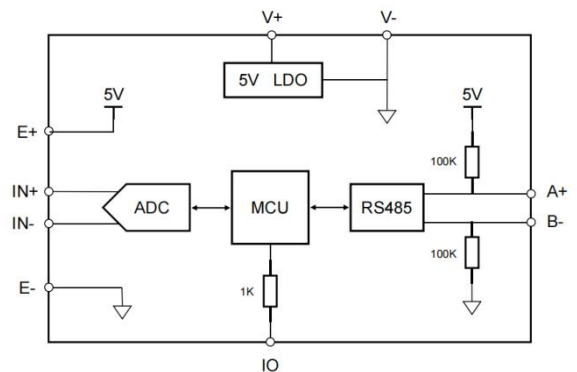


DM010

特性:

- 有效分辨率: 100 万码 (typ)
- 无噪声分辨率: 20 万码 (typ)
- 采样速度: 10~160 (次/秒)
- 零点温漂: 3ppm/°C (typ)
- 增益温漂: 4ppm/°C (typ)
- 通讯接口: Modbus-RTU (RS485)
- 传感器激励电压: 5V, Max 20mA
- 供电电源: 6~12VDC
- 自身功耗: 4mA (typ)
- 尺寸: 18mm×12mm×5mm
- 低成本

功能框图:



应用场合:

- 数字式称重传感器
- 重量变送器
- 模转数接线盒
- 称重仪表

概述:

DM010 数字模块主要应用于称重传感器（特别是小量程称重传感器）的数字化，通过它可以把模拟输出的称重传感器（简称模拟传感器）转变为数字信号输出的称重传感器（简称数字传感器）。

DM010 具备低成本和小尺寸的特点，因此非常适合于小量程模拟传感器的数字化。同时由于它出色的模拟性能、强抗干扰能力和高可靠性使得其组成的数字传感器具备优异的产品性能。

DM010 采用标准的 Modbus-RTU (RS485) 通讯接口可以方便的和 PC、PLC、HMI、安卓屏、嵌入式系统等连接，组成低成本高性能的称重系统，可省去昂贵的称重仪表。

DM010 集成了 AD 采样、智能滤波算法、称重数据处理（包括去皮、置零、零位跟踪等），用户无需太多的称重专业知识就能实现符合要求的称重系统。

1 技术规格	4
2 绝对最大额定值	5
3 引脚配置和功能描述	5
4 外形尺寸	6
5 传感器接口	6
6 数字接口	6
7 安装要求	6
8 Modbus-RTU 协议	7
8.1 保持寄存器简介	7
8.2 000 区寄存器（称重信息）	7
8.3 100 区寄存器（通讯参数）	9
8.4 200 区寄存器（用户参数）	10
8.5 300 区寄存器（标定功能）	11
8.6 400 区寄存器（标定参数）	12
8.7 500 区寄存器（高速采样）	12
9 常用操作	12
9.1 读取称重数据	12
9.2 去皮操作	13
9.3 设皮操作	13
9.4 置零操作	13
9.5 重量锁定	14
9.6 获取有效的采样	14
9.7 高速采样	15
9.8 IO 功能	15
10 通讯相关设置	15
10.1 主机发送间隔	15
10.2 通讯参数	16
10.3 通讯参数恢复	16
10.4 通讯发送延时	17
10.5 响应时间	17
10.6 低字在前访问	17
11 称量相关说明	17
11.1 标定操作	17
11.2 称重参数	18

DM010

11.3 置零范围 18

11.4 零点跟踪 19

11.5 蠕变补偿模式 19

11.6 用户修正系数 19

11.7 用户重力加速度修正 20

11.8 目标重量修正 20

12 参数相关设置 20

12.1 恢复默认用户参数 20

12.2 参数可靠性 21

12.3 备份和恢复标定参数 23

13 工具软件简介 24

14 定制软件和远程升级 24

版本信息	1.00
修订内容	首版
发布日期	2024 年 10 月 16 日

1 技术规格

所有参数测试在：环境温度 20℃、供电电源 6V、传感器阻抗 350 Ω、传感器灵敏度 2mV/V 的条件下进行，特别申明除外。

表 1 技术规格

参数	DM010	单位	测试条件/注释
模拟性能			
有效分辨率	100 万（typ）	码	
无噪声分辨率	20 万（typ）	码	
采样速度	10、20、40、80、160	次/秒	
零点误差	0.01（typ）	% F.S	
零点温漂	3（typ）	ppm/℃	
增益误差	0.5（typ）	% F.S	
增益温漂	3（typ）	ppm/℃	
线性度	0.001（typ）	% F.S	
信号输入范围	-19~+19	mV	
通讯接口			
通讯协议	Modbus-RTU（从机）		
硬件接口	RS485（半双工）		
波特率	1200~115200	bps	8 种
校验方式	无校验、奇校验、偶校验		
最大传输距离	1200	米	波特率≤9600
总线偏置电阻	100	k Ω	
总线最大个数	128	个	通讯距离≤2 米
ESD 防护	15	kV	HBM
功能参数			
非线性修正段数	5	段	折线拟合
参数写保护模式	软件写保护		
标定参数备份数	6	份	其中 2 份需手动备份
其他参数备份数	4	份	
其他			
工作温度	-10~40	℃	
工作湿度	≤65	%RH	无冷凝
供电电压	6~12	VDC	
自身功耗	4（typ）	mA	
尺寸	18×12×5	mm	±0.2mm

DM010

2 绝对最大额定值

环境温度 20℃，别申明除外。

参数	范围
V+至 V-	-30V ~ +30V
A+/B-至 V-	-7V ~ 12V
其他引脚至 V-	-0.3V ~ 5.3V
E+至 E-输出电流	≤100mA
最大功率耗散	200mW
工作温度范围	-20℃ ~ 60℃
存储温度范围	-40℃ ~ 85℃
相对湿度	≤80%RH（无冷凝）

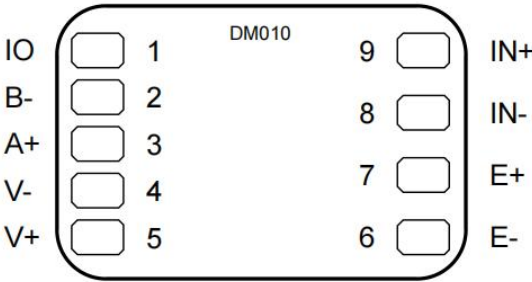
ESD 警告



ESD（静电放电）敏感器件。
带电器件和电路板可能会在没有察觉的情况下放电。尽管本产品具有专利或专有保护电路,但在遇到高能量ESD时,器件可能会损坏。因此，应当采取适当的 ESD 防范措施，以避免器件性能下降或功能丧失。

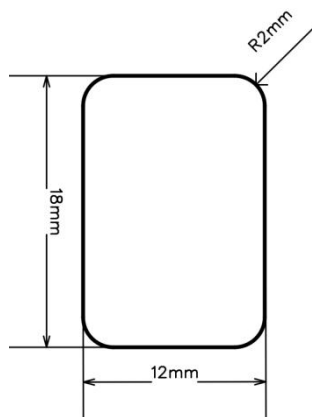
注意，超出以上所列的绝对最大额定值可能导致模块永久性损坏。这只是额定应力值，不意味着器件可以在这个范围内正常工作，长期在绝对最大额定值条件下工作会影响模块的可靠性。

3 引脚配置和功能描述



序号	引脚名称	类型	说明
1	IO	数字输入/输出	开关量接口，输入/输出可配置
2	B-	数字输入/输出	RS485 通讯接口反向端
3	A+	数字输入/输出	RS485 通讯接口同向端
4	V-	地	电源输入负极
5	V+	电源	电源输入正极（6~12VDC）
6	E-	模拟输出	传感器激励电压输出负极
7	E+	模拟输出	传感器激励电压输出正极
8	IN-	模拟输入	传感器信号输入负极
9	IN+	模拟输入	传感器信号输入正极

4 外形尺寸



厚度: 5mm

5 传感器接口

DM010 的传感器接口连接电阻应变式称重传感器，要求传感器的输入阻抗 $\geq 250\Omega$ ，输出灵敏度 1~3mV/V。E+、E-接传感器的激励电压输入，IN+和 IN-接传感器的信号输出。

因为 DM010 具备数字化标定功能，模拟传感器在生产时可以省略零点补偿、灵敏度补偿、非线性补偿、输入输出阻抗调整等步骤，简化生产工艺，降低生产成本。

但由于数字化补偿的范围比较宽，容易掩盖质量瑕疵。（比如一个灵敏度 2mV/V 的传感器实际灵敏度变为 1mV/V，这个时候通过数字化标定，也可使用。但是掩盖了实际的故障，降低了产品的可靠性。）这个也是在使用时需要注意避免的。

6 数字接口

DM010 的数字接口包括电源、RS485 和 IO 接口。一般情况下 IO 接口不用。引出 4 根线：V+、V-、A+、B-。

建议采用 4 芯屏蔽线，屏蔽层必须和传感器弹性体连接，使用时屏蔽层需要连接大地，否则将降低数字传感器的抗干扰能力，使之更容易损坏。

推荐导线颜色：

V+	红
V-	黑
A+	绿
B-	白/蓝

7 安装要求

安装模块时，要求全程使用手套，禁止用手直接和模块接触。

注意做好防静电措施。模块对外接口具备防静电能力，但是内部电路并不具备防静电能力，对内部电路的静电放电可能损坏模块，或者降低模块的可靠性。

安装环境注意保持干燥，建议湿度控制在 $\leq 65\%RH$ 。

安装时需固定好模块然后再焊线，注意保持模块和周围金属部件之间的距离，建议 2mm 以上。

焊接时，注意保证焊点饱满，建议焊接完成后，清理助焊剂残留。

模块建议采用整体封胶处理。

8 Modbus-RTU 协议

DM010 支持标准的 Modbus-RTU 协议。全部功能都是映射到保持寄存器。保持寄存器以字（16bit）为单位，可读写。根据功能不同划分为六个不同的区，每个区包含 50 个字，用最小的地址来作为区的名字，000 区表示地址 00~49 的区、100 区表示 100~149 的区，以此类推。

表 2 Modbus 功能分区

000 区	称重信息	读取重量、状态，执行置零、去皮
100 区	通讯参数	读写通讯参数
200 区	用户参数	读写用户标定参数，执行用户标定
300 区	标定功能	执行厂商标定
400 区	标定参数	读写厂商标定参数
500 区	高速采样	高速采样缓冲区

本手册是基于 Modbus 寄存器读写来说明的。关于 Modbus 协议细节可以参考《DM010 Modbus 说明书》、《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）。

8.1 保持寄存器简介

保持寄存器是以字为单位（16bit），对于 32bit 的变量则会占据两个字的空间，默认是高字在前。比如毛重值它占用了保持寄存器地址 0 和地址 1 这两个寄存器，那么地址 0 里面保存的就是毛重的高 16bit，地址 1 里面保存的就是低 16bit。

模块的寄存器地址是以 0 起始的，某些 PLC 的寄存器地址是从 1 起始的，而且会根据不同的寄存器类型加上前缀（比如保持寄存器就会加 4 起始的前缀）。毛重我们的地址是 0 和 1，某些 PLC 中就是 40001 和 40002。

读保持寄存器的功能码是 3，写保持寄存器的功能码是 16。

8.2 000 区寄存器（称重信息）

地址	字数	功能	读写	解释
0	2	毛重值	只读	毛、皮、净重值为“重量数值”。“重量数值”为对应重量不包括小数点的数值（小数点位数需从保持寄存器地址 6 中获取）。比毛重值加上小数点后就得到毛重。
2	2	皮重值	可读写	
4	2	净重值	只读	
6	1	小数点位数	只读	
7	1	状态标志 ^a	只读	状态标志共 16 位，具体含义下表
8	2	毛重内码	只读	内码提供了高分辨率的重量信息，主要用于标定调试和计量检定。
10	2	皮重内码	只读	

DM010

地址	字数	功能	读写	解释
12	2	净重内码	只读	内码 = 重量 × 20 ÷ 分度值
14	2	AD 码	只读	内部 ADC 输出的数值，偏移二进制码
16	2	锁定重量	只读	锁定重量配合锁定功能使用
18	1	传感器输出电压	只读	为传感器输出的信号电压值，单位为 uV。
19	1	模块类别	只读	本模块固定为 2
20	1	模块型号	只读	本模块固定为 0
21	1	硬件版本	只读	模块的硬件版本
22	1	软件版本	只读	模块的软件版本
23	1	改制版本	只读	模块的软件改制版本，标准版为 0
24	1	备用	只读	
25	1	软件校验和	只读	软件校验和主要用于判断程序的完整性
26	1	用户参数校验和	只读	通讯参数和用户参数的校验和
27	1	标定参数校验和	只读	标定参数的校验和
28	2	用户参数设置次数	只读	反映了对应的参数被写入的次数。只访问 000 区是不会增加这个次数的。
30	2	标定参数设置次数	只读	
32	1	参数健康状态	只读	参数各个副本的保存情况
33	1	写操作允许状态	只读	100 区~400 区写操作的允许状态
34	4	模块 ID 号	只读	模块的唯一标识
38	2	备用	只读	
40	1	锁定命令	只可单独写	决定锁定重量的内容，详见重量锁定小节
41	1	去皮命令	只可单独写	执行去皮操作，详见去皮操作小节
42	1	置零命令	只可单独写	执行置零操作，详见置零操作小节
43	1	固定通讯参数模式	只可单独写	详见通讯参数恢复小节
44	1	IO 模式	只可单独写	详见 IO 功能小节
45	5	备用	不要操作	

a 状态标志（保持寄存器地址 7）

位	名称	说明
0	零位	当毛重小于等于 0.25 个分度值时，零位标志位为 1
1	稳定	重量变化小于判稳范围 1.6 秒后，稳定标志位为 1
2	去皮	当皮重不为 0 时，皮重标志位为 1
3	超载	当毛重大于满量程加 9 个分度值时，超载标志位为 1 当超载时，主机必须停止显示重量，并做出相应的报警提醒用户
4	开机	模块上电会执行开机置零（如果设置开机置零有效的话） 此时开机标志位为 1，重量信息无效（固定为 0）
5	模拟部分异常	传感器或模块 ADC 模拟部分损坏时，此标志位为 1，重量信息无效
6	ADC 芯片损坏	模块 ADC 芯片损坏时，此标志位为 1，重量信息无效
7	备用	
8	锁定	锁定重量处于锁定状态时，锁定标志位为 1
9	固定通讯参数	详见通讯参数恢复小节
10	开关量输入状态	IO 模式设置为弱上拉输入时，可读取 IO 的输入引脚的状态
11	新 AD 采样有效	详见获取有效的采样小节

8.3 100 区寄存器（通讯参数）

地址	字数	功能	默认值	解释
100	1	通讯模式	0	0: Modbus-RTU 1: 定制通讯模式（无效）
101	1	通讯地址	1	1~246
102	1	波特率	5	0:1200 1:2400 2:4800 3:9600 4:19200 5:38400 6:57600 7:115200
103	1	校验方式	2	0:无校验 1:奇校验 2:偶校验 3:“1”校验
104	1	通讯发送延时	10	0~120 （单位为 ms）

8.4 200 区寄存器（用户参数）

地址	字数	功能	默认值	解释
200	1	ADC 转换速度	0	0: 10 次/秒 1: 20 次/秒 2: 40 次/秒 3: 80 次/秒 4: 160 次/秒
201	1	滤波模式	0	0: 滑动滤波
202	1	滤波强度	16	1~32 滑动滤波的平均次数
203	1	判稳门限	1	1~99
204	1	开机置零范围	20	0~100（FULL%）
205	1	手动置零范围	4	0~100（FULL%）
206	1	备用	0	
207	1	零点跟踪范围	1	0~9（单位为分度值）
208	1	零点跟踪速度	0	0~3 数值越大修正速度就越快
209	1	补偿模式	1	0: 无补偿 1: 蠕变补偿 2: 闪变点抑制 3: 蠕变补偿 + 闪变点抑制
210	1	备用	0	
211	1	用户分度值	标定值	0~23（见 8.4.1）
212	2	用户满量程值	标定值	100~12000000（见 8.4.2）
214	2	用户修正系数	1000000	250000~4000000 固定为 6 位小数
216	4	备用	0	
220	2	用户重力加速度值		9300000~10300000 固定为 6 位小数
222	2	目标重量修正		800~用户满量程值
224	2	恢复默认用户参数		写入 4444444，初始化用户参数为默认值

8.4.1 用户分度值

分度值决定了模块的分辨率，分为标定分度值和用户分度值，标定分度值是在标定时候选择的分度值，用户分度值则是实际使用的分度值，用户分度值设置范围受标定分度值限制。

标定分度值可设置 0~23。

用户分度值允许设置范围：标定分度值 ± 3 。

分度值参数 0~23，分别表示不同的分度值，见下表。

表 3 参数-分度值对应表

参数	0	1	2	3	4	5	6	7
分度值	0.0001	0.0002	0.0005	0.001	0.002	0.005	0.01	0.02
参数	8	9	10	11	12	13	14	15
分度值	0.05	0.1	0.2	0.5	1	2	5	10
参数	16	17	18	19	20	21	22	23
分度值	20	50	100	200	500	1000	2000	5000

8.4.1 用户满量程

满量程决定模块的最大称量，分为标定满量程和用户满量程，标定满量程是在标定的时候设定的满量程，用户满量程是实际使用的满量程。用户满量程受标定满量程的限制，设置范围如下： $100 \leq \text{用户满量程} \leq \text{标定满量程}$ 。

满量程值是一个不带小数点的数值，如果分度值选用的是带小数点的，那么满量程使用对应的小数点位数。比如分度值设置为 3（0.001），它是 3 位小数点。那么满量程也是三位小数点，这个时候满量程值如果为 30000，它代表的含义就是 30.000。

8.5 300 区寄存器（标定功能）

地址	字数	功能	默认值	解释
300	1	标定分度值	3	0~23（见注 1）会同时修改用户值
301	1	备用	0	
302	2	标定满量程值	8000	100~12000000（见注 2）会同时修改用户值
304	2	标定点重力加速度值	0	9300000~10300000 固定为 6 位小数
310	2	标定零点功能		
312	2	标定加载点 1 功能		
314	2	标定加载点 2~5 功能		
320	2	备份标定参数功能		只可单独写
322	2	恢复标定参数功能		只可单独写
340	2	开启/退出写保护状态		只可单独写

8.6 400 区寄存器（标定参数）

地址	字数	功能	默认值	解释
400	1	标定分度值	3	
401	1	备用	0	
402	2	标定满量程	8000	
404	2	标定零点 AD 码	8388608	
406	10	标定加载点 AD 码	2097152	
416	10	标定加载点目标值	8000	
426	2	标定点重力加速度	0	

本区为标定参数读写区，一般执行标定操作都应该通过 300 区实现，而不是直接写本区。直接写本区，不会修改相关的参数，例如修改本区的标定分度值，不会修改用户分度值。

为了提高参数的可靠性，建议上位机读取并保存本区的参数，当发生参数丢失时，可以写回。

8.7 500 区寄存器（高速采样）

地址	字数	功能	默认值	解释
500	1	采样 1 的序列号		见高速采样小节
501	2	净重高速采样 1		
503	2	净重高速采样 2		
.....				
531	2	净重高速采样 16		

9 常用操作

9.1 读取称重数据

保持寄存器地址 0~7 保存称重数据和状态。其中某些状态标志，决定着重量数据是否有效，一般是需要同时读取的。

对于某些简化场合，也可以只读取其中的部分寄存器。比如不需要状态信息，小数点位数固定，不关心皮重值或不使用去皮功能，那么只读取一个净重（地址 4、5）也是可以的。

例：读净重值（读取保持寄存器地址 4、5）

地址	字数	功能	读写
4	2	净重值	只读

净重值为双字变量，对应的保持寄存器地址 4~5。

这里的净重值，是不含有小数点的值。比如 1.000kg，读取到的值就是 1000。

通讯示例：

主机发送（十六进制）：01 03 00 04 00 02 85 CA

模块应答（十六进制）：01 03 04 00 00 03 E8 FA 8D

00 00 03 E8 为净重值 0x000003E8 = 1000

9.2 去皮操作

去皮操作是通过写去皮命令寄存器（地址 41）实现的，见下表

写入内容	作用	备注
22200	取消皮重	将皮重设置为 0
22211	判稳去皮	稳定的话，则将皮重设置为当前毛重，否则失败
22222	不判稳去皮	将皮重设置为当前毛重

例：不判稳去皮（写保持寄存器地址 41 为 22222）

地址	字数	功能	读写
41	1	去皮命令	只可单独写

通讯示例：

主机发送（十六进制）：01 10 00 29 00 01 02 56 CE 1F 9D

模块应答（十六进制）：01 10 00 29 00 01 D0 01

9.3 设皮操作

设皮操作就是直接改写皮重寄存器（地址 2 和 3）。比如要将当前的皮重值改为 0.500kg（小数点位数为 3 时），那么直接向皮重寄存器写入 500 即可。

例：设皮重为 500（写保持寄存器地址 2、3 为 500）

地址	字数	功能	读写
2	2	皮重值	可读写

通讯示例：

主机发送（十六进制）：01 10 00 02 00 02 04 00 00 01 F4 72 61

模块应答（十六进制）：01 10 00 02 00 02 E0 08

9.4 置零操作

置零操作是通过写置零命令寄存器（地址 42）实现的，见下表

写入内容	作用	备注
33300	取消置零	恢复开机零点
33311	置零	毛重处于手动置零范围，置零成功，否则失败
33322	强制置零	不判置零范围，直接置零。

例：置零（写保持寄存器地址 42 为 33311）

地址	字数	功能	读写
42	1	置零命令	只可单独写

通讯示例：

主机发送（十六进制）： 01 10 00 2A 00 01 02 82 1F 80 F2

模块应答（十六进制）： 01 10 00 2A 00 01 20 01

9.5 重量锁定

置零操作是通过写锁定命令寄存器（地址 40）实现的，见下表

写入内容	作用	备注
11100	取消锁定	锁定重量设置为为 0，锁定重量 0
11111	锁定毛重	锁定重量设置为当前毛重，锁定标志为 1
11122	锁定净重	锁定重量设置为当前净重，锁定标志为 1
11133	锁定毛重内码	锁定重量设置为当前毛重内码，锁定标志为 1
11144	锁定净重内码	锁定重量设置为当前净重内码，锁定标志为 1
11155	锁定 AD 码	锁定重量设置为当前 AD 码，锁定标志为 1

发送锁定命令后，就可以通过锁定重量寄存器（地址 16、17）读取锁定命令指定的锁定内容。
在总线上有多个设备时通过锁定命令读取重量数据，可以保证各个模块的同步采样。

例：锁定净重内码（写保持寄存器地址 40 为 11144）

读取锁定的净重内码（读保持寄存器地址 16、17）

地址	字数	功能	读写
16	2	锁定重量	只读
40	1	锁定命令	只可单独写

通讯示例：

主机发送（十六进制）： 01 10 00 28 00 01 02 2B 88 BE EE

模块应答（十六进制）： 01 10 00 28 00 01 81 C1

主机发送（十六进制）： 01 03 00 10 00 02 C5 CE

模块应答（十六进制）： 01 03 04 00 00 00 31 3B E7

9.6 获取有效的采样

大多数的应用场合，无需关心是否是有效采样，可以忽略本小节内容。
而对于某些实时性要求比较高的场合，希望能够获取每一次有效的 AD 采样，那么就需要关心是否是有效采样了。
当通讯读取重量数据的周期小于模块 ADC 采样周期，读取到的重量数据可能是上一次的采样结果（我们称为无校采样数据）。通过读取状态标志寄存器（地址 7），判断其中的“新 AD 采样

有效”标志（bit11），如果为 0，则表明本次读取的是无校采样数据，可以丢弃。

模块最高采样速度可以到达 160 次/秒，对于很多主机而言，很难做到每秒钟 160 次以上的读数。这个时候就需要使用高速采样缓冲区，见下一小节。

9.7 高速采样

对于嵌入式应用，采用高通讯波特率，用上一小节的方式，可以获取全部的有效采样。但对于大部分主机而言，通讯速度一般为每秒几十次，高于 40 次/秒的采样，想要获取全部有效采样，就要用到高速采样缓冲区（500 区）。

高速采样缓冲区里面保存了最新的 16 个净重采样值（地址 501~532），和第一个采样值的序列号（地址 500）。用户需要自动拼接。

比如第一次读取的序列号是 1，十六个采样值时 A1~A16；第二次读取的序列号是 7，采样值是 B1~B16。那么这个时候实际完成了 22 次采样，采样数据为 A1~A16，B11~B6。其中 B1~B10 和 A7~A16 是重复的丢弃。

9.8 IO 功能

IO 功能是通过设置 IO 模式寄存器（地址 44）实现的。标准版的 IO 功能只能实现通过 Modbus 指令读取 IO 的状态，或者输出 0 或者 1。如果需要更多的功能（如上下限报警、按键置零等）则需定制软件实现。IO 的状态是不保存的，每次上电都是默认为“弱上拉输入”。配置为“弱上拉输入”时，IO 的输入状态可以在状态标志（地址 7）的 bit10 获取。

写入内容	作用	备注
55500	弱上拉输入	上电默认
55511	推挽输出 0	
55522	推挽输出 1	

例：推挽输出 1（写保持寄存器地址 44，323 为 55522）

地址	字数	功能	默认值
44	1	IO 模式	

通讯示例：

主机发送（十六进制）：01 10 00 2C 00 01 02 D8 E2 7B B5

模块应答（十六进制）：01 10 00 2C 00 01 C0 00

10 通讯相关设置

10.1 主机发送间隔

Modbus-RTU 标准要求，任何数据帧之间要间隔 3.5T。也就是说主机发送完等待 3.5T 后模块才开始应答，模块应答完后间隔 3.5T 主机才能发送下一条数据。对于嵌入式应用，往往会忽略模块应答后的 3.5T 的等待，这个需要避免的。

由于模块命令响应延时的不确定性，在一个总线上接多模块时，在模块应答后主机需要在等

待 3.5T 的基础上额外再等待 5ms，这个是推荐的。

10.2 通讯参数

通讯参数由通讯模式、通讯地址、波特率、校验方式和通讯发送延时组成

标准模块通讯模式固定为 0（Modbus-RTU），如果是定制协议的模块使用定制通讯协议时需要设置为 1。

通讯地址、波特率、校验方式需要和上位机相同。这三个参数修改后，会用原参数应答。然后就用新的参数，接受指令。比如波特率是 38400，改写为 9600，那么模块会用 38400 应答指令，然后切换为 9600 波特率，这个时候再发命令就要用 9600 了。

例：设置通讯地址为 10（写保持寄存器地址 101 为 10）

地址	字数	功能	默认值
101	1	通讯地址	1

主机发送（十六进制）： 01 10 00 65 00 01 02 00 0A 2F A2

模块应答（十六进制）： 01 10 00 65 00 01 11 D6

10.3 通讯参数恢复

通讯地址有 246 种、波特率有 8 种、校验方式有 4 种，一共有 $246 \times 8 \times 4 = 7872$ 种可能，如果默认通讯参数被意外改写了，很难通过扫描的方式找回来。这个时候就需要使用通讯参数恢复功能。

模块开机后的 300ms 内会使用固定的通讯参数（通讯模式=Modbus-RTU，通讯地址=247，波特率=38400，校验方式=偶校验，发送延时=10）等待接收命令，如果在 300ms 内接收到进入固定通讯参数模式命令，则将继续使用这些通讯参数。否则的话就会使用设置的通讯参数，进入正常通讯模式。

状态标志（保持寄存器地址 7）的 bit9 用来区分是否用的是固定通讯参数。

进入或退出固定通讯参数模式可通过写保持寄存器地址 43 来实现。

写入内容	作用	备注
44400	退出	写入任何通讯参数也会退出固定参数通讯模式
44411	进入	只有在开机 300ms 内发有效

例：进入固定通讯参数模式（写保持寄存器地址 43 为 44411）

地址	字数	功能	默认值
43	1	固定通讯参数模式	

主机发送（十六进制）： F7 10 00 2B 00 01 02 AD 7B B3 5C

模块应答（十六进制）： F7 10 00 2B 00 01 65 57

恢复时主机以 38400，偶校验的方式循环发送 F7 10 00 2B 00 01 02 AD 7B B3 5C，然后给模块重新上电，此时模块就进入了固定通讯参数模式，可以通讯了。

发送退出固定通讯参数模式命令，或者设置任意通讯参数都会退出固定通讯参数模式。

10.4 通讯发送延时

通讯发送延时（0~120），单位为 ms。是在模块发送数据前额外添加的延时，他会使响应时间变长。对于某些 RS232 转 RS485 或者 USB 转 RS485 的转换器或者数据线，他们的发送和接收切换需要一个延时，模块应答的太快会造成不能通讯或者通讯不可靠。这个时候就需要设置通讯发送延时，一般推荐设置为 10（ms）。对于原生的 RS485 接口，发送接收切换没有延时，通讯发送延时推荐设置为 0。

10.5 响应时间

模块读操作的最大响应时间为 20ms，写操作的最大响应时间为 160ms。所以建议设置 Modbus 超时时间为 200ms 以上。

10.6 低字在前访问

对于双字（32bit）变量，模块采用高字在前的方式映射。比如毛重值它占用了保持寄存器地址 0 和地址 1 这两个寄存器，那么地址 0 里面保存的就是毛重的高 16bit，地址 1 里面保存的就是低 16bit。大部分主机是可以设置这个循序的。对于某些不可设置顺序，而且使用的是低字在前的主机，则无法简单使用。

为了解决这个问题，模块提供了低字在前的映射区：1000~1500 区，功能和 000~500 区一致，只是双字变量低字在前。也就是说地址 1000 保存的是毛重的低 16bit，地址 1001 保存的是高 16bit。

11 称量相关说明

11.1 标定操作

首先设定标定参数：标定分度值和标定满量程，设定标定分度值时会自动修改用户分度值，设定标定满量程时会自动修改用户满量程。

然后再标定零点和各个加载点。模块支持最多五段，也就是 5 个点的非线性修正。其中加载点 1 是必须标定的，即至少使用一段标定。标定加载点 1 功能就是用来确定加载点 1 的，通过写入加载点 1 的值，来执行标定功能。

标定加载点 2~5 功能就是用来确定加载点 2~5 的，通过写入加载点的值，来执行标定功能。每个加载点需依次标定，就是标定过加载点 1，然后在执行的就是标定加载点 2，然后才是标定加载点 3。如果要修改加载点 2，或者标定一个小于当前加载点的点，就必须重新标定加载点 1，重新开始。

各加载点之间至少间隔 10 个分度值。否则会执行错误。

例：标定零点（写保持寄存器地址 310、311 写入 5555555）

地址	字数	功能	默认值
310	2	标定零点功能	

通讯示例：

保持空秤，然后

主机发送（十六进制）：01 10 01 36 00 02 04 00 54 C5 63 2F A8

模块应答（十六进制）：01 10 01 36 00 02 A0 3A

例：标定加载点 1（写保持寄存器地址 312、313 写入 1000）

地址	字数	功能	默认值
312	2	标定加载点 1 功能	

通讯示例：加载 1kg 砝码，然后

主机发送（十六进制）：01 10 01 38 00 02 04 00 00 03 E8 FC 33

模块应答（十六进制）：01 10 01 38 00 02 C1 F9

11.2 称重参数

为了方便用户使用，模块采用两层称重参数模式，就是由传感器厂商或者衡器厂商标定时确定标定参数。而由用户设定用户参数，用户参数是实际使用的称重参数，而标定参数只决定了用户参数的默认值和设置范围。

用户可以在一定范围内修改传感器厂商或衡器厂商的设定。比如标定分度值为 0.001kg，用户觉得精度不够，那么可以把用户分度值改成 0.00001kg。

例：把用户分度值设置为 0.0001kg（写保持寄存器地址 211 为 0）

地址	字数	功能	默认值
211	1	用户分度值	标定值

通讯示例：

主机发送（十六进制）：01 10 00 D3 00 01 02 00 00 B5 F3

模块应答（十六进制）：01 10 00 D3 00 01 F0 30

11.3 置零范围

开机置零范围（地址 204），单位为满量程的百分比，例如满量程为 500kg，开机置零范围为 20，那么开机置零的范围值就是±100kg，模块重新上电后，如果毛重小于±100kg，模块就把该重量作为零点，输出毛重为 0。

手动置零范围（地址 205），单位为满量程的百分比，它影响着置零命令能否成功。如果当前毛重（相对于开机零点）小于手动置零范围值时，置零命令有效，模块把当前毛重作为零点，输出毛重为 0。

例：设置开机置零范围为 50%、手动置零范围为 20%（写保持寄存器地址 204 为 50，地址 205 为 20）

地址	字数	功能	默认值
204	1	开机置零范围	20
205	1	手动置零范围	4

通讯示例：

主机发送（十六进制）：01 10 00 CC 00 02 04 00 32 00 14 5E 6A

模块应答（十六进制）：01 10 00 CC 00 02 81 F7

11.4 零点跟踪

模拟传感器的零点会随时间和温度缓慢变化。当秤处于零位附近时，可以通过零位跟踪修正这种变化。零位跟踪有两个参数：零位跟踪范围和零位跟踪速度。零位跟踪范围（0~9），单位是0.5个分度值。比如分度值是0.001kg，零位跟踪范围设的是4，跟踪范围就是2个分度值0.002kg。当毛重小于±0.002kg，模块就会把毛重慢慢修正为0。

零点跟踪速度（0~3），数值越大零点修正的速度就越快。零点跟踪速度快，可能会造成加载小分量，被跟掉一定的数值，使重量稍微偏轻。

例：设置零点跟踪范围为2，零点跟踪速度1（写保持寄存器地址207为2，地址208为1）

地址	字数	功能	默认值
207	1	零点跟踪范围	1
208	1	零点跟踪速度	0

通讯示例：

主机发送（十六进制）：01 10 00 CF 00 02 04 00 02 00 01 DF BF

模块应答（十六进制）：01 10 00 CF 00 02 71 F7

11.5 蠕变补偿模式

模拟传感器长时间加载，会引起零点的缓慢变化，当加载的重量处于零点跟踪范围外时，这种变化就不能自动的被修正。蠕变补偿就是用于修正这种缓慢变化的。对于重量缓慢变化的场合（如气体罐装），必须关闭蠕变补偿，否则会引起计量误差。

当重量处于四舍五入的临界点时，很小的噪声就会使输出上下跳动一个分度值，称之为闪变点。如果不希望看到这种跳动，则可以选择闪变点抑制。

例：同时开启蠕变补偿和闪变点抑制（写保持寄存器地址209为3）

地址	字数	功能	默认值
209	1	补偿模式	1

通讯示例：

主机发送（十六进制）：01 10 00 D0 00 01 02 00 03 F5 C1

模块应答（十六进制）：01 10 00 D0 00 01 00 30

11.6 用户修正系数

用户修正系数（地址214，215）用于修正标定的误差，它固定6位小数点。标定完成后，用户修正系数为1000000，就是1.000000，代表不修正。如果把它设置为别的值，则重量就会成比例的变化。比如设置为900000（0.900000），那么重量就会变为原来的0.9倍。它的设置范围是250000~4000000（0.25~4）。

执行用户修正有三种方法，一是直接修改用户修正系数这个参数，二是通过用户重力加速度修正功能来自动的修改这个系数，三是通过目标重量修正功能来自动的修改这个系数。

例：设定用户修正系数为 0.9（写保持寄存器地址 214、215 写入 9000000）

地址	字数	功能	默认值
214	2	用户修正系数	1000000

通讯示例：

主机发送（十六进制）：01 10 00 D6 00 02 04 00 0D BB A0 9D 92

模块应答（十六进制）：01 10 00 D6 00 02 A0 30

11.7 用户重力加速度修正

使用地重力加速度如果和标定地重力加速度不同，会带来测量误差，通过输入用户地的重力加速度，自动的计算用户修正系数，从而修正这个误差。

用户重力加速度值，固定六位小数点，设置范围是 9300000~10300000（10.3~11.3）。如果写入的用户重力加速度在这个范围内，同时标定点也设置标定重力加速度，那么就会执行修正功能。否则会返回 Modbus 设备错误。

例：执行用户重力加速度为 11.0 修正（写保持寄存器地址 220、221 写入 10000000）

地址	字数	功能	默认值
220	2	用户重力加速度值	

通讯示例：

主机发送（十六进制）：01 10 00 DC 00 02 04 00 98 96 80 11 49

模块应答（十六进制）：01 10 00 DC 00 02 80 32

注意：必须首先设定了标定点重力加速度，才能使用用户重力加速度修正功能。

11.8 目标重量修正

当发现模块输出值和实际重量有了偏差，除了可以手动的修改用户修正系数来修正，还可以通过目标重量修正功能自动的来修改用户修正系数。

目标重量修正功能，就是通过写入当前的正确重量值，来修改用户修正系数。比如当前重量是 900，但模块输出是 1000，那么可以写入 900，这样就会把用户修正系数变为 0.9。其实这相当于实现了用户标定功能。

需要注意的是输入的正确重量的小数点位数，需要和用户分度值的一致。目标重量的范围（不考虑小数点）是 800~用户满量程。

12 参数相关设置

12.1 恢复默认用户参数

恢复默认用户参数寄存器（地址 224，225），写入 44444444，可以初始化用户参数，把用户参数都设置为默认值。如果用户参数设置乱了，不知道怎么恢复，那么就可以使用这个功能恢复为默认值，然后调整关注的参数即可。

例：恢复默认用户参数（写保持寄存器地址 224、224 写入 4444444）

地址	字数	功能	默认值
224	2	恢复默认用户参数	

通讯示例：

主机发送（十六进制）： 01 10 00 E0 00 02 04 00 43 D1 1C 51 AA

模块应答（十六进制）： 01 10 00 E0 00 02 40 3E

12.2 参数可靠性

参数可靠性对于系统是非常重要的，一旦称重相关参数丢失或被意外改写，有可能会带来计量误差。下面提出几个方法，可以提高参数可靠性。

12.2.1 检查参数校验

每次开机读取参数校验和和设置次数，和标定完成后的进行比对。如果不一致就可以判断参数丢失或被意外改写。

例：读取参数校验和（读保持寄存器地址 26~30）

地址	字数	功能	默认值
26	1	用户参数校验和	
27	1	标定参数校验和	
28	2	用户参数设置次数	
30	2	标定参数设置次数	

通讯示例：

主机发送（十六进制）： 01 03 00 1A 00 04 65 CE

模块应答（十六进制）： 01 03 08 B7 6B D5 4D 00 00 03 8B 7A DF

12.2.2 检查参数健康状态

每次开机后读取参数健康状态寄存器，发现有损坏的块（不为 0）则可以重新备份，防范于未然。

例：读取参数健康状态寄存器（读保持寄存器地址 32）

地址	字数	功能	默认值
32	1	参数健康状态	

通讯示例：

主机发送（十六进制）： 01 03 00 20 00 01 85 C0

模块应答（十六进制）： 01 03 02 00 00 B8 44

参数健康状态寄存器（地址 32）含义如下

位	含义	备注
bit0	用户参数副本 1	0: 正常 1: 损坏 开机后如果这四个副本都被破坏则模块使用默认用户参数
bit1	用户参数副本 2	
bit2	用户参数副本 3	
bit3	用户参数副本 4	
bit4	标定参数副本 1	0: 正常 1: 损坏 开机后如果这四个副本都被破坏则模块使用备份标定参数副本
bit5	标定参数副本 2	
bit6	标定参数副本 3	
bit7	标定参数副本 4	
bit8	备份标定参数副本 1	如果备份标定参数副本也被都被破坏则模块使用默认标定参数
bit9	备份标定参数副本 2	
bit13	内部 RAM 硬件损坏	需更换模块
bit14	内部 XRAM 硬件损坏	需更换模块
bit15	非法软件	需更换模块

用户参数副本或标定参数副本，有 1~3 块损坏时，可以通过重新写入任意一个用户参数或标定参数来恢复备份。备份标定参数副本损坏则需要发送备份命令恢复，见下一小节。

12.2.3 软件写保护

开启软件写保护，避免参数被意外改写。

100 到 400 区的写入操作可以开启软件写保护，这个状态是断电保持的。

开启软件写保护后，要写某一个区就必须先写这个区的最后一个寄存器为 12345。比如 300 区，那就写地址 349 为 12345。这个断电不保持的。也就是说开启了软件写保护后。如果需要写 100~400 区寄存器，每次上电都要写一次 12345。

例：开启软件写保护（写保持寄存器地址 340，341 为 13142012，写其他值退出）

地址	字数	功能	默认值
340	2	开启/退出写保护状态	

通讯示例：

主机发送（十六进制）：01 10 01 54 00 02 04 00 C8 87 FC 18 BF

模块应答（十六进制）：01 10 01 54 00 02 01 E4

12.2.4 备份对比

在上位机备份全部用户和标定参数，开机后比对，发现不一致则写回。

例：读取全部用户参数（读保持寄存器地址 200~215）

通讯示例：

主机发送（十六进制）： 01 03 00 C8 00 10 C5 F8

模块应答（十六进制）： 01 03 20 00 00 00 00 00 10 00 01 00 14 00 0A 00 00 00 01
00 03 00 01 00 00 00 00 00 00 1F 40 00 0E F4 20 58 12

例：读取全部标定参数（读保持寄存器地址 400~427）

通讯示例：

主机发送（十六进制）： 01 03 01 90 00 1C 45 D2

模块应答（十六进制）： 01 03 38 00 03 00 00 00 00 1F 40 00 82 F8 0D 00 07 D9 2E
00 FF FF FF 00 FF FF FF 00 FF FF FF 00 FF FF FF 00 00 03
E8 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
95 89 40 5D B0

12.3 备份和恢复标定参数

备份标定参数副本 1 和 2，需要手动操作备份。备份的标定参数除了可以手动恢复以外，也可以在 4 份标定参数都被破坏时，被作为有效参数使用。

例：备份标定参数（写保持寄存器地址 320，321 为 11111111）

地址	字数	功能	默认值
320	2	备份标定参数功能	

通讯示例：

主机发送（十六进制）： 01 10 01 40 00 02 04 00 10 F4 47 FD 38

模块应答（十六进制）： 01 10 01 40 00 02 41 E0

例：恢复标定参数（写保持寄存器地址 322，323 为 22222222）

地址	字数	功能	默认值
322	2	恢复标定参数功能	

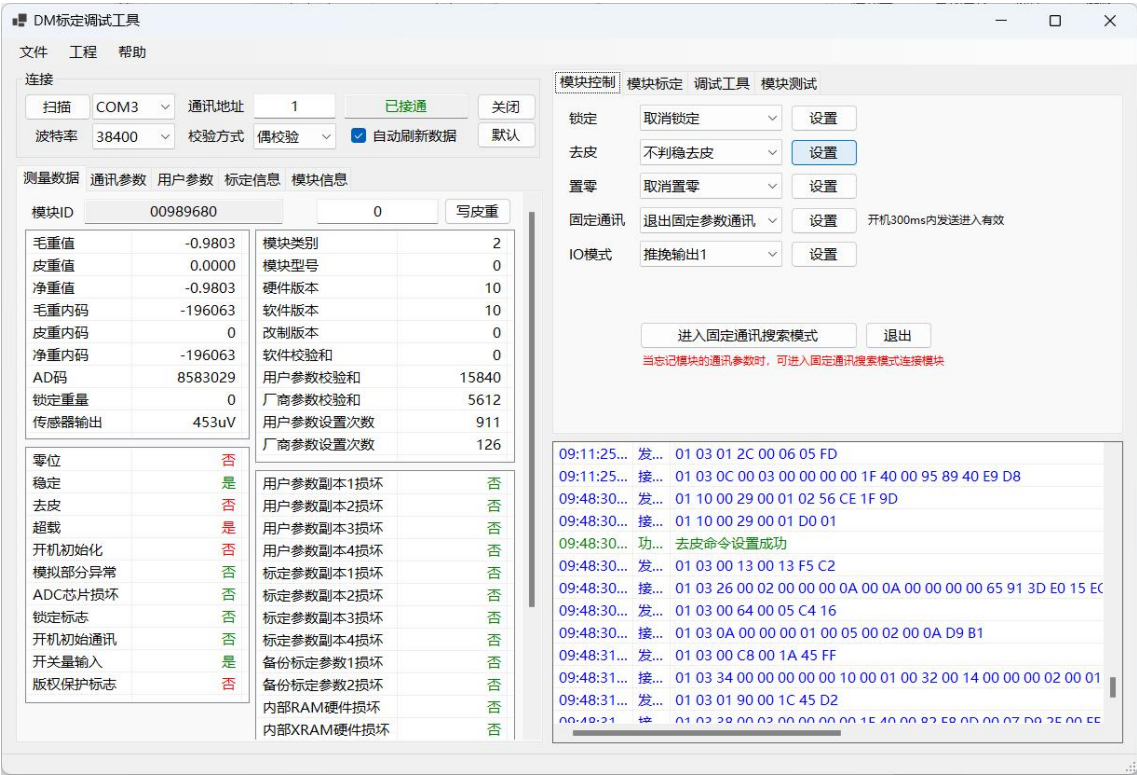
通讯示例：

主机发送（十六进制）： 01 10 01 40 00 02 04 00 21 E8 8E 64 61

模块应答（十六进制）： 01 10 01 40 00 02 41 E0

13 工具软件简介

模块提供免费的标定调试软件《DM 标定调试软件》，通过它可以方便的配置模块的参数，标定模块和执行各种功能。



它既可以作为一个工具软件使用，也是一个编程演示工具。软件收发的每条数据都可以在右下的数据收发框看到。通过软件执行一条功能，数据收发框就会用绿色的文字显示执行结果。这个上面两行就是这条命令实际收发的数据。

右键点击某条收发数据，然后点击“复制行”可以复制该数据。本手册的通讯示例大部分都是这么获得的。本手册的通讯示例通讯地址用的都是 1，对于别的通讯地址，可以用这个工具软件来获得收发数据。

14 定制软件和远程升级

模块支持定制软件和远程升级功能。例如定制 IO 口功能，通讯协议等。
受硬件限制，定制通讯协议只支持读称重数据、去皮命令、置零命令等，不支持更改参数配置、标定操作等。