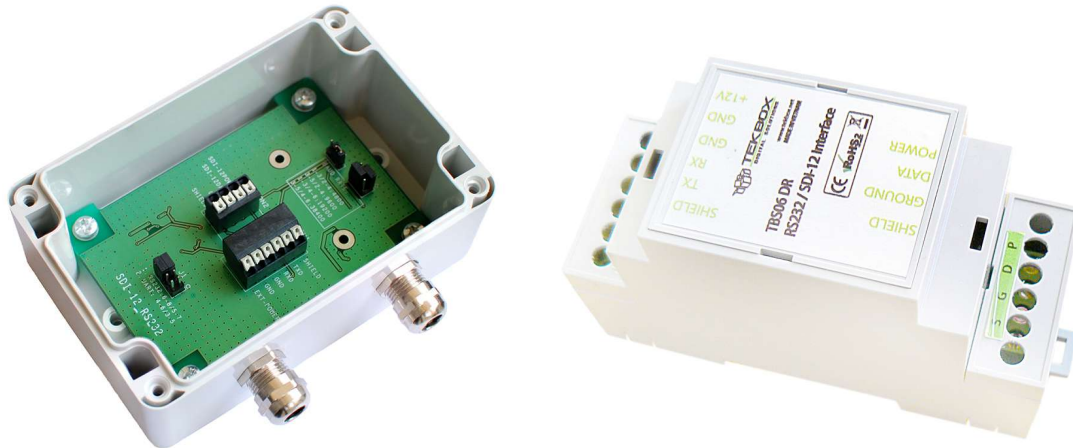


RS232 to SDI-12 Interface

The TBS06 (DR) RS232 to SDI-12 Converter is an interface for connecting a PC, data logger or RTU to one or more sensors with SDI-12 interface. The TBS06 connects to the RS232 interface and provides a SDI-12 compliant data interface. The TBS06 is a versatile tool for everyone who designs sensors and data recorders or who installs, tests or maintains SDI-12 based data acquisition systems.

It is available in a IP67 housing from Fibox or in a Dinrail housing.



TBS06 RS232 to SDI-12 Interface – Fibox- and Dinrail variant

Features

- RS232 toSDI-12 Interface
- Transfer Mode
- Plug and play
- No driver required
- Selectable data rate: 4800 - 38400 Baud
- 5V ... 12V supply voltage
- Low power mode
- Spring loaded terminal blocks

- IP67 housing or Dinrail housing
- Operating Temperature Range:
- 40°C ... + 85°C

Target Applications

- SDI-12 Data Recording
- SDI-12 Field Installations
- SDI-12 Interface Debugging
- SDI-12 Sensor Testing

RS232 to SDI-12 Interface

Contents

1	INTRODUCTION	3
1.1	PRODUCT FEATURES	3
1.2	MECHANICAL DIMENSIONS	3
2	APPLICATION	4
3	FUNCTIONAL DESCRIPTION	4
3.1	OVERVIEW	4
3.2	INTERFACE WITH PC	4
4	HARDWARE DESCRIPTION	5
4.1	BOARD OVERVIEW	5
4.2	CONNECTIONS – FIBOX VARIANT	6
4.3	CONNECTIONS – DIN-RAIL VARIANT	7
5	SDI-12 BASICS	9
6	SETTING UP THE TBS06	10
6.1	REQUIREMENTS	10
6.1.1	User Interface	10
6.1.2	Hardware Interface	10
6.1.3	Read Data from sensor	10
7	ORDERING INFORMATION	11
8	HISTORY	11

RS232 to SDI-12 Interface

1 Introduction

SDI-12 is a standard for interfacing data recorders with microprocessor-based sensors. SDI-12 stands for serial/digital interface at 1200 baud. It can connect multiple sensors with a single data recorder on one cable. It supports up to 60 meter cable between a sensor and a data logger.

The SDI-12 standard is prepared by

**SDI-12 Support Group
(Technical Committee)
165 East 500 South
River Heights, Utah
435-752-4200
435-752-1691 (FAX)
<http://www.sdi-12.org>**

The latest standard is version V1.3 and dates from July 18th, 2005. The standard is available on the web site of the SDI-12 Support Group.

TBS06 implements all functions for interfacing SDI-12 sensors to a RTU, data logger or PC with RS232 interface.

It is a plug and play solution for controlling or testing of sensors with SDI-12 interface.

1.1 Product Features

- RS232 to SDI-12 Interface based on TBS01 SDI-12 module
- Transfer Mode
- Plug and play
- No driver required
- Selectable data rate: 4800 - 38400 Baud
- 6V ... 16V supply voltage
- Current consumption: 6 mA in "always on mode"; less than 250µA in "auto power management mode"
- Fibox IP67 housing variant: Weidmueller spring loaded terminal blocks
- Dinrail housing variant: screw terminal blocks
- Operating Temperature Range: - 40°C ... + 85°C

1.2 Mechanical dimensions

Fibox housing: Fibox PC 081206

http://www.fibox.com/catalog/1999/product/611/7032580_ENG3.html

Dinrail housing: Bud Industries DMB-4770

<http://www.budind.com/pdf/hb4770.pdf>

RS232 to SDI-12 Interface

2 Application

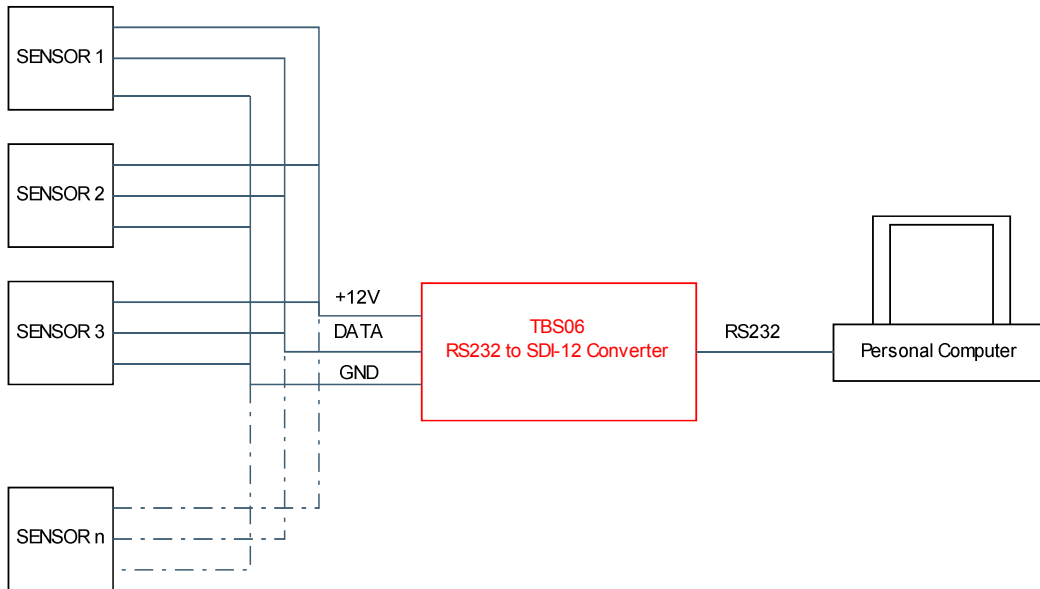


Figure 1 –TBS06 Application, standard setup for controlling / testing sensors

3 Functional Description

3.1 Overview

The SDI-12 standard defines a set of commands to configure sensors and to initiate measurements. Upon receiving specific commands, the sensor may carry out internal tasks, respond with information on conversion time or send measurement data.

SDI-12 commands typically are ASCII strings generated by the data recorder/controller firmware. TBS06 can be controlled by a PC application or hyper terminal and converts the command strings to the logic levels and baud rate specified by the SDI-12 standard. Furthermore TBS06 handles breaks, marks and all other details of the SDI-12 protocol.

Upon receiving data or status information originated by a Sensor, TBS06 extracts the corresponding ASCII strings and sends it to COM Port of the PC.

3.2 Interface With PC



Figure 2 – TBS06 basic application setup

The application is built based on a SDI-12 Recorder Protocol Stack Module. It receives commands from the RS232 Interface (e.g. via data logger, RTU or PC), and transfers the commands to the SDI Interface, waits for sensor response and transfers the response (measurement results, etc.) back to the RS232 Interface of the data logger, RTU or PC. All SDI-12 commands are supported.

RS232 to SDI-12 Interface

4 Hardware Description

4.1 Board overview

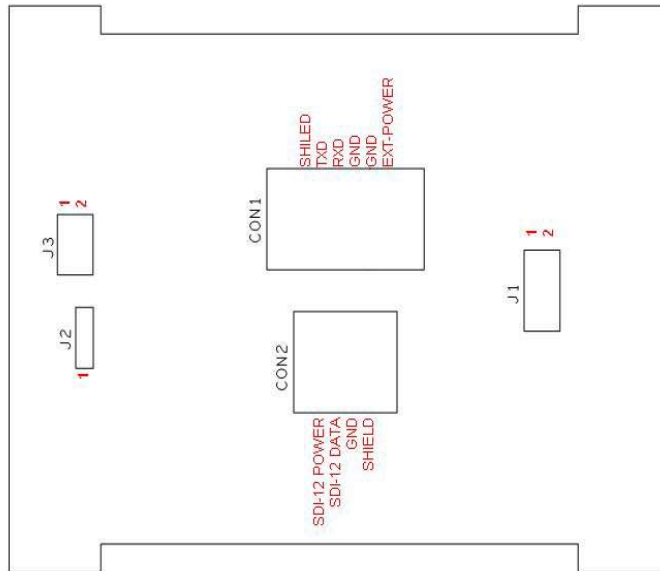


Figure 3 – Board Connections, Jumper Settings of Fibox variant

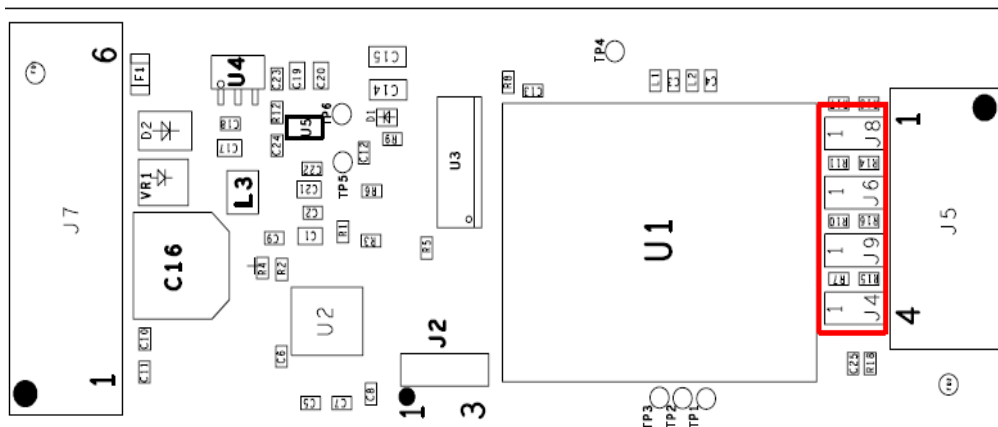


Figure 4 – Board Connections, Jumper Settings of Dinrail variant

RS232 to SDI-12 Interface

4.2 Connections – Fibox Variant

TBS06 supports RS232 and gives access to the UART interface of the on board SDI-12 module.

CON2: RS232 interface connector; Pin 1-2 of J1 : access to UART interface

4 Pin terminal block:

CON2 – SDI-12 Interface

Shield: connect to the shield of the SDI-12 cable or leave it unconnected

Ground: connect to the GND wire of the SDI-12 cable

SDI-12 data: connect to the data wire of the SDI-12 cable

SDI-12 Power: connect to the positive supply voltage wire of the SDI-12 cable; the SDI-12 supply voltage is directly connected to the RS232 supply line

6 Pin terminal block:

Shield: connect to the shield or leave it unconnected

Ground1: connect to the GND of supply for TBS06

EXT- Power: connect to an external power supply (6V...16V)

Ground2: connect to the GND of COM port (Pin 5 of DB9)

TxD: connect to RXD of the RS232 interface (Pin 2 of DB9)

RxD: connect to TxD of the RS232 interface (Pin 3 of DB9)

Jumper 1 (SDI12 module connectivity)

Connect to RS232 Interface:

Jumper: 5-7 and 6-8

Connects to RS232 lines of CON1

Access to UART Interface of the on-board TBS01 SDI-12 module:

Jumper: 3-5 and 4-6

Connect UART to pin 1-2 of J1

Jumper 2 (Power management)

Jumper 1-2: Always On (6 mA); default factory setting

Jumper 2-3: Power saving mode (>250uA), TBS06 will wakeup automatically when data is present at the Rx pin, and switches into sleep mode 2.5 seconds after RX becomes idle

Jumper 3 (Baud Rate selection)

4800 Baud: jumper 1-3, 2-4

9600 Baud: jumper 3-5, 2-4

19200 Baud: jumper 1-3, 4-6; default factory setting

38400 Baud: jumper 3-5, 4-6

RS232 to SDI-12 Interface

Other communication settings:

- 8 Bits
- No Parity
- 1 Stop Bit
- No Handshake

4.3 Connections – DIN-rail variant

RS232 Side, from left to right:

Cable shield
RS232 TX (TBS06DR output)
RS232 RX (TBS06DR input)
Ground
Ground
6-12V external supply input



SDI-12 Side, from left to right:

Cable shield
Ground
SDI-12 data line
SDI-12 supply voltage output

Figure 5 – Connections

TBS06DR is shipped with RS232 jumpered to 19200 Baud.

Following other Baud rates can be factory - jumpered upon specification in the order: 4800, 9600 and 32400 Baud.

RS232 to SDI-12 Interface

Other communication settings:

- 8 Bits
- No Parity
- 1 Stop Bit
- No Handshake

4 Pin terminal block:

CON1 – SDI-12 Interface

Shield: connect to the shield of the SDI-12 cable or leave it unconnected

Ground: connect to the GND wire of the SDI-12 cable

SDI-12 data: connect to the data wire of the SDI-12 cable

SDI-12 Power: connect to the positive supply voltage wire of the SDI-12 cable; the SDI-12 supply voltage is directly connected to the RS232 supply line

6 Pin terminal block:

CON2 – Power supply & RS232 interface

+12V: connect to an external power supply (7V...12V)

Ground1: connect to the GND of supply for TBS06DR

Ground2: connect to the GND of COM port (Pin 5 of DB9)

RxD: connect to TxD of the RS232 interface (Pin 3 of DB9)

TxD: connect to RXD of the RS232 interface (Pin 2 of DB9)

Shield: connect to the shield or leave it unconnected

Shield, Ground 1, Ground 2 and SDI-12 Ground are internally connected together

Jumper settings

Refer to Figure 4

Baud rate

4800 Baud	set jumpers J6, J8
9600 Baud	set jumpers J6, J9
19200 Baud	set jumpers J4, J8
38400 Baud	set jumpers J4, J9

Power management:

Jumper J2, position 1-2 jumpered: Always On (6 mA); default factory setting

Jumper J2, position 2-3 jumpered: Power saving mode (>250uA), TBS06 will wakeup automatically when data is present at the Rx pin, and switches into sleep mode 2.5 seconds after RX becomes idle

RS232 to SDI-12 Interface

5 SDI-12 Basics

SDI-12 is a serial data communication standard for interfacing multiple sensors with a data recorder

SDI-12 uses a shared bus with 3 wires: power (12V), data, ground

Data rate: 1200 baud

Each sensor at the bus gets a unique address which is in the range ASCII [0-9, a-z, A-Z]. The default address of every sensor is ASCII[0]. When setting up a SDI-12 sensor network, every sensor needs to be configured with a unique address. This can be done using the Change Address Command.

A sensor typically can measure one or more parameters.

Sensor manufacturers usually specify Extended Commands to configure or calibrate sensors. This commands are specified by the manufacturer, but they follow the command structure specified by SDI-12.

A typical recorder/sensor measurement sequence proceeds as follows:

- 1) The data recorder wakes all sensors on the SDI-12 bus with a break.
- 2) The recorder transmits a command to a specific, addressed sensor, instructing it to make a measurement.
- 3) The addressed sensor responds within 15.0 milliseconds returning the maximum time until the measurement data will be ready and the number of data values it will return.
- 4) If the measurement is immediately available, the recorder transmits a command to the sensor instructing it to return the measurement result(s). If the measurement is not ready, the data recorder waits for the sensor to send a request to the recorder, which indicates that the data are ready. The recorder then transmits a command to get the data.
- 5) The sensor responds, returning one or more measurement results

SDI-12 command structure:

Each SDI-12 command is an ASCII string with up to 5 characters, starting with the sensor address and terminated by a ! character.

Example:

Send Identification Command **0I!**

0 is the sensor address (sensor zero). Upon receiving this command, the sensor will send an ASCII string containing sensor address, SDI-12 compatibility number, company name, sensor model number, sensor version number and sensor serial number.

The standard process to carry out a measurement is to send a measurement request upon which the sensor responds with the time that is required to carry out the measurement and the number of data items being returned. After waiting the time that the sensor requires to carry out the measurement, the data recorder sends a Read Command to get the measurement results.

Example:

Start Measurement Command **0M1!**

Sensor 0 might respond **00302** which means the measurement will take 30 seconds and deliver 2 values.

After min. 30 seconds, the data recorder can send the Read Data Command **0D0!** to which Sensor 0 might reply **0+27+1050**. +27+1050 is the two measurement results which may be 27°C and 1050 milibar.

The response string of a sensor is always in ASCII format and may contain up to 40 or up to 80 characters, depending on the type of command. Out of 40 or 80 characters, the values part of the response string may contain up to 35 or 75 characters.

RS232 to SDI-12 Interface

6 Setting up the TBS06

6.1 Requirements

6.1.1 User Interface

Any RTU, data logger or PC hyper terminal (e.g.: Windows Hyper Terminal, Terminal V1.9B, RealTerm, Docklight or specific application software such as LabVIEW)

- Set hyper terminal to 19200 baud, 8 Bits, No Parity, 1 Stop Bit, No Handshake

6.1.2 Hardware Interface

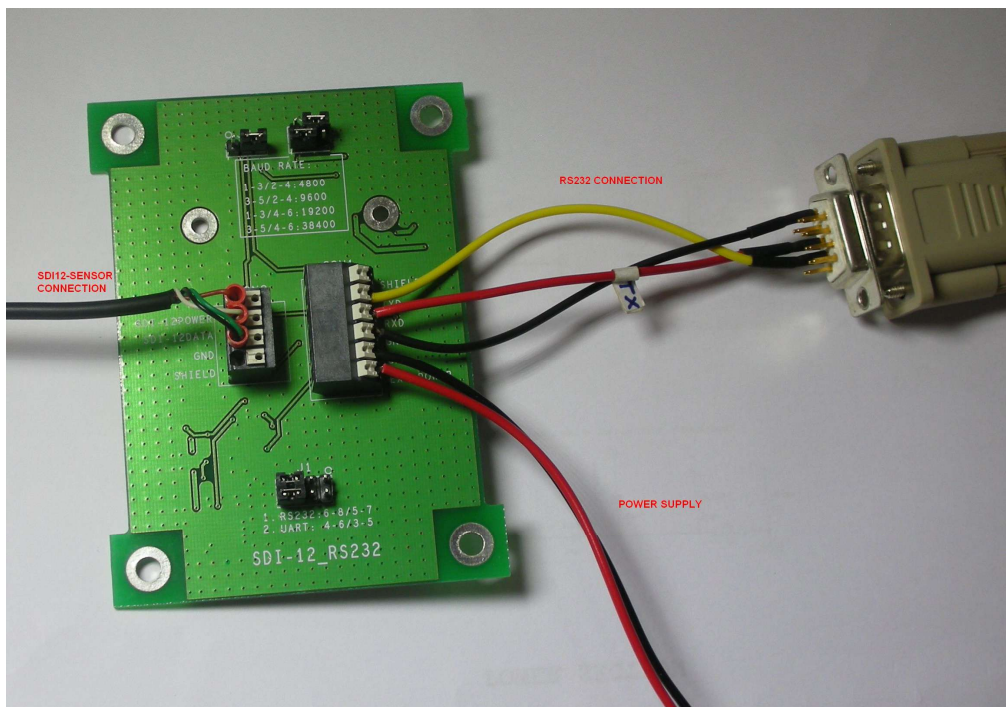


Figure 6 – RS232 connection

6.1.3 Read Data from sensor

On hyper terminal, send SDI-12 commands to read data from sensor.

Notice: every SDI-12 command must be terminated by **CR/LF** character (0x0D, 0x0A)

RS232 to SDI-12 Interface

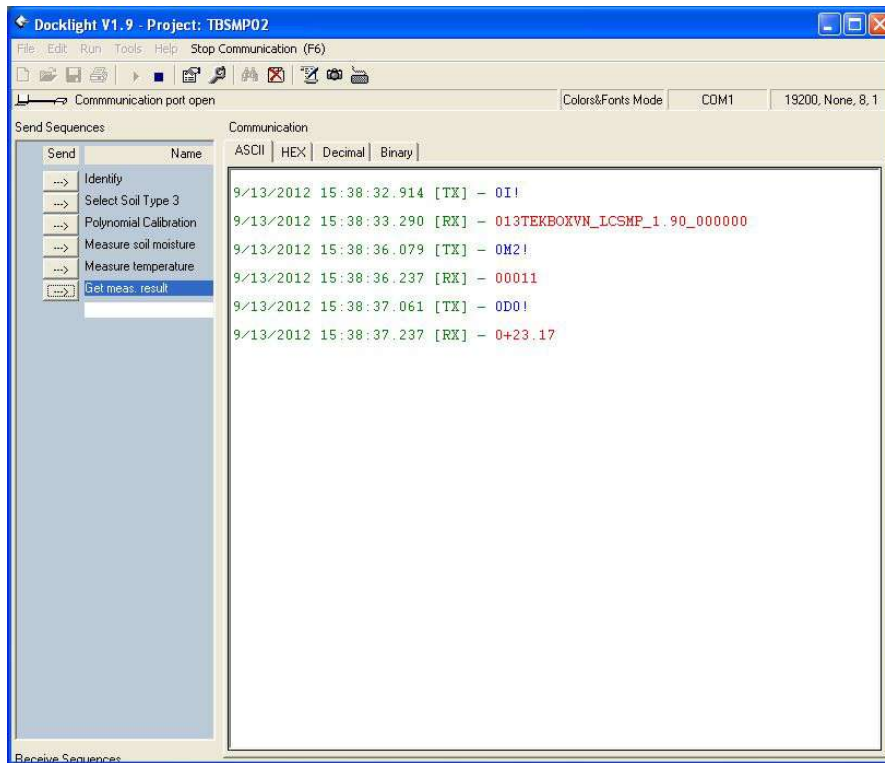


Figure 7 – Example: SDI-12 command flow on a terminal application

7 Ordering Information

Part Number	Description
TBS06	RS232 to SDI-12 interface in IP67 housing
TBS06-DR	RS232 to SDI-12 interface in DIN-rail housing
Default baud rate is 19200 Baud; default power management setting is “Always On” If you want different settings, please specify in your order	

Table 1 – Ordering Information

8 History

Version	Date	Author	Changes
V1.0	13.09.2012	ThinH	Creation of the document
V1.1	07.03.2014	Mayerhofer	Communication settings added
V1.2	23.09.2015	Mayerhofer	Fibox and Dinrail variant manuals merged

Table 1 – History