

**SDM-SIO1  
Serial Input/Output  
Module**



*Issued: 5.1.15*

Copyright © 2006-2015 Campbell Scientific Ltd.



# Guarantee

---

This equipment is guaranteed against defects in materials and workmanship. This guarantee applies for twelve months from date of delivery. We will repair or replace products which prove to be defective during the guarantee period provided they are returned to us prepaid. The guarantee will not apply to:

- Equipment which has been modified or altered in any way without the written permission of Campbell Scientific
- Batteries
- Any product which has been subjected to misuse, neglect, acts of God or damage in transit.

Campbell Scientific will return guaranteed equipment by surface carrier prepaid. Campbell Scientific will not reimburse the claimant for costs incurred in removing and/or reinstalling equipment. This guarantee and the Company's obligation thereunder is in lieu of all other guarantees, expressed or implied, including those of suitability and fitness for a particular purpose. Campbell Scientific is not liable for consequential damage.

Please inform us before returning equipment and obtain a Repair Reference Number whether the repair is under guarantee or not. Please state the faults as clearly as possible, and if the product is out of the guarantee period it should be accompanied by a purchase order. Quotations for repairs can be given on request. It is the policy of Campbell Scientific to protect the health of its employees and provide a safe working environment, in support of this policy a "Declaration of Hazardous Material and Decontamination" form will be issued for completion.

When returning equipment, the Repair Reference Number must be clearly marked on the outside of the package. Complete the "Declaration of Hazardous Material and Decontamination" form and ensure a completed copy is returned with your goods. Please note your Repair may not be processed if you do not include a copy of this form and Campbell Scientific Ltd reserves the right to return goods at the customers' expense.

Note that goods sent air freight are subject to Customs clearance fees which Campbell Scientific will charge to customers. In many cases, these charges are greater than the cost of the repair.



Campbell Scientific Ltd,  
80 Hathern Road,  
Shepshed, Loughborough, LE12 9GX, UK  
Tel: +44 (0) 1509 601141  
Fax: +44 (0) 1509 601091  
Email: [support@campbellsci.co.uk](mailto:support@campbellsci.co.uk)  
[www.campbellsci.co.uk](http://www.campbellsci.co.uk)



# Precautions

---

DANGER — MANY HAZARDS ARE ASSOCIATED WITH INSTALLING, USING, MAINTAINING, AND WORKING ON OR AROUND **TRIPODS, TOWERS, AND ANY ATTACHMENTS TO TRIPODS AND TOWERS SUCH AS SENSORS, CROSSARMS, ENCLOSURES, ANTENNAS, ETC.** FAILURE TO PROPERLY AND COMPLETELY ASSEMBLE, INSTALL, OPERATE, USE, AND MAINTAIN TRIPODS, TOWERS, AND ATTACHMENTS, AND FAILURE TO HEED WARNINGS, INCREASES THE RISK OF DEATH, ACCIDENT, SERIOUS INJURY, PROPERTY DAMAGE, AND PRODUCT FAILURE. TAKE ALL REASONABLE PRECAUTIONS TO AVOID THESE HAZARDS. CHECK WITH YOUR ORGANIZATION'S SAFETY COORDINATOR (OR POLICY) FOR PROCEDURES AND REQUIRED PROTECTIVE EQUIPMENT PRIOR TO PERFORMING ANY WORK.

Use tripods, towers, and attachments to tripods and towers only for purposes for which they are designed. Do not exceed design limits. Be familiar and comply with all instructions provided in product manuals. Manuals are available at [www.campbellsci.eu](http://www.campbellsci.eu) or by telephoning +44(0) 1509 828 888 (UK). You are responsible for conformance with governing codes and regulations, including safety regulations, and the integrity and location of structures or land to which towers, tripods, and any attachments are attached. Installation sites should be evaluated and approved by a qualified engineer. If questions or concerns arise regarding installation, use, or maintenance of tripods, towers, attachments, or electrical connections, consult with a licensed and qualified engineer or electrician.

## General

- Prior to performing site or installation work, obtain required approvals and permits. Comply with all governing structure-height regulations, such as those of the FAA in the USA.
- Use only qualified personnel for installation, use, and maintenance of tripods and towers, and any attachments to tripods and towers. The use of licensed and qualified contractors is highly recommended.
- Read all applicable instructions carefully and understand procedures thoroughly before beginning work.
- Wear a **hardhat** and **eye protection**, and take **other appropriate safety precautions** while working on or around tripods and towers.
- **Do not climb** tripods or towers at any time, and prohibit climbing by other persons. Take reasonable precautions to secure tripod and tower sites from trespassers.
- Use only manufacturer recommended parts, materials, and tools.

## Utility and Electrical

- **You can be killed** or sustain serious bodily injury if the tripod, tower, or attachments you are installing, constructing, using, or maintaining, or a tool, stake, or anchor, come in **contact with overhead or underground utility lines.**
- Maintain a distance of at least one-and-one-half times structure height, or 20 feet, or the distance required by applicable law, **whichever is greater**, between overhead utility lines and the structure (tripod, tower, attachments, or tools).
- Prior to performing site or installation work, inform all utility companies and have all underground utilities marked.
- Comply with all electrical codes. Electrical equipment and related grounding devices should be installed by a licensed and qualified electrician.

## Elevated Work and Weather

- Exercise extreme caution when performing elevated work.
- Use appropriate equipment and safety practices.
- During installation and maintenance, keep tower and tripod sites clear of un-trained or non-essential personnel. Take precautions to prevent elevated tools and objects from dropping.
- Do not perform any work in inclement weather, including wind, rain, snow, lightning, etc.

## Maintenance

- Periodically (at least yearly) check for wear and damage, including corrosion, stress cracks, frayed cables, loose cable clamps, cable tightness, etc. and take necessary corrective actions.
- Periodically (at least yearly) check electrical ground connections.

WHILE EVERY ATTEMPT IS MADE TO EMBODY THE HIGHEST DEGREE OF SAFETY IN ALL CAMPBELL SCIENTIFIC PRODUCTS, THE CUSTOMER ASSUMES ALL RISK FROM ANY INJURY RESULTING FROM IMPROPER INSTALLATION, USE, OR MAINTENANCE OF TRIPODS, TOWERS, OR ATTACHMENTS TO TRIPODS AND TOWERS SUCH AS SENSORS, CROSSARMS, ENCLOSURES, ANTENNAS, ETC.



# PLEASE READ FIRST

---

## About this manual

Some useful conversion factors:

**Area:** 1 in<sup>2</sup> (square inch) = 645 mm<sup>2</sup>

**Length:** 1 in. (inch) = 25.4 mm  
1 ft (foot) = 304.8 mm  
1 yard = 0.914 m  
1 mile = 1.609 km

**Mass:** 1 oz. (ounce) = 28.35 g  
1 lb (pound weight) = 0.454 kg

**Pressure:** 1 psi (lb/in<sup>2</sup>) = 68.95 mb

**Volume:** 1 UK pint = 568.3 ml  
1 UK gallon = 4.546 litres  
1 US gallon = 3.785 litres

## Recycling information



At the end of this product's life it should not be put in commercial or domestic refuse but sent for recycling. Any batteries contained within the product or used during the products life should be removed from the product and also be sent to an appropriate recycling facility.

Campbell Scientific Ltd can advise on the recycling of the equipment and in some cases arrange collection and the correct disposal of it, although charges may apply for some items or territories.

For further advice or support, please contact Campbell Scientific Ltd, or your local agent.



Campbell Scientific Ltd, 80 Hathern Road, Shepshed, Loughborough, LE12 9GX, UK Tel: +44 (0) 1509 601141 Fax: +44 (0) 1509 601091  
Email: [support@campbellsci.co.uk](mailto:support@campbellsci.co.uk)  
[www.campbellsci.co.uk](http://www.campbellsci.co.uk)





# Contents

---

<b>1. Introduction.....</b>	<b>1</b>
<b>2. Specifications .....</b>	<b>2</b>
2.1 Supported data rates and protocols .....	2
2.2 Electrical parameters.....	3
2.2.1 SDM-SIO1 current consumption .....	3
2.2.2 SDM-SIO1 voltage specifications.....	4
2.2.3 EMC compliance .....	4
2.3 Temperature and humidity ranges .....	4
2.4 Physical parameters.....	5
2.5 Datalogger compatibility.....	5
<b>3. Installation .....</b>	<b>5</b>
3.1 Connections .....	6
3.2 Safety considerations .....	8
3.3 Examples for connecting the SDM-SIO1 to other equipment.....	9
3.3.1 RS-485 one to one connection example.....	9
3.3.2 RS-485 multi unit / in line example .....	10
3.3.3 RS-485 half duplex wiring example.....	11
3.3.4 RS-485 internal circuit diagram.....	11
3.3.5 RS-232 wiring example with handshaking.....	12
3.3.6 RS-232 basic 3-wire example .....	12
3.3.7 Connecting a 9 way socket to the SDM-SIO1.....	13
3.4 Power conservation.....	13
<b>4. Programming the datalogger.....</b>	<b>14</b>
4.1 Special information about the Serial I/O CRBasic commands when used with the SDM-SIO1 .....	14
4.1.1 SerialOpen .....	14
4.1.2 SerialClose.....	16
4.1.3 SerialIn .....	17
4.1.4 SerialOut .....	17
4.1.5 SerialInBlock .....	17
4.1.6 SerialOutBlock.....	17
4.1.7 SerialInChk .....	18
4.1.8 SerialInRecord .....	18
4.1.9 SerialFlush.....	18
4.1.10 Serial Input Errors .....	18
4.2 Configuring handshaking and receive only modes.....	18
4.2.1 Using RTS/CTS and automatic handshaking .....	18
4.2.2 RS-485 half-duplex mode .....	19
4.2.3 Using the RS-232 link in receive only mode .....	19
4.3 Example datalogger programs.....	19
<b>5. Firmware upgrades and flash signature errors.....</b>	<b>22</b>
5.1 Upgrading the firmware .....	22
5.2 Firmware signature errors .....	22
5.3 Checking the firmware version and signature .....	22

## **Appendix**

A. Using the Handshaking lines for general input/output .....	A-1
A.1 The input pin (CTS/pin 8) .....	A-1
A.2 The output pin (RTS/pin 9) .....	A-2

## **Tables**

3-1. SDM address settings .....	6
3-2. SDM-SIO1 connections (left to right as viewed from the front of the unit) .....	7
3-3. SDM-SIO1 Functional description of the connections .....	7
4-1. Communications port parameters RS-232 .....	15
4-2. Communications port parameters RS-485 full duplex .....	15
4-3. Communications port parameters RS-485 half duplex .....	16
4-4. Communications port parameters RS-232 receive only mode .....	16

# SDM-SIO1 Serial Input/Output Module

---

*The SDM-SIO1 Module is designed to allow expansion of the number of serial ports available on a datalogger for communicating with intelligent sensors or driving external displays.*



*SDM-SIO1 Serial Input/Output Module*

## 1 Introduction

The SDM-SIO1 Module connects to Campbell Scientific dataloggers using the SDM port and communications protocol. It connects to the remote serial device using industry standard hardware that can be set to RS-232, RS-485 or RS-422 signal levels. When operating in RS-232 mode it also supports hardware handshaking. RS422 mode is functionally the same as RS485 mode except the connection is limited to a point to point system. Connections and programming for RS422 are otherwise identical to RS485.

The SDM-SIO1 will accept serial data and store it in its buffer which is 2047 bytes in size allowing remote equipment to transmit large amounts of data without needing to stop other processes in the datalogger.

Up to 15 SDM-SIO1s can be connected to a single logger using the SDM port, allowing the user to connect 15 different items of equipment to their logger with ease, in addition to any connections made to the dataloggers other serial ports.

To start using the SDM-SIO1 it is first necessary to work out how data will be exchanged with a sensor. In the case of a sensor there are basically two options, either the datalogger requests data and then picks up the response, or the sensor transmits data “one-way” using its own timebase. The latter mode is more common but can lead to problems with synchronising the sensor measurements with the logger program and can also lead to the occasional missing data value as there are two independent clocks. Once the method of communication and the communications standard is defined then refer to Section 3 below on how to install the module and connect it to the datalogger and the serial device.

The SDM-SIO1 Module is implemented in such a way that it looks like a built-in serial port to the user when writing programs in CRBasic. This means all the user needs to do is define the address of the SDM-SIO1, which is easily accomplished by the use of a rotary switch on the side of the unit. The serial port can then be used as if it were built into the datalogger. The only difference in operation between the SDM-SIO1 and a built-in port is that there will be a small delay when transferring data to and from the device via the SDM connection (see Section 4.1).

The SDM-SIO1 can also be used in 'talk-through' mode to allow a user to talk, via a terminal module, to a sensor connected to the SDM-SIO1 for test and diagnostic purposes. Please refer to the logger manual for further details.

Section 4 of this manual gives the differences in the use of this module compared to the datalogger standard serial ports, plus there are some simple examples.

## 2 Specifications

### 2.1 Supported data rates and protocols

Data rates and protocols are set up using the SerialOpen command in CRBasic. The SerialOpen command is discussed elsewhere in this document.

#### Supported data rates

300, 1200, 2400, 4800, 9600, 19200, 38400, 57600, and 115200 bits/s

#### Supported modes of operation

RS-232 (Full duplex and receive only), RS-485 (Half and full duplex), RS-422 (Half and full duplex). Hardware CTS/RTS flow control is supported in RS-232 mode, the handshaking lines can also be used as general purpose I/O lines.

#### Supported data format

8, 7 bit data size\*; none, odd or even parity; one or two stops bits.

\* In 7 bit mode with no parity the user must ensure that the characters received by the SDM-SIO1 have a delay of at least one bit period or greater between them. This does not affect any other configuration and does not affect transmissions out of the SDM-SIO1.

#### Miscellaneous information

- The SDM-SIO1 does not support auto baud rate detection nor the use of the serial port for DNP, or general PakBus communications.
- Use of the serial port for general Pakbus communications is not currently supported.

#### Buffer sizes

- Transmit buffer size: 767 Bytes (Buffer from the logger to the sensor)
- Receive buffer size: 2047 Bytes (Buffer from the sensor to the logger)

Both transmit and receive buffers are fill and discard type, i.e. once the buffers become full no new information is accepted and all further data is

discarded until space is made when the logger requests data from the SDM-SIO1.

## 2.2 Electrical parameters

### 2.2.1 SDM-SIO1 current consumption

	Nominal	Max	Notes
<b>General currents</b>			
Standby current	70 uA	100 uA	Current after SerialClose has been called.
<b>RS-232 and RS-485 current consumption <sup>(1)</sup></b>			
Idle current	5.5 mA	6 mA	After SerialOpen has been called
Idle current (receive only)	4.1 mA	4.5 mA	After SerialOpen in receive only mode
Active current (RS-232)	11.5 mA	12 mA	Active RS-232 command
Active current (RS-485)	12.5 mA	13 mA	Active RS-485 command (no termination resistors)
<b>Line load currents</b>			
RS-232 line load	2 mA per load	3 mA per load	Average expected increase in drawn current per RS-232 line connected in idle or active modes (no extra current in stand-by mode).  Both TX and RTS are considered to be RS-232 loads.
RS-485 line load <sup>(2)</sup>	40 mA <sup>(3)</sup>	77 mA <sup>(4)</sup>	This extra current is only present when actively transmitting

- (1) All currents are measured with no loads connected
- (2) The RS-485 transmit pair is disabled when not transmitting in order to save power higher value resistors can be used to save power dependent upon the application. For many applications, especially with shorter cable runs, no load/termination resistors will be needed.
- (3) Single 100R load between transmit lines. Two 100R resistors (one on each end) is the maximum recommended loading. Removing any termination resistance should dramatically decrease current consumption during transfer of data
- (4) The RS-485 interface is protected against short circuits via a 44R resistance making this the maximum current possible even during short circuit. This resistance is part of the ESD protection circuitry and will be present at all times; it shouldn't affect normal circuit operations. The 'RS-485 internal circuit diagram' on p.10 of this manual shows the circuit in detail.

## 2.2.2 SDM-SIO1 voltage specifications

Connection	Minimum voltage <sup>(1)</sup>	Nominal voltage <sup>(1)</sup>	Maximum voltage <sup>(1)</sup>
Power supply, +12V connection <sup>(2)</sup>	7V	12V	20V
RS-232 input threshold Low	0.8V	-	-
RS-232 input threshold High	-	-	2.4V
RS-232 input absolute maximum <sup>(2)</sup>	-	+/-15V	+/-18V
RS-232 input resistance	3 K $\Omega$	5 K $\Omega$	7 K $\Omega$
RS-232 output voltage swing <sup>(3)</sup>	+/-5V	+/-5.4V	-
RS-232 output absolute maximum	-	-	+/-13.2V
RS-485 input (Differential)	200mV <sup>(4)</sup>	-	6V
RS-485 output (Differential)	2V (at 50 Ohms)	-	-
SDM lines (high level)	4.3V	5V	5.7V
SDM lines (low level)	0V	-	0.7V

- (1) Values are volts D.C. (except resistances)
- (2) It is NOT recommended that the user runs their SDM-SIO1 at maximum ratings for extended periods of time
- (3) Assuming a worst case 3 K $\Omega$  load
- (4) It is not recommended that the user allows such low input voltages as there will be an increased chance that external noise may cause errors in the incoming data

## 2.2.3 EMC compliance

The SDM-SIO1 has been tested and shown to comply with IEC 61326. The device incorporates transient and surge protection that is designed to meet IEC61000-4-5, level 4, providing the device is adequately grounded.

## 2.3 Temperature and humidity ranges

Temperature range	Minimum	Maximum	Notes
Standard range	-25°C	+50°C	
Extended range (optional)	-40°C	+80°C	(Contact Campbell Scientific Ltd for further extended temperature requirements)

Humidity	Minimum	Maximum	Notes
Standard range	0%	95%	(non-condensing)

## 2.4 Physical parameters

### Main body

Height:	5.4H cm (2.2”H)
Width:	8.0W cm (3.1”W)
Depth:	2.5D cm (1.0”D)

### Main body including base mounting flange

Height:	5.4H cm (2.2”H)
Width:	11.2W cm (4.5”W)
Depth:	2.5D cm (1.0”D)
Weight:	80g (approximately)

**Mounting centres are 10.2cm (4”)**

## 2.5 Datalogger compatibility

The SDM-SIO1 is compatible with our CR800, CR850, CR1000, CR3000, CR5000, and CR9000X dataloggers. The CR5000 and CR9000X’s operating system must be OS 6 or higher.

## 3 Installation

The SDM-SIO1 is normally mounted on the backplane of a Campbell Scientific enclosure using the screws and plastic inserts provided. The SDM-SIO1 is designed to be installed in a dry, non-condensing environment.

Before fixing it, select and set the SDM address as this requires access to the side of the case. The SDM address is set with a screw driver. Below is a list of the possible SDM addresses and their relationships to the COMport number in the SerialOpen command.

There can be up to 15 SDM-SIO1s on a single SDM bus. Each SDM-SIO1 will need to be set to a unique address before they are powered up; if any other equipment is present on the bus, whether it’s an SDM-SIO1 or not, then the user will have to ensure their addresses are not the same.

Table 3-1. SDM address settings		
Rotary switch position	SDM address	SerialOpen command comm. port number
0	0	32
1	1	33
2	2	34
3	3	35
4	4	36
5	5	37
6	6	38
7	7	39
8	8	40
9	9	41
A	10	42
B	11	43
C	12	44
D	13	45
E	14	46
F <sup>(1)</sup>	15 <sup>(1)</sup>	47 <sup>(1)</sup>

(1) Address 'F' is not available as it's the broadcast address. Setting this address will result in the SDM-SIO1 having an address of '0' not 'F'.

### 3.1 Connections

Connection to the SDM-SIO1 is achieved via the 15 terminals arranged along the top of the unit. The terminals are spring loaded providing an easy and reliable method of connection. Wires should be stripped 7-9 mm, twisted and inserted in the round hole whilst opening the clamp by pushing a screwdriver in the adjacent rectangular hole or by pushing a screwdriver into the hole on the side of the connector. Remove the screwdriver to close the clamp making sure the clamp grips the wire rather than the plastic insulation.

If there is a need to insert more than one wire in each terminal, if using multi-strand wire twist the conductors together first. If using solid wires, either solder or crimp multiple pairs together before insertion. For RS-485 connections note that the RS-232 terminals double up as a secondary connection, so it is rare to have multiple conductors in one terminal as both sets of connections can be used either for forming an RS-485 daisy-chain or for adding termination resistors at each end of a network.

When making connections to the datalogger always ensure power to the datalogger is switched off and connect the ground (G) connection first.

Below is a table showing all the connections on the SDM-SIO1. There are a number of pins that are common within the unit, these are also shown below.



**NOTE**

The A, B, X, Y notation used in the labelling of the SDM-SIO1 follows the practice of the driver chip manufacturer (Maxim Inc.) used in this product. Unfortunately, the convention of A/B and X/Y labelling differs between different designers of RS485 products. If you cannot get the connection to work with another device, or the data is completely corrupted, please first check the polarity of the connections (see below) for the SDM-SIO1 relative to the device it is being used with as they give a more certain indicator of the correct connection than the A/B notation. If the polarity is not indicated then reversing the A relative to B connections (and the X and Y if used) can be done to check for the correct connection with no risk of damage to either device.

SDM			Power connections			RS-232 connections				RS-485/RS422				
C1	C2	C3	G	+12V	G	RX-A	CTS-B	RTS-Y	TX-Z	0V	Z	Y	B	A
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
						<i>15</i>	<i>14</i>	<i>13</i>	<i>12</i>		<i>10</i>	<i>9</i>	<i>8</i>	<i>7</i>

*Note: Italic numbers indicate which pins are connected internally (for use with RS-485 termination resistors etc.). For example the signals present on pin 7 will also be present on pin 15.*

	Case text	Description
1	C1	SDM data line – connect to datalogger SDM C1
2	C2	SDM clock line – connect to datalogger SDM C2
3	C3	SDM enable line – connect to datalogger SDM C3
4	G	Connect to the datalogger power ground (G) <sup>(1)</sup>
5	+12V	Main power supply – connect to logger 12V
6	G	RS-232 0V reference/second G connection <sup>(2)</sup>
7	RX-A	RS-232 receive line
8	CTS-B	RS-232 CTS hardware handshaking line / output
9	RTS-Y	RS-232 RTS hardware handshaking / input
10	TX-Z	RS-232 transmit line
11	0V	RS-485 0V reference line <sup>(3)</sup>

12	Z	'-' RS-485 output line, line Z*
13	Y	'+' RS-485 output line, line Y*
14	B	'-' RS-485 input line, line B*
15	A	'+' RS-485 input line, line A*

\*See note above

- (1) At least one of the two G terminals ('G') must be connected to the loggers ground terminal or earth boss.

---

### CAUTION

**The ground connection to the datalogger should be made with large gauge wire, e.g. 16/0.2 to provide a low impedance path to ground to allow full protection from static and electrical transients.**

---

- (2) The 'G' (pin 6) can be used for the RS-232 zero volt reference or any other ground connection needed, e.g. shields.
- (3) There is a 100R resistor in series with the dataloggers ground connection. This connection should be used when connecting RS-485 equipment by long wire lengths. It ensures both systems have a common ground reference point. See Section 3.2 below before connecting.

As shown above in Table 3.3 connections 1 to 5 need to be made to the datalogger, either connecting the SDM lines to the matching control ports or the dedicated SDM port when fitted, e.g. the CR3000. The connection of the wires to the remote serial device will vary with type of device and method of communication. It is necessary to work out the best mode of operation of the serial device, taking into consideration issues such as power consumption, cable lengths (RS-485 being better than RS-232), synchronisation of data collection etc.

In Section 3.3 there are some examples of different connection schemes for the serial devices. Further discussion of different modes of operation are given in Section 4.

### NOTE

Do not connect both RS-232 and RS-485 interfaces to the SDM-SIO1 at the same time as this may cause a bus contention and even possibly cause damage.

---

## 3.2 Safety considerations

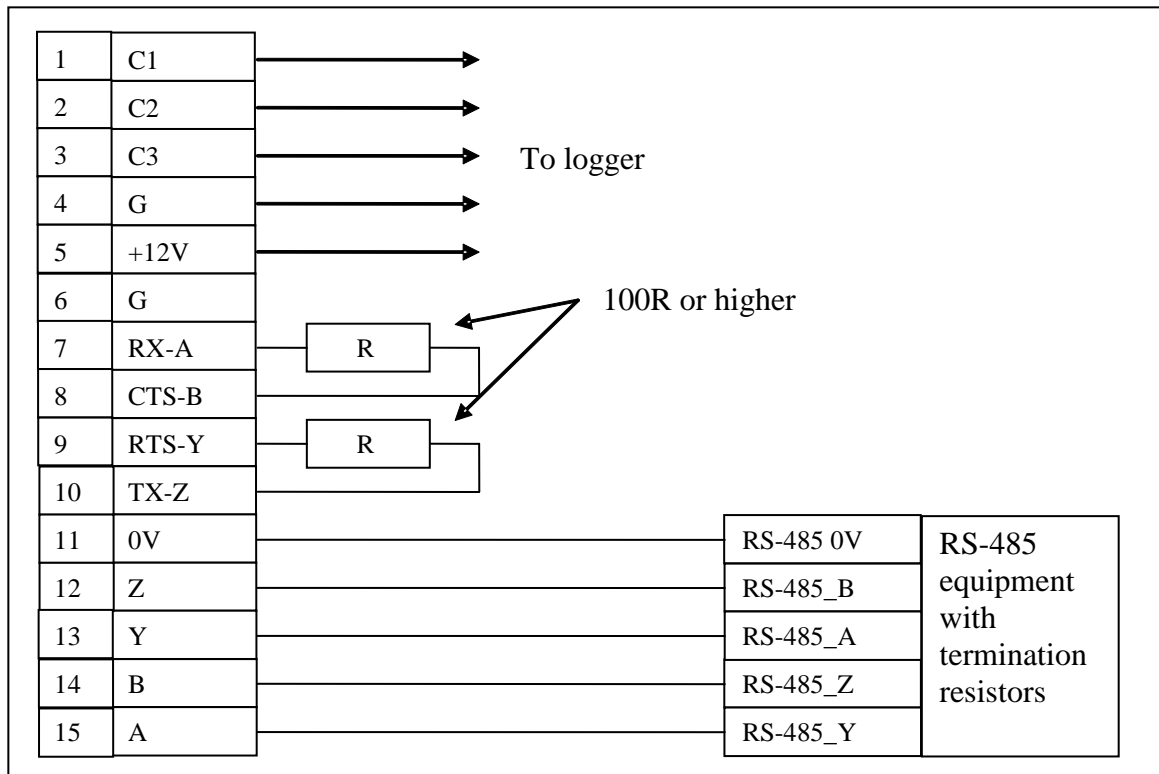
The SDM-SIO1 is considered to be a component of a measurement system that is installed in an enclosure and wired up in accordance with this manual. Due to space considerations full details of the maximum ratings of the connections are not given on the device. Instead the user should study this manual and in particular Section 2.1.2 to determine the maximum voltages that are applicable to any terminal before starting an installation.

The RS-485 0V ground reference connection may be needed to ensure all units are referenced to a common ground voltage. This is more often needed with long cable runs.

It is advisable to check the difference in ground potential with some caution before connecting any wires, to ensure the potential differences are reasonably close and excessive current will not flow between the two ground wires. There is a current limiting resistor fitted in the 0V line in the SDM-SIO1, but this will not be adequate in the event of a serious ground fault, e.g. the ground references being 240 V apart, due to faulty AC wiring. If a large potential difference is found please seek the advice of a qualified electrician before continuing with the installation.

### 3.3 Examples for connecting the SDM-SIO1 to other equipment

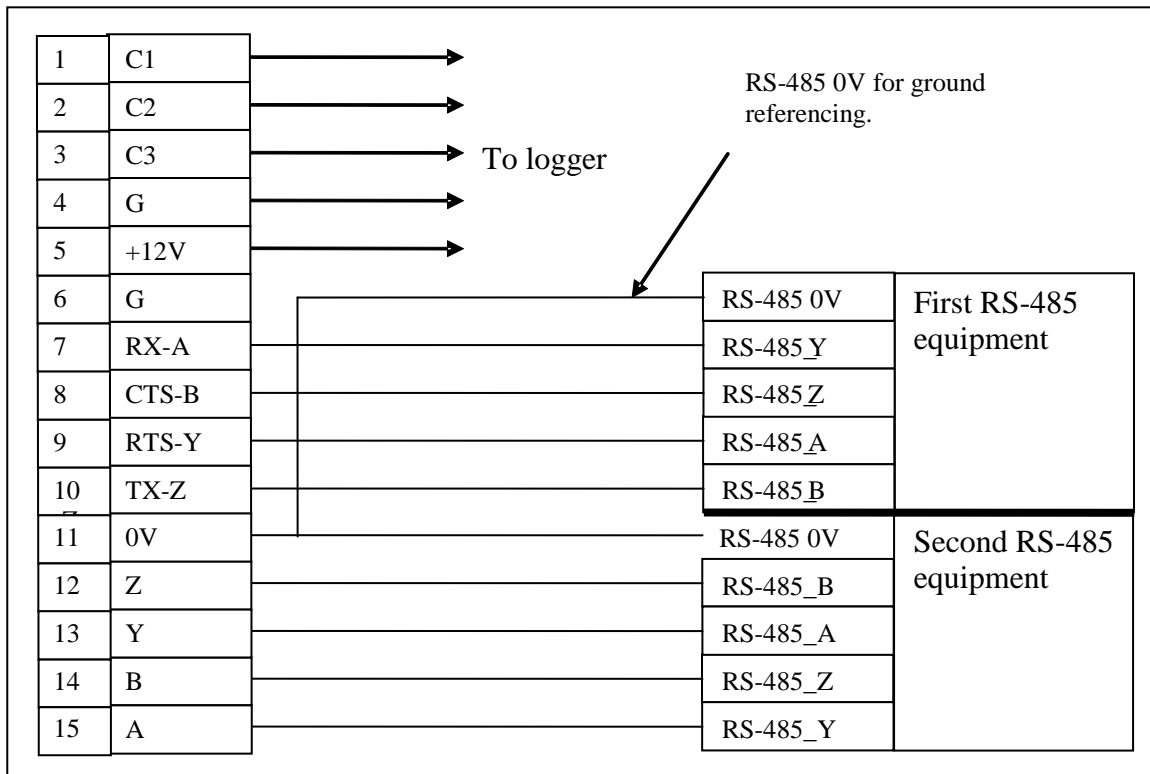
#### 3.3.1 RS-485 one to one connection example



Notes:

- Z, Y, B and A are connected to their corresponding differential wire pairs when in RS-485/RS422 mode. Where A and Y are the '+' lines and B and Z are the '-' lines.
- Connections 7-10 are connected internally to connections 12-15. This allows for terminations resistors to be added when in RS-485 mode (if needed) or, more than one connection to the I/O lines or TX and RX lines in RS-232 mode.
- If the use of termination resistors is required in RS-485 mode then they should be connected between pins TX-Z and RTS-Y for the ZY line and CTS-B and RX-A for the A B line (see diagrams elsewhere in document).
- In half duplex RS-485/RS-422 mode the Z Y and A B pairs are connected internally by the hardware without the need for any user interaction. The user should connect their wires to ZX and YS, as 'A' and 'B' are disabled.

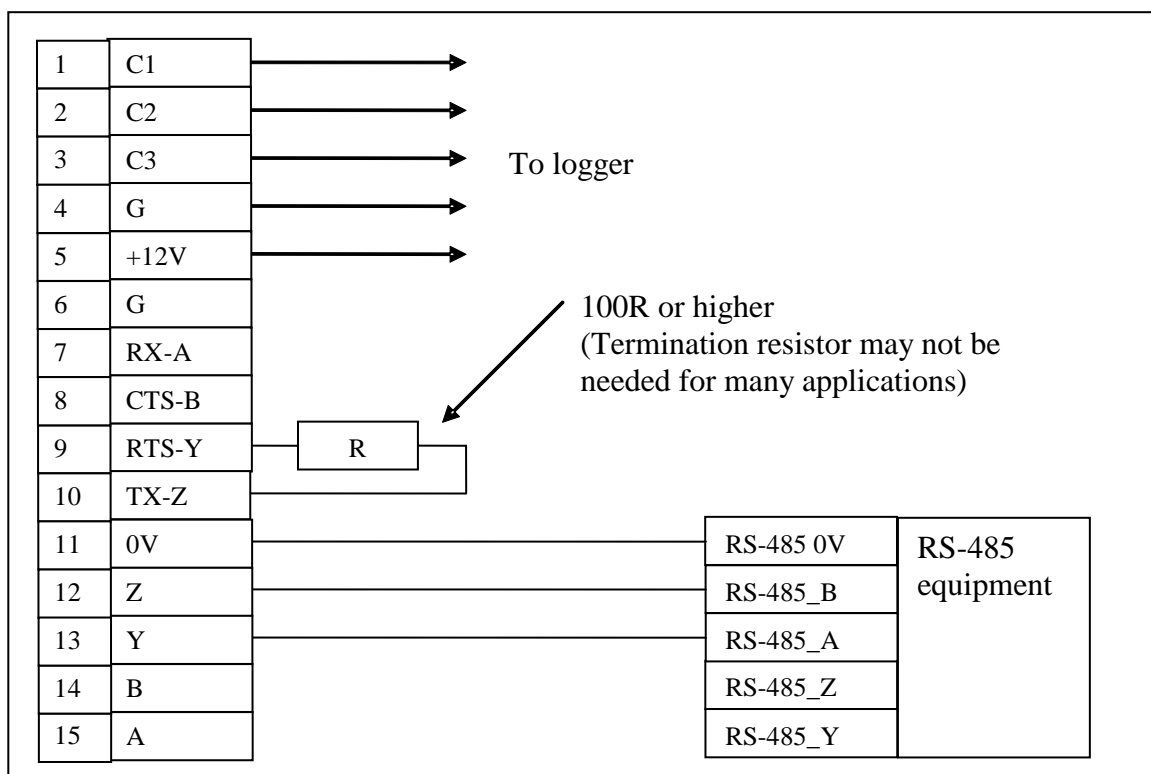
### 3.3.2 RS-485 multi unit / in line example



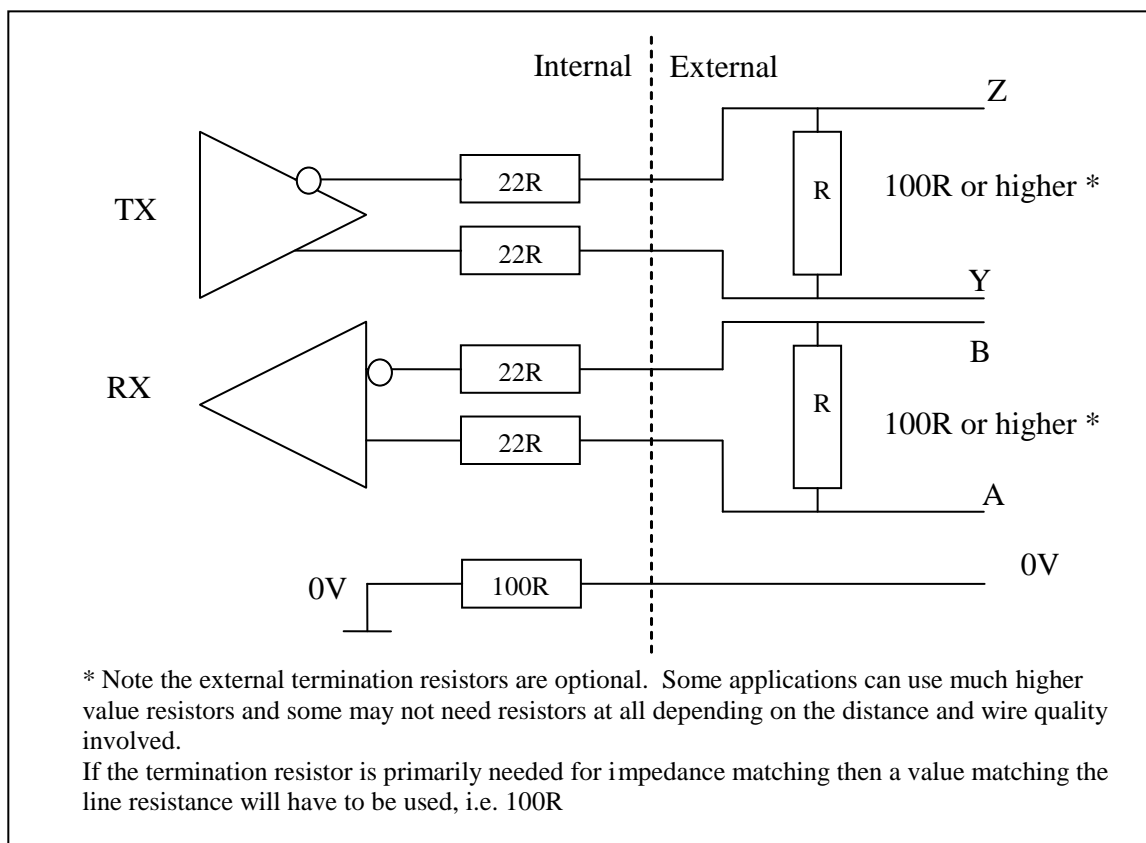
**NOTE**

As with all RS-485 configurations the use of termination resistors is optional. They tend not to be required with shorter cable runs and with modern slew-rate limited driver technology (as used in the SDM-SIO1).

### 3.3.3 RS-485 half duplex wiring example

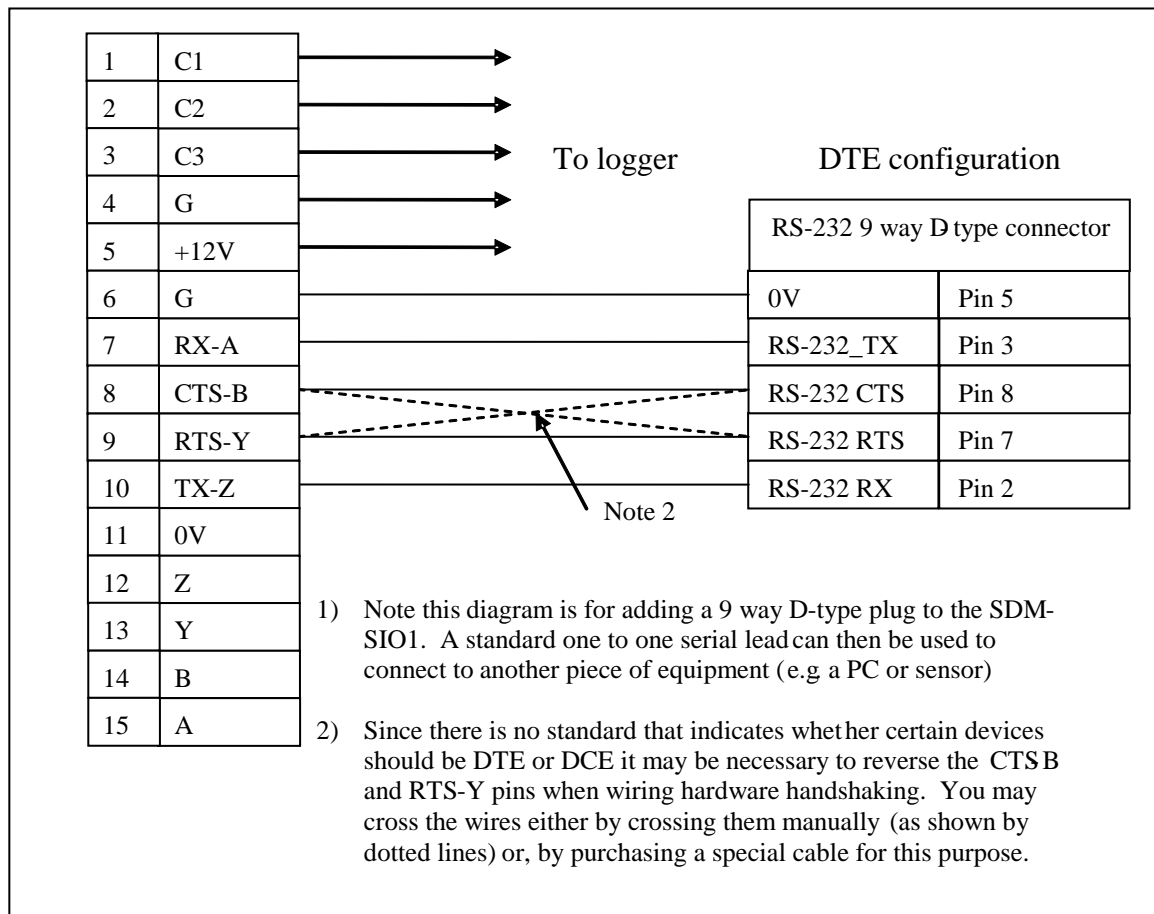


### 3.3.4 RS-485 internal circuit diagram





### 3.3.7 Connecting a 9 way socket to the SDM-SIO1



#### Notes:

- The CTS-B (CTS) and RTS-Y (RTS) lines can also be used to trigger external circuitry if desired when not being used in handshaking mode. Note that when using CTS-B and RTS-Y lines as input and output that the voltage levels are +/-12V NOT 0 and 5V.
- CTS-B and RTS-Y can't be used as separate input and outputs when in RS-485 mode.

## 3.4 Power conservation

The SDM-SIO1 features an industry standard RS-232/RS-485 driver chipset (Maxim 3160) which ensures maximum likelihood of compatibility with all other devices. When the driver is powered on it uses more power than one of the datalogger's control port based "com ports" – typically 6 mA minimum, partly because it generates the correct signal levels which in itself requires power and partly because, in the case of RS-232 signals, the resting state of ~-6V driving a nominal RS-232 ~3k load implicitly wastes ~2 mA of current per line.

To avoid excess current wastage, the chip can be turned off when not in use, e.g. between polled measurements, simply by closing the serial port, using Serialclose (see below). When running in RS-232 mode with a sensor that only sends data one-way to the datalogger, run the SDM-SIO1 in “receive only mode” as this does not turn on the output drivers.

If using RS-232 mode also avoid connecting handshaking lines that are not required to avoid the unnecessary 2 mA drain per line.

## 4 Programming the datalogger

The normal Serial port instructions such as SerialOpen and SerialIn are used with the SDM-SIO1. Aside from changes to the COMPort and the SerialOpenFormat parameter a program written for a standard serial port will usually work with the SDM-SIO1, providing the timing of the measurements is not very critical.

### 4.1 Special information about the Serial I/O CRBasic commands when used with the SDM-SIO1

The following section gives further information about the CRBasic programming language and how the serial commands are to be used with the SDM-SIO1.

This section assumes the user has knowledge of the CRBasic programming language; if not then please contact Campbell Scientific for further assistance.

Only the parameters that need special explanation or where their functionality has changed are listed below. Any other parameters should be taken as not having changed. Please refer to the datalogger manual and/or the CRBasic help system for more guidance.

#### 4.1.1 SerialOpen

All internal buffers in the SDM-SIO1 will be flushed when this command is called resulting in the loss of any data that might have been in them at that point in time.

##### **COMPort parameter**

The ‘COMPort’ numbers are defined in the ‘Address configuration’ section (see Table 3.1). Comport numbers in the range of 32..47 are reserved for use with the SDM-SIO1.

##### **BaudRate parameter**

Baud rate is used to set up the SDM-SIO1s baud rate as you would with any RS-232 interface. The SDM-SIO1 does not support automatic baud rate recognition. Instead setting the rate to a negative number sets the automatic flow control system (RTS/CTS). This system is discussed in greater detail elsewhere in the document.



### SerialOpenFormat parameter

This parameter defines not only the data format but with the SDM-SIO1 is used to set whether the module should work in normal RS-232, listen only RS-232, full or half-duplex RS-485 modes as defined in the tables below.

Code	Parity	No. stop bits	No. data bits
0	None	1	8
1	Odd	1	8
2	Even	1	8
3 (default)	None	1	8
4	<i>Not used</i>		
5	Odd	2	8
6	Even	2	8
7	None	2	8
8	<i>Not used</i>		
9	Odd	1	7
10	Even	1	7
11 <sup>(1)</sup>	None	1	7
12	<i>Not used</i>		
13	Odd	2	7
14	Even	2	7
15	None	2	7

(1) This mode is only supported if there is at least a one bit delay between characters received by the SDM-SIO1

Code	Parity	No. stop bits	No. data bits
16	None	1	8
17	Odd	1	8
18	Even	1	8
19	None	1	8
20	<i>Not used</i>		
21	Odd	2	8
22	Even	2	8
23	None	2	8
24	<i>Not used</i>		
25	Odd	1	7
26	Even	1	7
27 <sup>(1)</sup>	None	1	7
28	<i>Not used</i>		
29	Odd	2	7
30	Even	2	7
31	None	2	7

(1) This mode is only supported if there is at least a one bit delay between characters received by the SDM-SIO1

<b>Table 4-3. Communications port parameters RS-485 half duplex</b>			
<b>Code</b>	<b>Parity</b>	<b>No. stop bits</b>	<b>No. data bits</b>
48	None	1	8
49	Odd	1	8
50	Even	1	8
51	None	1	8
52	<i>Not used</i>		
53	Odd	2	8
54	Even	2	8
55	None	2	8
56	<i>Not used</i>		
57	Odd	1	7
58	Even	1	7
59 <sup>(1)</sup>	None	1	7
60	<i>Not used</i>		
61	Odd	2	7
62	Even	2	7
63	None	2	7

(1) This mode is only supported if there is at least a one bit delay between characters received by the SDM-SIO1

<b>Table 4-4. Communications port parameters RS-232 receive only mode</b>			
<b>Code</b>	<b>Parity</b>	<b>No. stop bits</b>	<b>No. data bits</b>
64	None	1	8
65	Odd	1	8
66	Even	1	8
67	None	1	8
68	<i>Not used</i>		
69	Odd	2	8
70	Even	2	8
71	None	2	8
72	<i>Not used</i>		
73	Odd	1	7
74	Even	1	7
75 <sup>(1)</sup>	None	1	7
76	<i>Not used</i>		
77	Odd	2	7
78	Even	2	7
79	None	2	7

(1) This mode is only supported if there is at least a one bit delay between characters received by the SDM-SIO1

#### 4.1.2 SerialClose

This will place the SDM-SIO1 unit into shutdown mode where only SDM communications will operate. This means any data coming into the SDM-SIO1 on the RS-232/RS-485 interface will be lost.

This is the lowest possible power mode and for optimum power efficiency the SDM-SIO1 should be placed in this mode whenever possible.

### 4.1.3 SerialIn

The primary difference when using this command compared to when it is used with a serial port built into the dataloggers is timing. If the data has already been sent by the remote sensor/system, the instruction will run in a few tens of microseconds as the data is read from internal memory. However, when using the SDM-SIO1 extra time is required to transfer data from the module and into the dataloggers own memory.

The extra time (in microseconds) taken to transfer data from the SDM-SIO1 to the logger can be calculated using the formula below.

$$\text{Time} = (C + 1) * (8 * \text{'SDMRate'})$$

Where,

C = Number of characters to transfer from the SDM-SIO1

SDMRate = Rate set using the 'SDMSpeed' command in CRBasic giving the time in microseconds for one bit period.

It's worth noting that the bit rate defined by SDMSpeed is not exact and will vary slightly depending on the logger used. If you require more accurate information about SDM data rates consult your loggers' documentation.

#### Example

The command 'SerialIn(Dest,32,1,0,10)' using a 30 $\mu$ s bit rate would transfer its 10 bytes of data from the SDM-SIO1 to the logger in approximately 2.7 ms.

$$\text{Time} = (10 + 1) * (8 * 30)$$

Note: that 30 $\mu$ s per bit is the default data rate for most Campbell loggers. It is possible to reduce this time and the transfer time by using the SDMSpeed instruction. This can be done if using short cable runs between the logger and all SDM devices.

### 4.1.4 SerialOut

Transmission from the SDM-SIO1 will commence once the first byte of user information is received from the logger. There will be a total delay of 16 bit periods (at the SDMSpeed) before transmission commences. Normally this delay can be ignored as it is short (0.48 ms), but some applications may need to account for it.

### 4.1.5 SerialInBlock

The **SerialInBlock()** instruction can be used as described in the datalogger manual. However, if using the option to capture only the most recent data, make sure the instruction is called often enough to avoid filling up the SDM-SIO1's buffer. The SDM-SIO1's buffer operates in a fill and stop mode, and therefore will not include the most recent data if it gets full.

### 4.1.6 SerialOutBlock

No special information. Can also be used to set the general purpose output line (see Appendix A).

### 4.1.7 SerialInChk

This returns the number of characters that have been received by the SDM-SIO1 and that are currently held in its buffer (0-2047).

### 4.1.8 SerialInRecord

No special information.

### 4.1.9 SerialFlush

This command will purge all information in the logger and SDM-SIO1 transmit and receive buffers.

### 4.1.10 Serial Input Errors

For any of the serial input instructions above, the character '?' will be returned in place of the expected data whenever a parity, framing or overrun error is detected. These errors are only flagged for RS-232/RS-485/RS-422 data coming into the SDM-SIO1. A large number of such characters could indicate an incorrectly setup protocol configuration using the SerialOpen command, or one of the wires is loose or incorrectly terminated.

## 4.2 Configuring handshaking and receive only modes

### 4.2.1 Using RTS/CTS and automatic handshaking

Handshaking is a method used by RS-232 to ensure communications equipment is free to receive or transmit data. This interface is often called RTS/CTS (Hardware handshaking) or DTR/DTE.

Automatic handshaking for the SDM-SIO1 is activated through use of the 'SerialOpen' commands 'BaudRate' setting. When the 'BaudRate' value is set to a negative number the SDM-SIO1 will enable automatic handshaking. It is worth noting that normally setting the 'BaudRate' to a negative number will enable automatic baud rate detection, this is not the case for the SDM-SIO1.

When enabled the two I/O ports (pins 8 and 9) will operate as RTS and CTS lines. It is important that the remote equipment supports handshaking as no data will be sent if handshaking is enabled but isn't supported by the equipment the SDM-SIO1 is connected to.

Once handshaking is enabled pins 8 and 9 are no longer available for general use and are dedicated to the auto-handshaking system.

Enabling handshaking will increase active current consumption due to the extra RS-232 load. It will not affect sleep current however as the RS-232 chip is disabled (sleep is set by calling the 'SerialClose' function).

For connection diagrams and further information on using handshaking consult Section 3.3 within this manual.

**NOTE**

- 
- Auto flow control should only be selected in RS-232 mode
  - When Autoflow control is enabled the user cannot set or clear the spare I/O ports (pins 8 and 9)
- 

**4.2.2 RS-485 half-duplex mode**

In RS-485 half duplex mode the SDM-SIO1 will wait for approximately 2.5ms after a character is received before it tries to transmit out any data in its transmit buffer. This is to ensure that a contention does not occur on the data line which will cause data corruption. There is also a delay of at least the time to send one character after the last character has been transmitted before the SDM-SIO1 will listen for new incoming data.

**NOTE**


---

When in either RS-485 half duplex or full duplex modes the user can't set or clear the spare I/O ports (Pins 8 and 9).

---

**4.2.3 Using the RS-232 link in receive only mode**

To place the SDM-SIO1 into RS-232 receive only mode, use the 'SerialOpen' command with the 'SerialOpenFormat' value set within the range of 64 to 79. Consult Section 4.1.1 in this manual for more detailed information about these settings.

In receive only mode the SDM-SIO1 will consume less current than normal but still can receive new information on its RS-232 port.

The example below will set the SDM-SIO1 with the address 0 into receive only mode. All normal baud rates and buffer sizes are supported.

**Example**

```
SerialOpen (32,115200,64,100,10000)      'Set receive only mode'
```

**4.3 Example datalogger programs**

There follows some simple examples of how to write programs in CRBasic to send and receive data using the SDM-SIO1. These programs are not extensive and are fundamentally no different to those written for use when reading data from a standard serial port.

The exact program to be used will vary with the serial device being used. Unfortunately the number of possible variations of reading different sensors is almost infinite. If you are struggling to write code please contact Campbell Scientific who may already have experience with the sensor and may be able to offer advice on how to deal with it.

## 4.3.1.1 Example using RS-232 mode

```

'-----
' Example use of the SDM-SIO1.
' This example shows how to open the a serial port using an SDM-SIO1.
' A prompt is sent from the logger to the sensor and it then waits for a response
' before reading the data.
' The logger then retrieves the data and places it into a string
'-----

Public ReturnedData as string * 100      'string where the data from the logger is stored

BeginProg

    Const SensorPort = 32                'Declare the serial port the sensor is set to
    'The sensors address switch should be set to position 0
    SDMSpeed (30)                        'Optionally set the SDMSpeed - not normally needed

    Scan(1000,mSec,0,0)

        ' Open serial port to RS-232 mode, 115200bps, 8-bit data, 1 stop bit and no parity
        SerialOpen (SensorPort,115200,3,100,10000)      'open the serial port to the sensor

        'Request data' will need to be replaced with the correct command for your sensor
        'In this example we wait for the response Start, for up to 1 second before continuing
        SerialOut (SensorPort,"Request data","Start",1,100) 'Send data to the sensor

        SerialIn (ReturnedData,SensorPort,100,0,100)    ' Get data from the sensor

        SerialClose (SensorPort) 'Close the serial port to the sensor
        '(this places the SDM-SIO1 into its lowest power mode)

        'Now there would be code to read the data out of the ReturnedData string and either store
        'it as strings or convert the string into number(s).

    Next Scan

EndProg

```

### 4.3.1.2 Example using RS-485 mode

```

'-----
' Example use of the SDM-SIO1.
' This example shows how to open the RS-485 serial port using an SDM-SIO1.
' Data is sent from the logger to the sensor.
' The program then sits in a loop until the SDM-SIO1 reports it had data
' available
' The logger then retrieves the data and places it into a string
' The returned string is then converted to a float and an offset is applied
'-----

Public ChkValReturned           'value returned by the SerialInChk function
Public AvailableData            'amount of data in the SDM-SIO1s buffer at present
Public ReturnedData as string * 100 'string where the data from the logger is stored
Public ConvertedValue as float  'floating point value returned by sensor

Sequentialmode

BeginProg

    Const SensorPort = 32        'Declare the serial port the sensor is set to
                                'The sensors address switch should be set to position 0
    SDMSpeed (30)               'Optionally set the SDMSpeed not normally needed

    Scan(1000,mSec,0,0)

        ' Open serial port to RS-485 mode, 115200bps, 8-bit data, 1 stop bit and no parity
        ' note that the 'SerialOpenFormat' parameter is 19 for RS-485 mode
        SerialOpen (SensorPort,115200,19,100,10000)      'open the serial port to the sensor

        'Request data' will need to be replaced with the correct command for your sensor
        SerialOut (SensorPort,"Request data", "",0,10)    'Send data to the sensor

        ' wait for the sensor to respond using a loop this time – this may be useful if there
        ' is not a predictable response from the sensor.
        Do
            ChkValReturned = SerialInChk (SensorPort)    ' Get available data
            AvailableData = ChkValReturned AND 4095      ' mask off the input pin flag (bit 16)
        Loop until AvailableData <> 0                    ' wait until data is available

        SerialIn (ReturnedData,SensorPort,100,0,100)    ' Get data from the sensor

        'Convert string to float
        ConvertedValue = ReturnedData

        ' add an offset to the returned floating point value
        ConvertedValue = ConvertedValue + 100

        SerialClose (SensorPort)                        'Close the serial port to the sensor
        '(this places the SDM-SIO1 into its lowest power mode)

        'Now there would be code to read the data out of the ReturnedData string and either store
        'it as strings or convert the string into number(s).

    Next Scan
EndProg

```

## 5 Firmware upgrades and flash signature errors

### 5.1 Upgrading the firmware

In the unlikely event that the operating system of the SDM-SIO1 needs to be upgraded. This can be achieved by connecting it to a PC and using the Campbell Scientific Device Configurator program to load the operating system. (This program can be downloaded free of charge from [www.campbellsci.com/downloads](http://www.campbellsci.com/downloads)).

The SDM-SIO1 needs a reliable source of 12V power connected in the standard way and a cable made similar to that shown in Section 3.3.7 to connect to an RS-232 port on the PC, except there is no requirement to connect any handshaking lines.

Start the Device configurator program and select the SDM-SIO1 device (the SDM-SIO1 device may not be available in older versions of the software. New versions can be downloaded from the Campbell website). Follow the instructions for that device to load a new operating system. At the end of the process a success message will be shown if successful. During the loading of the operating system do not disturb or disconnect power to the SDM-SIO1 otherwise it may need to be returned to the factory for repair.

### 5.2 Firmware signature errors

The operating system is stored in “flash” memory. When a new version is loaded a checksum signature is automatically created the first time its run and stored in memory too. This signature value can be read back using the logger and compared to the signature supplied with the version that was just loaded (contact Campbell Scientific if you require to do this).

If the SDM-SIO1 is not operational or is exhibiting random faults then the flash memory may be corrupted. This is a very unlikely event due to the robust nature of the flash device used. The unit automatically checks the flash memory against its signature upon power up. If an error is found it will send out the string ‘sig error:XXXX:YYYY’ when an error is detected, where ‘XXXX’ is the signature as it’s being read and ‘YYYY’ is the stored signature.

The error string is sent out via the RS-232 port automatically and, will also be returned to the datalogger the next time any attempt is made to read information from the sensor. The message will be output at the default baud rate of 9600bps, 8-bits, 1 stop and no parity.

If this error is seen contact Campbell Scientific to obtain a copy of the latest operating system and load it into the SDM-SIO1 using the above procedure. If this does not correct the fault then the unit may be faulty and will need to be returned to the factory for repair.

### 5.3 Checking the firmware version and signature

It is not possible to check the current firmware version using the device configuration program as the SDM-SIO1 is designed as a transparent RS232 device. Instead the firmware version can be checked by loading a program into the logger to which it is attached. The program below shows use of the SDMGeneric command to ask the SDM-SIO1 with a specific address what the firmware version is (Ver\_value) and the signature of the firmware (Sig\_value).



```

'-----
' Simple OS version and OS sig check for the SDM-SIO1
'-----

'Variables and constants for the version number and signature checking
Public Ver_Value As String * 25 'Holds version as text string
Public Sig_Value As String * 4 'Holds SIG of OS as four byte HEX string
Public Sig_Value_Dec          'Holds sig as a decimal number

'Change this address to match the SDM-SIO1 SDM address.
Const SDM_Address=0

Dim Src As String * 1
Const cmd = 5           'constant cmd = 0..7
Const bytes_out = 1    'constant number of bytes out
Const bytes_val = 1    'constant bytes per value (1,2,4)
Const big_endian = 1   'constant 1 (big endian) or 0 (little endian)
Const delay_usec = -0  'constant delay between outgoing bytes (negative means delay also for incoming
bytes)
Const Ver_values_in = 20 'constant number of values to receive
Const Sig_values_in = 4  'constant number of values to receive

Sequentialmode

BeginProg
SDMSpeed (30) 'Fix the speed
Ver_Value = ""
Sig_Value = ""

Scan(1,Sec,0,0)
'Use the generic SDM command to get extra info from the SDM-SIO1
'Ask for the firmware version
Src = CHR(1)
SDMGeneric(Ver_Value,SDM_Address,cmd,bytes_out,Src,Ver_values_in,bytes_val, big_endian,delay_usec)
' Read signature
Src = CHR(2)
SDMGeneric(Sig_Value,SDM_Address,cmd,bytes_out,Src,Sig_values_in,bytes_val,big_endian,delay_usec)
Sig_Value_Dec = HexToDec (Sig_Value) 'Convert sig to decimal too.
Next Scan

EndProg

```



# Appendix A. Using the Handshaking lines for general input/output

---

*This Appendix describes how to use the CTS and RTS lines for input and output ports.*

The I/O pins (CTS/pins 8 and RTS/9) can be read or set by the user as required allowing unique protocols to be created, or they can simply be used as flags or enable lines.

If the user enables automatic handshaking as discussed earlier then pins 8 and 9 can't be accessed.

## A.1 The input pin (CTS/pin 8)

The state of the input line (pin 8) can be read by using the special SDMGeneric instruction in CRBasic.

Bit 16 of a value returned to the logger using the SDMGeneric command reports whether the input is logic high or low. The example below shows how to call the command with the required parameters.

```
'CR1000 Example program showing how to detect the state of the input line on an SDM-SIO1
'Using the SDMGeneric instruction

Public sio1response As Long, Inputstate As Boolean
Const addr = 0 'constant sdm address CHANGE with SDM-SIO1 address
Const cmd = 1 'constant, command 1 returns the input state in bit 16
Const bytes_out = 0 'constant number of bytes out - none sent
Const bytes_val = 2 'two bytes returned
Const big_endian = 1 '1=(big endian)
Const delay_usec = -0 'delay between outgoing bytes (negative means delay also for incoming bytes)
Const Values_in = 1

'Main Program
BeginProg
  Scan (1,Sec,0,0) 'Or whatever scan rate is being used

  ' Use the SDMGeneric instruction to send command 0x01 which returns the input line state as bit 16
  SDMGeneric(sio1response,addr,cmd,bytes_out,"",Values_in,bytes_val, big_endian,delay_usec)
  'bit 16 set for input line high or low so check the value and set a boolean var TRUE or FALSE accordingly
  Inputstate = sio1response AND &H8000
NextScan
EndProg
```

The input line can accept 0 and 5V logic inputs OR -12 and +12V RS232 level inputs. Below is a break down of the different input voltages allowed and the state of the input line flag.

Voltage on the input line	State of bit 16
+12V	1
-12V	0
5V	1
0V	0

## A.2 The output pin (RTS/pin 9)

To set the output pin (RTS/pin 9) you must use the ‘SerialOutBlock’ command.

The spare I/O lines are RS232 lines NOT logic lines. This means that the output line voltage is -12V and +12V (approximately) not 0 and 5V.

The following examples show you how to set the output pin using the ‘SerialOutBlock’ instruction.

### CRBasic example for setting the output pin

`SerialOutBlock(32, 1, 0)` ‘ This will set the spare output pin high

`SerialOutBlock(32, 0, 0)` ‘ This will set the spare output pin low

The two example lines of code above will set the output pin on the SDM-SIO1 high then low respectively on the SDM-SIO1 device set to address 0 on its rotary switch.

Any value greater than 1 will set the output pin high, but usually the number 1 is used, as this will improve the readability of the code.

Value of ‘SerialOutBlock’ ‘SerialExpression’ variable	Voltage on the output line
1	+12V <sup>(1)</sup>
0	-12V <sup>(1)</sup>

(1) Approximate voltage; do not drive relays or high current loads directly from this pin



## CAMPBELL SCIENTIFIC COMPANIES

### **Campbell Scientific, Inc. (CSI)**

815 West 1800 North  
Logan, Utah 84321  
UNITED STATES

[www.campbellsci.com](http://www.campbellsci.com) • [info@campbellsci.com](mailto:info@campbellsci.com)

### **Campbell Scientific Africa Pty. Ltd. (CSAf)**

PO Box 2450  
Somerset West 7129  
SOUTH AFRICA

[www.csafrica.co.za](http://www.csafrica.co.za) • [sales@csafrica.co.za](mailto:sales@csafrica.co.za)

### **Campbell Scientific Australia Pty. Ltd. (CSA)**

PO Box 8108  
Garbutt Post Shop  
QLD 4814 AUSTRALIA

[www.campbellsci.com.au](http://www.campbellsci.com.au) • [info@campbellsci.com.au](mailto:info@campbellsci.com.au)

### **Campbell Scientific do Brazil Ltda. (CSB)**

Rua Apinagés, nbr. 2018 - Perdizes  
CEP: 01258-00 São Paulo SP BRAZIL

[www.campbellsci.com.br](http://www.campbellsci.com.br) • [vendas@campbellsci.com.br](mailto:vendas@campbellsci.com.br)

### **Campbell Scientific Canada Corp. (CSC)**

14532 – 131 Avenue NW  
Edmonton, Alberta T5L 4X4  
CANADA

[www.campbellsci.ca](http://www.campbellsci.ca) • [dataloggers@campbellsci.ca](mailto:dataloggers@campbellsci.ca)

### **Campbell Scientific Centro Caribe S.A. (CSCC)**

300N Cementerio, Edificio Breller  
Santo Domingo, Heredia 40305  
COSTA RICA

[www.campbellsci.cc](http://www.campbellsci.cc) • [info@campbellsci.cc](mailto:info@campbellsci.cc)

### **Campbell Scientific Ltd. (CSL)**

80 Hathern Road, Shepshed, Loughborough LE12 9GX  
UNITED KINGDOM

[www.campbellsci.co.uk](http://www.campbellsci.co.uk) • [sales@campbellsci.co.uk](mailto:sales@campbellsci.co.uk)

### **Campbell Scientific Ltd. (France)**

3 Avenue de la Division Leclerc  
92160 ANTONY  
FRANCE

[www.campbellsci.fr](http://www.campbellsci.fr) • [info@campbellsci.fr](mailto:info@campbellsci.fr)

### **Campbell Scientific Spain, S. L.**

Avda. Pompeu Fabra 7-9  
Local 1 - 08024 BARCELONA  
SPAIN

[www.campbellsci.es](http://www.campbellsci.es) • [info@campbellsci.es](mailto:info@campbellsci.es)

### **Campbell Scientific Ltd. (Germany)**

Fahrenheitstrasse 13, D-28359 Bremen  
GERMANY

[www.campbellsci.de](http://www.campbellsci.de) • [info@campbellsci.de](mailto:info@campbellsci.de)

### **Campbell Scientific (Beijing) Co., Ltd.**

8B16, Floor 8 Tower B, Hanwei Plaza  
7 Guanghua Road, Chaoyang, Beijing 100004  
P.R. CHINA

[www.campbellsci.com](http://www.campbellsci.com) • [info@campbellsci.com.cn](mailto:info@campbellsci.com.cn)

*Please visit [www.campbellsci.com](http://www.campbellsci.com) to obtain contact information for your local US or International representative.*