DORGEAN
Beijing Dorgean Electronic Technology Co.,Ltd

Instrument Expert
Original factory packaging
www.dorgean.com

KEANLEY

TRS-1203

SDI-12 Interface Protocol Converter/User Manual

Beijing Keanley Technology Co.,Ltd



For more informations, please visit www.keanley.com

CONTENTS

01 . What are the characteristics/application scenarios of SDI-12/TRS-1203..... 1

02 . Description of interface indication..... 2

03 . Devices connection diagram..... 3

04 . Basic parameters and dimensions..... 4

05 . Absolute maximum rating/electrical characteristics..... 5

06 . Introduction of SDI-12/RS232/RS485..... 6

07 . Set TRS-1203..... 7

08 . Introduction of obtaining and using SDI-12 Interface conversion software 8

09 . Modbus-RTU mode..... 9

10 . Obtain corresponding sensor data via Modbus..... 11

11 . Obtain analysis of corresponding sensor data via Modbus..... 13

12 . Commonly used function codes of Modbus..... 14

13 . Exception codes of Modbus..... 15

14 . Optional SDI-12 transparent transmission protocol..... 16

15 . Start sensor measurement in SDI-12 mode..... 21

16 . Obtain sensor data in SDI-12 mode..... 22

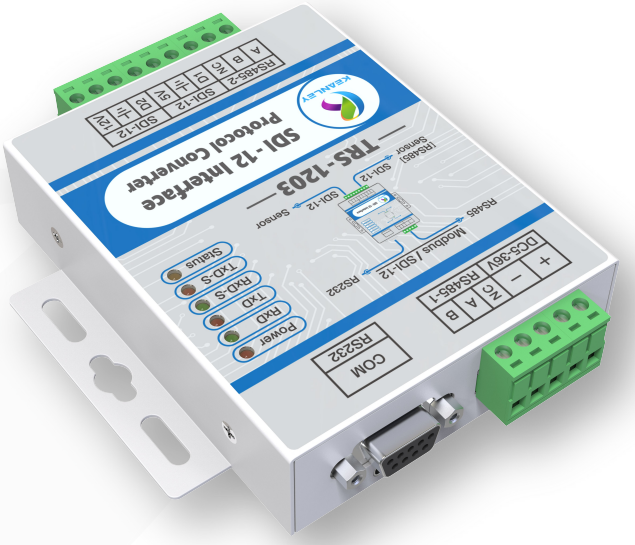
17 . Recommended SDI-12 circuit..... 24

18 . Commands and replies of SDI-12..... 25

19 . Connectable sensor brands..... 27

What is SDI-12?

SDI-12 (Serial Digital Interface) is a serial digital interface with a baud rate of 1200. The SD-12 communication standard, a serial data communication interface protocol widely applied in environmental monitoring around European and American countries in recent years, is now widely used in multi-parameter measurement and control of industry and agriculture, hydrological and meteorological monitoring of rivers, lakes, and seas, as well as in animal husbandry and food production, through which data can be transmitted over long distances.



TRS-1203 Interface Protocol Converte

Features of TRS-1203

- Low energy consumption.
- Using single or multiple sensors in one cable.
- Rich variety of communication interfaces.
- Multiple indicator lights of operational communication status.
- Long-term unmanned operations.

Application scenarios

The SDI-12 Interface Protocol Converter, an advanced device widely used in fields such as agriculture, environmental monitoring, meteorological research, and hydrological measurement, etc., has features of high reliability, stable performance, and easy operation, which can effectively convert data between SDI-12 interface and other interfaces, providing users with a convenient and fast user experience.

In the field of agriculture, the SDI-12 Interface Protocol Converter can convert the soil moisture, temperature, gas concentration and other data measured by sensors into general analog or digital signals for output through the SDI-12 interface, so that users can better control the growth environment of crops and improve yield and quality. In addition, it can also be integrated into irrigation systems, automatically adjusting the irrigation amount based on soil moisture conditions, thus, achieving intelligent agricultural management.

In terms of environmental monitoring, the SDI-12 Interface Protocol Converter can output measurement data from various environmental parameter sensors through the SDI-12 interface, including air temperature and humidity, air pressure, wind speed and direction, to meet the needs of environmental monitoring. For example, in meteorological research, researchers can integrate data from various meteorological parameter sensors into the meteorological station system via this product, accurately monitoring and predicting meteorological changes, and providing convenience for meteorological forecasting.

In addition, the SDI-12 Interface Protocol Converter also plays an important role in the field of hydrological measurement. By converting the data measured by sensors such as water level, flow rate, and rainfall into SDI-12 interface outputs, real-time transmission and remote monitoring of hydrological observation data can be achieved, which is of great significance for water resource management, flood warning, and river flow regulation, etc.

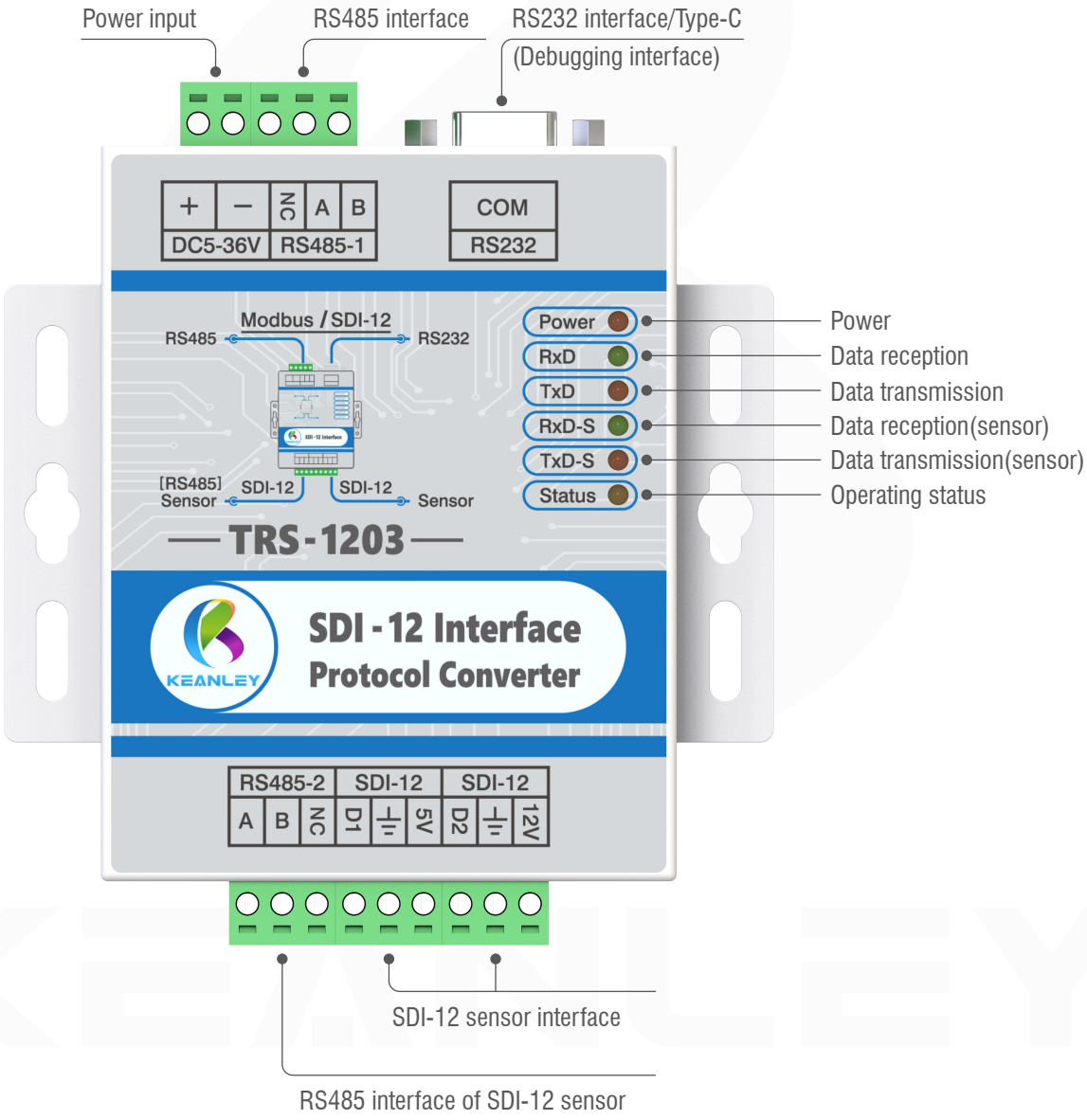
The SDI-12 Interface Protocol Converter is simple to operate. Data transmission and processing can be achieved only by connecting the sensor to the converter which is connected to a data acquisition system or a computer. Meanwhile, it supports data recording and storage functions, allowing users to query historical data at any time and perform data analysis and comparison. In addition, various output interface options are provided, including RS485, RS232, USB, etc., so that users can choose the interface according to their actual needs.

In summary, the SDI-12 Interface Protocol Converter, a high-performance, reliable, and easy to operate device, could be widely applied in fields like agriculture, environmental monitoring, meteorological research, and hydrological measurement. Fast and accurate data transmission and analysis processing can be achieved by seamlessly integrating with various sensors, providing users with a comprehensive solution. No matter you are an agricultural producer, an environmental researchers, or a hydrological observation professionals, you can definitely rely on this powerful device to better carry out your work and achieve better results.

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

info@keanley.com

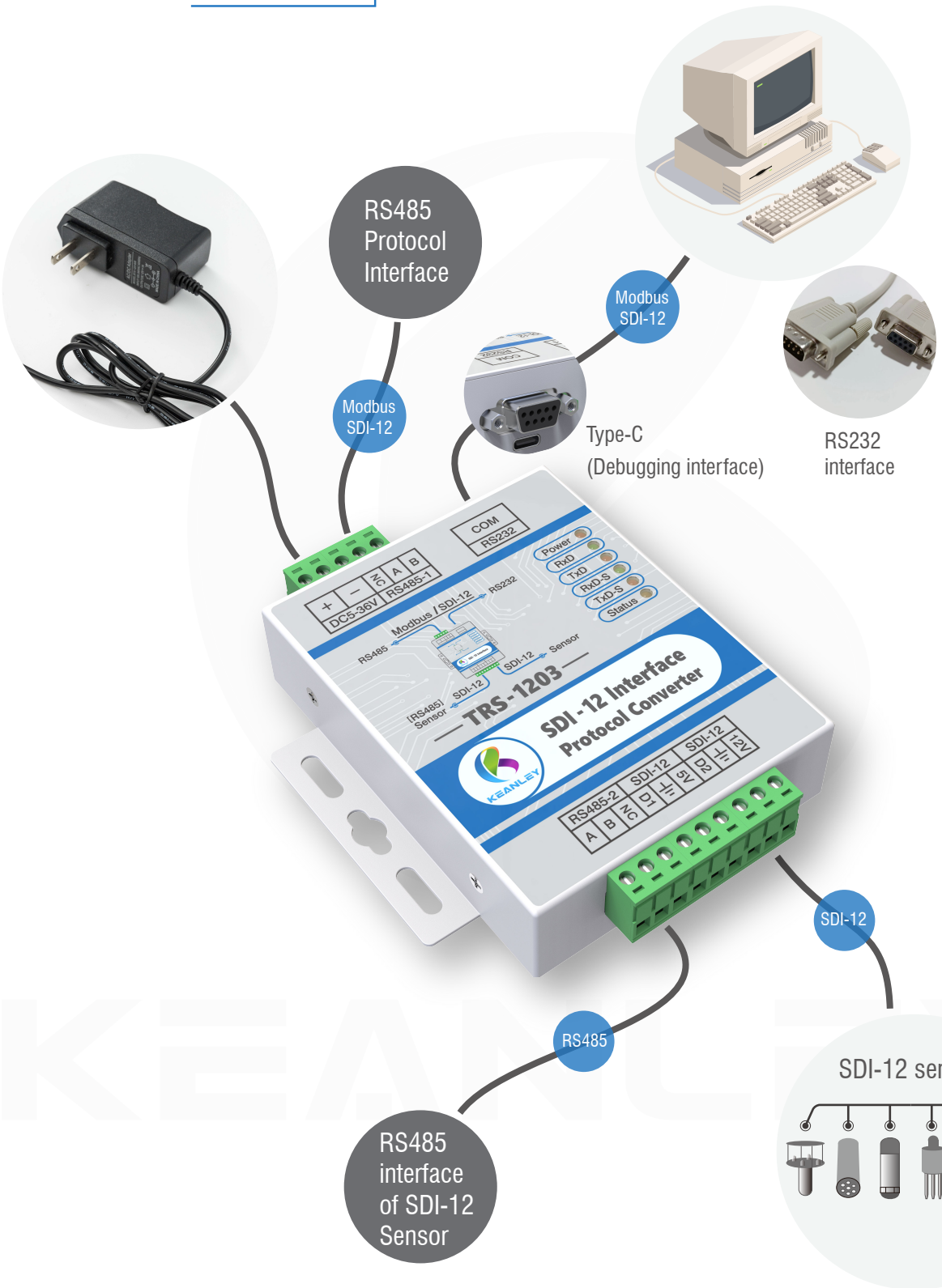
Description of interface indication



Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

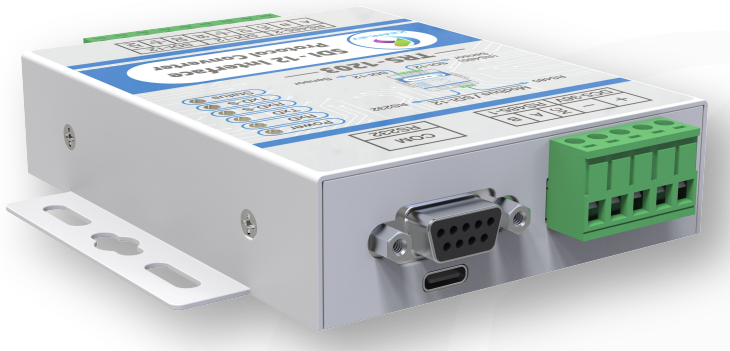
info@keanley.com

Devices connection diagram



Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

info@keanley.com

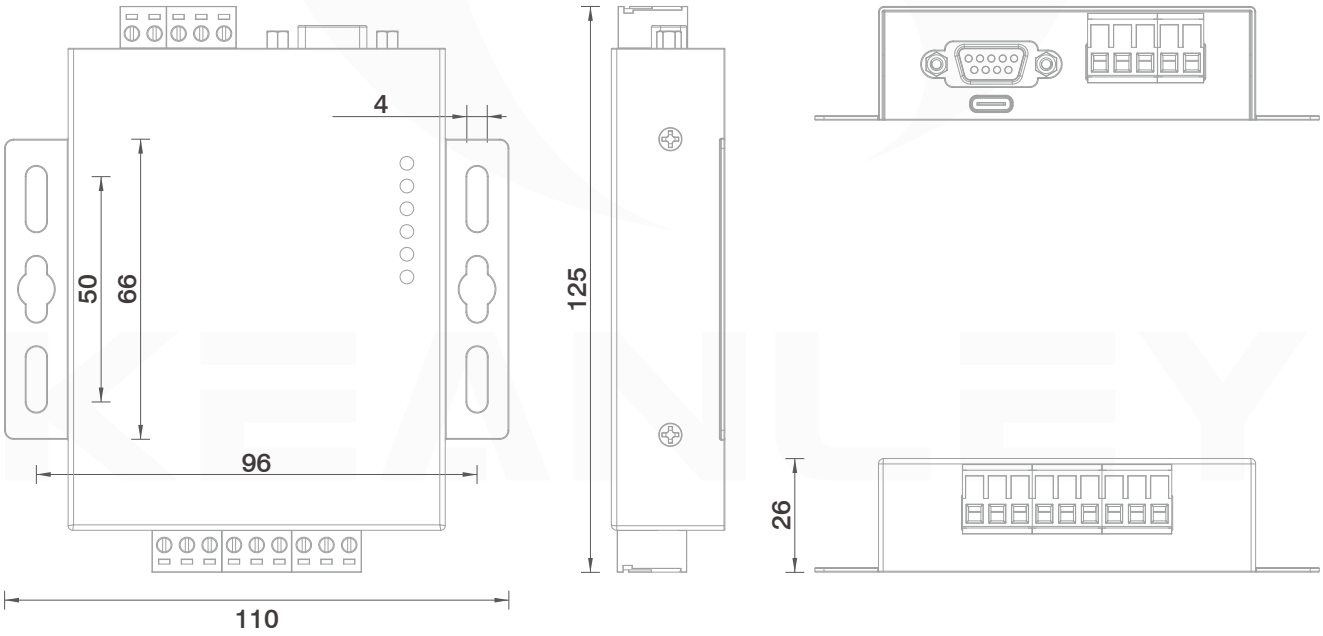


Basic parameters

- Input voltage：DC5V-36V
- Static power consumption：＜30mA@12V
- Transmission distance：1200 meters
- Baud rate：1200 or 9600(Optional)
- Communication protocol: Standard Modbus or SDI-12 transparent transmission
- Working temperature：- 40℃~70℃
- Function：SDI-12 to RS232/RS485/USB
- Sensor Interface：SDI-12 and RS485
- Communication Interface：RS232/RS485/USB

Dimensions

Unit：mm



Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

info@keanley.com

Absolute maximum rating/electrical characteristics



Figure 1. TRS-1203 interface block diagram

Stress exceeding the limits listed in the table below may lead to permanent failure. Long term exposure to absolute ratings may affect equipment reliability. The limit values comply with the absolute maximum rating system (IEC 134). All voltages are based on the board GND.

Symbol	Parameter	Condition/Status	MIN	MAX	UNIT
	SDI-12 D1/D2		-0.3	5.1	V
	SDI-12/5V POWER		4.8	5.2	V
	SDI-12/12V POWER		11.2	12.1	V
	DC 5-36V		4.6	37	V
VES	Electrostatic treatment		-	2000	V

Table 1. Absolute maximum rating

Temperature TA=20℃, Load electrical characteristics.

Symbol	Parameter	Condition/Status	MIN	TYP	MAX	UNIT
SDI-12 Data Terminal						
VIL	Input low voltage level		-0.3	0.0	0.8	V
VIH	Input high voltage level		2.8	3.6	5.0	V
VOL	Output low voltage level		-0.3	0.0	0.8	V
VOH	Output high voltage level		4.5	4.8	5.1	V
SDI-12 Power Terminal （5V）						
V_SDI_Out	Sensor power supply voltage		4.9	5	5.1	V
I_SDI_Out	Sensor power supply current		-	200	-	mA
SDI-12 Power Terminal （12V）						
V_SDI_Out	Sensor power supply voltage		11.8	12	12.1	V
I_SDI_Out	Sensor power supply current		-	200	-	mA
Working temperature range			-40 ~ 70			℃

Table 2. Electrical characteristics

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

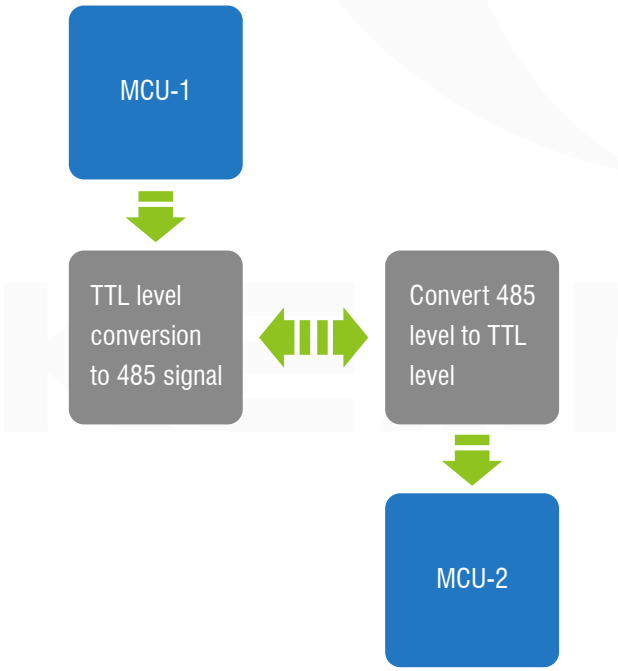
info@keanley.com

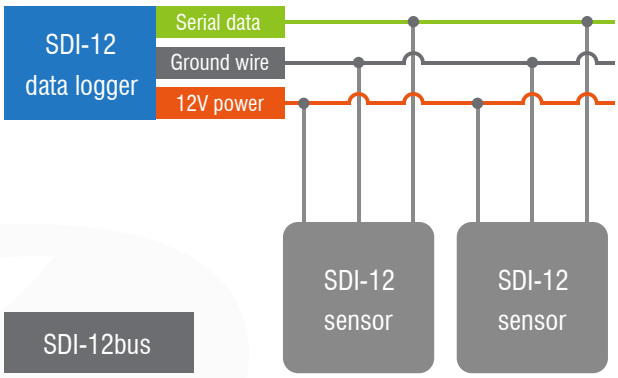
Introduction of SDI-12

SDI-12 is specifically designed for communication between sensors and data loggers. It fully defines the hardware, network and application layers. It has the function of assigning addresses to sensors and synchronously and asynchronously requesting readings. Any sensor with SDI-12 enabled can be connected to any data logger with SDI-12 enabled. The only practical limitations are the power consumption and voltage of the sensor.

The SDI-12 bus has a common power supply cord for all sensors. The comprehensive power consumption should not exceed the capacity of the data logger. All sensors should be able to work within the power supply cord voltage. If the power supply cord of the data logger can't meet the power requirements of the sensor, an external power source can be introduced. However, this will make the system more complicated and less practical. SDI-12 is a master-slave network. Each sensor has a unique address. There can be up to 62 different addresses (0.1, a.. z, a.. z) with a maximum cable length of approximately 70-100 meters.

The most unique feature of this protocol is its sleep and wake-up mechanisms identified as 12 ms and 8.33 ms. Due to the low power consumption mode of the SD1-12 sensor, it enters sleep mode when not in operation. Therefore, a 12 ms identification signal needs to be added before each command to wake up all sensors on the bus. After the sensor wakes up from the low power mode and receives the next 8.33 ms identification signal, it begins to search for a matching address, and the address matching begins to measure. If the address is invalid or idle for over 100 ms, it will return to low-power consumption standby mode. Note: This feature is automatically executed by the data logger or SDI-12 protocol converter.





SDI-12 has the following advantages over other commonly used communication standards:

1. The hardware structure is simple. Only one power supply cord and one data cable are required. And the level definition of the data cable is basically the same as that of the MCU, without the need for complex conversion circuits.
2. SDI-12 has a hardware wake-up function, which requires a wake-up operation before each communication. Therefore, SDI-12 sensors can enter sleep mode when not awakened to reduce sensor power consumption.
3. The SDI-12 protocol commands are in readable text format and the command structure is simple, making it easy for on-site technicians to directly use the console for debugging, as shown in the above figure.

Introduction of RS232 and RS485

The RS232 interface, complying with the interface standards for serial data communication established by the Electronic Industries Association of the United States (EIA), is widely used to connect computer serial interface peripherals.

The working mode of RS232 is single ended, which is an unbalanced transmission mode. The logical level of the signal at the receiving and transmitting ends is relative to the signal ground. RS232 was originally used for one-on-one communication between DET (Digital Terminal Equipment) and DCE (Data Communication Equipment), which is point-to-point. It is generally used for duplex transmission, and half duplex transmission as well. In addition, RS232 is negative logic, with a logic level of ±5～±15 V. The transmission distance is only limited to 15 meters. which can reach up to 50 meters in practical applications.

To improve the disadvantages of short communication distance and low speed of RS232, EIA has developed the RS485 interface standard based on RS422. RS485 is a balanced transmission and differential reception standard, therefore it has the capability to suppress common mode interference. Its maximum transmission distance is 1200 meters, which can reach up to 3000 meters in practical applications. And the transmission rate can reach up to 10 Mbit/s. RS485 adopts a half duplex working mode, allowing for multi-point and bidirectional communication on a simple pair of shielded twisted pairs. However, only one point can be in the transmitting state at any time. Therefore, the transmitting circuit must be controlled by an enable signal. Generally, master-slave communication is adopted, which means one host carries multiple slaves, as shown in the figure on the left.

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

info@keanley.com

Set TRS-1203

1. TRS-1203 Converter Free Optional Combination and Customer Customization

(check relevant parameters for reference device selection)

■ Device mode baud rate selection, such as selecting SDI-12 mode with a baud rate of 9600Bps.

Product model	Mode selection	Baud rate selection	Protocol customization
TRS-1203	Mode ☑ SDI-12 ☐ Modbus	Baud Rate ☑ 9600 Bps ☐ 1200 Bps	Standard protocols & customer

2. Hardware connection of TRS-1203 converter.

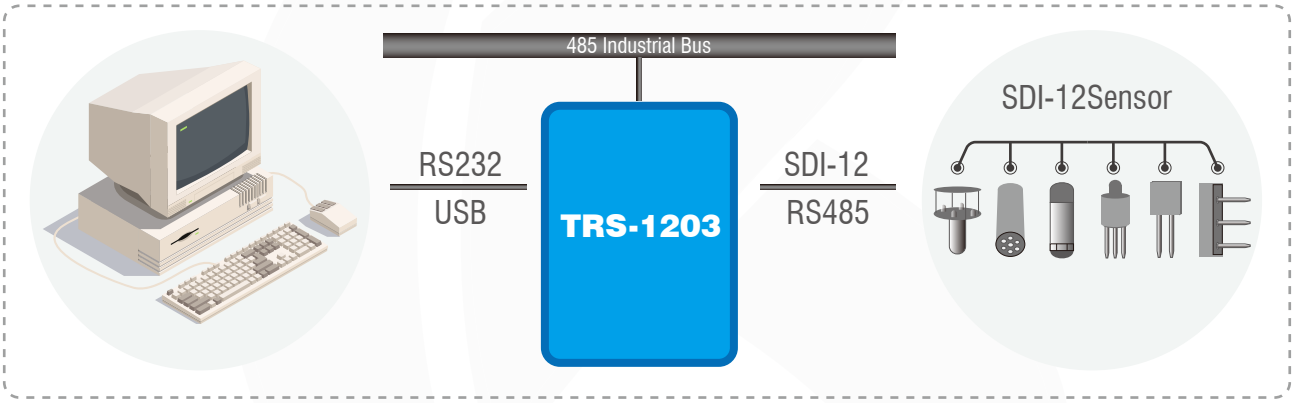


Figure 2 . Converter hardware connection diagram

TRS-1203 has three hardware communication interfaces.The first two communication methods are generally used.

■ RS232 is used to connect computers or industrial control computers with COM interfaces. 232 hardware port has power off control function(DB9 male and female plugs are plugged into cables to connect PC and TRS-1203).

■ Type-C interface is used to connect a computer's USB for data transfer (USB end connected to computer, Type-C end connected to TRS-1203).

■ RS485 interface is used to connect industrial bus devices (A connected to A, B connected to B).

There are two hardware connection methods for TRS-1203 and sensors, SDI-12 and RS485.

■ SDI-12 interfaces connect various types of corresponding sensors.

■ RS485 interfaces connect RS485 interface sensors.

3. TRS-1203 converter has two modes of communication configuration.

■ The default communication interface number is COM1.You can click to open or close this port in the serial port tool to achieve the power on and off function of the 1203 device (the communication interface connects the COM interface of device RS232).

■ The specific serial port number can be viewed in the device manager (Type-C port of the USB connected to the communication interface).The communication configuration is shown in Table 3.

Communication Configuration				
Start bit	Data bits	Parity bit	Stop bit	Baud rate
1	8	None	1	1200/9600 (optional)

Table 3. Communication configuration

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

4. Introduction of obtaining and using SDI-12 interface conversion software.

The SDI-12 interface conversion software is specifically developed for communication debugging of SDI-12 series products. It has a layout setup interface by region, simple and easy to use settings, and meets all operations of TRS-1203 products.

■ The download link of SDI-12 interface conversion software and USB driver (CH340) is listed in the following box.

Download link of serial port debugging software driver (Chinese/English) and USB driver (CH340):
http://www.keanley.com/download.html

■ Double click to open the downloaded file “SDI-12 interface conversion software.exe” (Using RS232 interface by default)as shown in Figure 3 below.

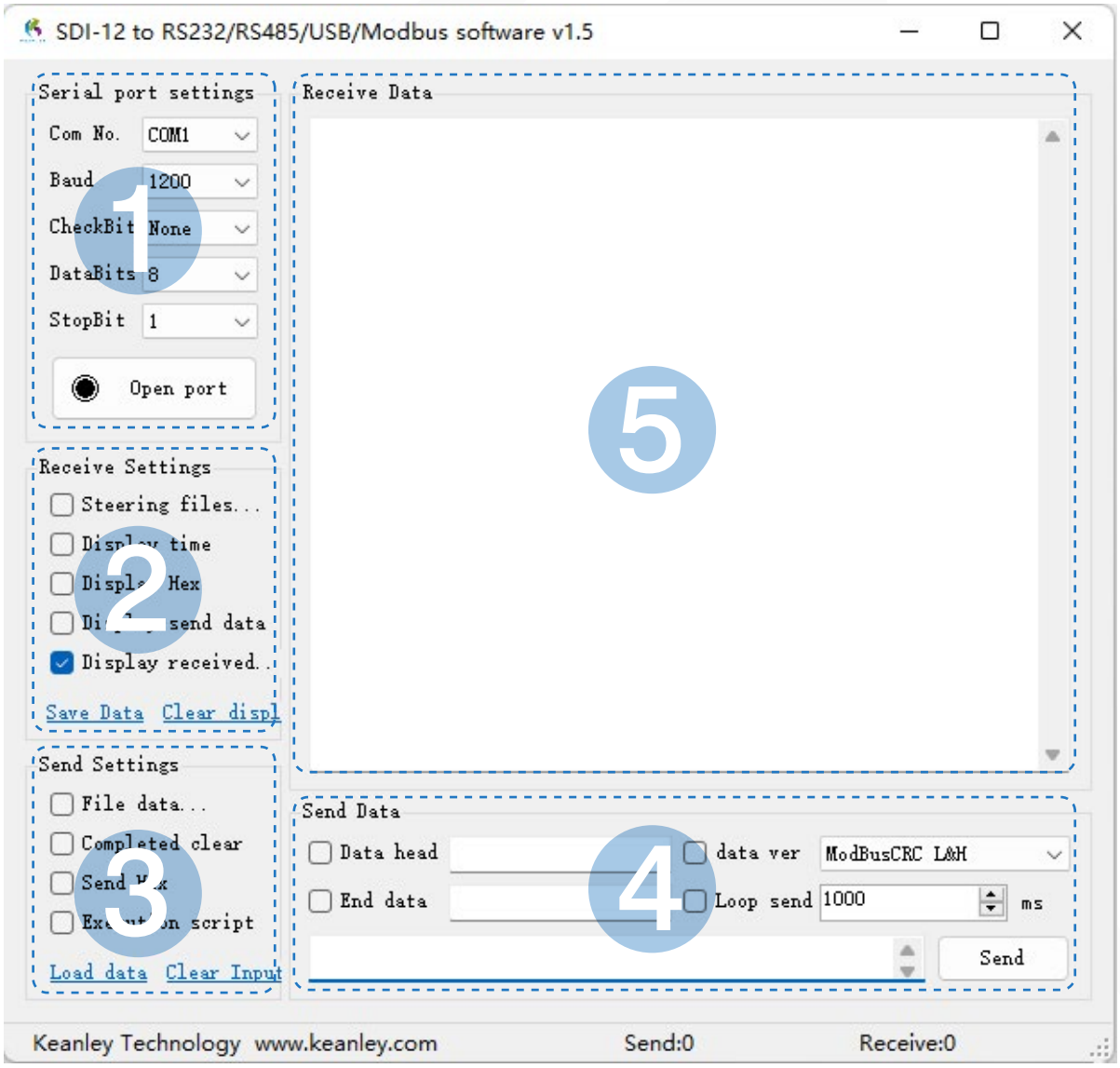


Figure 3. Software initial interface

As shown in Figure 3, the interface conversion software is divided into five regions.

■ Please refer to Table 3 on page 7 for the serial interface settings.

■ Check the SDI-12 mode in the setting area of receiving in order of 2-4-5 from top to bottom, and check 2-4-5 for Modbus.

■ No need to check when sending the SDI-12 mode in the settings area. Check the third option for Modbus mode.

■ To send the SDI-12 of data area , only input the response command in the sending bar. No need to check the rest. In Modbus mode, check the data verification and select “ModbusCRC low byte first”.

■ The receiving data area will display the current sending data and the instantly returned data.

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

info@keanley.com

Modbus-RTU Mode

1. Introduction of Modbus.

Modbus, a bus protocol standard based on serial communication, adopts master-slave communication, in which the host sends and the slave responds. Its communication process is as follows: there is a host with multiple slaves on a bus, and the host queries one of them. First, addresses must be assigned to these slaves (in order to distinguish which slave it is, and each address must be unique). After assigning an address, the host needs to query. Afterwards, the data is distributed, and the slave receives the data sent by the host. Then, the corresponding address of the slave replies, and the host receives the slave data. This is a communication process from the host to the slave.

2. Modbus-RTU Protocol.

(1). Frame structure.

Address (1 byte)	Function Code (1 byte)	Data (may vary depending on the	Verification

- Address: occupies one byte, ranging from 0 to 255, with a valid range of 1 to 247.
- Function code: occupies one byte, the meaning of the function code is to explain the meaning of the command.
- Data: there are different structures depending on different function codes.
- Verification: the purpose of calculating the transmitted data is to ensure the accuracy and consistency of the transmitted data.

(2). Practical examples.

The most commonly used function codes for the Modbus RTU protocol are 03, 16, and 17. 03 was used mostly to query information on sensors, for example, we use function code 03 to query sensor data.

Host send: 02 03 00 00 00 02 C4 38

02	03	00	00	00	02	C4	38
Current sensor address	Function code (read address)	Starting address		Register data count (2 bytes)		Modbus-CRC16 verification	

Table 4. Host sending Code

- 02 - Address， current sensor address.
- 03 - function Code, 03 represents the query function, which queries sensor data.
- 00 00 - The starting register address representing the query indicates that starting from 0x0000, Modbus stores data in registers and obtains different variables by querying registers.
- 00 02 - Represents querying a register, combined with the previous 00 00, which means querying the values of two registers starting from 0.
- C4 38 - Cyclic redundancy check is a verification formula for Modbus, which verifies data starting from the first letter and ending before C4.

Slave reply： 02 03 04 79 6A 00 B6 71 C5

02	03	04	79	6A	00	B6	71	C5
Sensor address	Function code (read address)	Number of returning bytes	One set of data		Two sets of data		Modbus-CRC16 verification	

Table 5. Slave reply code

- 02 - Address, current sensor address.
- 03 - Function code, 03 represents the query function, which queries sensor data (The function code sent by the host to the slave is returned intact by the slave).
- 04 - Represents the number of bytes in the following data. A register has 2 bytes, so the number of bytes in the following must be 2 times the number of query registers.
- 79 6A - Represents the value of register 01; 00 B6- represents the value of register 02.
- 71 C5 - Cyclic redundancy check, a verification formula for Modbus, verifies data starting from the first letter and ending before 71.

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

If the nameplate at the back of the device is in Modbus mode and 1200 baud rate, please refer to pages 9-15 for communication content. The commonly used function codes in the Modbus protocol include not only the 03 query sensor data introduced earlier, but also reading sensor address 16, modifying sensor address 17, etc. The CRC check is 2 bytes, and the data length is not fixed. Different function codes have different data lengths. Examples of data formats corresponding to function codes 16 and 17 are listed.

■ The host sends the instruction to read the sensor address through TRS-1203 (function code 16)

Transmit：02 16 00 00 00 01 89 FA

02	16	00	00	00	01	89	FA
Arbitrary value	Function code (read address)	Starting address		Read a sensor value		Modbus-CRC16 verification	

Table 6. Read command.

Return：00 16 01 00 EA 39

00	16	01	00	EA	39
Current online sensor address	Function code (read address)	Byte count (address count)	Online address	Modbus-CRC16 verification	

Table 7. Return command

■ Command of host modifying sensor address (when modifying sensor address, the bus can only connect one sensor function code 17)

Transmit: 00 17 00 00 00 02 F5 D9 (address 00 changed to 02)

00	17	00	00	00	02	F5	D9
Current sensor address	Function code (write address)	Starting address		Change to address range 0002 (0-105)		Modbus-CRC16 verification	

Table 8. Modify command

Return：02 17 01 02 D3 DA

02	17	01	02	D3	DA
Current online sensor address	Function code (write address)	Return data bytes count (address count)	Modified sensor number	Modbus-CRC16 verification	

Table 9. Return Command

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Obtain corresponding sensor data via Modbus

Communication description: In the transmitting area, only data bits need to be input, and no need to input the last two verification bits. Simply check data verification and select ModbusCRC low byte first. After sending data, the last two digits are automatically generated based on the data information (this data interface is a demonstration of a sensor connected to measure pressure and temperature).

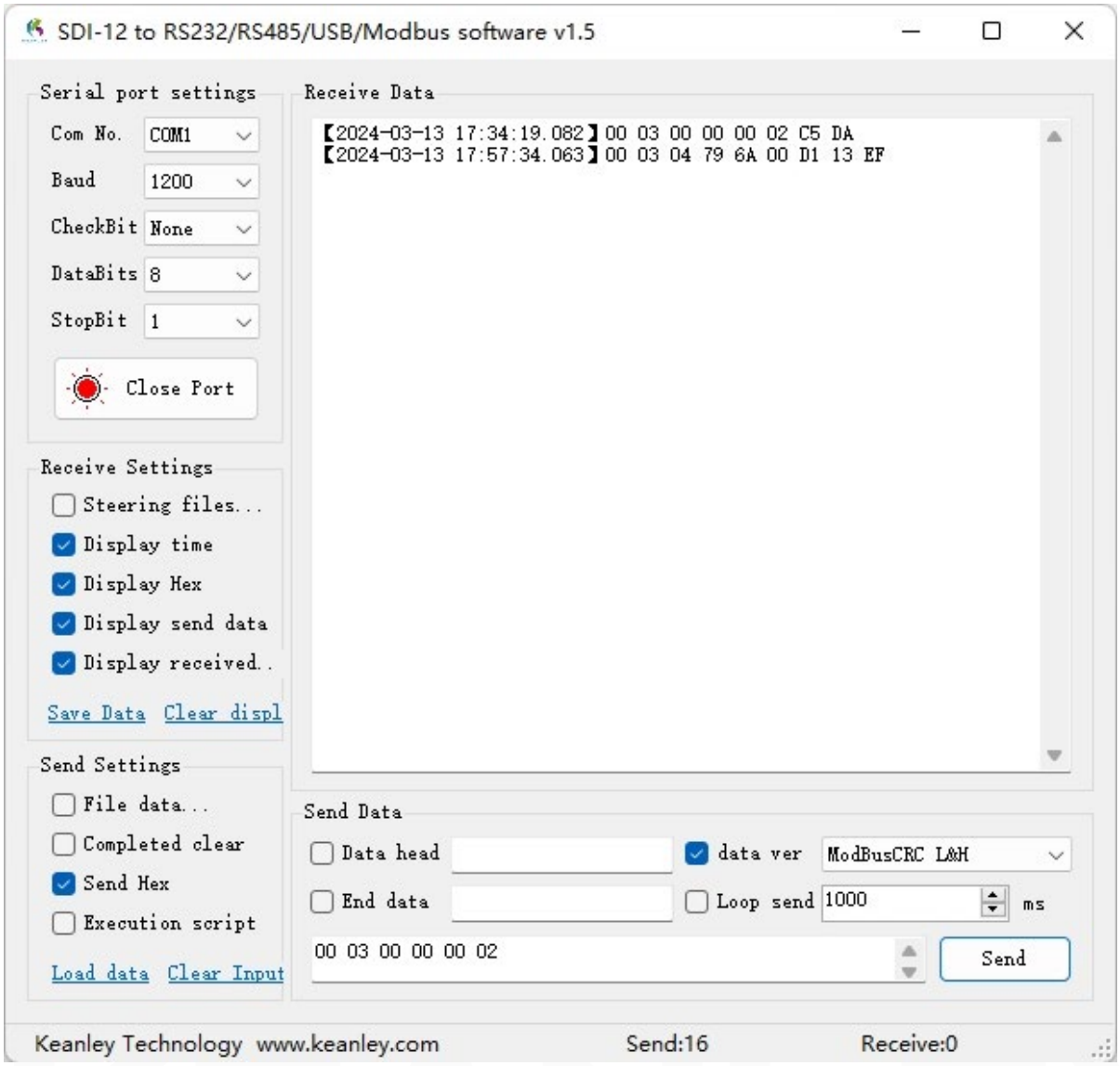


Figure 4 . Data acquisition interface

After connecting the device, open the above serial port software interface and set the parameters according to Table 3 on page 7 and contents on page 8 (RS232 interface communication).

Step 1: Set the serial port number COM1.

Step 2: Set the baud rate to 1200 (1200 and 9600 optional), refer to the label on the back of the device.

Step 3: Data bit 8.

Step 4: Check Bit None.

Step 5: Stop bit 1.

Step 6: Check the corresponding position as shown in the above figure and click the button to open the serial port tool (the red mark indicates that the serial port is open).

Step 7: Enter the read pressure/temperature command (00 03 00 00 00 00 02) in the sending area.

Step 8: Click the send button with the mouse to return the data (00 03 04 79 6A 00 D1 13 EF).

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Description of three types of TRS-1203 function codes:

■ 03-Read sensor data (note: all returned hexadecimal register data converted to decimal is 10 times of the actual value. If there are other requirements for the display format of the returned data, customization can be made).

■ 16-Read sensor address (note: only one sensor can be connected on the SDI-12 bus at this time).

■ 17-Modify sensor address (note: only one sensor can be connected on the SDI-12 bus at this time).

According to the above description, for example, command 03:

Read the pressure and temperature values of the corresponding sensors.

Send in sending area: 00 03 00 00 00 02 C5 DA

Return in returning area: 00 03 04 79 6A 00 D1 13 EF

The blue color indicates: 79 6A

Hexadecimal (HEX) 79 6A converted to decimal (DEC) : 31082

Actual pressure value: 31082/10=3108.2

00 D1

Hexadecimal (HEX)00 D1converted to decimal (DEC) : 209

Actual temperature value: 209/10=20.9

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

info@keanley.com

Obtain analysis of corresponding sensor data via Modbus

The meaning of the two sets of data returned refers to the data manual of the connected sensors (these data is a demonstration of a sensor connected to measure pressure and temperature).

03: Read data

00 03 00 00 00 02 C5 DA

- CRC check, low 8 bits, high 8 bits
- Register count, high 8 bits, low 8 bits
- Starting address, high 8 bits, low 8 bits
- 03 Function code (read data)
- Sensor address

Return data parsing

00 03 04 79 6A 00 D1 13 EF

- CRC check low 8 bits, high 8 bits
- Register count data 0x00D1 converted to decimal: 209 Actual value: 209/10=20.9 (temperature)
- Register count data 0x796A converted to decimal: 31082 Actual value: 31082/10=3108.2(pressure)
- Return Bytes count
- 03 Function code (read data)
- Sensor address

16: Read number

02 16 00 00 00 01 89 FA

- CRC check, low 8 bits, high 8 bits
- Read 1 sensor (can be any value)
- Starting address
- 16 Function code (read data)
- Sensor address (can be any value)

Response:

01 16 01 01 2A 05

- CRC check, low 8 bits, high 8 bit
- Current online sensor address
- Return data bytes count
- 16 Function code (read data)
- Current online sensor address

17: Write number

01 17 00 00 00 02 F4 08

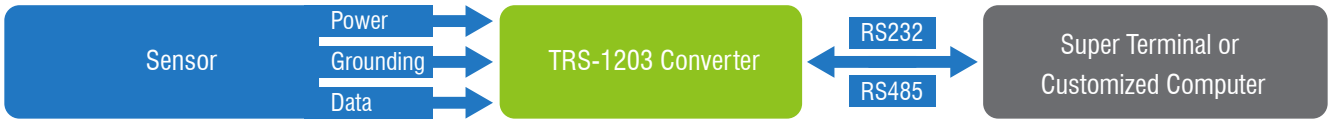
- CRC check, low 8 bits, high 8 bits
- Change to :0002, the range of address values is 0-105
- Starting address
- 17 Function code (write data)
- Current sensor address

Return a response after successfully modifying the number:

02 17 01 02 31 C9

- CRC check, low 8 bits, high 8 bit
- Modified sensor number
- Return data bytes count
- 17 Function code (write data)
- Current online sensor address

Product wiring diagram



Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

info@keanley.com

Commonly used function codes of Modbus

Code	Name	Register PLC Address	Bit/Word Operation	Operation Count
01	Read coil status	00001～09999	Bit operation	Single or multiple
02	Read discrete input status	10001～19999	Bit operation	Single or multiple
03	Read holding register	40001～49999	Word operation	Single or multiple
04	Read input register	30001～39999	Word operation	Single or multiple
05	Write a single coil	00001～09999	Bit operation	Single
06	Write a single holding register	40001～49999	Word operation	Single
15	Write multiple coils	00001～09999	Bit operation	Multiple
16	Write multiple holding registers	40001～49999	Word operation	Multiple

Table 10. Commonly used function codes of Modbus

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Exception codes of Modbus

The abnormal response messages consist of slave addresses, function codes, and exception codes. Among them, function codes are different from the normal response messages. In the abnormal response messages, the highest bit (MSB) of the function code is set to 1. Because the function code occupies one byte in the Modbus protocol, it is described as: abnormal function code=normal function code+0x80.

Exception Code	Name	Description
01	Illegal function code	The slave device does not support this function code.
02	Illegal data address	The specified data address does not exist in the slave device.
03	Illegal data value	The specified data exceeds the range or is not allowed to be used.
04	Slave device failure	Unknown errors occur during the response processing of the slave device.

Table 11. Exception code of Modbus

Communication example: control the slave machine to power on, if the register address does not exist.

0	1	2	3	4	5	6	7
Device address	Function code	Starting address		Register data count (2 bytes)		Modbus-CRC16 verification	
0x01	0x06	0x00	0xFF	0x00	0x01	0x78	0x3A

Table 12. Communication example

In the abnormal response, the highest position of the function code returned by the slave is 1, and the exception code is 02 which indicates an illegal address.

0	1	2	3	4
Device address	Function code	Exception code	Modbus-CRC16 verification	
0x01	0x86	0x02	0xC3	0xA1

Table 13. Communication example

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Optional SDI-12 transparent transfer protocol

The nameplate pasted on the back of the device is in SDI-12 mode and 1200 baud rate. Please refer to pages 16-26 for use instructions.

1. Analysis of the composition of SDI-12 transparent transport protocol:

1. All commands sent on the SDI-12 bus must be ASCII codes, with a maximum of 5 characters. All commands sent must start with an address code and end with “!”.
2. The specification of SDI-12 stipulates that the address range is the digits “0-9, a-z, A-Z”, and generally the initial address is the digit "0".
3. Command format, such as aM!, is a measurement command, where "a" is the address, "M" is the command to start the sensor measurement, and "!" is the end symbol of the command. These three symbols constitute a complete command.
4. After the command is issued, the sensor will respond accordingly. Still giving the aM! example above, the response result of the sensor is "atttn" at this point, in which "a" is the address, "ttt" is the measurement time with unit in second, and "n" is the quantity of data.

2. The standard SDI-12 byte frame format is as follows (with a fixed baud rate of 1200).

SDI-12 byte frame format			
Starting Bit	Data Bit	Parity Check Bit	Stopping Bit
1	7	1	1

Table 14. SDI-12 byte frame format

3. SDI-12 time sequence.

All SDI-12 commands and responses must follow the format shown in Figure 5 of the data line. The two commands and responses, preceded by an address and combined with a carriage return and line feed (<CR><LF>), are operated in the time sequence shown in Figure 6.

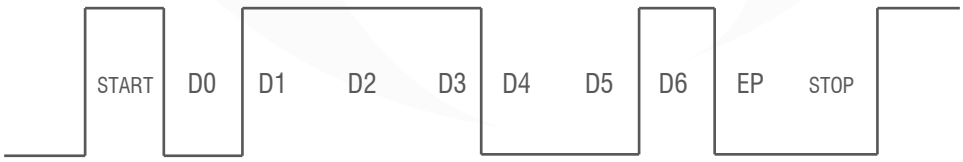


Figure 5. Example of SDI-12 transmission character 1 (0x31)

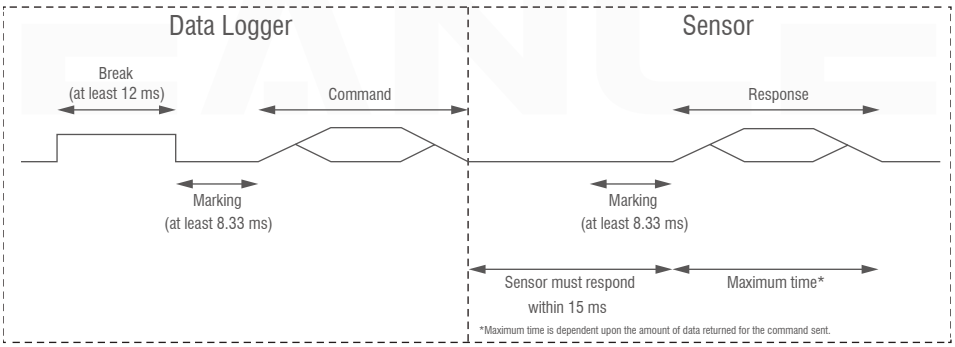


Figure 6. Communication example of the data logger and the sensor

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

4. Device address.

ASCII codes "0" to "9" are standard addresses that must be supported by all sensors and data loggers. When more than 10 sensors are needed, A～Z (65-90) and a～z (97-122) will be used.

ASCII Address (Single Character)	Decimal	Hexadecimal	Description
“0”	48	30	The factory default addresses of all sensors are set to 0 for the convenience of a single sensor systems
“1”～“9”	49 ～ 57	31 ～ 39	Multiple sensor addresses on the SDI-12 bus

Table 15. Device address

5. The measurement sequence process of a typical TRS-1203 data loggers and sensors is as follows.

Step 1. The data logger wakes up all sensors on the SDI-12 bus through an interrupt signal (at least 12 ms of continuous blank numbers).
Step 2. The data logger sends commands to the sensor identified at a specific address, indicating it to conduct a measurement.
Step 3. The sensor at the specified address responds within 15 ms, including the maximum time required to obtain measurement data and the number of measurement data to be returned.
Step 4. If the measurement data is instantly available, the data logger sends a command to the sensor to indicate that it returns a measurement data ready request. The data logger sends a measurement data return command.
Step 5. Sensor response, returns one or more measurement data.

6. Commonly used SDI-12 commands and replies.

Name	Command	Reply
Interrupt signal	Continuous 12ms blank number	None
Confirm activation	a!	a<CR<LF>
Send verification	aI!	AIlccccccmmmmmmvvvxx…xx<CR><LF>
Address change	aAb!	B<CR><LF>(This command is only supported when the sensor supports software to change the address)
Address query	?!	a<CR><LF>
Start measurement	aM!	atttn<CR><LF>
Send data	aD0!...aD9!	a<values><CR><LF>
Continuous measurement	aR0!...aR9!	a<values><CR><LF>

Table 16. Basic SDI-12 commands/replies (more commands can be found in the final attachment

The first character of all commands and replies is always the device address. The last character of the command is "!". The last two bytes of the reply are carriage return and line feed (<CR><LF>). In replies to command D, the maximum number of characters for the <values> section is 35 or 75.

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

■ Confirming activation command (a!)

This command is used to confirm that the sensor can respond properly to data logger or other SDI-12 devices to determine its presence on the SDI-12 bus, as shown in Table 17 below.

Command	Reply
a!	a<CR><LF>
a-Sensor address	a-Sensor address
!-Command end symbol	<CR><LF> Response end symbol

Table 17. Confirming activation command (a!)

■ Sending authentication command (aI!)

This command is used to query the SDI-12 version, model, and firmware version number of the sensor, as shown in Table 18 below.

Command	Reply
aI!	allccccccmmmmmmvvvxxx…xxx<CR><LF>
a-Sensor address	a-Sensor address
I-Sending verification command	II - SDI-12 version number, displaying the compatibility of SDI-12 version. For example, version 1.3, encoded as 13 xxx Xx - optional area, up to 13 characters, is used for serial number or other sensor information unrelated to data logger
!-Command end symbol	<CR><LF>Response end symbol

Table 18. Sending authentication command (aI!)

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

■ Address query command (?!)

If the sensor has an SDI-12 bus address, it needs to reply to the command. For example, if the sensor receives a query command ?!, reply “a” no matter what its address is. This command allows the user to confirm the address of the sensor. There is only one sensor on the SDI-12 bus.

■ Changing address command (aAb!)

This command changes the address of the sensor. If the sensor supports software to change the address, it will support the address change command, as shown in Table 19. After this command is sent and responded, the sensor is allowed to not reply to other commands for 1 second. This is because time is needed for the sensor to write the new address into non-volatile memory. There is only one sensor on the SDI-12 bus.

Command	Reply
aAb!	b<CR><LF>
a-Sensor address	b-Sensor address (new address if successfully changed, old address if address unchangeable)
A- Changing address command	
b-New modified address	
!-Command end symbol	<CR><LF> Response end symbol

Table 19. Changing address command (aAb!)

■ Starting measurement command （aM!）

This command tells the sensor to make measurements. But the sensor will not instantly reply to the measurement data after receiving this command. Instead, it will first reply to the time required to return the measurement data and the quantity of measurement data to be returned. To obtain the measurement data, the sending data command (aD0!) must be used. The starting measurement command is shown in Table 20.

Command	Reply
aM!	atttn<CR><LF>
a-Sensor address	a-Sensor address
M-Starting measurement command	ttt-Time measured in seconds, the time taken by the sensor to complete the measurement.
!-Command end symbol	n-The number of measured values, which will be returned in one or multiple D commands. (“n” is an integer number from 0 to 9).

Table 20. Starting measurement command (aM!)

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

■ Sending data command（aD0!,aD1!,…aD9!）

This command is used to obtain a set of data within the sensor. D0! is sent after M, MC, C, CC, V, or HA commands. The sensor sends returning data as a response. If all expected measurement data is not returned in D0 reply, the data logger will keep sending D1!, D2!,etc., until all measurement data is received. Sending data command is shown in Table 21 below.

Command	Reply
aD0!(aD1!...aD9!)	a<values><CR><LF>
a-Sensor address	a-Sensor address
D0-sending data command， D1…D9-additional data transmission	Polarity symbol+7-digit number+decimal point
!-Command end symbol	<CR><LF>Response end symbol

Table 21. Data Command(aD0!,aD1!,…aD9!)

If the response of the D command is correct but no data is returned, it indicates that the sensor has exited the measurement process. If the data logger wants to obtain data, an additional M, C, or V command must be sent. The maximum number of characters that can be returned in the <values> section of the D command reply is 35 or 75. If the data sending command D is used to receive data after concurrent measurement command or high-capacity ASCII measurement command, the maximum value is 75. Otherwise, the maximum value is 35.

■ Continuous measurement (aR0!…aR9!)

If the sensor can continuously monitor the value to be measured, such as the shaft encoder, there is no need to start measurement commands (M!, M1!... M9!). They can be read directly using the R command.

For example:

If (the sensor is in continuous measurement mode) aR0!,it will obtain and return the current value read from the sensor.

The R command (R0!... R9!) works similarly to the D command. The main difference is that the R command does not require the use of the M command to inform the sensor to perform measurement. The maximum number of characters that can be returned in the <values> section of the R command reply is 75.

Each R command is an independent measurement process, for example, there is no need for R0... R4 before the R5 command. If the sensor can't perform continuous measurement, it will return "a+<CR><LF>" as a reply. If CRC verification is required, the CRC verification code must be included before <CR><LF>, such as 0AP@<CR><LF.

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

info@keanley.com

Start sensor measurement in SDI-12 mode

Refer to the devices connection diagram (this data demonstration is sensor CTD-100). After a successful connection, open the interface conversion software on the computer and manually set the communication parameters, as shown in Figure 7 below.

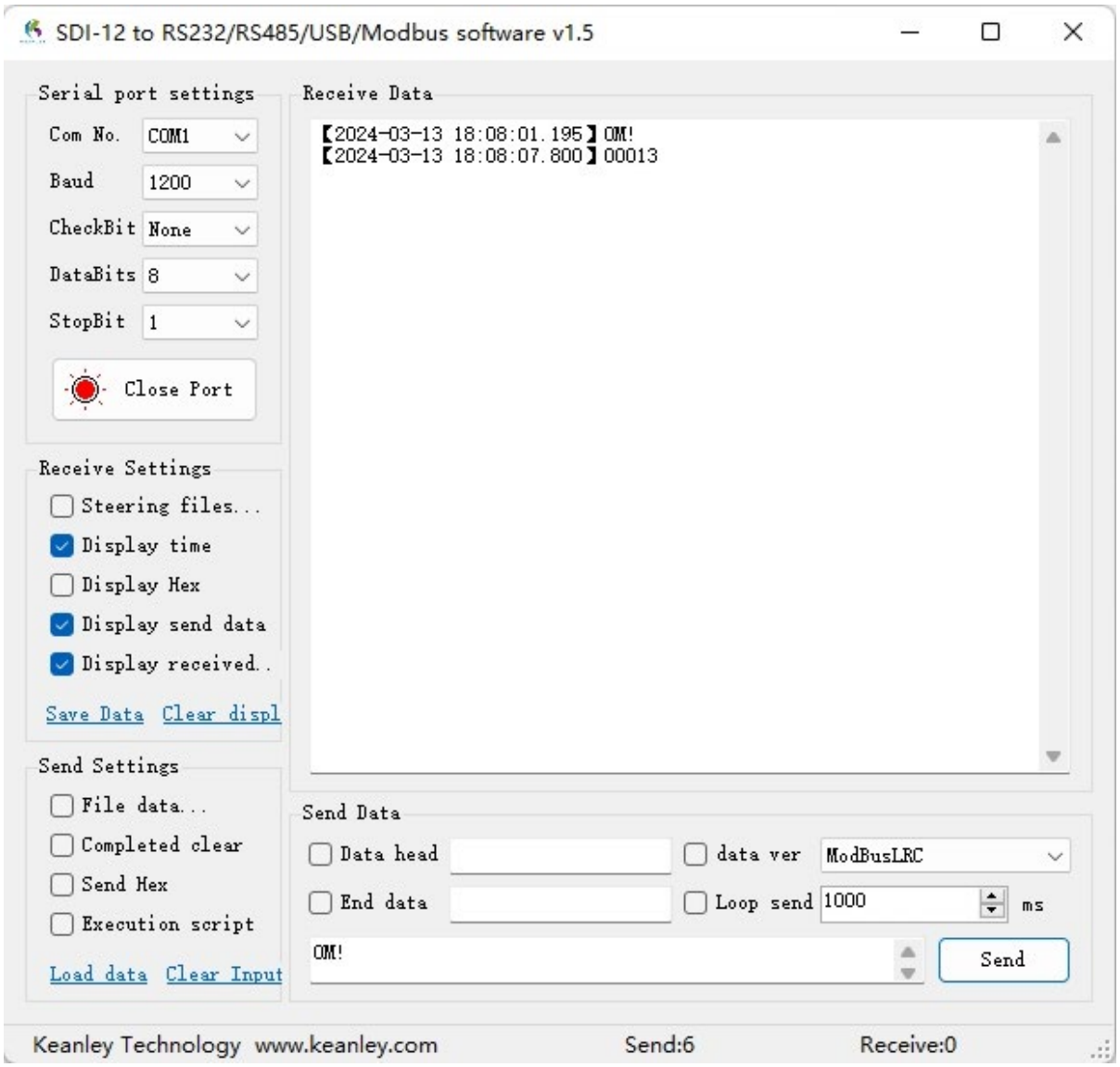


Figure 7. Setting communication parameter interface

After connecting the devices, open the serial interface tool and set the communication parameters (232 interface communication) according to Table 3 on page 7 and steps on page 8:

Step 1: The default option of serial interface number is COM1.

Step 2: Set the baud rate to 1200 (1200 and 9600 optional), refer to the label on the back of the device.

Step 3: Data bit 8.

Step 4: None (no verification).

Step 5: Stop bit 1.

Step 6: Check the corresponding position as shown in the above figure and then click the button to open the serial interface tool (the red mark “on” indicates that the serial interface is open).

Step 7: The default initial address of the sensor is number "0". After manual sending 0M! in the sending area, the second entry in the receiving data area is the returning data (0001).

After following steps 1-7 above, return to 00013. According to Table 5.5 on page 11, the parsing of the returning instruction “atttn” shows that “a” corresponds to sensor address “0”, “ttt” corresponds to time 1 second, and “n” corresponds to data 3.

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Obtain sensor data in SDI-12 mode

Obtain CTD-100 data in SDI-12 mode.

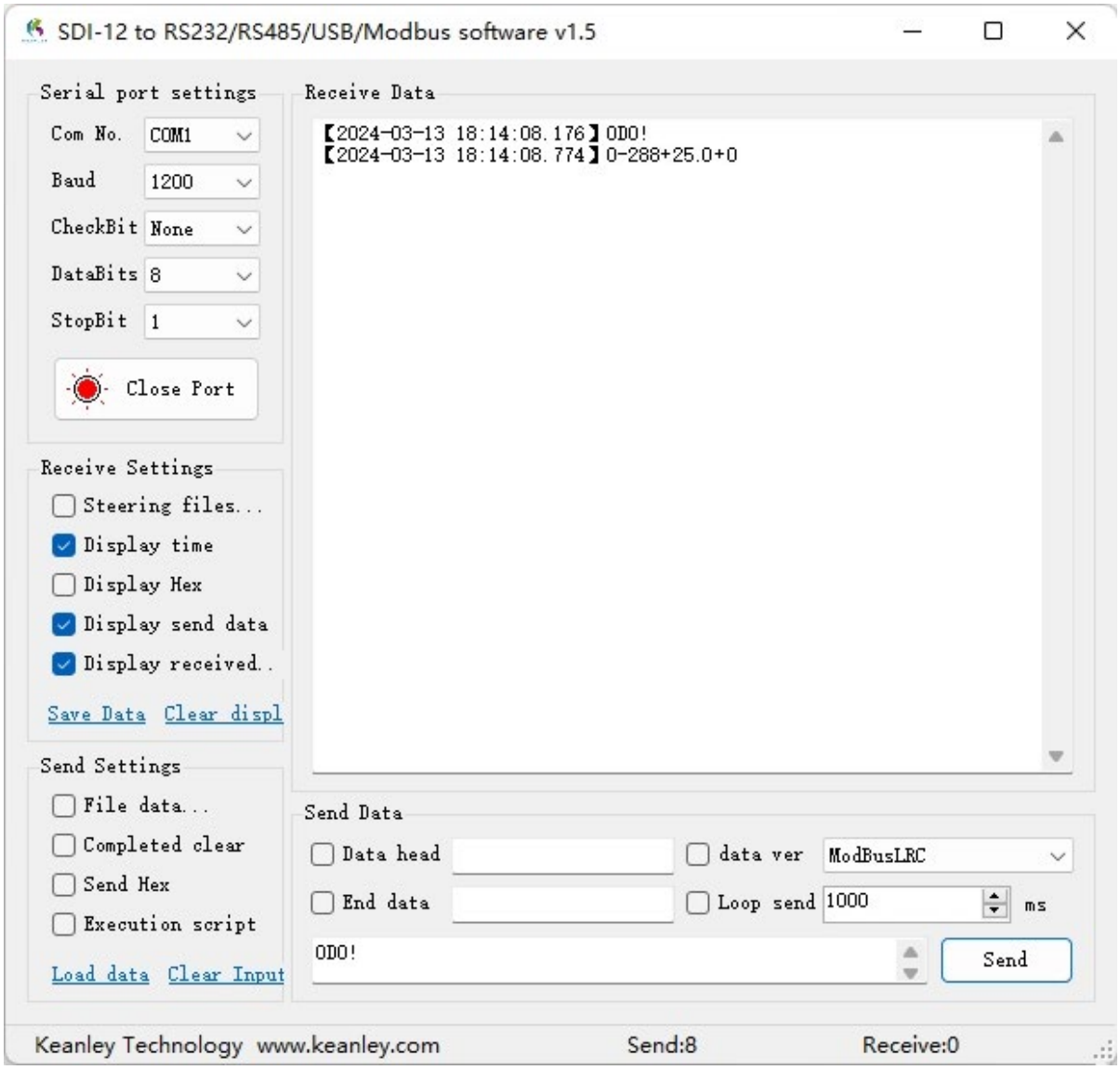
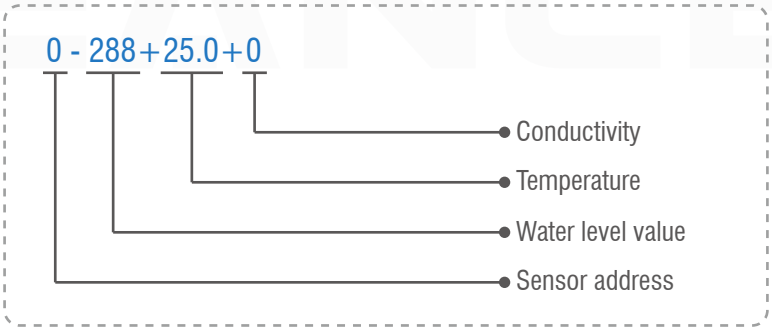


Figure 8. Obtaining sensor data interface

When using the sensor for the first time, the default address is 0. Enter 0D0! in the sending area and clicking send button, the data obtained is 0-288+25.0+0. According to the CTD-100 sensor instruction manual, parsing the returned data is as follows.



Refer to page 25 for other SDI-12 commands

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Send: The oscillogram feedback of 0D0! is shown in Figure 9.

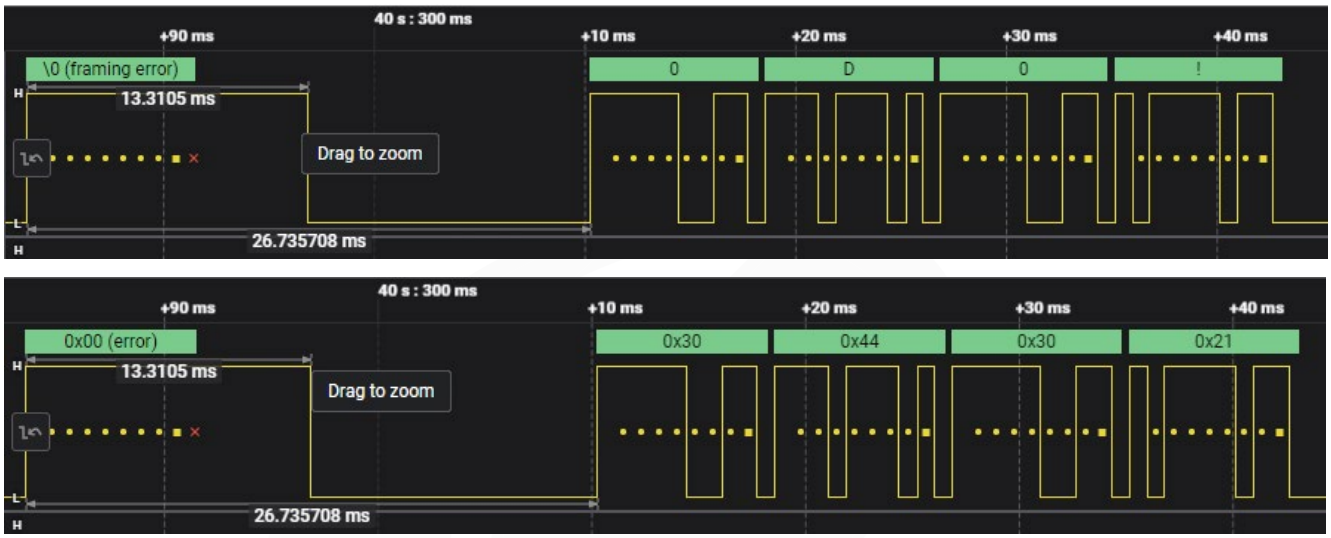


Figure 9. Oscillogram

The oscillogram feedback of the returning value characters 0-288+25.0+0 is shown in Figure 10.

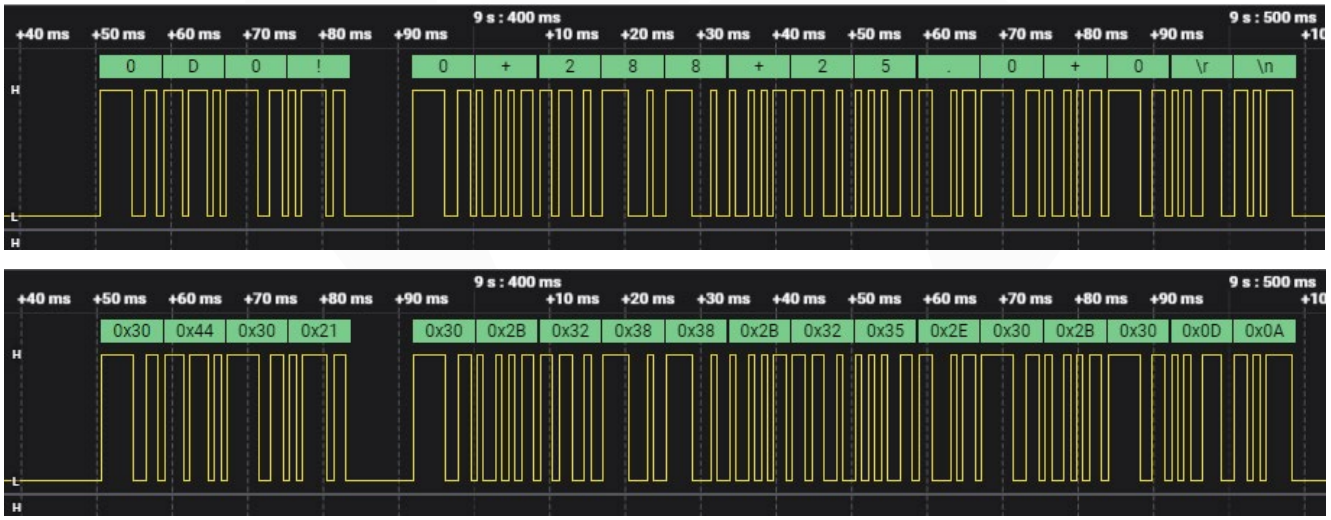


Figure 10. Oscillogram

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Recommended SDI-12 circuit

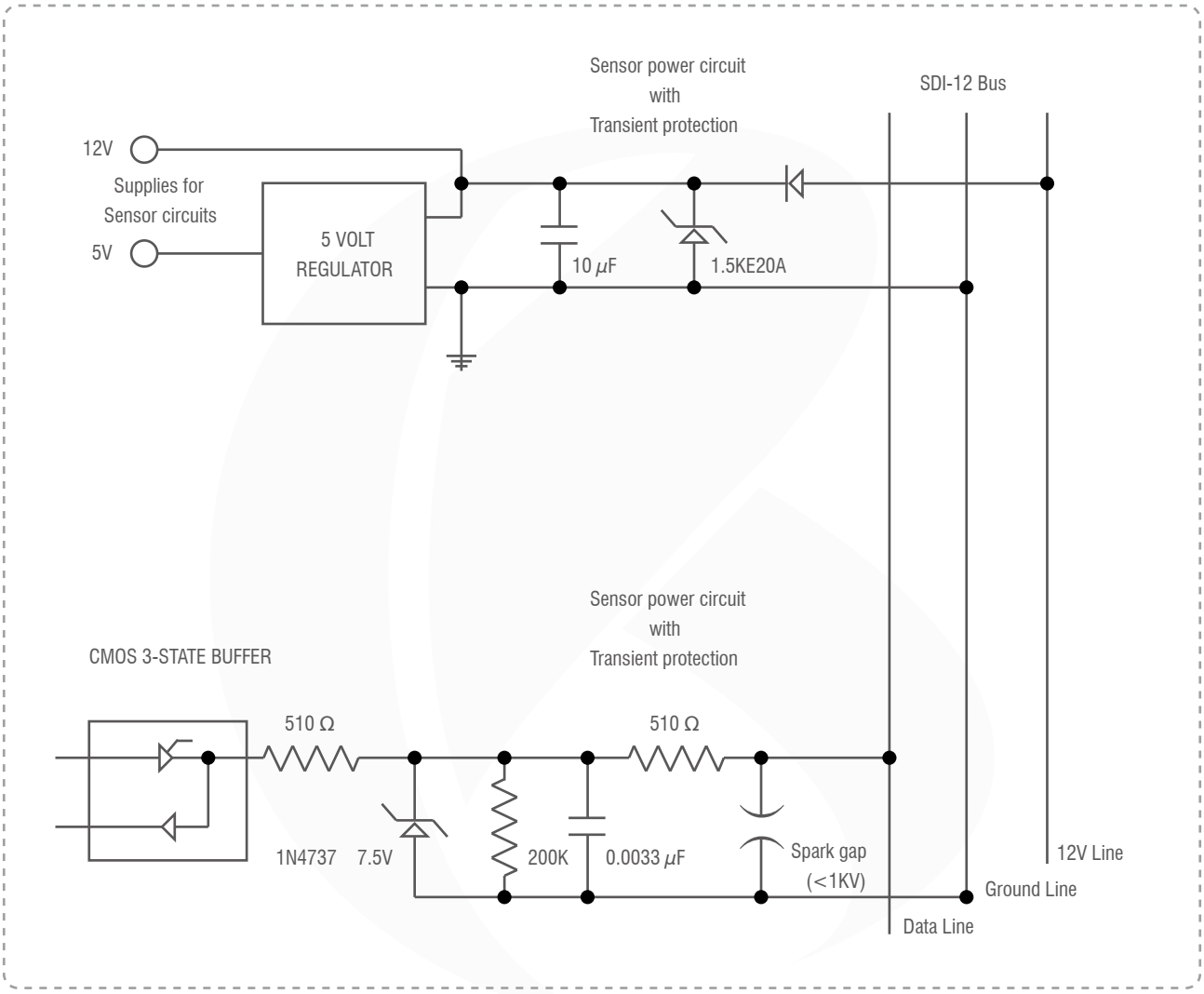


Figure 11. Recommended SDI-12 circuit

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Commands and replies of SDI-12

Name	Command	Reply
Interrupt signal	Continuous 12ms blank number	None
Confirm activation	a!	a<CR<LF>
Send verification	aI!	AIlccccccmmmmmmvvvxx…xx<CR><LF>
Address change	aAb!	B<CR><LF> (This command is only supported when the sensor supports software to change the address)
Address query	?!	a<CR><LF>
Start measurement	aM!	atttn<CR><LF>
Start measurement with CRC*	aMC!	atttn<CR><LF>
Send data	aD0!	a<values><CR><LF> or a<values><CRC><CR><LF>
	-	a<values><CR><LF> or a<values><CRC><CR><LF>
	-	a<values><CR><LF> or a<values><CRC><CR><LF>
	-	a<values><CR><LF> or a<values><CRC><CR><LF>
	aD9!	a<values><CR><LF> or a<values><CRC><CR><LF>
Additional measurements	aM1!	atttn<CR><LF>
	-	atttn<CR><LF>
	-	atttn<CR><LF>
	-	atttn<CR><LF>
	aM9!	atttn<CR><LF>
Additional measurement with CRC	aMC1!...aMC9!	atttn<CR><LF>
Start verification	aV!	atttn<CR><LF>

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Name	Command	Reply
Start concurrent measuremen	aC!	atttn<CR><LF>
Start concurrent measurement with CRC	aCC!	atttn<CR><LF>
Additional concurrent measurement	aC1!	atttn<CR><LF>
	-	atttn<CR><LF>
	-	atttn<CR><LF>
	-	atttn<CR><LF>
	aC9!	atttn<CR><LF>
Additional concurrent measurement with CRC	aCC1!...aCC9!	atttn<CR><LF>
Continuous measuremen	aR0!...aR9!	a<values><CR><LF> (same format as D command)
Continuous measurement with CRC	aRC0!...aRC9!	a<values><CRC><LF> (same format as D command)

Table 22. Commands and replies of SDI-12

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.

Connectable sensor brands

Mode	Function Classification	Brand	Logo
SDI-12/Modbus	Environmental monitoring instruments	Vaisala (Finland)	VAISALA
		GILL (U.K)	GILL
		R.M.Young (U.S)	YOUNG
	Measuring and metrologic instruments	METER (U.S)	METER
		Campbell (U.S)	CAMPBELL SCIENTIFIC
		Apogee (U.S)	apogee INSTRUMENTS
		ACCLIMA (U.S)	Acclima
		Stevens (U.S)	STEVENS
		OTT (Germany)	OTT HydroMet
Notes	Any sensor that meets the SDI-12/Modbus protocol standard can be connected (* only partial brands are shown). Non-standard protocols can be customized.		

Table 23. Connectable sensor brands

Any logo or product name is a trademark of Keanley or its cooperation partners. The information contained in this document is strictly prohibited from any copying, transfer, distribution, or storage, and all rights are reserved.All specifications (including technical specifications) are subject to change without prior notice.