

Interfacing the eosGP to a Campbell Scientific CR1000 logger

Introduction

The Campbell Scientific CR1000 is a popular data logger which provides a plethora of options for connecting and recording multiple sensors. The Eosense eosGP CO₂ gas probe has multiple output options. This note is provided to help you decide which connection method is right for your application and demonstrate the connection process.

Required Components

- eosGP
- Campbell Scientific CR1000 data logger
- Short Cut software (CR1000)
- SSC/SLC
- Power Connector
- Serial Extension Cable (Serial Only)
- Ground Connector (Analog Only)

Serial Mode

Why would you want to use your data logger's serial input with the Eosense CO₂ gas probe? Unlike basic analog sensors such as thermistors, the eosGP uses an on-board computer to make measurements and apply calibrations to calculate a highly accurate CO₂ concentration reading. For the convenience of users with basic analog-only data loggers, we supply simulated analog output signals, but for maximum resolution and noise immunity, it is always preferable to use the serial output.

The CR1000 allows up to 4 serial inputs on its control I/O ports in addition to analog inputs, allowing synchronization between serial and analog sensor readings. Simply connect pin 2 of the eosGP's serial extension cable to the Rx input of the CR1000's Com port, and pin 5 to a digital ground (usually labelled 'G') as shown in Figure 1.

The GP can either be powered from the logger's 12 V supply or via a separate supply. In this example, Com4 is used, but with a small change to the program, any of the 4 control I/O Com ports can be used. Using the CR1000's dedicated RS232 port is more difficult since this connector is used to upload programs and download data from the logger and is beyond the scope of this application note.

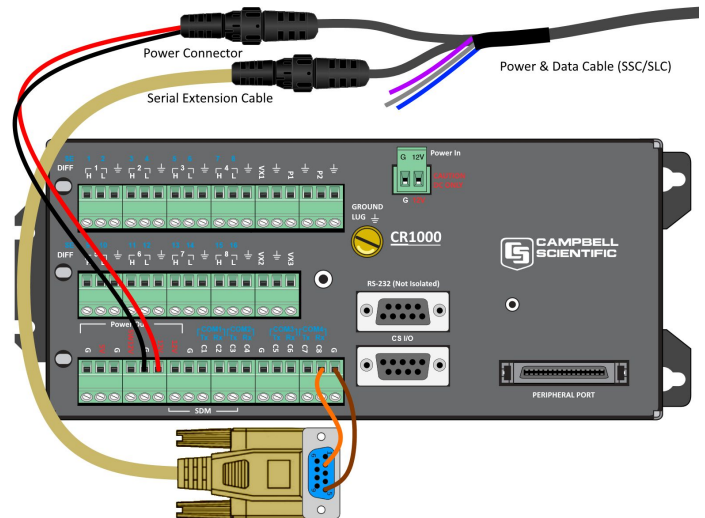


Figure 1: Connecting the eosGP to a CR1000 in serial mode ([printable image](#))

A demonstration program is shown in listing below (and also available as a [download](#)). The blue text represents a basic CR1000 program which in the case of the demo measures and records battery voltage and internal temperature, though it will obviously vary based on your specific application. The red text represents the lines of program code which are added to extract and record the CO₂ concentration, high range CO₂ concentration, and temperature from the eosGP.

The lines starting with "SerialOpen" and "SerialIn" contain references to "Com4". These can be changed to "Com1", "Com2" or "Com3" to make use of the other serial ports. The eosGP outputs a new serial reading approximately every 5 seconds. When each new serial message arrives, the concentrations and temperature are updated ("Scan" should have a frequency of less than 5 seconds). Calling the data table will store these most recent values. If the "DataTable" frequency is faster than 5 seconds, the same readings will be stored multiple times since the data is only updated every 5 seconds. While the demonstration program was developed specifically for the CR1000, the techniques are applicable to other data logger models like the CR800, CR6 and CR3000 with minor changes to the program.

```

'CR1000 GP Serial Demo
'Created by Eosense

'Declare Variables and Units
Public BattV
Public PTemp_C
Public SerialInput As String * 50
Public SerialFields(6) As String * 16
Public GP_CO2Conc as FLOAT
Public GP_CO2HiConc as FLOAT
Public GP_Temp as FLOAT

Units BattV=Volts
Units PTemp_C=Deg C
Units GP_CO2Conc=ppm C02
Units GP_CO2HiConc=ppm C02
Units GP_Temp=Deg C

'Define Data Tables
DataTable(Table1,True,-1)
DataInterval(0,1,Min,10)
Sample(1,BattV,FP2)
Sample(1,PTemp_C,FP2)
Sample(1,GP_CO2Conc,FLOAT)
Sample(1,GP_CO2HiConc,FLOAT)
Sample(1,GP_Temp,FLOAT)
EndTable

DataTable(Table2,True,-1)
DataInterval(0,1440,Min,10)
Minimum(1,BattV,FP2,False,False)
EndTable

'Main Program
BeginProg
'Initialize Serial Port
SerialOpen(Com4,115200,0,0,100)
'Main Scan
Scan(2,Sec,1,0)
'Default Datalogger Battery Voltage measurement

'BattV'
Battery(BattV)
'Default Wiring Panel Temperature measurement

'PTemp_C'
PanelTemp(PTemp_C,_60Hz)
'Eosense GP CO2 measurements
'Read GP Serial Data
SerialIn(SerialInput,Com4,6,&h0D,80)
'If new data is available, parse and convert
if InStr(1,SerialInput,"["",2) then
'Separate input into fields
SplitStr(SerialFields(),SerialInput,
)"],["",6,0)
'Store CO2 Concentration as floating point number
GP_CO2Conc=SerialFields(2)
'Store Temperature as floating point number
GP_Temp=SerialFields(3)
'Store CO2 High Range Concentration as fpn
GP_CO2HiConc=SerialFields(5)
endif
'Call Data Tables and Store Data
CallTable Table1
CallTable Table2
NextScan
EndProg

```

Text 1: Source code listing (GP_Serial_Demo.CR1)

Differential Mode

If serial mode is unavailable, the eosGP's analog outputs can be connected to the CR1000's differential inputs. Remove the serial extension cable from the power & data cable and replace it with the included ground connector. This supplies a separate ground reference for each analog output. The analog/ground pairs can then be connected to high/low (H/L) differential input pairs on the CR1000 as shown in Figure 2. The power can remain connected to 12 V output of the CR1000 or can use a separate supply.

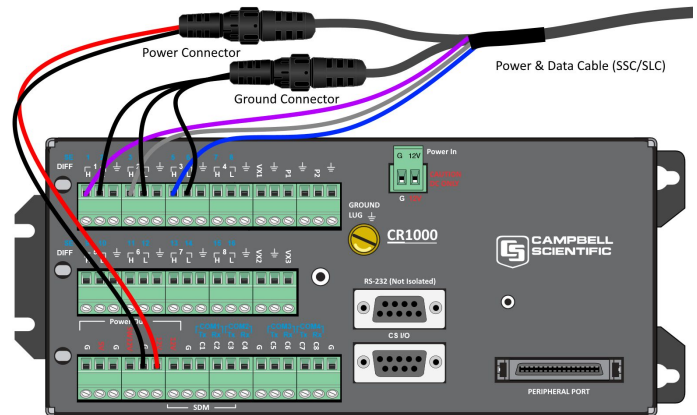


Figure 2: Connecting the eosGP to a CR1000 in differential mode ([printable image](#))

This setup can be configured using Campbell's Short Cut program as there is no complicated programming involved. In the sensor section, select "Differential Voltage" and add 3 sensors. Adjust the properties for each sensors as per Table 1 if you have a standard dual-range eosGP (5000 ppm and 20000 ppm).

Table 1: Sensor Properties

| Sensor Property | Purple Wire | Grey Wire | Blue Wire |
|----------------------------|--------------|--------------|--------------|
| Measurement Result (name) | CO2_Low | CO2_High | Temperature |
| Measurement Result (units) | ppm | ppm | Degrees C |
| Range of Sensor Voltage | 0 to 5000 mV | 0 to 5000 mV | 0 to 5000 mV |
| Multiplier | 1.2 | 4.8 | 0.014 |
| Offset | 0 | 0 | -20 |

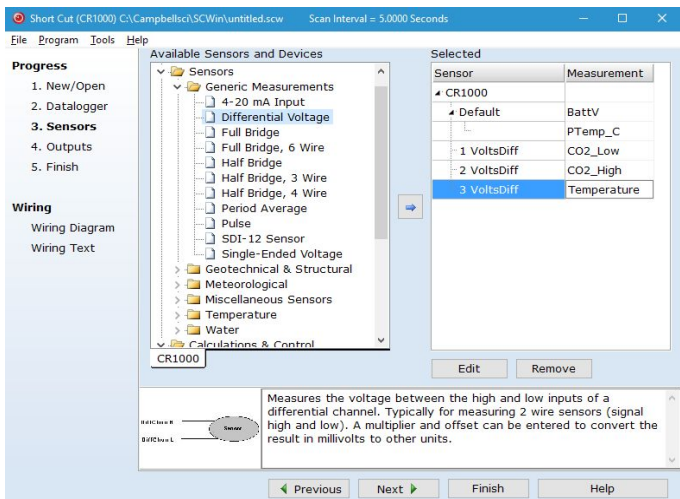


Figure 3: Short Cut differential set-up

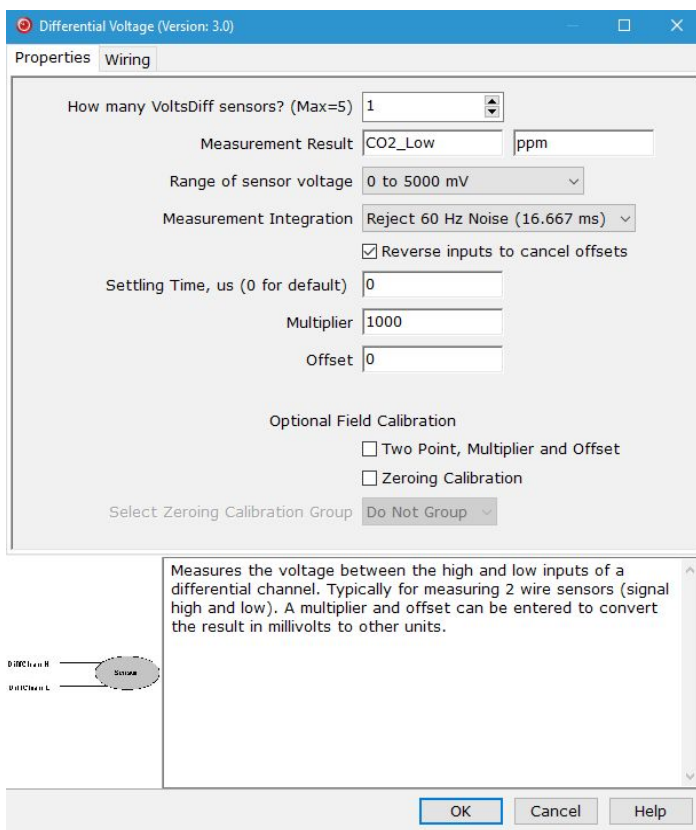


Figure 4: Short Cut properties box

Single-Ended Mode

If serial and differential mode are unavailable (perhaps due to lack of unused inputs) the eosGP can be connected in single-ended mode, however this is the least accurate mode and should only be used when there are no other options. Because the eosGP draws large but short-duration current

pulses to power its NDIR sensor, a varying amount of voltage is dropped across the ground lead of the cable (an effect that gets worse with longer cables). This time-varying noise signal can be diminished by moving the eosGP power connector to an isolated power supply (i.e. separate from the CR1000's supply with no common ground connection) as shown in Figure 5. The configuration in Short Cut uses "Single-Ended Voltage" as the sensor type, but all other settings are the same as for differential mode (Table 1).

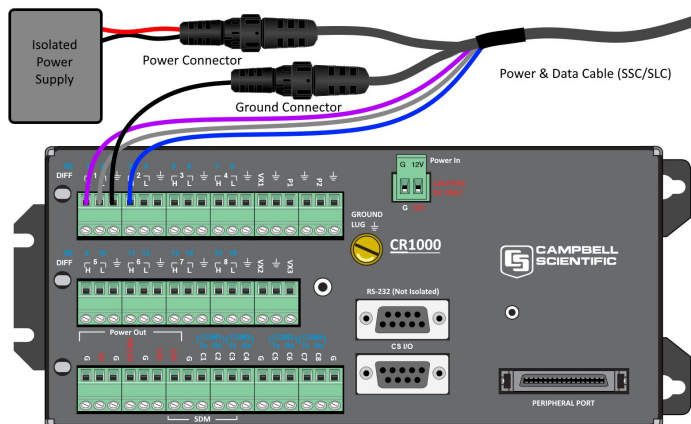


Figure 5: Connecting an eosGP to a CR1000 in single-ended mode ([printable image](#))

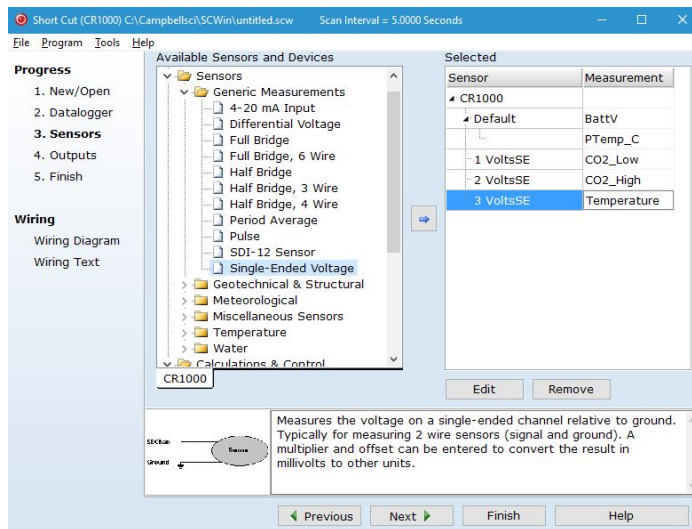


Figure 6: Short Cut single-ended set up

Conclusion

Whether you choose to use the serial, differential or single-ended mode to set up your eosGP with the Campbell Scientific CR1000 data logger, set up should be straight forward to get up and running. Any further questions can be directed to our support team.