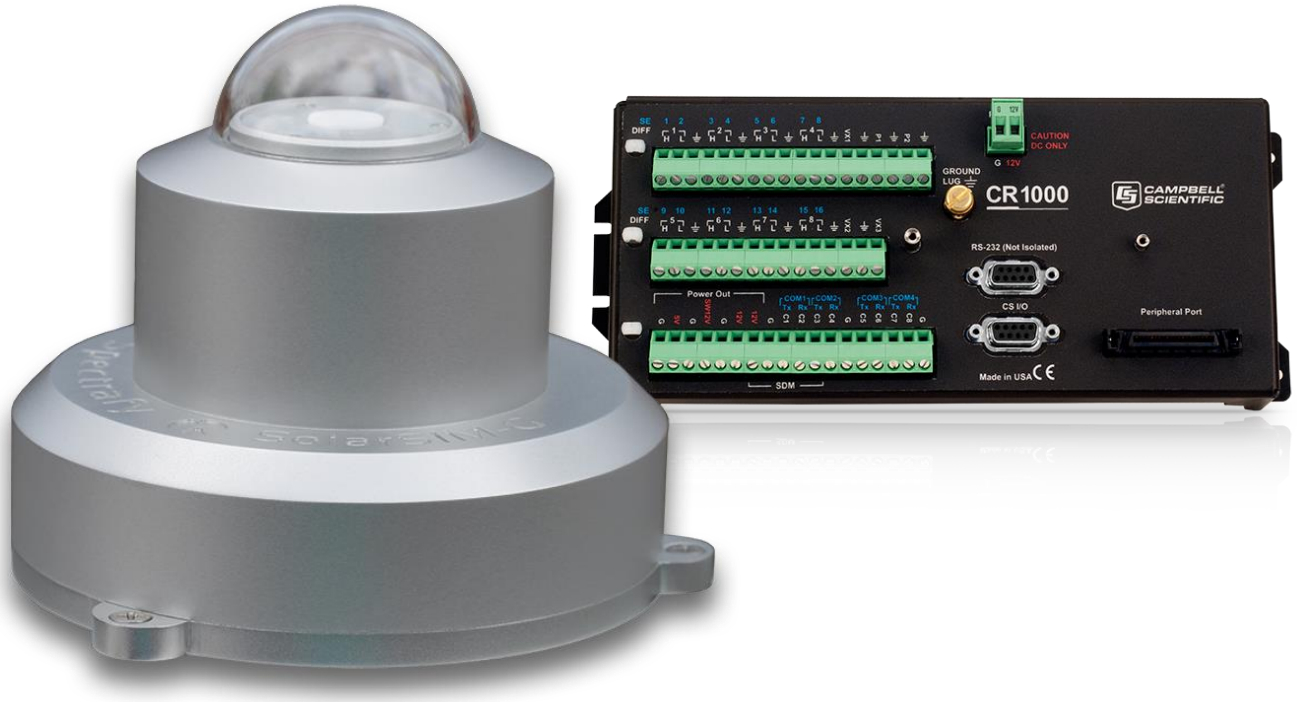
Spectrafy™
solar spectral sensors

Application Note:
Connecting the SolarSIM-G to a Campbell Scientific CR1000 datalogger



1. Introduction

This application note describes the options and provides sample code controlling a SolarSIM-G through a Campbell Scientific (CS) datalogger (e.g. CR1000, CR3000).

2. Components required

The SolarSIM-G uses the RS-485 communication protocol. Unfortunately, this protocol is not a standard feature on CR1000 dataloggers and the user must therefore employ an RS-485/RS-232 converter of some sort. Currently, Spectrafy does not offer an RS-485/RS-232 adapter accessory. As such, a third-party solution must be employed. Two options exist:

1. Use the MD485 RS-485 Multidrop Interface, by Campbell Scientific.

Figure 1. Campbell Scientific MD485 Multidrop interface

2. Assemble the following components (or their equivalents):
    a. The Port-Powered RS232-RS485 Converter, by US Converters (PN: XS201A).
    b. The DB9 M/M mini Gender Changer, by US Converters (PN: CN05B).
    c. The DB9 Terminal Block Header, by US Converters (PN: DBT9)

Figure 2. From left to right - Port-Powered RS232 to RS485 Converter, DB9 M/M Gender Changer, and DB9 Terminal Block Header

4 Florence Street, Suite 204, Ottawa, Canada K2P 0W7 | +1 (613) 237-2020 | info@spectrafy.com | www.spectrafy.com

3. Wiring

The wiring guide for the SolarSIM-G is as follows:

Colour	Label	Function
Blue	V_{in}	Input voltage (+12 VDC)
White	GND	Common ground
Black	D-	Negative RS-485 input
Brown	D+	Positive RS-485 input

a. Wiring to an MD485

On the MD485, the RS-485 port pins are labelled as G, A and B. Connect the SolarSIM-G as follows:

SolarSIM-G wire colour	MD485 pin	Function
White	G	Common ground
Black	A	Negative RS-485 input
Brown	B	Positive RS-485 input

The SolarSIM-G's blue wire must be connected to a 12VDC power supply - typically the 12VDC Power Out on the CS datalogger.

b. Wiring to a RS232 to RS485 converter (RS-485 side)

The SolarSIM-G is wired into the terminal board (RS-485 side) of the Port-Powered RS-232 to RS485 converter as follows:

SolarSIM-G wire colour	XS201A terminal block pin	Function
White	3	Common ground
Black	2	Negative RS-485 input
Brown	1	Positive RS-485 input

The SolarSIM-G's blue wire must be connected to a 12VDC power supply – for example the 12VDC Power Out on the underlying CS datalogger.

Note: the XS201A RS232-RS485 converter itself, is powered through the RS232 port. In the event that the RS232 port is not capable of supplying sufficient power, Pin 4 on the XS201A terminal board can be connected to an external 5-12V power supply – such as that provided by the CS datalogger.

4 Florence Street, Suite 204, Ottawa, Canada K2P 0W7 | +1 (613) 237-2020 | info@spectrafy.com | www.spectrafy.com

c. Wiring to a RS232-RS485 converter (RS-232 side)

Interface the RS-232 side of the RS-232 to RS-485 converter with a DB9 Terminal Block Header with the aid of the DB9 M/M Gender Changer. The DB9 Terminal Block Header is wired to CR1000's COM1 serial port as follows:

DBT9 terminal block pin	CR1000 pin	Function
2	C2	RS-232 receive terminal
3	C1	RS-232 transmit terminal
5	GND	Common ground

4. Datalogger communication

a. Serial commands

There is only one serial command required to acquire data from the SolarSIM-G:

```
N1000_E
```

In return, the SolarSIM-G sends an ASCII string with the ambient temperature, pressure and relative humidity, the internal temperature and relative humidity, and nine voltages. The following ASCII string is a sample output:

N1010_2500.000,1013.120,4750.000,2600.000,1050.000,2500.032,4999.999,
0000.001,1274.004,2746.321,3291.214, 3924.385,1900.500,0500.123/r/n

The aforementioned string can be parsed as:

N"serial number"_"(Tout + 50) x 75" , "P_{out} x 10" , "H_{out} x 100" , "(T_{in} + 50) × 75" , "H_{in} x 100" , "V1", "V2" , "V3" , "V4" , "V5" , "V6" , "V7" , "V8" , "V9" , "end of line character"

Where:
T_{out} = ambient temperature
P_{out} = ambient atmospheric pressure
H_{out} = ambient relative humidity
T_{in} = internal SolarSIM-G temperature
H_{in} = internal SolarSIM-G relative humidity

4 Florence Street, Suite 204, Ottawa, Canada K2P 0W7 | +1 (613) 237-2020 | info@spectrafy.com | www.spectrafy.com

The aforementioned example string is parsed in the table below:

Parameter	Symbol	Value	Units
Ambient temperature	T_{out}	-16.67	℃
Ambient pressure	P_{out}	101.312	kPa
Ambient humidity	H_{out}	47.50	%
Internal temperature	T_{in}	-15.33	℃
Internal humidity	H_{in}	10.50	%
Voltage channel 1	V_1	2500.032	mV
Voltage channel 2	V_2	4999.999	mV
Voltage channel 3	V_3	0.001	mV
Voltage channel 4	V_4	1274.004	mV
Voltage channel 5	V_5	2746.321	mV
Voltage channel 6	V_6	3291.214	mV
Voltage channel 7	V_7	3924.385	mV
Voltage channel 8	V_8	1900.500	mV
Voltage channel 9	V_9	500.123	mV

b. Serial port configuration

The serial port for SolarSIM-G should be configured as follows:

Parameter	Value
Baud rate	9600
Parity	None
Data bits	8
Stop bits	1

4 Florence Street, Suite 204, Ottawa, Canada K2P 0W7 | +1 (613) 237-2020 | info@spectrafy.com | www.spectrafy.com

5. Sample CS Datalogger code

WARNING: this code has been supplied by a customer where it was successfully used on a CR1000 to communicate with a SolarSIM-D2. It has been modified here for use with a SolarSIM-G. It has been successfully compiled in the CRBasic Editor, but not field-tested in this application.

Example code for CR1000 datalogger:

```
'Description: Acquires raw data from the SolarSIM-G
'Datalogger:  CR1000 from Campbell Scientific
'Author:      Spectrafy Inc.
'Date:        April 16, 2018

'Defines user constants
Const Timezone     = -5 'hrs
Const DAQRate      = 60 's
Const SamplingRate = 5  's

'Defines program constants
Const TerminationCharacter = CHR(10) + CHR(13)
Const SerialCommand = "N1000_E" + TerminationCharacter

'Declares public variables
Public SerialData As String *256
Public OutputData(14)

'Declares data table column names
Alias OutputData(1)  = AmbientTemperature   'C
Alias OutputData(2)  = AmbientPressure      'kPa
Alias OutputData(3)  = AmbientHumidity      '%
Alias OutputData(4)  = InternalTemperature  'C
Alias OutputData(5)  = InternalHumidity     '%
Alias OutputData(6)  = ChannelVoltage1      'mV
Alias OutputData(7)  = ChannelVoltage2      'mV
Alias OutputData(8)  = ChannelVoltage3      'mV
Alias OutputData(9)  = ChannelVoltage4      'mV
Alias OutputData(10) = ChannelVoltage5      'mV
Alias OutputData(11) = ChannelVoltage6      'mV
Alias OutputData(12) = ChannelVoltage7      'mV
Alias OutputData(13) = ChannelVoltage8      'mV
Alias OutputData(14) = ChannelVoltage9      'mV

'Defines data table
DataTable (Spectrafy_G,1,-1)      'Autoallocates table size
  DataInterval (0,DAQRate,Sec,10) 'Sets the DAQ rate
  Sample (1,Timezone,FP2)         'Stores timezone
  FieldNames("Timezone")          'Names "Timezone" column
  Average (14,OutputData,IEEE4,0) 'Stores raw data
EndTable
```


4 Florence Street, Suite 204, Ottawa, Canada K2P 0W7 | +1 (613) 237-2020 | info@spectrafy.com | www.spectrafy.com

```
'Executes main program
BeginProg
  'Initializes serial port
  SerialOpen (Com1,9600,0,0,256)     'Serial communication on port "Com1"
                                     'Baud rate: 9600 bps
                                     'Buffer size: 256 bytes
  'Sets a 5 s scan interval
  Scan (SamplingRate,Sec,0,0)

      'Transmits the broadcast command
            SerialOut (Com1,SerialCommand,"",0,0)

            'Receives serial data with a 1000 ms timeout
            SerialIn (SerialData,Com1,100,TerminationCharacter,256)

      'Clears the serial buffer
            SerialFlush (Com1)

      'Removes the header from the serial data
            SerialData = Mid(SerialData,7,256)

      'Parses the serial data into numeric values
            SplitStr (OutputData(),SerialData,"",14,0)

      'Converts raw data into meteorological data
       AmbientTemperature = (AmbientTemperature / 75.0) - 50.0
       AmbientPressure = AmbientPressure / 10.0
       AmbientHumidity = AmbientHumidity / 100.0
       InternalTemperature = (InternalTemperature / 75.0) - 50.0
       InternalHumidity = InternalHumidity / 100.0

      'Passes raw data to "Spectrafy" table
            CallTable Spectrafy_G
  NextScan
EndProg
```

6. Support

If you have any questions regarding your specific application, don't hesitate to contact Spectrafy at info@spectrafy.com.

4 Florence Street, Suite 204, Ottawa, Canada K2P 0W7 | +1 (613) 237-2020 | info@spectrafy.com | www.spectrafy.com