**封面**



振动电阻式传感器测量模块

RM501/502

用

户

手

册

* 高精度同步采样
* 全自动正测反测

河北稳控科技股份有限公司（固件版本：V1.00）

2024年06月（文档版本：V1.01）

**前言**

感谢使用稳控科技生产的RM502模块。本产品采用了高精度模拟信号驱动和采集技术，特别适用于差动电阻式传感器以及其它对电阻精度要求较高的传感器的驱动和测量。恒流驱动以及4路实时电流测量、高精度采样电阻，有效规避了因环境温度改变而导致的测量误差；10路高精度差分AD转换，降低测量环境中的共模干扰；可编程增益放大功能，使本产品对小信号也有极高灵敏度；另外，本产品在设计过程中充分考虑了电路不对称以及测量过程中传感器连接不可靠等实际因素，采用硬件方式实现了传感器的正反测自动切换功能。

用户接口方面，提供了UART和IIC数字接口，提供了两路模拟电压输出接口。

本产品体积小、集成度高，适用于进行终端设备开发，例如：手持/便携式测量仪器、全自动信息化监测仪等。

RM502设计及生产过程中，充分参考、执行了以下标准（规范），符合这些文件中对于读数仪的直接或者间接的指标要求。

《DL/T 1742-2017 差动电阻式仪器测量仪表》

《GB T 3408.1-2008大坝监测仪器应变计第1部分》（差动电阻式传感器）

请务必按照本手册具体说明操作或设计外围电路，本公司不承担由于不正常操作造成的财产损失或者人身伤害责任。请严格按照手册中的技术规格和参数使用或设计开发相应的产品。

RM502的目标，是测量多种类型的基于电阻的传感器，例如：差阻式传感器、惠斯通电桥、NTC、RTD、热电偶等。目前版本仅实现了差阻式传感器的测读。本公司承诺，在产品的升级过程中，会最大限度保证硬件、软件接口的兼容性。

**主要特性**

* **外形尺寸：**  
  RM501/2 30.0mmX26.0mmX4.3mm 贴插封装-20
* **供电电压：**DC3.3V单电源
* **数字接口：**UART+I2CUART：TTL/RS232/RS485，通讯速率9.6~460.8kbps（默认9600bps）  
  I2C：通讯速率50~500kHz
* **模拟输出：**两路电压信号输出，分辨率1/4095
* **测量范围：**总电阻：0~500Ω（线阻+传感器电阻R1+R2）  
  线缆电阻：0~150Ω（@0.4平方线缆1000米，或者1平方线缆2000米）  
  传感器内部电阻：0~150Ω  
  Z值：计算值，等于R1/R2
* **示值分辨率（有效分辨率）：**电流：0.00001mA（0.01uA）  
  电阻：0.0001Ω（0.1mΩ）  
  Z值：0.00001（0.001%/十万分之一）
* **测量精度：**电阻：优于±0.001Ω  
  Z值：优于±0.01%（示值的万分之一）
* **测量速率：**0.2~100次/秒（默认1Hz）
* **测量模式：**可正测、反测、正反测（取平均）
* **通讯协议：**MODBUS、字符串指令集
* **其它特性：**供电电压测量、环境温度测量  
  利用正反测评定测量精度  
  测线电阻测量
* **唯一识别码：**全球唯一识别码
* **工业温度范围：**-40℃~+85℃

**应用领域**

差阻式传感器测量仪器仪表开发

差阻式传感器生产检测

自动化、信息化监测系统

**订购信息**

附加编码

D

C

B

RM

A

通道数量

封装样式

0：贴片（或贴片、直插兼容）

1：直插

系列代码

5：小体积封装模块

类别标识

固定为“RM”（电阻测量模块）

RM501：单通道测量模块

RM502：双通道测量模块

**版权及商标**

本手册版权属于河北稳控科技股份有限公司，所有内容受著作权保护，且为我公司之财产。任何个人或团体未经我公司书面同意，严禁重制、复制、引用或者修改本手册中的任何部分，稳控科技保留一切解释权利。

我公司拒绝任何超出法律保证的赔偿要求。对于手册内容正确性，不负担任何责任。本手册内容或手册所述之产品（或固件程序）若有变动，恕不预先通知。

为我公司注册商标，对本商标造成危害、损失的一切行为，本公司保留所有权利。

联系电话：0316-5999728 400-096-5525

企业网址：[www.winkooo.cn](http://www.winkooo.cn)

邮箱： [info@GEO-INS.com](mailto:info@GEO-INS.com)

[info@GEO-Explorer.cn](mailto:info@GEO-Explorer.cn)

河北稳控科技股份有限公司

**关于本手册**

本手册是系列产品RM5XX模块的使用指导书，专门提供给终端客户和电子产品设计工程师，完成不同层面的应用。

RM系列模块具有近乎相同的使用方法，以下介绍中，如无特殊说明，均以RM502为例。

***产品默认出厂参数已设置为适用于绝大多数传感器，一般无需修改。***

本手册中表示不同进制数字时会使用明确的标识符号，以0x为前缀的数字为16进制数据，以B为后缀的为2进制数据，无任何前缀或后缀的数字为10进制数据。“\r\n”表示非可见的回车符和换行符。

寄存器通常以简称表示（多个大写字母组合），某寄存器中的某些“位”则以“寄存器简称.[高位数字:低位数字]”形式表示。

带有“\*”的章节表示功能尚不稳定或未经充分测试，请在咨询后使用。

由于产品版本升级或其他原因，本文档内容会不定期进行更新，请根据本手册对应的硬件及固件程序版本使用，必要时向我们索要与您实际使用产品相匹配的技术手册。

除非另有约定，本文档仅作为使用指导，本文档中的所有陈述、建议和举例不构成任何明示或暗示的担保，对此造成的损失不承担任何责任，本手册所述内容最终解释权归我公司所有。

**基本概念**

**差动电阻式传感器：**（Elastic wire sensor）是以两个电阻的比值来反映被测物理量的传感器。由于是由卡尔逊教授在1932年研发成功，所以也称为卡尔逊式传感器。因为采用的是两个电阻的电阻比来计算物理量，所以可以抵消环境温度引起的不利影响。目前主流的测量方式为五线制，可以去除传感器线缆电阻的影响。

**积分时长：**在进行传感器信号采样过程中，对信号的持续连续采样时长。一般而言，积分时长越长，采样值越精准，对周期性的干扰的滤除作用越明显（例如工频率50Hz干扰）。积分时长直接影响到采样的转换速率，与转换速率是反比关系。

**采样数量：**（N Samples），采样时，独立进行多少次采样再进行平均值计算。上述“积分时长”专指单次采样，而“采样数量”是规定每次测量时进行多少次独立采样之后再计算最终结果。

**滤波：**在本手册中专指使用历史测量数据进行滤波处理。如果采用了滤波功能，数值的稳定性会有所提升，但会存在一定的滞后性。

**测量质量：**在本手册中专指利用正测和反测之后得到的两个结果的不一致性。

目录

[**封面** 1](#_Toc168585383)

[**前言** 2](#_Toc168585384)

[**主要特性** 3](#_Toc168585385)

[**应用领域** 4](#_Toc168585386)

[**订购信息** 4](#_Toc168585387)

[**版权及商标** 5](#_Toc168585388)

[**关于本手册** 6](#_Toc168585389)

[**基本概念** 7](#_Toc168585390)

[**一、总体说明** 10](#_Toc168585391)

[1.1绝对最大值 10](#_Toc168585392)

[1.2特性及指标 10](#_Toc168585393)

[1.3管脚定义 12](#_Toc168585394)

[1.3.1RM501/502管脚定义 12](#_Toc168585395)

[1.4封装尺寸 13](#_Toc168585396)

[**二、硬件接口** 14](#_Toc168585397)

[2.1电源接口 14](#_Toc168585398)

[2.2参数复位管脚 14](#_Toc168585399)

[2.3数字接口1（UART） 14](#_Toc168585400)

[2.4数字接口2（IIC） 14](#_Toc168585401)

[2.4.1设备地址 15](#_Toc168585402)

[2.4.2IIC协议硬件层信号类别及说明 15](#_Toc168585403)

[2.5传感器接口 15](#_Toc168585404)

[2.5.1差阻式传感器不同线制及线缆颜色定义 15](#_Toc168585405)

[2.5.2非五线制传感器改为五线制 15](#_Toc168585406)

[2.6模拟信号输出接口 16](#_Toc168585407)

[**三、模块使用** 17](#_Toc168585408)

[3.1模块启动 17](#_Toc168585409)

[3.1.1启动信息 17](#_Toc168585410)

[3.1.2启动流程 17](#_Toc168585411)

[3.1.3获取版本信息及序列号 17](#_Toc168585412)

[3.2模块复位（重启） 17](#_Toc168585413)

[3.3恢复出厂参数 18](#_Toc168585414)

[3.3.1恢复出厂参数 18](#_Toc168585415)

[3.3.2修改出厂参数 19](#_Toc168585416)

[3.3.2恢复默认参数 19](#_Toc168585417)

[3.4通讯协议 19](#_Toc168585418)

[3.4.1UART通讯协议 20](#_Toc168585419)

[3.4.2通讯协议（IIC） 24](#_Toc168585420)

[3.4.3主动上传测量数据（UART） 24](#_Toc168585421)

[3.5寄存器概述（汇总） 25](#_Toc168585422)

[3.6模块地址操作（UART） 27](#_Toc168585423)

[3.6.1修改已知设备地址的地址 27](#_Toc168585424)

[3.6.2读取未知设备地址的地址 27](#_Toc168585425)

[3.6.3修改未知设备地址的地址 27](#_Toc168585426)

[3.7通讯速率和软件握手（UART） 27](#_Toc168585427)

[3.7.1通讯速率 27](#_Toc168585428)

[3.7.2软件握手 28](#_Toc168585429)

[3.7.3修改已知设备地址的通讯速率 28](#_Toc168585430)

[3.7.4读取未知设备地址的通讯速率 28](#_Toc168585431)

[3.7.5修改未知设备地址的通讯速率 28](#_Toc168585432)

[3.8参数保存 28](#_Toc168585433)

[3.9系统状态 29](#_Toc168585434)

[3.10测量模式 29](#_Toc168585435)

[3.11测量流程 29](#_Toc168585436)

[3.12传感器接入检测 29](#_Toc168585437)

[3.13采样积分时长 29](#_Toc168585438)

[3.14采样内容 30](#_Toc168585439)

[3.15采样次数 30](#_Toc168585440)

[3.16历史数据滤波 30](#_Toc168585441)

[3.17正测和反测 31](#_Toc168585442)

[3.18模拟信号输出 31](#_Toc168585443)

[3.19辅助功能寄存器 32](#_Toc168585444)

[3.20数据更新速率 32](#_Toc168585445)

**一、总体说明**

1.1绝对最大值

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **参数** | **条件（备注）** | **最小值①** | **典型值** | **最大值①** | **单位** |
| **环境温度** |  | -40 |  | 85 | ℃ |
| **储存温度** |  | -65 |  | 150 | ℃ |
| **VDD** |  | -0.3 |  | 3.6 | V |
| **VI/O** |  | -0.3 |  | VDD+0.3 | V |
| **IIN** |  |  |  | 100 | mA |
| **IOUT** |  |  |  | 100 | mA |
| **最大结温** |  |  |  | 125 | ℃ |
| **注1：长时间在最大允许值或超过最大允许值的条件下工作可能导致器件永久性损坏。** | | | | | |

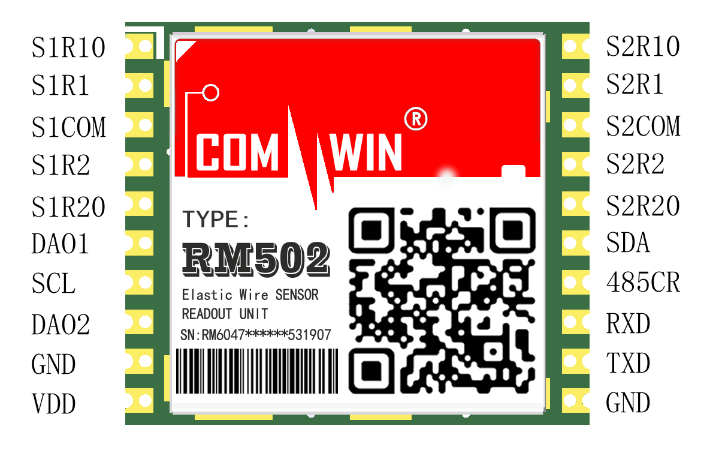
1.2特性及指标

供电电压3.3V，环境温度25℃

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 参数 | 条件 | 最小值 | 典型值 | 最大值 | 单位 |
| 电源 | | | | | |
| VDD |  | 3.0 | 3.3 | 3.6 | V |
| IVDD① | 空闲（未连接传感器） |  | 25 |  | mA |
| 测量（连接两路传感器） |  | 35 |  | mA |
| 休眠 |  |  |  | mA |
| 传感器激励与测量 | | | | | |
| 激励电流 | 常规测量 |  | 1.0 |  | mA |
| 低热测量 |  | 0.2 |  | mA |
| 电阻测量范围 | 差阻式传感器③ | 0 |  | 500 | Ω |
| 单个电阻最大检测③ | 120 |  | 150 | Ω |
|  |  |  |  |  |
| 模数转换  分辨率 | 有效位数② |  | 24 |  | bits |
| 平均有效值RMS② |  | 22 |  | bits |
| 峰峰有效值② |  | 20 |  | bits |
| 电阻分辨率 | 采样分辨率 |  | 0.00002 |  | Ω |
| 示值分辨率 |  | 0.0001 |  | Ω |
| 电阻测量误差 |  |  | ±0.0003 |  | Ω |
| 温漂 |  |  | 10 |  | ppm |
| 环境测量分辨率 | | | | | |
| 供电电压 |  |  |  |  | mV |
| 环境温度 |  |  |  |  |  |
| 模拟量输出 | | | | | |
| 输出分辨率 |  |  | 1/4095 |  |  |
| 驱动能力 |  |  | 1 |  | mA |
| 电压范围 |  |  | 2048 |  | mV |
|  |  |  |  |  |  |
| UART | | | | | |
| 通讯速率 |  | 9600 |  | 460800 | bps |
| 高电平 | TTL接口 | 1.2 | 3.3 | 3.6 | V |
| 低电平 | TTL接口 | 0 |  | 0.8 | V |
| 注①：连接传感器电阻为100Ω（R1+R2）  注②：采样速率0.2Hz条件下  注③：总电阻是4个电阻之和，两个线缆电阻L1、L5以及两个传感器电阻R1、R2，总电阻值检测范围为0~500Ω，单个电阻最大检测范围为0~150Ω。 | | | | | |

1.3管脚定义

1.3.1RM501/502管脚定义

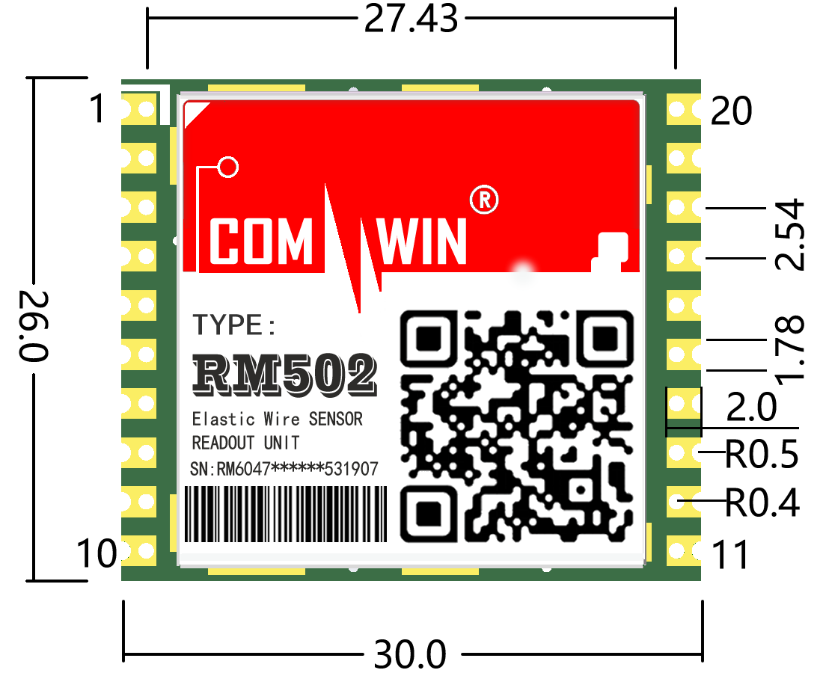


RM501/502管脚定义

RM501/RM502管脚定义

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **符号** | **编号** | | **类型** | **说明** |
| **501** | **502** |
| **S1R10** | 1 | 1 | 双向 | 连接差阻传感器白线 |
| **S1R1** | 2 | 2 | 输入 | 连接差阻传感器绿线 |
| **S1COM** | 3 | 3 | 输入 | 连接差阻传感器红线 |
| **S1R2** | 4 | 4 | 输入 | 连接差阻传感器黑线 |
| **S1R20** | 5 | 5 | 双向 | 连接差阻传感器蓝线 |
| **DAO1** | 6 | 6 | 输出 |  |
| **SCL** | 7 | 7 | 双向 | IIC接口时钟 |
| **DAO2** | 8 | 8 | 输出 |  |
| **GND** | 9 | 9 | 电源 |  |
| **VDD** | 10 | 10 | 电源 | 核心电源2.5~3.6V |
| **GND** | 11 | 11 | 电源 |  |
| **TXD** | 12 | 12 | 输出 | UART发送管脚 |
| **RXD** | 13 | 13 | 输入 | UART接收管脚 |
| **485CR** | 14 | 14 | 输出 | UART发送指示，可用作485收发控制 |
| **SDA/RST** | 15 | 15 | 双向 | IIC接口数据 |
| **S2R20** | 16 | 16 | 双向 | 连接差阻传感器蓝线 |
| **S2R2** | 17 | 17 | 输入 | 连接差阻传感器黑线 |
| **S2COM** | 18 | 18 | 输入 | 连接差阻传感器红线 |
| **S2R1** | 19 | 19 | 输入 | 连接差阻传感器绿线 |
| **S2R10** | 20 | 20 | 双向 | 连接差阻传感器白线 |
| **注：** | | | | |

1.4封装尺寸



RM501/502机械尺寸

**二、硬件接口**

2.1电源接口

RM501/502为单电源供电模块，在VDD及GND之间连接DC3.3V电源即可。

RM501/502为高精密模拟信号转换模块，对周边噪声十分敏感，建议使用低噪声、低纹波的LDO为模块供电。

建议靠近电源管脚（VDD）使用一个10µF钽电容（低ESR)和一个0.1µF的陶瓷电容并联。增加并联电容可以有效去除高频干扰。同时为防止浪涌对芯片的损坏，建议在模块电源输入管脚使用一个适合电压的500mW的齐纳二极管防止模块的超压损坏。PCB布局时，电容和二极管应尽可能靠近模块的电源输入管脚。

***注：模块没有反接电源及超压保护措施，反接电源及超压使用会导致永久性损坏。***

2.2参数复位管脚

Pin15（SDA/RST）管脚为双向管脚，在不同运行阶段具有不同功能：

上电启动时，RST管脚为输入，当检测到管脚为低电平时复位参数为出厂值。详见“3.3恢复出厂参数”。在启动完成后此管脚为IIC-SDA功能。

为了防止上电时参数复位，外部必须连接2k~4.7k上拉电阻。

2.3数字接口1（UART）

RM系列模块提供全双工串行TTL电平的UART接口，并支持外接RS485电平转换芯片。默认端口设置为“9600,N,8,1”，支持由软件修改为9600~460800bps通讯速率。

UART的TTL电平逻辑高为VDD，逻辑低为GND，与非3.3V单片机进行连接时，要注意逻辑电平的转换。

TXD为强推挽输出管脚，RXD为输入管脚。

管脚485CR为数据收发指示管脚，模块向外发送数据时管脚485CR输出高电平（强推挽），非发送时输出低电平。利用这一逻辑特性，当在UART外部连接RS485电平转换芯片时，发送指示管脚可作为半双工485芯片的收发控制管脚使用。

2.4数字接口2（IIC）

RM5XX支持双向IIC总线和数据传输协议，支持最高500kHz的通讯速率。

在总线中，RM5XX为从设备，与之通讯的上位机为主设备，主设备控制整个通讯过程。向总线发送数据的设备定义为发送器，接收数据的设备定义为接收器。总线必须由主设备来控制，由主设备产生串行时钟（SCL）、控制总线访问以及产生开始和停止信号（条件）。

RM5XX设备通过SCL和SDA线与总线连接，两根数据线均为漏极开路，与非3.3V单片机进行连接时，要注意逻辑电平的转换。

当使用IIC接口时，必须外接上拉电阻。

2.4.1设备地址

使用IIC总线时，RM5XX使用IIC专用的设备地址，设备的IIC地址默认为0xA0(160)，可通过特殊的UART接口指令来完成IIC地址的修改，修改后的地址永久保存。

IIC地址修改指令为：$IICA=xxx\r\n

需要注意的是，IIC地址必须为偶数，否则修改不能成功。

2.4.2IIC协议硬件层信号类别及说明

为了降低使用难度，RM系列模块的I2C接口采用了与AT24C02完全相同的读写时序，寄存器定义详见“3.5寄存器汇总”。

2.5传感器接口

RM模块可测量两个通道差阻式传感器。

2.5.1差阻式传感器不同线制及线缆颜色定义

目前，差阻式传感器几乎全部为5线制，少部分为4线制，极少有3线制（精度无法保证）。

R2

R1

R20

蓝

R2

黑

COM

红

R1

绿

R10

白

L1

L5

R2

R2

黑

COM

红

R1

绿

R10

白

R1

R20

蓝

R2

R20

蓝

R2

黑

COM

红

R1

绿

R1

R10

白

L5

三线制 四线制 五线制

上图为常规的差阻式传感器的线制、线色定义，以及与RM模块的连接示意图。

在实际测量时，应以传感器厂家提供的接线定义为准（而不能仅以本文档中标识的颜色为准）。

五线制测量方法可以从测量原理上排除线缆电阻对测量结果的不利影响，所以在高精度应用时或者线缆过长时，必须使用五线制测量。

2.5.2非五线制传感器改为五线制

若传感器厂家所提供的传感器为三线或者四线制时，可以将传感器线缆断开（断开点为传感器出线端），使用五芯线缆将其改为五线制（各颜色接法见上图）。

2.5.3传感器接口注意事项

差阻式传感器测量过程中，信号较为微弱，且行业规范中对电阻的测量精度要求很高，所以在使用RM模块时，必须使用短而粗的测量线缆或者线路（PCB布线），必须注意：五个信号线必须远离其它信号、电源。

在传感器接口的5个接线端，建议分别对地（GND）连接100pF~100nF电容，相邻引脚间连接100pF~100nF电容。

S1R10

S1R1

S1COM

S1R2

S1R20

2.6模拟信号输出接口

RM502模块有两路模块电压输出接口，Pin6（DAO1）和Pin8（DAO2），分别用于将两路差阻传感器的测量结果（Z1和Z2）以电压形式输出。

DAOx的输出电压范围为0~2048mV，分辨率为4095（12位DA）。

**三、模块使用**

本章主要内容为RM模块基本工作原理以及工作参数、实时数据解释说明。

模块出厂时的默认参数值能够满足大部分差阻传感器的数据读取，***无特殊情况不需要修改参数***。若需要修改某些参数时，务必详细阅读本章内容以了解参数含义。错误的参数值可能导致模块无法正常工作，必要时请使用参数复位功能将参数恢复为出厂值。

3.1模块启动

3.1.1启动信息

模块上电自行启动，初始化完成后输出如下启动信息（UART接口）

RM502 //模块系列名称  
HW:1.00 //硬件版本号  
SF:1.00\_240514 //固件版本号  
ADDR:001 //模块地址（基于UART和MODBUS协议地址）  
IICA:A0H(160) //I2C地址  
SN=46705731203936BA //模块机器码（序列号）

3.1.2启动流程

（1）读取存储的工作参数，进行参数校验，若校验错误则自动恢复为出厂值；

（2）将参数加载到对应的寄存器；

（3）若RST引脚为低电平则恢复寄存器值为出厂值；

（4）初始化各部分功能模块，期间检测各功能模块，发生错误时输出错误信息；

（5）经由UART接口输出启动信息；

（6）根据寄存器定义的工作模式，开始工作（输出BOOT OK）。

模块上电启动过程为200~500ms不等，建议在模块上电后500ms再开始向模块发起数据通讯操作。

3.1.3获取版本信息及序列号

向系统功能寄存器SYS\_FUN写入功能码03，模块返回固件版本信息及唯一序列号，输出信息格式详见“3.1.1启动信息”。

模块的版本信息和唯一码同时也保存于寄存器中，可以直接读取这些寄存器获取信息，详见“3.5寄存器定义”。

3.2模块复位（重启）

以下几种情况（或操作）可使模块产生复位动作，重新启动。

（1）在模块正常工作期间，向寄存器SYS\_FUN发送软复位指令0x01；

（2）内核电压过低或受到强电磁干扰；

（3）未知的非法参数设置，导致的工作异常；

3.3恢复出厂参数

设备内部存在三类系统参数，分别为：用户系统参数、出厂系统参数和默认系统参数。

用户系统参数：也称“工作参数”，可修改可保存，每次上电时自动加载并按照此参数运行。在设备使用过程中对参数的修改、设备运行逻辑均是指用户参数，用户参数是使用最频繁的参数类别。

出厂系统参数：保存于独立分区中的一组系统参数，仅当收到“恢复出厂参数”指令或者设备检测到用户参数异常而无法工作时才会读取并覆盖用户系统参数。出厂参数可使用特殊指令进行修改。设备出厂时此参数已由厂家进行了设置，建议不要修改（慎用）。

默认系统参数：默认参数是仅能保证设备能完成基本通讯工作的一组系统参数，此参数为固定参数，用户无法通过任何途径修改。当恢复出厂参数后，内部判断设备仍无法正常工作时会自动加载默认系统参数，以使基本的数字通讯可以进行。

$RSTP

$STDF

$STFC

3.3.1恢复出厂参数

有两种方法实现所有参数（寄存器）恢复为出厂值

（1）硬件参数复位：在模块启动时，检测到参数复位管脚RST为低电平时复位参数为出厂值，在管脚电平未恢复为高电平前，模块一直处于暂停状态。复位后，由UART输出提示信息“RST\r\n”。

（2）软件参数复位：向寄存器SYS\_FUN写入0x02。

以下几种情况时，模块会自动恢复为出厂参数

（1）参数CRC错误：上电过程中，检测到参数区校验码错误，自动恢复为出厂参数值，由UART输出提示信息“CRC Err\r\n”。

（2）UART通讯速率错误：上电过程中，检测到参数BAUD值为非法的通讯速率值，自动恢复为出厂参数值，由UART输出提示信息“BAUD Err\r\n”。

3.3.2修改出厂参数

使用当前的用户参数写入到出厂参数区。***此操作建议专业人员使用，普通用户不要轻易修改出厂参数。***

**使用指令设置**

$STFC\r\n

设备响应后返回字符串：OK\r\n

也可向系统功能寄存SYS\_FUN器写入0x0A实现相同的功能。

3.3.2恢复默认参数

将设备内部预先设置的一组固定参数加载到用户系统参数。指令如下：

$STDF\r\n

设备响应后返回字符串：OK\r\n

也可向系统功能寄存SYS\_FUN器写入0x0B实现相同的功能。

3.4通讯协议

通讯协议是上位机通过RMXXX模块支持的数字接口完成信息交互的数据格式、传输步骤、通讯速率等的一系列预先约定。上位机必须按照本章描述的通讯协议规则来完成与RMXXX的数据交互工作。

**寄存器机制**

RMXXX内部维护有若干寄存器，模块在寄存器参数值的控制下完成差阻传感器的测量工作。寄存器的值总是以整数形式存在，基本操作单位为“字”（2字节整数，大端模式），有掉电保存和掉电丢失两种类型（对应“读/写”和“只读”两种属性）。通过模块的数字接口可完成对寄存器的读取和写入（修改）操作。寄存器写（修改）寿命典型值为10万次，读取次数没有限制。

**数据模式**

寄存器数据值采用大端模式，数据的高字节保存在内存的低地址中，而数据的低字节保存在内存的高地址中，数据帧传输时先传输低地址字节后传输高字节。每个寄存器对应两个字节，则单个寄存器的值=低字节值\*256+高字节值。

**冲突解决**

当模块收到上位机指令时原则上会立即处理并返回应答信息，若模块“正忙”时，RMXXX采用测量优先的原则，等待当前测量周期结束才会响应。不同的工作参数会有不同的“忙”时长（详见“3.15测量时长与优化”），在发送指令后，若模块未能及时回复，应继续等待其响应，向模块发送多条指令没有意义，模块仅会响应最先收到的一条指令。

注：数字接口通讯会在一定程度上影响差阻传感器频率采集精度，不要过于频繁的对数字接口发送指令。

3.4.1UART通讯协议

UART接口支持标准的工业MODBUS通讯协议（03、04、06、16指令码）和自定义的简单AABB协议以及$字符串指令集。三种协议均支持基于模块地址和总线连接的一主多从应用结构，在总线中，RMXXX模块始终作为从机使用（被动等待指令，不主动上传数据，但“自动上传数据”和“软件握手”除外，详见后续对应章节说明）。

根据下述通讯协议规则，读写寄存器。

在使用MODBUS或AABB通讯协议时，请确认模块的软件握手参数处于关闭状态（默认）。

（一）MODBUS通讯协议

在MODBUS协议下，RMXXX内部所有寄存器被定义为“保持寄存器”（详见MODBUS通讯协议标准说明），模块支持基于MODBUS协议的多个连续寄存器读取、单个寄存器写入、多个连续寄存器写入三种指令，对应指令码分别为0x03（或0x04）、0x06、0x10。下面逐一说明每种指令码的指令和返回数据帧格式。

（1）03/04（0x03/0x04）指令码：读取多个连续的寄存器数据，指令格式如下

指令数据帧结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 地址码 | 功能码0x03或0x04 | 开始地址 | 寄存器数量 | CRC校验 |
| 1字节 | 1字节 | 2字节 | 2字节 | 2字节 |

返回数据帧结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 地址码 | 功能码0x03或0x04 | 数据长度 | 数据 | CRC校验 |
| 1字节 | 1字节 | 1字节 | n字节 | 2字节 |

例：读取地址为0x01的模块寄存器值，寄存器开始地址为0，连续读取10个寄存器

主机发送指令：0x01 0x03 0x00 0x00 0x00 0x0A ***0xC5 0xCD***

从机返回应答：0x01 0x03 0x14 0x00 0x01 0x00 0x60 0x00 0x18 0x00 0x00 0x01 0x01 0x00 0x01 0x00 0x0A 0x1F 0x1F 0x00 0x00 0x1F 0x1F ***0x2E 0x75（下划线为读取到的10个寄存器值）***

主机发送指令：0x01 0x04 0x00 0x00 0x00 0x0A ***0x70 0x0D***

从机返回应答：0x01 0x04 0x14 0x00 0x01 0x00 0x60 0x00 0x18 0x00 0x00 0x01 0x01 0x00 0x01 0x00 0x0A 0x1F 0x1F 0x00 0x00 0x1F 0x1F ***0x18 0x93（下划线为读取到的10个寄存器值）***

读取多个连续寄存器时，单次读取不要超过64个寄存器，不要试图读取不存在的寄存器。

（2）06（0x06）指令码：修改单个寄存器的值，指令格式如下

指令数据帧结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 地址码 | 功能码0x06 | 寄存器地址 | 寄存器值 | CRC校验 |
| 1字节 | 1字节 | 2字节 | 2字节 | 2字节 |

返回数据帧结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 地址码 | 功能码0x06 | 寄存器地址 | 寄存器值 | CRC校验 |
| 1字节 | 1字节 | 2字节 | 2字节 | 2字节 |

例：将地址为0x01的模块中的寄存器8的值修改为100

主机发送指令：0x01 0x06 0x00 0x08 0x00 0x64 ***0x09 0xE3***

从机返回应答：0x01 0x06 0x00 0x08 0x00 0x64 ***0x09 0xE3***

（3）16（0x10）指令码：修改连续的多个寄存器的值，指令格式如下

指令数据帧结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 地址码 | 功能码0x10 | 起始地址 | 寄存器数量 | 字节数 | 寄存器值 | CRC校验 |
| 1字节 | 1字节 | 2字节 | 2字节 | 1字节 | n字节 | 2字节 |

返回数据帧结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 地址码 | 功能码0x06 | 起始地址 | 寄存器数量 | CRC校验 |
| 1字节 | 1字节 | 2字节 | 2字节 | 2字节 |

例：修改地址为0x01的模块中0~9寄存器的值

主机发送指令（16进制指令）：01 10 00 00 00 0A 14 00 01 00 60 00 18 00 00 01 01 00 01 00 0A 1F 1F 00 00 1F 1F ***FD EC（下划线数据为10个寄存器值）***

从机返回应答：0x01 0x10 0x00 0x00 0x00 0x0A ***0x40 0x0E***

RMXXX的UART接收缓存为80字节，向模块发送多寄存器写指令时，单帧指令长度不要超限。

（二）AABB通讯协议

AABB通讯协议是一种非标准自定义协议，相较于MODBUS通讯协议，结构更简单，指令生成方法更容易，便于进行快速测试。AABB通讯协议支持单寄存器读写两种指令。

（1）读取单个寄存器

指令数据帧结构

|  |  |  |  |
| --- | --- | --- | --- |
| 指令头0xAA 0xBB | 地址码 | 寄存器地址 | 和校验 |
| 2字节 | 1字节 | 1字节 | 1字节 |

返回数据帧结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 指令头0xAA 0xBB | 地址码 | 寄存器地址 | 寄存器值 | 和校验 |
| 2字节 | 1字节 | 1字节 | 2字节 | 1字节 |

**指令头：**固定为16进制AABB

地址码：RMXXX模块的地址（1~255，其中地址255为通用地址，详见后续“通用模块地址”说明）

**寄存器地址：**要访问的寄存器地址（0~127），寄存器地址字节最高位是读写标志位，为0时表示读寄存器，为1时表示写寄存器。

**和校验：**之前所有数据之和，0xAA+0xBB+地址码+寄存器地址，校验和超过255时，仅使用低字节。如下例中，校验和=0xAA+0xBB+0x01+0x08=0x016E，则只使用0x6E作为最终和校验码。

例：读取地址为0x01的模块寄存器值，寄存器地址为8

主机发送指令：0xAA 0xBB 0x01 0x08 ***0x6E***

从机返回应答：0xAA 0xBB 0x01 0x08 0x00 0x60 ***0xCE***

（2）修改单个寄存器

指令数据帧结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 指令头0xAA 0xBB | 地址码 | 寄存器地址|0x80 | 寄存器值 | 和校验 |
| 2字节 | 1字节 | 1字节 | 2字节 | 1字节 |

写寄存器指令中，寄存器地址字节的最高位应为1，即地址值与0x80做“或”运算。

返回数据帧结构

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 指令头0xAA 0xBB | 地址码 | 寄存器地址 | 寄存器值 | 和校验 |
| 2字节 | 1字节 | 1字节 | 2字节 | 1字节 |

例：修改地址为0x01的模块寄存器值，寄存器地址为8，修改值为100

主机发送指令：0xAA 0xBB 0x01 0x88 0x00 0x64 ***0x52***

从机返回应答：0xAA 0xBB 0x01 0x08 0x00 0x64 ***0xD2***

（3）通用模块地址

AABB通讯协议支持模块通用地址，无论模块的当前地址为何值，使用0xFF作为地址对模块发送读写指令，均可得到模块正确应答。

例：使用通用地址，读取任一模块的寄存器8

主机发送指令：0xAA 0xBB 0xFF 0x08 ***0x6C***

从机返回应答：0xAA 0xBB 0x01 0x08 0x00 0xC8 ***0x36***

***注：当总线上连接有多个模块时（通常为RS485总线），使用通用地址时总线上所有模块均会响应指令，导致指令无法正常使用。***

***注：严禁在连接有多个RM模块的总线中使用通用地址修改模块地址。***

（4）特殊模块地址

模块地址保存于寄存器ADDR.[7:0]，取值范围为1~255，这些地址中，255在AABB协议中作为通用地址使用，地址128（0x80）用于特殊用途，故此对模块地址进行修改时可使用的地址有：1~127、129~254，共253个。

模块地址寄存器（0x00）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **位** | **符号** | **值** | **描述** | **默认值** |
| **bit15:8** |  |  | 暂未定义 | 0 |
| **bit7:0** |  | 1~254 | 模块地址 | 1 |

（三）校验码算法

无论是向模块发送指令还是接收模块返回的答应数据，均应严格进行数据校验。极少情况下，模块返回的应答数据会存在错误，通过数据帧的校验码验证可避免读取到错误的数据。

（1）CRC16-MODBUS算法

unsigned int crc16(unsigned char \*dat, unsigned int len)

{

unsigned int crc=0xffff;

unsigned char i;

while(len!=0)

{

crc^=\*dat;

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

{

if((crc&0x0001)==0)

crc=crc>>1;

else

{

crc=crc>>1;

crc^=0xa001;

}

}

len-=1;

dat++;

}

return crc;

}

（2）和校验算法

unsigned char AddCheck(unsigned char \*dat,unsigned char count)

{

unsigned char i,Add=0;

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

Add+=dat[i];

return Add;

}

（四）$字符串指令集

$字符串指令集使用更加易于手工操作的字符串做为控制指令，方便对模块进行快捷的功能测试与演示。字符串指令集见下表。

|  |  |  |
| --- | --- | --- |
| **指令符** | **功能描述** | **参数说明** |
| **$GETP=A,R** | 读取寄存器值 | A：模块地址，R：寄存器地址 |
| **$SETP=A,R,B** | 修改寄存器值 | A：模块地址，R：寄存器地址，B：寄存器值 |
| **$SAVE** | 保存参数 |  |
| **$RSTP** | 恢复出厂参数 | 详见“3.3节 恢复出厂参数” |
| **$STFC** | 设置出厂参数 |
| **$STDF** | 恢复默认参数 |
| **注：字符串指令必须以回车结尾**  **注：B值可以“H”结尾，表示数值为16进制。**  **例如：$SETP=1,7,1F1FH\r\n等同于$SETP=1,7,7967\r\n** | | |

3.4.2通讯协议（IIC）

IIC通讯协议本身即是基于设备地址和寄存器的物理层通讯协议，使用IIC接口对模块寄存器的访问，请遵循前述硬件接口时序及协议说明即可。

IIC访问时使用与UART相同的寄存器地址，地址定义请详见“3.5寄存器概述”。需要注意的是寄存器的类型为“字”，占用2字节，在使用IIC读写寄存器时，读取和写入的字节数必须为偶数。

**IIC中的STOP信号是不可省略的，若省略则当次访问不会生效。**

向寄存器写操作时，在每次发送STOP后应延时10ms后再发起下次访问。

通过IIC修改单个寄存器后会立即保存，而连续寄存器的修改不会自动保存（仅当时生效），下次上电自动恢复，若要保存所有寄存器的当前值，需要向FUN寄存器写入功能码0x0C（详见“7.1功能码汇总”）。

IIC为同步串行接口，**在IIC总线中作为从设备的RMXXX无法主动输出任何数据**，本手册中所述的与自动上传有关的特性均是指UART接口。

3.4.3主动上传测量数据（UART）

默认情况下RMXXX模块总是以从机身份与主机完成数据交互，在这种主从结构中，RMXXX从不主动上传数据，可通过修改自动上传寄存器（ATSD\_SEL）来实现模块主动输出测量数据功能，ATSD\_SEL寄存器的每1位对应了一种数据类型，见下表。

自动上传寄存器ATSD\_SEL（0x07）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **位** | **符号** | **值** | **描述** | **默认值** |
| **bit15** |  | 0/1 |  | 0 |
| **bit14** |  | 0/1 |  | 0 |
| **bit13** |  | 0/1 |  | 0 |
| **bit12** |  | 0/1 | 通道2的电流值，数据前缀“I=” | 0 |
| **bit11** |  | 0/1 | 通道2的L5电阻值，数据前缀“L5=” | 0 |
| **bit10** |  | 0/1 | 通道2的R1电阻值，数据前缀“R1=” | 0 |
| **bit9** |  | 0/1 | 通道2的R2电阻值，数据前缀“R2=” | 0 |
| **bit8** |  | 0/1 | 通道2的L1电阻值，数据前缀“L1=” | 0 |
| **bit7** |  | 0/1 |  | 0 |
| **bit6** |  | 0/1 |  | 0 |
| **bit5** |  | 0/1 |  | 0 |
| **bit4** |  | 0/1 | 通道1的电流值，数据前缀“I=” | 0 |
| **bit3** |  | 0/1 | 通道1的L5电阻值，数据前缀“L5=” | 0 |
| **bit2** |  | 0/1 | 通道1的R1电阻值，数据前缀“R1=” | 0 |
| **bit1** |  | 0/1 | 通道1的R2电阻值，数据前缀“R2=” | 0 |
| **bit0** |  | 0/1 | 通道1的L1电阻值，数据前缀“L1=” | 0 |
| *注：主动上传数据破坏了主从通讯机制，为避免主动上传数据与响应主机指令的应答数据同时发送导致的冲突，当收到主机指令后会暂停自动发送5秒。* | | | | |

若设置了自动上传，则在每个通道测量完成后集中输出此通道的当前测量结果，数据字符串举例如下：

CH1 I=1.0005 L5=0.2597 R1=50.001 R2=49.999 L1=0.2539 Z=1.00004

CH1：当前输出的是通道1的测量结果

I=1.0005：当前驱动电流为1.0005mA

L5=0.2597：传感器线缆L5的电阻为0.2597Ω

R1=50.001：传感器内部电阻R1的电阻值为50.001Ω

R2=49.999：传感器内部电阻R2的电阻值为49.999Ω

L1=0.2539：传感器线缆L1的电阻为0.2539Ω

Z=1.00004：Z=R1/R2=1.00004

**自动上传与主从协议冲突解决**

当设置了自动上传功能，又需要向模块发送主从结构的MODBUS或AABB协议指令时，模块采用主从协议优先的原则，在自动发送过程中，若收到了主从指令，则自动上传立即暂停5秒钟。这样导致的一个现象是，当向模块发送了修改自动发送寄存器后（主从协议指令），并不会立即收到模块的自动上传数据，而是在约5秒后才会收到。

3.5寄存器概述（汇总）

RXXX模块的工作过程完全依赖于寄存器（参数）值，寄存器是16位二进制表示的整数，分为读写寄存器与只读寄存器，读写寄存器又分为掉电保存和上电复位两种类型，可以通过UART或IIC数字接口访问这些寄存器来修改模块各种参数，实现控制模块、与模块交互的目的。

以下汇总表列出了所有寄存器及概要功能描述，更加详细的寄存器使用会在后续功能介绍时一一具体说明。

不同的固件版本对寄存器的定义可能有微小不同，操作寄存器前应确认固件版本是否与本手册对应。

寄存器总表（固件版本3.50）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 地址 | 符号 | 读写 | 数据类型 | 名称 | 默认值 | 单位 |
| 0x00(0) | ADDR | 读/写/存 |  | 模块地址（UART） | 0x0001 |  |
| 0x01(1) | BAUD① | 读/写/存 |  | 通讯速率（UART） | 0x0060 | 100bps |
| 0x02(2) | AUX | 读/写/存 |  | 辅助功能寄存器 | 0x0018 |  |
| 0x03(3) | SYS\_FUN | 读/写/复 |  | 系统功能寄存器 | 0x0000 |  |
| 0x07(7) | ATSD\_SEL | 读/写/复 |  | 自动上传 | 0x0000 |  |
| 0x09(9) | ITEMS\_EN | 读/写/存 |  | 测量项 | 0x1F1F |  |
| 0x0A(10) | INTG\_TIME | 读/写/存 |  | 积分时长 | 50 | mS |
| 0x0B(11) | NSAMPLES | 读/写/存 |  | 采样数量 | 1 | 个 |
| 0x0C(12) | FILT | 读/写/存 |  | 历史滤波 | 0 | 个 |
| 0x15(21) | MEAS\_PN | 读/写/存 |  | 测量模式（正反测） | 0x0101 |  |
| 0x1F(31) | CRC | 只读 |  | 参数CRC校验 |  |  |
| 0x20(32) | SYS\_STA | 读/写/复 |  | 系统状态寄存器 | 0 |  |
| 0x23(35) | V\_POW | 只读 |  | 供电电压 |  | 0.1mV |
| 0x24(36) | TEMP | 只读取 |  | 环境温度 |  | 0.1℃ |
| 0x29(41) | S1\_I | 只读 | 单精浮点② |  |  | mA |
| 0x2B(43) | S1\_L5 | 只读 | 单精浮点 |  |  | Ω |
| 0x2D(45) | S1\_R1 | 只读 | 单精浮点 |  |  | Ω |
| 0x2F(47) | S1\_R2 | 只读 | 单精浮点 |  |  | Ω |
| 0x31(49) | S1\_L1 | 只读 | 单精浮点 |  |  | Ω |
| 0x33(51) | S1\_R1+R2 | 只读 | 单精浮点 |  |  | Ω |
| 0x35(53) | S1\_Z | 只读 | 单精浮点 |  |  |  |
| 0x3D(61) | S2\_I | 只读 | 单精浮点 |  |  | mA |
| 0x3F(63) | S2\_L5 | 只读 | 单精浮点 |  |  | Ω |
| 0x41(65) | S2\_R1 | 只读 | 单精浮点 |  |  | Ω |
| 0x43(67) | S2\_R2 | 只读 | 单精浮点 |  |  | Ω |
| 0x45(69) | S2\_L1 | 只读 | 单精浮点 |  |  | Ω |
| 0x47(71) | S2\_R1+R2 | 只读 | 单精浮点 |  |  | Ω |
| 0x49(73) | S2\_Z | 只读 | 单精浮点 |  |  |  |
| 0x51(81) | HW\_VER | 只读 |  | 硬件版本 |  |  |
| 0x52(82) | SF\_VER | 只读 |  | 固件版本 |  |  |
| 0x53(83) | ID | 只读 |  | 模块ID（8字节） |  |  |
| 读：可读取；写：可修改；存：掉电不遗失；复：上电后复位为默认值  不要修改预留寄存器（位）的上电初始值。  ①：这些寄存器（参数）在下次启动时生效。  ②：单精度浮点数的4字节按照IEEE754的逆序排列，例如：  0XA7 0XF9 0x21 0x41，颠倒顺序后为0x41 0x21 0xF9 0xA7，按照IEEE754规则转换为浮点数为10.12345 | | | | | | |

3.6模块地址操作（UART）

将地址为0x01的模块地址修改为0x02

3.6.1修改已知设备地址的地址

模块地址0x01

MODBUS指令：0x01 0x06 0x00 0x00 0x00 0x02 ***0x08 0x0B***

AABB指令：0xAA 0xBB 0x01 0x80 0x00 0x02 ***0xE8***

3.6.2读取未知设备地址的地址

未知地址的模块仅可使用AABB协议的通用地址0xFF

AABB指令：0xAA 0xBB 0xFF 0x00 ***0x64***

3.6.3修改未知设备地址的地址

AABB指令：0xAA 0xBB 0xFF 0x80 0x00 0x02 ***0xE6***

***修改模块地址后，新的地址立即生效，收到的应答信息中的模块地址码部分是新的地址值。后续的指令应使用新的地址对模块进行操作。***

3.7通讯速率和软件握手（UART）

3.7.1通讯速率

RMXXX的UART接口支持9600~460800bps通讯速率，通过设置寄存器BAUD.[13:0]来改变通讯速率，BAUD.[13:0]的单位为“每秒百位”或“百bps”。寄存器取值与对应的通讯速率如下表。

UART通讯速率寄存器BAUD（0x01）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **位** | **符号** | **值** |  |  | **描述** | **默认值** |
| **bit15** |  |  | 是否启用软件握手协议功能 | | | 0 |
| **bit14** |  |  | 是否忽略“测量正忙”标志而立即响应指令 | | | 0 |
| **bit13:0** |  | 值 | 速率 | 值 | 速率 |  |
|  | 96 | 9600bps（默认） | 768 | 76800bps（NR） | 96 |
|  | 128 | 12800bps | 1152 | 115200bps |
|  | 144 | 14400bps | 1280 | 128000bps |
|  | 192 | 19200bps | 1536 | 153600bps |
|  | 288 | 28800bps（NR） | 2304 | 230400bps |
|  | 384 | 38400bps（NR） | 2560 | 256000bps |
|  | 560 | 56000bps（NR） | 4608 | 460800bps |
|  | 576 | 57600bps（NR） |  |  |
|  |  |  |  |  |
| **通讯速率在下次启动时生效，非法的通讯速率值会引起参数恢复出厂值。NR为不推荐** | | | | | | |

在条件允许情况下，尽量使用较高的通讯速率，缩短数据传输时间。

***注：关于UART通讯的其它参数（校验位、数据位、停止位），详见“辅助功能寄存器说明”。***

3.7.2软件握手

模块开始一次测量时，从UART接口主动发送XOFF信号（0x13），表示模块开始忙于测量数据，当测量完成时主动发送XON信号（0x11），表示模块本次测量完成，正处于空闲状态。

在开启模块的软件握手功能后，若需要向模块发送指令，建议UART的通讯流程为：首先等待模块返回XON信号（0x11），当收到XON信号或等待超时后立即向模块发送指令。

注：在一主多从的总线应用中，严禁开启模块的软件握手功能。

3.7.3修改已知设备地址的通讯速率

模块地址0x01，将波特率修改为115200bps

MODBUS指令：0x01 0x06 0x00 0x01 0x04 0x80 ***0xDB 0x6A***

AABB指令：0xAA 0xBB 0x01 0x81 0x04 0x80 ***0x6B***

3.7.4读取未知设备地址的通讯速率

未知地址的模块仅可使用AABB协议的通用地址0xFF

AABB指令：0xAA 0xBB 0xFF 0x01 ***0x65***

模块返回：0xAA 0xBB 0x01 0x01 0x00 0x60 ***0xC7***

3.7.5修改未知设备地址的通讯速率

AABB指令：0xAA 0xBB 0xFF 0x81 0x04 ***0x80 0x69***

***修改UART通讯速率后，新的值会在下次启动时生效，未重启之前模块会继续使用修改前的通讯速率。重启后应使用新的通讯速率与模块进行通讯。***

***除上述模块地址、通讯速率寄存器外，其它所有寄存器的访问方法均相同，不再一一举例。***

3.8参数保存

使用通讯协议发送指令对模块的工作参数进行修改后，若需要永久性存储，需要向模块发送“参数保存”指令。

发送指令“$SAVE”或者使用其它通讯协议向寄存器3（功能寄存器）写入12（0x0C）。

MODBUS协议指令：01 06 00 03 00 0C 79 CF

AABB协议指令：AA BB 01 83 00 0C F5

3.9系统状态

暂无说明

3.10测量模式

暂无说明

3.11测量流程

暂无内容

3.12传感器接入检测

暂无内容

3.13采样积分时长

积分时长直接影响采样速率和采样精度（详见“名词解释”）。

寄存器INTG\_MS用于设置单个采样的耗时长短，数值单位为mS。

积分时长寄存器INTG\_MS（10）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **位** | **符号** | **值** | **描述** | **默认值** |
| **bit15:8** |  | 0 | 预留，不可写入非0数据 | 0 |
| **bit7:0** |  | 2 | 单个采样的耗时为2mS | 50 |
|  | 4 | 单个采样的耗时为4mS |
|  | 10 | 单个采样的耗时为10mS |
|  | 20 | 单个采样的耗时为20mS |
|  | 50 | 单个采样的耗时为50mS（默认） |
|  | 100 | 单个采样的耗时为100mS |
|  | 200 | 单个采样的耗时为200mS |
|  | | | | |

对采样速率没有过高要求时，建议积分时长为50mS，此时长下对工频干扰抑制效果最为明显。

当对某一信号进行测量时，首次测量的耗时会是积分时长的两倍，详见“采样次数”。

3.14采样内容

采样内容用于设置对哪些子项进行数据采集。

对于五线制差阻传感器的数据采样，可关闭对L1和L5的采样，从而提升传感器测量速度。

测量内容寄存器ITEMS\_EN（9）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **位** | **符号** | **值** | **描述** | **默认值** |
| **bit15:13** |  | 0 | 预留，不可写入非0数据 | 0 |
| **bit12** |  | 0/1 | 是否采样通道2传感器的驱动电流 | 1 |
| **bit11** |  | 0/1 | 是否采样通道2传感器的L5线阻 | 1 |
| **bit10** |  | 0/1 | 是否采样通道2传感器的R1电阻 | 1 |
| **bit9** |  | 0/1 | 是否采样通道2传感器的R2电阻 | 1 |
| **bit8** |  | 0/1 | 是否采样通道2传感器的L1线阻 | 1 |
| **bit7:5** |  | 0 | 预留，不可写入非0数据 | 0 |
| **bit4** |  | 0/1 | 是否采样通道1传感器的驱动电流 | 1 |
| **bit3** |  | 0/1 | 是否采样通道1传感器的L5线阻 | 1 |
| **bit2** |  | 0/1 | 是否采样通道1传感器的R1电阻 | 1 |
| **bit1** |  | 0/1 | 是否采样通道1传感器的R2电阻 | 1 |
| **bit0** |  | 0/1 | 是否采样通道1传感器的L1线阻 | 1 |
|  | | | | |

3.15采样次数

数值采样时，连续采集多少次后计算一次结果。

多次采样有利于提升测量精度、滤除突变噪声信号。

采样次数寄存器NSAMPLES（11）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **位** | **符号** | **值** | **描述** | **默认值** |
| **bit15:4** |  | 0 | 预留，不可写入非0数据 | 0 |
| **bit3:0** |  | 1~10 | 采样次数，必须为1~10之间的值 | 1 |
|  | | | | |

采样次数与采样速率的计算

对某一信号进行测量时，首次采样的耗时会是积分时长的两倍，之后的采样耗时为积分时长。

例如：设置采样次数为5时，对信号的测量耗时为：

积分时长50mS\*2+积分时长50mS\*4=100+200=300mS。

3.16历史数据滤波

使用最近多少次的历史结果进行滤波计算本次测量结果。

历史滤波可以抑制信号噪声、有利于提升测量结果稳定性，但会造成实时数据滞后。

历史滤波寄存器FILT（12）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **位** | **符号** | **值** | **描述** | **默认值** |
| **bit15:4** |  | 0 | 预留，不可写入非0数据 | 0 |
| **bit3:0** |  | 0 | 不进行历史数据滤波 | 0 |
|  | 0~10 | 使用1~10个历史数据进行滤波处理 |
|  | | | | |

历史滤波的数据滞后性，若设置历史滤波参数为5时，当信号产生变化后，因有前4次测量结果的作用，在信号改变后第5次测量完成后，才会是真实的信号数值。

3.17正测和反测

RM模块可以进行正向驱动电流加载和反向电流加载，实现对差阻传感器的正测和反测。

通过寄存器MEAS\_PN来设置两个通道的测量方向。

测量方式寄存器MEAS\_PN（21）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **位** | **符号** | **值** | **描述** | **默认值** |
| **bit15:10** |  | 0 | 预留，不可写入非0数据 | 0 |
| **bit9:8** |  | 0 | 通道2，不测 | 1 |
|  | 1 | 通道2，仅正测 |
|  | 2 | 通道2，仅反测 |
|  | 3 | 通道2，正反测（取平均） |
| **bit7:2** |  | 0 | 预留，不可写入非0数据 | 0 |
| **bit1:0** |  | 0 | 通道1，不测 | 1 |
|  | 1 | 通道1，仅正测 |
|  | 2 | 通道1，仅反测 |
|  | 3 | 通道1，正反测（取平均） |
|  | | | | |

3.18模拟信号输出

RM502有两个模块电压输出引脚，以电压形式输出通道1和通道2的Z值。

使用寄存器DAO1和DAO2来设置输出0~2048mV所对应的Z值范围。

电压输出配置寄存器DAOx（13/14）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 位 | 符号 | 值 | 描述 | 默认值 |
| bit15 |  | 0 |  | 0 |
| bit14:8 |  | 0~127 | Z值上限，输出电压2048mV所对应的Z值 | 110 |
| bit7 |  | 0 |  | 0 |
| bit6:0 |  | 0~127 | Z值下限，输出电压0mV所对应的Z值 | 90 |
| 设置值/100=实际的Z值，默认110、90对应Z值1.1和0.9 | | | | |

某Z值对应的输出电压，使用下式计算

例如，在默认参数条件下，若Z值为0.95时，DAOx引脚输出的电压应为：

当测量到DAOx引脚的输出后，反算其对应的Z值，公式为

例如，当测量到DAOx引脚输出电压为512mV时，则对应的Z值为：

3.19辅助功能寄存器

辅助功能寄存器AUX（0x02）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 位 | 符号 | 值 | 描述 | 默认值 |
| bit15 |  | UART数据位数定义 | | 0 |
| 0 | 8位 |  |
| 1 | 9位 |  |
| bit14:13 |  | UART停止位数定义 | | 0 |
| 0 | 1位 |  |
| 1 | 1.5位 |  |
| 2 | 2位 |  |
| bit12:11 |  | UART校验方法定义 | | 0 |
| 0 | 无，不校验 |  |
| 1 | 奇校验 |  |
| 2 | 偶校验 |  |
| bit10:0 |  | 0 | 预留，暂未定义功能 |  |

3.20数据更新速率

数据更新速率取决于：单个采样时长（积分时长）、每次测量时的采样次数、采样的内容多少，还要考虑通讯耗时。

以下以默认参数来计算单个通道的耗时（更新速率）。

模块的默认参数为：积分时长50mS，每次采样1个，采样内容为4个内容（L1、L5、R1、R2）。

对某一内容进行采样时，首次需要两倍时长，默认采样数量是1个，所以采样时长为100mS，对4个内容进行采样耗时为400mS，对两个通道进行测量耗时为800mS（大约）。

如果设置的自动输出，则还应考虑自动输出测量结果的耗时，在9600bps通讯速率下，输出两通道所有元素的测量结果，大约需要160mS，则相邻输出的间隔为800+160=960mS（约为1Hz左右）。

河北稳控科技股份有限公司

2024年5月