可以和变送器进行通讯，同时也可以读出，变送器当前的通讯参数。

c. 连接操作小结

先选择正确的端口号，然后按下表所述连接软件

变送器通讯模式	其他通讯参数	连接方法
Modbus-RTU	已知	按已知的通讯参数配置软件后点击“打开”
Modbus-RTU	未知	勾选“自适应”后点击打开
非 Modbus-RTU	已知	
非 Modbus-RTU	未知	

2. 测量值显示区

0.19 °C	总电阻 (mΩ)	-	导线电阻 (mΩ)	=	RTD电阻 (mΩ)
	101335		1260		100075

测量值显示区，用于显示测量的结果。包括温度、总电阻、导线电阻（三线制测量时）和 RTD 电阻。其中 RTD 电阻 = 总电阻 - 导线电阻。温度是根据 RTD 电阻，查对应的温阻表，计算所得。RTD 的类型不同温阻表也不相同。

导线电阻每 3 秒自动补偿一次，期间总电阻不刷新。可以通过高级参数来禁用导线电阻实时补偿功能，使总电阻的刷新不受影响。

3. 通讯参数区

通讯模式	波特率	校验方式	通讯地址 (1-246)
Modbus-RTU协议	38400	偶校验	1
发送延时 (0-255ms)	上报间隔 (0-65535*0.1s)		
0	10		

通讯参数区用于配置变送器的通讯参数。

a. 通讯模式

通讯模式有两种：**Modbus-RTU 协议**和**主动上报模式**。**Modbus-RTU 协议**是一种命令模式，需要上位机发送命令，变送器才会应答数据。而**主动上报模式**则是变送器主动的周期性的发送测量数据（发送周期通过“上报间隔”参数设置），无需上位机发送命令。关于这两种通讯模式的详细描述，见第四章。

与 **Modbus-RTU 协议**有关的其他通讯参数是：波特率、校验方式、通讯地址和发送延时。与**主动上报模式**有关的其他通讯参数是：波特率、校验方式和上报间隔。

b. 发送延时的作用

大部分的 **USB 或 RS232 转 RS485** 的通讯线，收发切换都会有一个延时。这样在上位机发送完数据后，通讯线还处于发送状态，接收不到变送器传回的数据，或造成接收数据不稳定。这个问题有一个明确的表现就是，变送器指示灯处于正常的高频闪烁（约 2Hz）状态，但是上位机接收不到数据，或者接收不稳定，就有可能是这个问题。

这个时候就可以设置“发送延时”参数，让变送器在收到 PC 的命令后，延时一会再发，就可以避免这个问题。推荐设置为 20ms 左右。如果是原生的 **RS485** 接口，则不存在这个问题，发送延时推荐设置为 0ms。

4. 测量参数区

RTD类型	采样速度	无效时输出(0.01℃)	使用用户校准
PT100(三线制)	20Hz	-20000	不使用

测量参数区主要用于配置测量相关的参数。

RTD 类型用于配置 RTD 的类型（Pt100、Cu100 或 Cu50）和测量模式（三线制或两线制）。

无效时输出的温度值，用于指定测量无效时（比如 RTD 断线），变送器输出的测量值，它的单位是 0.01℃。如果上位机需要对异常情况进行处理，那么把这个值设定为一个特定的不可能到达的温度，例如-20000（就是-200℃）。如果上位机需要忽略偶发的异常情况，那么可以把无效时输出值设定为一个特定的值 2147483647，那么出现异常时变送器就维持上一次的测量值不变。

用户校准默认为不使用，这里只能查看不能修改。如需使用用户校准，见下一节。

5. 功能按键区



a. 恢复默认参数

功能是把全部用户参数恢复到出厂默认值，包括通讯参数、测量参数、用户校准参数和高级参数等。请谨慎操作。

b. ADC 零点自校准

功能是手动触发 ADC 零点自校准。

c. 导线电阻补偿

功能是手动触发导线电阻补偿。只有在用高级参数禁用实时导线电阻补偿后，此功能才能使用。

d. 默认、读取和写入

“默认”按钮的功能是把当前页面的参数恢复成默认值，但不写入变送器。

“读取”按钮的功能是把变送器内部的参数读取到当前页面。

“写入”按钮的功能是把当前页面的参数写入变送器。

三、用户校准界面

1. 用户校准的使用条件

变送器出厂时已经进行过校准，一般无需再进行“用户校准”。应保持“使用用户校准”选项为“不使用”状态（见下图）。错误的用户校准，可能降低系统的精度，甚至造成测量错误。当使用用户校准时，发现测量误差比较大或者测量错误时，可以尝试选择不使用“用户校准功能”



对于具备专业能力和设备的用户，可以通过用户校准进一步提高测量精度。可以让变送器在特定的温度下，输出希望得到的测量值。

比如在冰水混合物，变送器输出为 0.03°C ，此时可以通过用户校准，使变送器输出 0.00°C 。但需要注意的是，冰水混合物的内的温度也不一定是 0.00°C ，如果存在对外热传导的话，冰水混合物就会有一个温度梯度，测得 0.03°C 可能是一个正确的值，把它校准成 0.00°C 反而带来误差。这也是不建议普通用户进行“用户校准”的原因之一。

2. 用户校准概述

测量误差分为零点误差和增益误差两种。比如说两线制测量时 RTD，导线电阻带来的误差，就是零点误差；RTD 电阻的制造偏差，就属于增益误差。零点误差不会随温度的变化而变化，而增益误差则是会随温度的变化，而发生变化。

如果进行“一点式校准”，就是用一个温度点来校准，就需要用户判断，这个误差是零点误差还是增益误差。如果无法确定为增益误差，一般默认为零点误差。而进行“两点式校准”，就是用两个温度点来校准，则软件会同时进行进行零点和增益的校准。

零点校准影响的是“零点偏移值”（初始值为 0），增益校准影响的是“增益修正系数”（初始值为 1.000000）（见下图）。根据执行的校准方式不同，软件会自动修改这些参数，并写入变送器。



根据信号源的不同，校准还分为“温度方式校准”和“电阻方式校准”，“温度方式校准”输入的目标值是温度，电阻方式校准输入的目标值是电阻。除了使用标准电阻来校准变送器的自身测量误差（这个在工厂内已经校准过），一般使用的是“温度方式校准”来校准 RTD 的误差。下面以“温度方式校准”来进行说明。

3. 一点式温度校准

将“校准方式”选择为“一点式校准”将 RTD 放置与已知的温度里，等待温度稳定，且变送器的温度输出稳定。

如果选择修正零点误差，则在“温度目标值 1”中输入目标温度然后点击右侧的“确定”按钮。（如下图），完成后“零点偏移值”会发生变化，“增益修正系数”恢复初始值 1.000000。

校准方式选择	温度方式修正	电阻方式修正
<input style="width: 100%;" type="text" value="一点式校正"/>	目标温度值1 (°C) <input style="width: 100%;" type="text" value="0.00"/>	目标电阻值1 (mΩ) <input style="width: 100%;" type="text"/>
零点方式修正:	<input type="button" value="确认"/>	<input type="button" value="确认"/>
增益方式修正:	目标温度值2 (°C) <input style="width: 100%;" type="text"/>	目标电阻值2 (mΩ) <input style="width: 100%;" type="text"/>
	<input type="button" value="确认"/>	<input type="button" value="确认"/>

如果选择修正零点误差，则在“温度目标值 2”中输入目标温度。然后点击右侧的“确定”按钮。（如下图）完成后“增益修正系数”会发生变化，“零点偏移值”恢复初始值 0。

校准方式选择	温度方式修正	电阻方式修正
<input style="width: 100%;" type="text" value="一点式校正"/>	目标温度值1 (°C) <input style="width: 100%;" type="text"/>	目标电阻值1 (mΩ) <input style="width: 100%;" type="text"/>
零点方式修正:	<input type="button" value="确认"/>	<input type="button" value="确认"/>
增益方式修正:	目标温度值2 (°C) <input style="width: 100%;" type="text" value="40.00"/>	目标电阻值2 (mΩ) <input style="width: 100%;" type="text"/>
	<input type="button" value="确认"/>	<input type="button" value="确认"/>

4. 两点式温度校准

将“校准方式”选择为“两点式校准”将 RTD 放置与已知的较低的温度里，等待温度稳定，且变送器的温度输出稳定。然后再“目标温度值 1”里输入这个较低的温度，然后点击右侧的“确定”按钮。

再将 RTD 放置与已知的较高的温度里，等待温度稳定，且变送器的温度输出稳定。然后再“目标温度值 2”里输入这个较高的温度，然后点击右侧的“确定”按钮。完成后“增益修正系数”和“零点偏移值”都会发生变化。

校准方式选择	温度方式修正	电阻方式修正
<input type="button" value="两点式校正"/>	目标温度值1 (°C) <input style="width: 100%;" type="text" value="0.00"/> <input type="button" value="确认"/>	目标电阻值1 (mΩ) <input style="width: 100%;" type="text"/> <input type="button" value="确认"/>
零点方式修正:	目标温度值2 (°C) <input style="width: 100%;" type="text" value="40.00"/> <input type="button" value="确认"/>	目标电阻值2 (mΩ) <input style="width: 100%;" type="text"/> <input type="button" value="确认"/>
增益方式修正:		

上述校准流程也可以反过来，先校准较高的温度，然后再校准较低的温度。注意：较低的温度都是输入在“目标温度值 1”，较高的温度都是输入在“目标温度值 2”，不会随校准顺序而改变。

为了保证增益修正的准确性，目标温度之差需 $\geq 40^{\circ}\text{C}$ ，目标电阻之差需大于 $40\ \Omega$ 。否则会校准失败。

完成第一点的校准后，对应的“确认”按钮会变绿色（见下图），以表示已经进行过该点的校准。这个校准数据，会在模块中保存。也就是说，两点校准过程中，模块可以断电，而不会影响校准结果。

校准方式选择	温度方式修正	电阻方式修正
<input type="button" value="两点式校正"/>	目标温度值1 (°C) <input style="width: 100%;" type="text" value="0.00"/> <input style="background-color: #90EE90;" type="button" value="确认"/>	目标电阻值1 (mΩ) <input style="width: 100%;" type="text"/> <input type="button" value="确认"/>
零点方式修正:	目标温度值2 (°C) <input style="width: 100%;" type="text"/> <input type="button" value="确认"/>	目标电阻值2 (mΩ) <input style="width: 100%;" type="text"/> <input type="button" value="确认"/>
增益方式修正:		

5. 手动方式校准

如果确切的知道误差的类别和大小，可以直接输入“零点偏移值”或“增益修正系数”，然后点击“写入”按钮，写入参数。

例如：两线制测量时，知道两根导线的电阻和为 $100\text{m}\ \Omega$ ，那么可以在“零点偏移值”内输入“4000”（因为单位是 $0.25\text{m}\ \Omega$ ），然后点击“写入”按钮。

例如：知道 PT100 在 0°C 时的初始电阻值为 $99.98\ \Omega$ ，那么可以在“增益修正系数”内输入“0.9998”，然后点击“写入”按钮

四、高级参数界面

1. 概述

高级参数包括“禁用闪变抑制功能”、“禁用开机 ADC 零点自校准”、“禁用开机导线电阻补偿”和“实时导线电阻补偿”四个。一般使用时不需要修改，正常显示这个配置页面也是隐藏的。如需打开高级参数配置页面，则需要按住“Alt”的同时用鼠标左键双击“参数设置”四个字的区域（见下图红圈）。



完成后会显示如下界面：



2. 闪变抑制功能

当温度处于一个临界值比如说 100.005°C ，此时变送器输出温度值会在 100.00°C 和 100.01°C 之间来回跳动。变送器默认会开启“闪变抑制功能”来消除这个跳动，让他停留在其中的一个数值上，这样看上去温度的稳定性会提高。如果想实时的反应温度的测量状态，则可以“禁用闪变抑制功能”。

3. 开机 ADC 零点自校准

ADC 芯片内部的零点会随着时间和温度进行漂移，从而影响测量的准确性。变送器默认会在上电的时候进行一次 ADC 零点自校准，这样就可以消除 ADC 零点的漂移。

但开机 ADC 零点自校准功能约需要 6 秒才能完成，会延长开机自检的时间。如果需要加快变送器开机的速度，可以设置“禁用开机 ADC 零点自校准”，使用工厂零点自校准的值。但是这样 ADC 的零点误差的变化就没办法消除了，不推荐使用。还需要注意的是 ADC 零点自校准，需在正确连接 PT100 后才能进行，否则可能会校准失败。

4. 实时导线电阻补偿

变送器默认开启导线电阻实时补偿功能，变送器每 3 秒采集一次导线电阻来进行补偿。这样可以实时修正导线电阻变化的误差。采集导线电阻时会停止对 RTD 电阻的采样，这样就会造成温度刷新会有一个小小的停顿。

因为导线电阻变化量非常小，如果忽略导线电阻在使用中的变化，则可以“禁用实时导线电阻补偿”，这样温度的刷新就是连续的。此时还可以在需要的时候发送“导线电阻补偿命令”，手动的进行补偿。“导线电阻补偿命令”约需要 3 秒钟才能完成，此时温度值不再刷新。

5. 开机导线电阻补偿

“禁用实时导线电阻补偿”时，最好能够取消“禁用开机导线电阻补偿”，而使用“开机导线电阻补偿”功能。

“开机导线电阻补偿”约需要 3 秒钟才能完成，会加长开机自检的时间。为了避免无效的开机延时，“开机导线电阻补偿”功能，只有在“禁用实时导线电阻补偿”时才有效。

第四章 串口通讯

一、概述

变送器提供一路 RS485 串口，支持 Modbus-RTU 协议和主动上报模式。通过《温度设置软件》中设置页面可以进行选择（见下图）。

The screenshot shows a software interface with two tabs: "参数设置" (Parameter Settings) and "用户校准" (User Calibration). The "参数设置" tab is active. It contains the following fields:

通讯模式	波特率	校验方式	通讯地址 (1-246)
Modbus-RTU协议 主动上报模式	38400	偶校验	1
	上报间隔 (0-65535*0.1s)		

二、Modbus-RTU

1. Modbus-RTU 协议

设备提供标准的 Modbus-RTU 协议，协议内容请参考以下文档：

《Modbus 应用协议》（国标 GB/T19582.1-2008）

《Modbus 协议在串行链路上的实现指南》（国标 GB/T19582.2-2008）

详见《TR030 Modbus 说明书》

2. 输入寄存器含义

Modbus 地址	PLC 地址	长度	含义	备注
0	30001	2	温度测量值	单位为 0.01℃（注 4-1） 例如：12345 表示 123.45℃
2	30003	1	状态字 1	（注 4-2）
3	30004	1	状态字 2	（注 4-3）
4	30005	2	PT100 的电阻值	单位 mΩ PT100 的电阻值 = 测量回路总电阻值 - 导线的电阻值
6	30007	2	测量回路总电阻值	
8	30009	2	导线电阻值	
10	30011	2	总电阻 AD	
12	30013	2	导线电阻 AD	
14	30015	1	变送器自身温度	单位为℃
15	30016	5	备用	
20	30021	2	模块型号	
22	30023	1	硬件版本号	
23	30024	1	软件版本号	
24	30025	1	程序校验和	
25	30026	1	用户参数校验和	
26	30027	1	厂商参数校验和	

注 4-1: Modbus 每个地址对应一个字（两个字节），长度为 1 表明这个数据为一个字（两个字节）的长度，长度为 2 表明这个数据为两个字（四个字节）的长

TR030 使用说明

度（其中高字在前）。

比如温度为 800℃时，温度测量值就是 80000（固定两位小数点），用十六进制表示就是 0x00013880。那么输入寄存器地址 0（保存的高字）就等于 0x0001，输入寄存器地址 1（保存的低字）就等于 0x3880。

注 4-2：状态字 1 含义

位	含义	备注
0	温度超范围	温度测量值无效
1	PT100 开路	温度测量值无效
2	PT100 短路	温度测量值无效
3	补偿导线开路	温度测量值无效
4	ADC 故障	温度测量值无效
5	开机自检中	温度测量值无效
6	内部校准中	温度测量值不刷新
7	其他异常	温度测量值无效
8	RTD 类型	0: PT100（三线制） 1: Cu100（三线制） 2: Cu50（三线制） 3: 定制（三线制） 4: PT100（两线制） 5: Cu100（两线制） 6: Cu50（两线制） 7: 定制（两线制）
9		
10		
11		
12-13	备用	
14	允许用户校准	详见《用户校准 Modbus 说明书》。
15	允许工厂校准	内部使用

注 4-3：状态字 2 含义

位	含义	备注
0-7	软件版本号	固定 1 位小数 例如 12 表示版本号为 1.2
8	是否使用用户校准	0: 不使用 1: 使用
9-15	备用	

3. 保持寄存器含义

Modbus 地址	PLC 地址	长度	含义	备注
0	40001	10	输入寄存器镜像	同输入寄存器地址 0~9，只读。（注 4-4）
10	40011	1	通讯模式	0 : Modbus-RTU 协议（注 4-5） 1 : 主动上报模式 2 : 外接显示屏 LED-035 通讯协议
11	40012	1	波特率	0 : 600 1 : 1200 2 : 2400 3 : 4800 4 : 9600 5 : 14400 6 : 19200 7 : 28800 8 : 38400 9 : 57600 10 : 76800 11 : 96000 12 : 115200 13 : 160000 14 : 200000 15 : 250000
12	40013	1	校验方式	0 : 无校验 1 : 奇校验 2 : 偶校验 3 : 零校验
13	40014	1	通讯地址	1 ~246
14	40015	1	通讯发送延时	单位为 ms，0~255，默认为 20
15	40016	1	主动上报模式间隔	单位为 0.1s，0~65535，默认为 10
16	40017	12	备用	建议不要修改
28	40029	1	RTD 类型	0 : PT100（三线制） 1 : Cu100（三线制） 2 : Cu50（三线制） 3 : 定制（三线制） 4 : PT100（两线制） 5 : Cu100（两线制） 6 : Cu50（两线制） 7 : 定制（两线制）
29	40030	1	AD 采样速度	0 : 10Hz、 1 : 20Hz、 2 : 80Hz
30	40031	1	备用	建议不要修改
31	40032	1	禁用闪变抑制功能	0 : 允许 1 : 禁用 （见第五小节）
32	40033	1	禁用开机 ADC 零点自校准	0 : 允许 1 : 禁用
33	40034	1	禁用开机导线电阻补偿	0 : 允许 1 : 禁用
34	40035	1	禁用实时导线电阻补偿	0 : 允许 1 : 禁用
35	40036	1	使用用户校准	0 : 不使用 1 : 使用 （见第六小节）
36	40037	2	温度测量无效时的输出值	单位为 0.01℃ 默认为 -20000 （表示 -200.00℃） 2147483647 表示测量无效时，维持上一次有效的值，慎用！

TR030 使用说明

Modbus 地址	PLC 地址	长度	含义	备注
38	40039	8	备用	建议不要修改
46	40047	2	ADC 零点自校准命令	只写，读取固定为 0 写入 0x12341111：执行校准
48	40049	2	导线电阻补偿命令	只写，读取固定为 0 写入 0x12342222：执行补偿 只有禁用实时导线电阻补偿才有效
50	40051	2	恢复默认参数	只写，读取固定为 0 先写入 0x56782222 再写入 0x87653333 把上面的参数和用户校准系数都设置为默认值

注 4-4：输入寄存器和保持寄存器的地址 0~地址 9，映射相同的内容。可以通过 04 命令（读输入寄存器）或 03 命令（读保持寄存器），二选一来读取需要的信息。保持寄存器地址 0~地址 9，为只读，写入无效。

注 4-5：**蓝色粗体**为出厂默认值，下同。

4. 不使用协议读取 Modbus-RTU 的温度测量值

如果用户不了解 Modbus-RTU 协议，也可以通过简单的串口收发，获得温度测量值。下面以变送器的通讯地址=1 为例：

上位发送：

0x01 0x04 0x00 0x00 0x00 0x02 **0x71** **0xCB**

0x01 对应的是变送器的通讯地址。

0x04 0x00 0x00 0x00 0x02 为固定值

0x71 **0xCB** 对应的是 CRC 校验，通讯地址不同 CRC 校验也不同。（注 4-6）

变送器应答：

0x01 0x04 0x04 **0x00** **0x01** **0x0A** **0xCC** **0xAC** **0xB1**

0x01 对应的是变送器的通讯地址。

0x04 0x04 为固定值

0x00 **0x01** **0x0A** **0xCC** 对应温度测量值（高字节在前）。（注 4-7）

温度测量值 = 0x00010ACC = 68300，温度为 683.00℃。

0xAC **0xB1** 对应的是 CRC 校验。

:注 4-6: 不同的通讯地址对应不同而读取命令，见下表：

地址	上位机（数据为十六进制）
1	01 04 00 00 00 02 71 CB
2	02 04 00 00 00 02 71 F8
3	03 04 00 00 00 02 70 29
4	04 04 00 00 00 02 71 9E
5	05 04 00 00 00 02 70 4F
6	06 04 00 00 00 02 70 7C
7	07 04 00 00 00 02 71 AD
8	08 04 00 00 00 02 71 52
9	09 04 00 00 00 02 70 83
10	0A 04 00 00 00 02 70 B0

注 4-7: 怎么得到测量数据的值

收到温度测量值数据为四个字节的补码, 计算机内部的有符号数也是用补码形式表示的, 所以可以直接把收到的四个字节拼成一个有符号的四字节整型,

以 C#语言为例:

```
byte[] rcvBuff= {0x01,0x04,0x04,0x00,0x01,0x0A,0xCC,0xAC,0xB1};

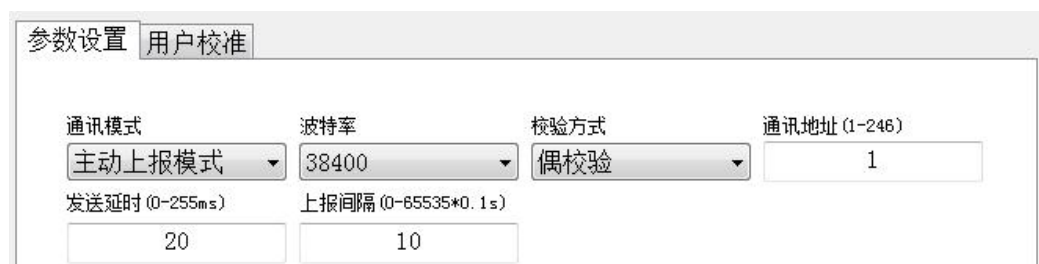
int temperature = 0;    //定义一个有符号的 32bit 的变量
for (int i = 3; i<= 6; i++) //把 rcvBuff[3]~ rcvBuff[6]拼成一个四字节整型
{
    temperature <<= 8;
    temperature += rcvBuff[i];
}
```


三、主动上报模式

1. 概述

变送器支持主动上报模式，在该模式下变送器，主动地向上位机发送测量结果，上位机无需发送数据，这样可以简化编程。主动上报的间隔可以通过《温度设置软件》中设置页面进行配置（见下图）。

上报间隔的单位为 0.1s，如需 1 秒上报一次，这个参数就应该设置为 10。



2. 协议格式

每帧数据由 10 个 ASC 字符组成，含义如下：

序号	含义	备注
1	起始字符	固定为 '='
2	测量值正负号	正值为 '0'，负值为 '-'
3	测量值的百位	'0' ~ '9', 没有的话填 '0'
4	测量值的十位	'0' ~ '9', 没有的话填 '0'
5	测量值的个位	'0' ~ '9', 没有的话填 '0'
6	小数点	固定为 '.'
7	测量值的十分位	'0' ~ '9', 没有的话填 '0'
8	测量值的百分位	'0' ~ '9', 没有的话填 '0'
9	回车符	固定为 0x0D
10	换行符	固定为 0x0A

可以使用串口调试软件，接收并保存测量这些测量数据，如下图：



第五章 技术规格

一、AD 特性

测量方法:	Σ - Δ 型 ADC
转换速度:	10~80 次/秒
增益漂移:	15ppm/ $^{\circ}$ C (典型值)

二、测量特性

RTD 类型:	Pt100、Cu100、Cu50
测量模式:	三线制、两线制
温度测量范围:	-200 $^{\circ}$ C~859 $^{\circ}$ C
温度分辨率:	0.01 $^{\circ}$ C
温度精度:	0.1 $^{\circ}$ C (典型值)
测量电流:	<0.55mA

三、使用环境

电源电压:	6~24VDC
工作温度:	-10 $^{\circ}$ C~40 $^{\circ}$ C
贮存温度:	-25 $^{\circ}$ C~55 $^{\circ}$ C
相对湿度	\leq 85%RH (无冷凝)