SJ-101CX

数字称重变送模块

使用说明书

V1.02

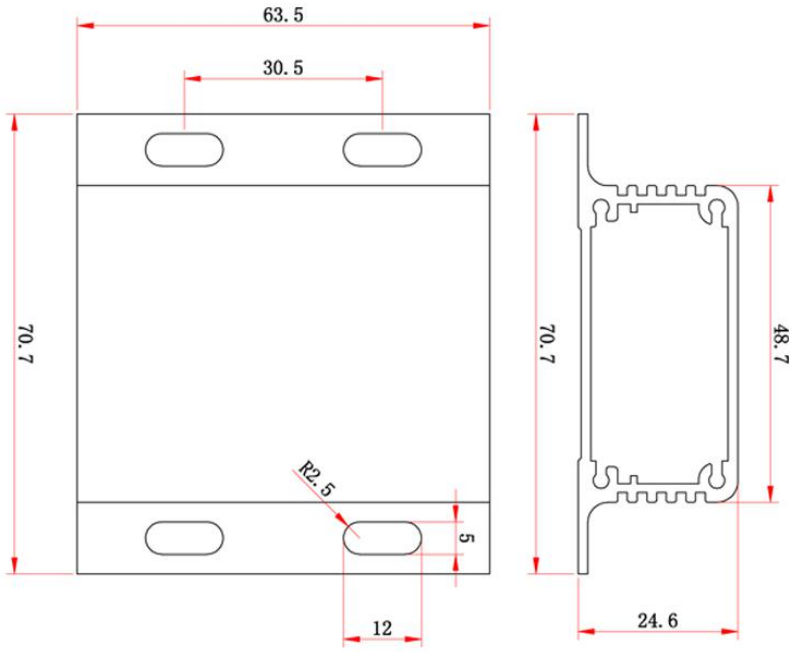
一、主要参数.....3
二、简介说明.....3
1、尺寸.....3
三、模块说明.....4
1、接口定义.....4
四、通信接口说明.....4
五、称重简介.....6
六、标定说明.....6
七、通信协议列表.....8
八、指令分解说明.....11
（一）、称重数据获取指令(建议一次性读取重量精度及状态).....11
（二）、称重操作指令.....13
（三） 、通讯操作指令.....15
（四） 、标定操作指令.....16
（五）、其它功能.....18
十、其它说明.....19
CRC 校验.....19
SJ101CX 测试工具使用.....21

一、主要参数

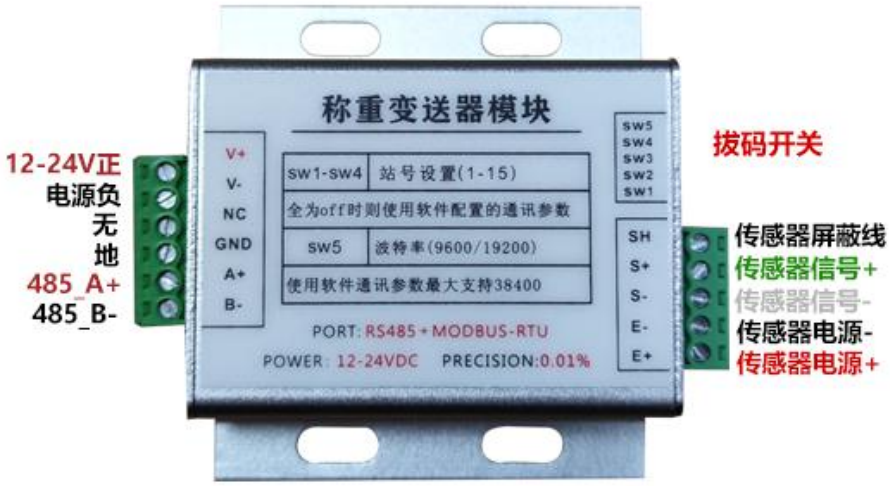
1.	型号：	SJ-101CX
2.	准确度等级:	3 级
3.	AD 采样速度：	10/40HZ
4.	分度数	n=3000, e=10d
5.	分度值：	1/2/5
6.	最小输入信号	1uV/e
7.	传感器灵敏度范围：	1.5～3mV / V
8.	传感器输入电压范围：	-15mV~+15mV
9.	传感器激励电压：	5V
10.	输入电源电压：	DC 12-24V
11.	串行通讯接口：	RS485 信号传输距离≤1000 米
12.	通信方式：	标准 MODBUS-RTU 协议
13.	使用温度	0-40℃
14.	储运温度：	-25～55℃

二、简介说明

1、尺寸



三、模块说明



可通过拨码开关设置相应的站号进行组网通讯。

1、接口定义

1、电源输入：

模块支持 12-24V 宽电压输入。（注意电源正负极性）

2、传感器接口

传感器采用四线屏蔽线接口。无长线补偿。

E(+)：传感器电源正　　E(-)：传感器电源负

S(+)：传感器信号正　　S(-)：传感器信号负

通常红线代表 E+ ，黑线代表 E-，绿线代表 S+，白线代表 S-。具体需要查看传感器的说明书。

四、通信接口说明

1、通信配置

模块的默认配置如下：

波特率：9600　　数据位：8 位

奇偶校验：无　　停止位：1 位　　设备地址：0x01

2、MODBUS-RTU 协议

模块支持 MODBUS-RTU 从机模式。**不熟悉协议也可以通过第九章指令解析实现通讯。**

支持 03，06，16 命令

MODBUS 采用最高频率不超过 20HZ，通信等待时间最小 50ms。

本模块采样 485 通信。

对于大部分应用除出厂用到校准外，使用过程中仅用到 2 条指令：读取重量精度状态和清零指令。

3、拔码开关设置

模块可设备不同的站号进行组网，可通过软件配置或者使用拔码开送设置通讯站和波特率。

通过拔码开关设置的参数需要重新上电才生效。

当 SW1-SW4 均为 off 时，将采用模块软件配置的参数进行通讯默认：9600,n,8,1，站号 1。软件配置通讯参数可通过通讯来修改。最高支持 115200。否则按下表通讯参数

sw4	sw3	sw2	sw1	站号(设备号)
off	off	off	off	使用软件配置参数
off	off	off	on	1
off	off	on	off	2
off	off	on	on	3
off	on	off	off	4
off	on	off	on	5
off	on	on	off	6
off	on	on	on	7
on	off	off	off	8
on	off	off	on	9
on	off	on	off	10
on	off	on	on	11
on	on	off	off	12
on	on	off	on	13
on	on	on	off	14
on	on	on	on	15

sw5	状态(sw1-sw4 不全为 off 时有效)
off	9600
on	19200

五、称重简介

一个完整的称重设备包括称重传感器、信号放大、信号处理、信号转换成重量、显示等。SJ101CX 是集成信号放大、处理、转换的一个装置，并将重量数据通过 RS485 接口与接口设备进行交互并组成完整的称重设备。

我们常用的应变片传感器是由电阻组成的电桥。通过压力变产生形变而改变阻值，从而造成输出的电压变化。SJ101CX 则是将采集到的电压信号转变成数字信号。再通过 RS485 通信对模块进行标定将数字信号转变成重量数据。

如何选择合适的传感器，以及如何判断当前重量可以做的精度。根据以下参数：

传感器灵敏度(传感器说明书上)2mV/V;

SJ101CX 最小输入信号 1uV/e、激励电压 5V(传感器输入电压)。可以确定出合适的称量参数。

满量程输出电压=激励电压*灵敏度

以 100kg 的传感器为例，加载 100kg 时传感器输出的电压= 2mV/V* 5V。即为 10mV=10000uV。

e 是计量时的检定分度值，d 是我们称重时的实际分度值。e 通常等于 10d（e=10d）。

而 SJ101C 的最小输入信号是 1uV/e。也就是可识别成 10000e(100000d)。

（满量程*$10^{精度}$）/分度值 < 100000

如 200kg 传感器做 0.01kg 的精度（200.00kg）。200*10^2/1=20000 在 100000 范围内，因此可以实现。

不过需要注意的是：以上参数仅供参考，实际传感器质量各不相同，需要根据实际的示值稳定情况进行选择合适的量程和精度。

六、标定说明

1、标定简介

任何一款称重设备在连接好传感器后都需要进行砝码标定(可通过传感器最大量程和灵敏度做免标定，但精度很低传感器的一置性和模块的基准电压一置性都将影响称重精度)。在现场如果没有砝码可以使用其它电子秤称(精度需要大于等于当前要校准的精度)一个物体作为参照物。通常选则传感器满量程的 1/4-3/4 的物体。

通常称重只校准一次即可，建议客户可以选择将该功能内置到软件中，方便之后修理换传感器后需要重新校准。

当然也可以在需要校准时连接我们的测试软件进行校准。

变送器是将称重传感器的电压信号转变成数字信号。该信号同传感器压的物体成线性关系。因此通过标定找出这样的关系来。

取两个点：

当重量为 0 时（秤台空载），保存变送器获取的传感器输出值。

当重量为我已知的砝码值时，保存变送器获取的传感器输出值。

2、标定参数简介

（1）、满量程：设置的最大称重值，称重超过该值称重数据无效，设置时建议小于传

感器的标注值，如 40kg 传感器设置 30kg 的满量程。

（2）、标定砝码值：该值为标定时加载的砝码重量。

（3）、精度：用于确认小数点的位置。

（4）、分度值：尾数据跳的步长，如分度值为 5，则重量末位以 0、5 起跳。

（5）、零点标定：用于获取空载时传感器输出值，标定零点后需要检查标定是否成功。只有标定成功后才可以下一步操作。

（6）、加载标定：加载[标定砝码值]设定的对应砝码值后，用于获取加载时传感器的输出值。

（7）、同步标定数据：用于保存当前标定数据。

4、标定步骤

要想改变满量程、精度、分度值、加载值必须进行一次标定操作才可以完成。校准步骤如下：

1、 设置满量程（也可以不用设置使用上一次的值）
2、 设置精度（也可以不用设置使用上一次的值）
3、 设置分度值（也可以不用设置使用上一次的值）
4、 设置加载砝码值（也可以不用设置使用上一次的值）
5、 清空秤台，发送校零点指令
6、 查询校准状态，只有当零点校准成功后进入下一步
7、 放上对应的砝码，发送校准砝码指令。只有零点校准成功后才有效。
8、 查询校准状态
9、 查询到校准成功后，发送校准数据同步生效指令。
10、 校准完成

注：

模块本身只有分辨率（30000）是没有单位的。实际单位和校准及传感器有关。如接 30kg 传感器设置满量程 30000，精度 0。则是以 g 为单位。

如果设置满量程 30000，精度 0.000，则以 kg 为单位（30.000）。

如果接的是 3t 的传感器，设置满量程为 3000，精度设置成 0，单位也是 kg。

七、通信协议列表

40001（十进制）是 PLC 保持继存器地址，可使用 03，06，16(十进制)指令，协议地址减去 40001 即可。**(通用功能一般仅可只用到 40001-40005)**

以下是协议所支持的功能：

MODBUS-RTU 格式说明					
PLC 地址 (十进制)	**协议地址 (十进制)**	**功能**	**说明**		**操 作方式**
40001 40002	00 01	获取当前重量值（不含小数点）	32 位重量数据高 16 位 32 位重量数据低 16 位		读
40003	02	小数点精度	0-3		读
40004	03	模块状态	寄 存 器 位	对应位值含义	读
			0	0 称重不稳定 1 称重稳定	
			1	0 非零点 1 零点	
			2	0 没有超重 1 超重	
			3	0 没有按键 1 有按键触发	
			4	0 开机有零点 1 开机无零点	
			5	0 称重无效 1 称重有效	
			6-15	保留	
40005	04	模块操作	写 1 称重示值清零		写
40006	05	开机置零	0 1 2	开机置零 开机不置零（以标定点作为零点） 开机不置零（以置零点作为零点）	读/写
40007	06	滤波设置	0 1 2	弱 中 强	读/写
40008	07	灵敏度设置 (蠕动处理)	0 1	灵敏度高(蠕变关) 灵敏度低(蠕变开,不适合慢加重量)	
40009	08	零点跟踪设置 (零点跟踪是当前重量在零点附近的范围内波动将自动清零)	0 1 2 3 4 5 6	0.5e 1.0e 1.5e 2.0e 2.5e 3.0e 5.0e	

40010	09	波特率设置	0 1 2 3 4	9600 14400 19200 38400 115200	
40011	10	校验位设置	0 1 2	无 奇 偶	
40012	11	设备地址（站号）	1-99（0 为广播地址）		
40013	12	通讯数据设置有效	发送 1 将立即生效当前设置的：波特率、校验位、设备地址		写
40014 40015	13 14	满量程设置	满量程高 16 位 满量程低 16 位		读/写
40016 40017	15 16	砝码值设置 校准值根据手中砝码而定，正常为满量程的一半（1/4 - 3/4）	校准值高 16 位 校准值低 16 位		读/写
40018	17	精度设置	0-3		读/写
40019	18	分度值设置	0 1d 1 2d 2 5d		读/写
40020	19	校准零点	写数据 1 进入零点校准 连接好秤台，并清空秤台，发送数据 1 进入零点校准。零点校准将获得空载时传感器输出的信号值。		写
40021	20	校准标定点	写数据 1 进入标定点校准 零点校准成功后，放上和砝码值对应重量的砝码后发送该指令，变送器将会获得加载后的传感器输出信号值。		写
40022	21	生效校准数据	写数据 1 生效：满量程、砝码值、精度、分度值、标定所计算的数据值。变送器将生效的数据运行。		写
40023	22	查询校准状态	0x04 零点校准成功 0x08 标定点校准成功		读
40024 40025	23 24	查看传感器输出值	输出传感器内码值，用于判断当前传感器连接是否正常或者损坏		读
40026	25	清除按键状态	写 1 清除按键状态		写
40027	26	重新启动变送器	写 1 将使变送器复位		写
40028	27	AD 采样速率选择	0 10hz 1 40hz		读/写
待扩展					
40059	58	保存用户数据	可用于记录当前称重单位或需要保存的数据		读/写
40060	59	查看当前版本	102		读

如果是和 PLC 进行通讯，则需要注意三菱 信捷等 PLC(西门子无需交换)读取出来 32 位数据存储方式是低 16 位在前，高 16 位在后，和模块正好相反。因此读取后需要将高低位切换后再访问 32 位数据。

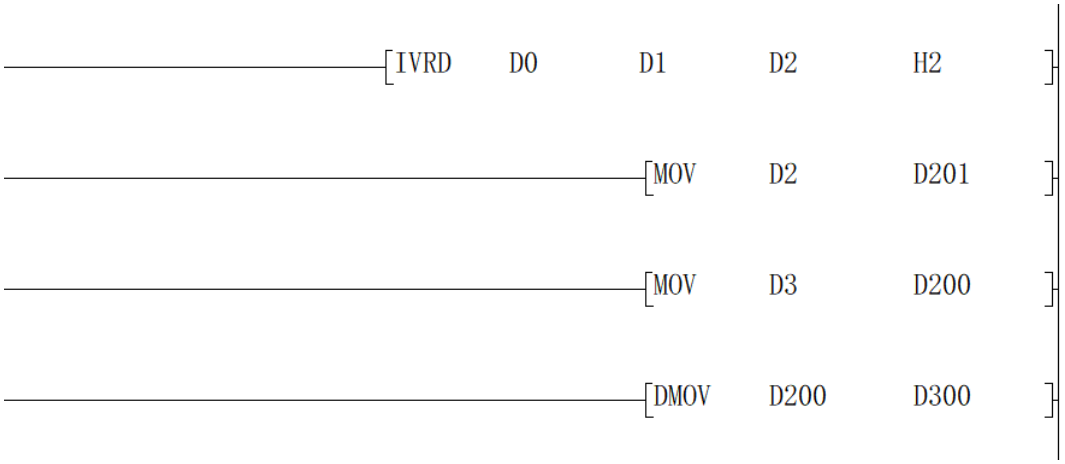
如读取数据存放在 D0 开始的软元：

MOV D0 D11

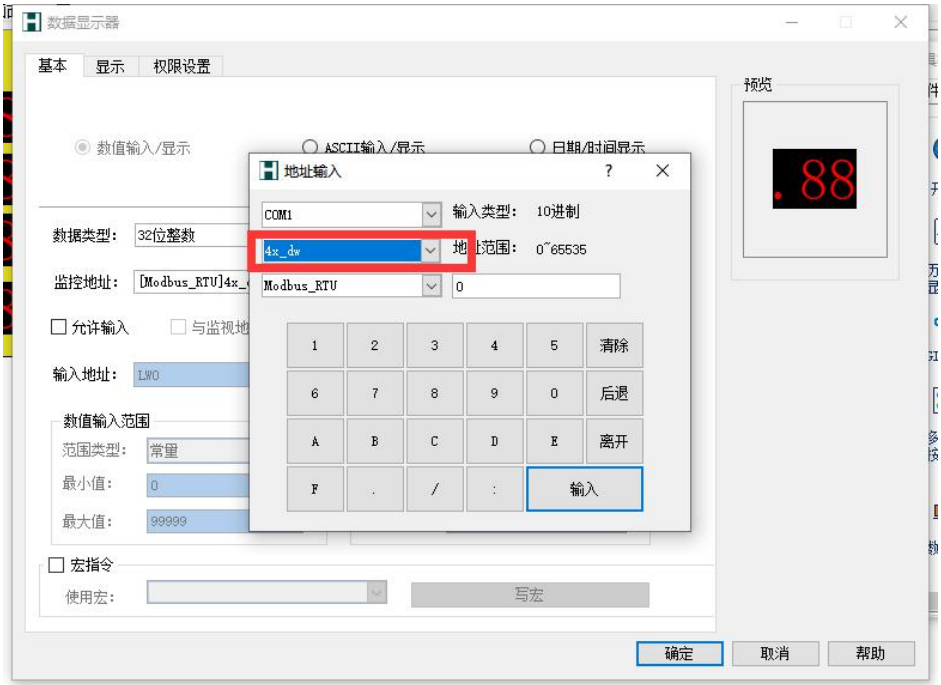
MOV D1 D10

DMOV D10 D20

先将数据高低位对调放到 D10 D11 中。然后用 DMOV 指令访问 32 位数据。



对于组态屏，可以直接设置显示控件高低位以及 32 位整数。



八、指令说明

MODBUS 协议需要注意的是没有帧头帧尾。因此每条指令之间必须有 3.5 以上字符的空闲时间用于软件判断一个完整的帧。且每条指令中间不能有间隔。

以下将以站号为 1 的为例作指令解析，如果需要组网，则修改对应站号及 CRC 校验即可。建议 CRC 校验用软件实现动态计算，方便扩展。

以下 CRC 用 XX 表示数据位是动态不确定的，因此 CRC 也是动态的。

对于大部分应用除出厂用到校准外，使用过程中仅用到 2 条指令：读取重量精度状态和清零指令。

（一）、称重数据获取指令(建议一次性读取重量精度及状态)

1、获取 32 位净重值【01 03 00 00 00 02 c4 0b】

	设备号	功能码	起始地址高位	起始地址低位	寄存器数高位	寄存器数低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x03	0x00	0x00	0x00	0x02	0xc4	0x0b

->返回 32 位净重值及相就状态位（变送器->上位机）。如 15.32 则返回 1532。

	设备号	功能码	返回数据个数	数据个数 4 个字节	CRC16 低位	CRC16 高位
上位机接收	0x01	0x03	0x04	32 位有符号重量数据	xx	xx

返回重量为 4 个字节 32 位的有符号型整数。在程序处理中将 4 个字节处理成 32 位有符号变量即可。如 long Wei=(long)wei[1]<<24|(long)wei[2]<<16|(long)wei[3]<<8|(long)wei[4]。

2、获取精度值 【01 03 00 02 00 01 25 ca】

	设备号	功能码	起始地址高位	起始地址低位	寄存器数高位	寄存器数低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x03	0x00	0x02	0x00	0x01	0x25	0xca

->返回精度值用于参考及计算（变送器->上位机）

	设备号	功能码	返回数据个数	数据个数 2 个字节	CRC16 低位	CRC16 高位
上位机接收	0x01	0x03	0x02	精度	xx	xx

(1)返回值意义

0x0000	0x0001	0x0002	0x0003
0	0.0	0.00	0.000

3、获取称重状态【01 03 00 03 00 01 74 0a】

	设备号	功能码	起始地址高位	起始地址低位	寄存器数高位	寄存器数低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x03	0x00	0x03	0x00	0x01	74	0a

->返回状态（变送器->上位机）

	设备号	功能码	返回数据个数	数据个数 2 个字节	CRC16 低位	CRC16 高位
上位机接收	0x01	0x03	0x02	状态	xx	xx

(1)状态位表示(bit0-bit15)

15-6	5	4	3	2	1	0
保留待扩展	0 称重无效 1 称重有效	0:开机有零点 1:开机无零点	0：无状态 1：有按键按下	0:没有超重 1:超重	0:非零点 1:零点	0:称重不稳定 1:称重稳定

Bit5:称重数据无效，即当前获取的重量值可能和实际加载值不等。如开机时未获得零点时或者标定状态时。

Bit4:开机零点，当开机时 10s 内稳重数据一直未稳定，将获取一个不稳定零点该位置 1

Bit3:按键状态标志，检测到该位置 1 后可通过清除按键状态指令清除该位。

4、可一次性获取重量、精度、状态【01 03 00 00 00 04 44 09】

	设备号	功能码	起始地址高位	起始地址低位	寄存器数高位	寄存器数低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x03	0x00	0x00	0x00	0x04	0x44	0x09

返回数据

	设备号	功能码	返回数据个数	数据个数 4 个字节	数据个数 2 个字节	数据个数 2 个字节	CRC16 低位	CRC16 高位
上位机接收	0x01	0x03	0x08	32 位有符号重量数据	精度	状态	xx	xx

返回数据意义同上

（二）、称重操作指令

1、对模块清零操作【01 06 00 04 00 01 09 cb】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x04	0x00	0x01	0x09	0xcb

清零是消除温度引起的零点漂移。

->返回数据为所发送的数据（变送器->上位机）

2、开机置零设置【01 06 00 05 00 00 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x05	0x00	低位	Xx	xx

操作位数据如下表

0x00	0x01	0x02
开机置零	以标定时的零点作为零点	以置零操作后的零点作为零点

0x00 开机置零：开机无论传感器加载多少重量，都当作称重零点。(默认)

0x01 开机不置零：开机时，选用标定时记录的零点作为开机点。(不常用，将丢弃)

0x02 开机不置零：设备将保存每一次的置零操作后的零点作为开机零点。（常用）

->返回数据为所发送的数据（变送器->上位机）

如果需要上电仍然保持断电前的重量可设置开机不置零。长时间放置可能会有误差，可通过空载时 清零操作清除误差

3、滤波强度设置【01 06 00 06 00 01 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x06	0x00	低位	Xx	xx

数据操作位

0x00	0x01	0x02
弱	中	强

滤波强度为了适应不同的环境称重的稳定性，越稳定则反应灵敏度变低。默认为中。

->返回数据为所发送的数据（变送器->上位机）

4、灵敏度设置【01 06 00 07 00 00 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x07	0x00	低位	Xx	xx

数据操作位

0x00	0x01
灵敏度高	灵敏度低

如果当前环境恶劣或者温度变化大请将灵敏度调低。如果需要秤量细小的物品，高精度的情况下可以将灵敏度调高。

当物品需要长时间放置在秤台上时，需要将灵敏度调低。可解决传感器蠕变及温度变化造成示值偏差。

->返回数据为所发送的数据（变送器->上位机）

5、零点跟踪设置【01 06 00 08 00 05 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x08	0x00	低位	Xx	xx

数据操作位

0x00	0x01	0x02	0x03	0x04	0x05	0x06
0.5e	1.0e	1.5e	2.0e	2.5e	3.0e	5.0e

零点跟踪用于，当秤台空载时由于环境或者温度造成求值不为零，且在上表所设置值的范围内时自动清除当前示值。

->返回数据为所发送的数据（变送器->上位机）

6、采样速率设置【01 06 00 1b 00 00 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x1b	0x00	低位	Xx	xx

数据操作位

0x01	0x00
40HZ	10HZ

默认 10HZ 速率

->返回数据为所发送的数据（变送器->上位机）

（三）、通讯操作指令

该操作只有当 SW1-SW4 都为 OFF 时，模块将用以下配置的通讯参数进行通讯。

1、波特率设置【01 06 00 09 00 00 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x09	0x00	低位	Xx	xx

数据操作位

0x00	0x01	0x02	0x03
9600	14400	19200	38400

设置后只有发送生效指令后才生效。

->返回数据为所发送的数据（变送器->上位机）

2、校验位设置【01 06 00 0A 00 00 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x0A	0x00	低位	Xx	xx

数据操作位

0x00	0x01	0x02
无	奇	偶

设置后只有发送生效指令后才生效。

->返回数据为所发送的数据（变送器->上位机）

3、设备站号设置【01 06 00 0B 00 01 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x0B	0x00	低位	Xx	xx

数据操作位：1-30

设置后只有发送生效指令后才生效。

->返回数据为所发送的数据（变送器->上位机）

4、生效通讯参数设置【01 06 00 0C 00 01 88 09】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x0C	0x00	0x01	0x88	0x09

发送 1 后将生效当前的波特率、校验位、站号设置。

->返回数据为所发送的数据（变送器->上位机）

（四）、标定操作指令

1、称重满量程设置

	设备号	功能码	起始地址高位	起始地址低位	寄存器数量		字节数	数据	CRC16 低位	CRC16 高位
上位机发送	0x01	0x10	0x00	0x0D	0x00	0x02	0x04	**4 字节**	Xx	xx

4 字节满量程值

即所要设置的满量程的值，该值为无符号整型。该值设置后需要发送标定数据同步指令后才有效（如称重满量程为 10000g 精度为 0.1g。则需要发送 100000 将它转换成 4 个字节）。

->返回数据为 01 10 00 0d 00 02 xx xx

2、标定砝码值设置

	设备号	功能码	起始地址高位	起始地址低位	寄存器数量		字节数	数据	CRC16 低位	CRC16 高位
上位机发送	0x01	0x10	0x00	0x0F	0x00	0x02	0x04	**4 字节**	Xx	xx

4 字节砝码重量值

即所要设置的砝码的值，该值为无符号整型。该值设置后需要发送标定数据同步指令后才有效（如砝码为 2000g 精度为 0.1g。则需要发送 20000）。

->返回数据为 01 10 00 0f 00 02 xx xx

3、称重精度设置

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x11	0x00	**低位**	Xx	xx

设置值如下，该值设置后需要发送标定数据同步指令后才有效

0x00	0x01	0x02	0x03
0	0.0	0.00	0.000

->返回数据为所发送的数据（变送器->上位机）

4、称重分度值设置

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x12	0x00	**低位**	Xx	xx

设置值如下，该值设置后需要发送标定数据同步指令后才有效

0x00	0x01	0x02	其它
1	2	5	保留

分度值，常用为 1 默认为 1。

如果不设置以上的值(满量程、精度、砝码值、分度值)而直接进行以下操作，模块将使用上一次校准时所使用的值。设置以上的值不会被立即保存，只有当校准完成后才会保存这些值，否则掉电后会恢复之前的值。

5、零点标定（上位机->变送器）【01 06 00 13 00 01 b9 cf】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x13	0x00	0x01	0xb9	0xcf

->返回数据为所发送的数据（变送器->上位机）

发送标定零点后，需要查询标定状态（为 0x04 时成功），零点标定成功后可进行下一步操作。零点标定失败的原因之一即是当前 AD 值一直不能稳定，检查连线或者使用环境是否会造成称重不稳定。

6、加载标定，放上砝码后发送该指令（上位机->变送器）【01 06 00 14 00 01 08 0e】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x14	0x00	0x01	0x08	0x0e

->返回数据为所发送的数据（变送器->上位机）

发送该指令后。随后查询标定状态（为 0xc0 时成功）。直到加载点标定成功后，再进行以下操作。如果标定失败，则需要检查连线或者使用环境是否会造成称重不稳定。

7、生效校准数据（上位机->变送器）【01 06 00 15 00 01 59 ce】

	设备号	功能码	起始地址高位	起始地址低位	2 字节操作数据位		CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x15	0x00	0x01	0x59	0xce

发送该指令完成标定操作，保存标定数据。

->返回数据为所发送的数据（变送器->上位机）

在零点标定、加载标定都成功后。发送该指令。保存当前标定数据。

8 标定状态查询（上位机->变送器）【01 03 00 16 00 01 65 ce】

	设备号	功能码	起始地址高位	起始地址低位	寄存器数高位	寄存器数低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x03	0x00	0x16	0x00	0x01	0x65	0xce

->返回 16 位状态位（变送器->上位机）

	设备号	功能码	数据位数	数据个数 2 个字节	CRC16 低位	CRC16 高位
上位机接收	0x01	0x03	0x02	2 字节状态	xx	xx

(1)状态位表示

15-4	3	2	1	0
保留	0:加载点无效 1:加载点有效	0:零点无效 1:零点有效	0:称重状态 2:校准成功	1:正在校准 3:校准失败

（五）、其它功能

1 获取传感器输出内码【01 03 00 17 00 02 74 0f】

	设备号	功能码	起始地址高位	起始地址低位	寄存器数高位	寄存器数低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x03	0x00	0x17	0x00	0x02	0x74	0x0f

->返回 32 位内码值（变送器->上位机）

	设备号	功能码	数据位数	数据个数 4 个字节	CRC16 低位	CRC16 高位
上位机接收	0x01	0x03	0x04	AD 输出值	xx	xx

该值用于判断传感器性能好坏或者是否接线有问题

2 清除按键标志【01 06 00 19 00 01 99 cd】

	设备号	功能码	起始地址高位	起始地址低位	数据高位	数据低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x19	0x00	0x01	0x99	0xcd

->返回数据为所发送的数据（变送器->上位机）

3 复位变送器【01 06 00 1A 00 01 69 cd】

	设备号	功能码	起始地址高位	起始地址低位	数据高位	数据低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x1A	0x00	0x01	0x69	0xcd

置 1 后，变送器将重新启动。

->返回数据为所发送的数据（变送器->上位机）

4 保存用户参数【01 06 00 3A 00 00 xx xx】

	设备号	功能码	起始地址高位	起始地址低位	数据高位	数据低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x06	0x00	0x3A	0x00	0x00	xx	xx

可用于记录用户需要保存的数据，如单位保存。比如组网的称重设备中有 0.00kg 也有 0.00g 的称重，此时可以用该位标记来区别。变送器本身不支持单位，看用户选择的传感器和标定时选用的砝码而定。

->返回数据为所发送的数据（变送器->上位机）

5 获取当前变送器的版本【01 03 00 3B 00 01 f5 c7】

	设备号	功能码	起始地址高位	起始地址低位	寄存器数高位	寄存器数低位	CRC16 低位	CRC16 高位
上位机发送	0x01	0x03	0x00	0x3B	0x00	0x01	0xf5	0xc7

返回的数值为当前的变送器的版本号。用于方便软件升级。

->返回数据为所发送的数据（变送器->上位机）

十、其它说明

量程、精度、分度值的设置需要根据传感器合理的设置。

CRC 校验

1、软件计算

可以搜索在线生成 CRC，在网页上生成。



需要注意高低位，01 03 00 00 00 02 C4 0B

2、C 函数

以下函数来自 FREEMODBUS。采用查表方式计算 CRC16。

```
static const UCHAR aucCRCHi[] = {
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
```

```
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x00, 0xC1, 0x81, 0x40,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40, 0x01, 0xC0, 0x80, 0x41, 0x01, 0xC0, 0x80, 0x41,
    0x00, 0xC1, 0x81, 0x40
};

static const UCHAR aucCRCLo[] = {
    0x00, 0xC0, 0xC1, 0x01, 0xC3, 0x03, 0x02, 0xC2, 0xC6, 0x06, 0x07, 0xC7,
    0x05, 0xC5, 0xC4, 0x04, 0xCC, 0x0C, 0x0D, 0xCD, 0x0F, 0xCF, 0xCE, 0x0E,
    0x0A, 0xCA, 0xCB, 0x0B, 0xC9, 0x09, 0x08, 0xC8, 0xD8, 0x18, 0x19, 0xD9,
    0x1B, 0xDB, 0xDA, 0x1A, 0x1E, 0xDE, 0xDF, 0x1F, 0xDD, 0x1D, 0x1C, 0xDC,
    0x14, 0xD4, 0xD5, 0x15, 0xD7, 0x17, 0x16, 0xD6, 0xD2, 0x12, 0x13, 0xD3,
    0x11, 0xD1, 0xD0, 0x10, 0xF0, 0x30, 0x31, 0xF1, 0x33, 0xF3, 0xF2, 0x32,
    0x36, 0xF6, 0xF7, 0x37, 0xF5, 0x35, 0x34, 0xF4, 0x3C, 0xFC, 0xFD, 0x3D,
    0xFF, 0x3F, 0x3E, 0xFE, 0xFA, 0x3A, 0x3B, 0xFB, 0x39, 0xF9, 0xF8, 0x38,
    0x28, 0xE8, 0xE9, 0x29, 0xEB, 0x2B, 0x2A, 0xEA, 0xEE, 0x2E, 0x2F, 0xEF,
    0x2D, 0xED, 0xEC, 0x2C, 0xE4, 0x24, 0x25, 0xE5, 0x27, 0xE7, 0xE6, 0x26,
    0x22, 0xE2, 0xE3, 0x23, 0xE1, 0x21, 0x20, 0xE0, 0xA0, 0x60, 0x61, 0xA1,
    0x63, 0xA3, 0xA2, 0x62, 0x66, 0xA6, 0xA7, 0x67, 0xA5, 0x65, 0x64, 0xA4,
    0x6C, 0xAC, 0xAD, 0x6D, 0xAF, 0x6F, 0x6E, 0xAE, 0xAA, 0x6A, 0x6B, 0xAB,
    0x69, 0xA9, 0xA8, 0x68, 0x78, 0xB8, 0xB9, 0x79, 0xBB, 0x7B, 0x7A, 0xBA,
    0xBE, 0x7E, 0x7F, 0xBF, 0x7D, 0xBD, 0xBC, 0x7C, 0xB4, 0x74, 0x75, 0xB5,
    0x77, 0xB7, 0xB6, 0x76, 0x72, 0xB2, 0xB3, 0x73, 0xB1, 0x71, 0x70, 0xB0,
    0x50, 0x90, 0x91, 0x51, 0x93, 0x53, 0x52, 0x92, 0x96, 0x56, 0x57, 0x97,
    0x55, 0x95, 0x94, 0x54, 0x9C, 0x5C, 0x5D, 0x9D, 0x5F, 0x9F, 0x9E, 0x5E,
    0x5A, 0x9A, 0x9B, 0x5B, 0x99, 0x59, 0x58, 0x98, 0x88, 0x48, 0x49, 0x89,
    0x4B, 0x8B, 0x8A, 0x4A, 0x4E, 0x8E, 0x8F, 0x4F, 0x8D, 0x4D, 0x4C, 0x8C,
    0x44, 0x84, 0x85, 0x45, 0x87, 0x47, 0x46, 0x86, 0x82, 0x42, 0x43, 0x83,
    0x41, 0x81, 0x80, 0x40
};

USHORT
usMBCRC16( UCHAR * pucFrame, USHORT usLen )
{
    UCHAR                 ucCRCHi = 0xFF;
    UCHAR                 ucCRCLo = 0xFF;
    int                   iIndex;
    while( usLen-- )
    {
        iIndex = ucCRCLo ^ *( pucFrame++ );
        ucCRCLo = ( UCHAR )( ucCRCHi ^ aucCRCHi[iIndex] );
        ucCRCHi = aucCRCLo[iIndex];
    }
    return ( USHORT )( ucCRCHi << 8 | ucCRCLo );
}
```

SJ101CX 测试工具使用

校准前工作

使用该软件需要注意的是。在模块第一次校准时称重值会上下波动。此时称重值可作为传感器接线是否正确。通过放物体到传感器上可看到值明显的变化。如果值没有明显的变化通常传感器接线没有接好。放物体后值如果是变小通常是信号线接反或者传感器装反了。未校准时值的变化不能判断称重是否稳定需要通过标志（窗口上红色的圆圈常亮表示稳定）只要显示稳定，且放东西在传感器上值会变大即可进行校准。

显示-----表示开机时未进入到称重状态。等待进入即可

显示 OUEr 表示超载（未校准时可能稍为按一下传感器便会显示）。可按清零返回零点。



1、 打开软件后设置好端口(我的电脑设备管理器中查看)、波特率、校验位和站号。

2、 设置成功后打开串口，软件后以 10/s 的速度向 SJ101CX 模块发送获取重量指令。如站号为 1 时发送的数据【01 03 00 00 00 04 44 09】，获取 4 个寄存器值，获取重量、精度、状态数据。

校准：

如:　设置满量程为 100kg 精度 10g(100.00）

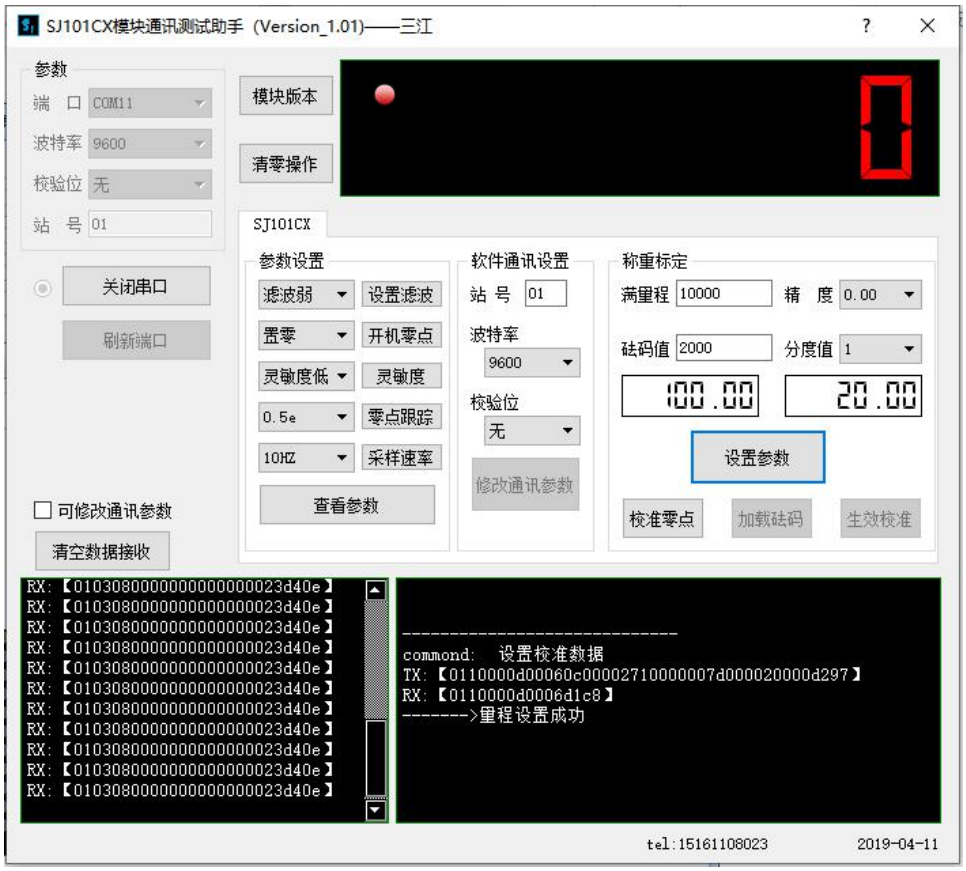
已知重量的砝码值为 20kg

站号为 1

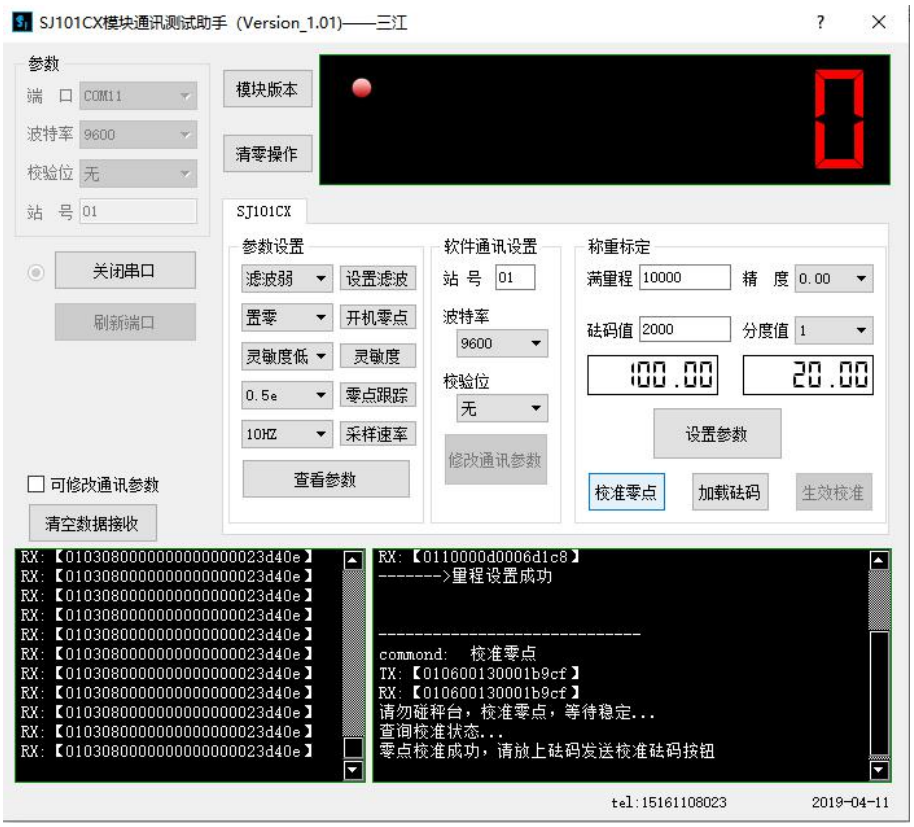
校准操作如下：

1、清空秤台，并不要碰触秤台。

2、设置满量程为 10000　精度设置为 0.00　砝码值设置为 2000，分度值为 1 后点击【设置参数】，等待返回状态显示成功。



3、 空载时点击【校准零点】，并等待校准成功。【加载砝码】按键有效。



4、 放上 20kg 砝码到称量台中，点击【校准砝码】按钮。并等待校准成功。



5、 返回成功状态后，此时【生效校准】按键有效。点击即完成校准。



注意：如指令发送超时请检查通讯参数或者硬件连接。