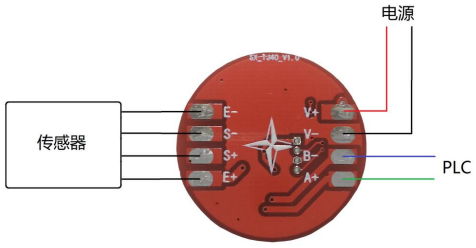
Embedded Digital Weighing User Manual

Feature

- Embedded micro digital transmitter
- A stable accuracy of 1/10,000
- Wide voltage range, 7-26VDC wide voltage operating range
- Multiple optional communication interfaces: RS485, RS232, TTL

1. Installation and wiring

1.1 Appearance



1.2 Interface Introduction

Name	Function
E+	Excitation signal+, It is generally red
E-	Excitation signal-, It is generally black
S+	Signal +, It is generally green
S-	Signal -, It is generally white
V+	The power supply for the transmitter is connected to a 7-26V DC power supply
V-	
A+	The RS485 communication interface is used to connect the upper computer PLC or computer, usually A to A, B to B, or "+" to "+", "-" to "-".
B-	

Definition of Indicator Light Function:

Name	Function
LED	It remains on by default after power-on, flashes during communication, and remains on again after communication stops

2. Communication interface

The module provides a USB interface that supports Modbus-RTU for external communication. The default communication parameters at the factory are as follows:

	Bautrate	protocol	Party	Stop bit
default	9600	Modbus-RTU	No Party	1

The instruction codes supported by the module in the Modbus protocol include:

03: Read multiple register
06: Write signal register
16: Write multiple register

3. Slave ID

The module uses Register No. 10 to set the station address. Modifying the value of this register can achieve the modification of the station address. The communication station address when the module leaves the factory is: 1

4. Detailed Explanation of Registers

All the Modbus registers inside the module are located in the 03 instruction code area (hold registers), with register addresses starting from 0 and a total of 30. The register allocation table is as follows:

Address	Name	Function
0,1	Real-time weight	The current real-time weight value, a 32-bit signed integer
2,3	Real-time voltage	The current sensor signal voltage value, 32-bit signed
4,5	Zero voltage	The voltage value corresponding to the zero point of the sensor during calibration
6,7	Sensitivity	The sensitivity value corresponding to the sensor weight point during calibration
8,9	The value displayed by the weights	The displayed value of the weights written during calibration
10	Slave ID	Set Slave ID, range:1-250
11	Baudrate	The current communication baud rate of the module,range:9600,19200,38400
12	Filtering length	The average filtering times are set within the range of 1 to 30, with the factory default value being 3. The larger this number is, the more stable the display will be, but the slower the dynamic response speed will be
13	--	
14	--	
15	--	
16	ADC Speed	Set the collection speed of the module. The factory default is 10Hz. 0:10Hz 1:40Hz This register takes effect immediately after being written
17	--	
18	Calibration cmd	Write instruction code to this register to trigger the corresponding calibration operation: 1:Zero point calibration (zeroing) 2: Calibration of weight points The written values do not need to be reset to zero; the module will reset automatically
19	Reset factory	Writing the numbers 12345 to this register can trigger the operation of restoring the module to factory Settings

5. Communication Programming methods

5.1 Software development for PC

For users developing software on Windows, Linux, Android, and other series of operating systems, it is first necessary to understand the control invocation and data sending and receiving operations of serial port devices, and then send and receive according to the instructions given in this chapter, ultimately achieving the data reading and writing functions of the module's registers

1. **Read the current weight, computer -> module**
   01 03 00 00 00 02 C4 0B
   01: Slave ID
   03: Read register command
   00 00: from register 0
   00 02: quantity 2
   C4 0B: Modbus CRC16
   Examples of module responses:
   01 03 04 E2 40 00 01 0C 5F
   01: Slave ID
   03: Read Register command
   04: quantity 4 byte
   E2 40 00 01:Four bytes of valid data can be converted to decimal as 123456. The conversion method is as follows: First, swap the high and low characters to become: 00 01 E2 40. Then, assign this value to a 32-bit signed number to directly print out 123456, or use a hexadecimal to decimal conversion tool such as a calculator to obtain 123456. Note: If the module reading is negative at this time, the output will be two's complement.
   0C 5F: Modbus CRC16
2. Zero-point calibration, computer -> module
   01 06 00 12 00 01 E8 0F
   01: Slave ID
   06: Write signal register command
   00 12: regisger Address 18
   00 01: write value 1
   E8 0F: Modbus CRC16
3. **Calibration of weight points**
   01 10 00 08 00 02 04 03 E8 00 00 72 79
   01: Slave ID
   10: Write multiple register
   00 08: form address 8

00 02：quartity 2 register

04：data lens 4 bytes

03 E8 00 00：The written value is converted to decimal to 1000. The conversion method is the same as the parsing method for the returned value within the read instruction segment.

6. Calibration method

This module supports the two-point calibration method for calibrating the module, namely the zero point and the weight point. Before performing the calibration operation, users need to determine the following key points first:

①：The sensor has been correctly installed and mechanically fixed firmly

②：The sensors, power supplies, communication and other wires of the module have been correctly connected

③：A heavy object with a known precise weight needs to be prepared as the calibration weight (the weight of the weight should not be less than one-tenth of the total range of the sensor).

7.1 Calibrate the module using the test software：

1：When the load on the scale pan/hook is 0, click the "Zero " button

2：Place the weight on the scale pan/hook, fill in the corresponding value of the weight in the "Set Value", note that only integers are supported. Then click the "Calibration" button, and the calibration is completed.

3：Repeatedly test with other heavy objects of known weights to show whether the weight is accurate.

7.2 Calibrate the module using configuration software or other upper computers：

1：When the load on the scale pan/hook is 0, write the number 1 to register 18

2：Place the weight on the scale pan/hook, write the corresponding display value of the weight into register No. 8, and the calibration is completed.

3：Repeatedly test with other heavy objects of known weights to show whether the weight is accurate

Attention：

①For any write program in the calibration operation, edge-triggered single write or trigger time within 1 second should be adopted. Do not allow the software to continuously send calibration instructions for a long time

②When debugging and calibrating the program for the first time on industrial control software, it is necessary to debug and read the program first. We strongly recommend that you calibrate the module with the test software first, and then debug and read the program with the PLC. Only when the software can display the same values as the official test software in both positive and negative numbers can you debug and calibrate the program

③The calibration values and real-time weight values of the module are both double-word registers, and the industrial control software needs to read and write according to the double-word format.

7. FAQ

Ask	Answer
How is the output number of the module converted into the actual weight value?	When the module is used for the first time after leaving the factory, the output value is incorrect. It must be connected to a sensor for calibration operation before the correct weight is output. The output number of the module is the actual weight and does not need to be converted
What is the digital unit of real-time weight?	The unit of the output number of the module is consistent with the display value of the weight filled in during calibration. For example, if a 10kg weight is used during calibration and the calibration value is filled in as 1000, then the output value 1000 of the module after calibration represents 10kg
Why does the test software show calibration failure, or the displayed value of the PLC after calibration is not the calibrated value?	It might be caused by the weight of the weights being too light during calibration. We suggest that the weight of the weights used during calibration should not be less than one-tenth of the total range of the sensor. The module will make a judgment on the calibration signal internally. If the weight is too small, calibration cannot be carried out or the default value will be written
I have a large-range sensor of XX tons. What should I do if I don't have such a weight-bearing weight?	t is not necessary to use standard weights. Heavy objects such as trucks and forklifts can also be used as temporary weights after precise weighing. If precise weighing is required, an object of sufficient weight must be used for calibration. If precise weighing is not necessary, calibration without weights can be used
What does it mean when the module outputs a very large weight value like FF FF FF FE?	Large numbers starting with "FF" are caused by the module's reading being negative. The negative values of the module are output in the form of two's complement (What is two's complement and how to convert it to a negative number: It is recommended to search for relevant explanations and methods using online resources).
What could be the reason for the test software showing that the signal is out of range?	Please remove all the sensor wiring first and then conduct the test. If the modules work properly after removing the sensors, please carefully check the wiring sequence of the sensors and whether there is a short circuit at the connection points

8. Working parameters

Project	parameters
Power supply	7-26VDC
Port	RS485
Power consumption	<1W（1x350Ω Sensor） <1.5W(4x350Ω Sensor)
Working temperature	-20 ~ 50℃
Working humidity	35~85%R·H
Module weight	≈0.1kg
Sensor excitation voltage	DC5V±0.1V 200mA(MAX)
ADC	24bit
Stable resolution	1/10000
Acquisition speed	10Hz / 40Hz (DFU-10Hz)
Nonlinearity	±0.05%F·S
Protocol	Modbus-RTU
baud rate	9600(默认),19200,38400
Party	No

9. Size