

How to Integrate Digiquartz® Intelligent Products with Campbell Scientific CR1000 Series Data Loggers



Paroscientific, Inc.
4500 148th Ave. N.E.
Redmond, WA 98052, USA
Tel: (425) 883-8700 Fax: (425) 867-5407
www.paroscientific.com
support@paroscientific.com

“The standard by which other standards are measured”

How to Integrate Digiquartz® Intelligent Products with Campbell Scientific CR1000 Series Data Loggers

by
Mustafa Yilmaz
support@paroscientific.com

1. Introduction

One important step in scientific research is to observe a phenomenon for a period of time and collect data. Conclusions are drawn after analyzing the data. As much as it is important to collect data, how data are collected is equally vital. The quality and reliability of the data acquisition system are of utmost importance since unreliable data can result in wrong conclusions. Besides being expensive and ineffective, such tests can even jeopardize human life and property. Therefore, it is crucial to design a proper test system with accurate, reliable, repeatable, stable data collection systems including sensors.

Campbell Scientific and Paroscientific, two leading companies in their fields, provide components of a scientific data acquisition system to collect data to observe geophysical and environmental applications. Campbell Scientific designs and manufactures data loggers. Paroscientific designs and manufactures high accuracy pressure sensors including depth sensors, portable pressure standards and meteorological measurement systems. Campbell Scientific data loggers and Paroscientific instruments are widely used to capture high accuracy, high resolution, and reliable data for scientific applications. The purpose of this technical note is to facilitate the integration of Digiquartz® Intelligent Instruments with Campbell Scientific CR1000 data logger.

2. Campbell Scientific CR1000 Data Logger

Data loggers are highly accurate voltmeters that are a part of a data acquisition system that converts electrical signals into data and stores them in its internal memory. Data loggers can measure most sensors with electrical output.

The CR1000 is a low power, wide temperature range data logger with flexible programming and telecommunication options. It comprises the most advanced data logging technology from Campbell Scientific. It provides precision measurement capabilities in a rugged, battery operated package. The data logger includes a CPU and analog and digital inputs and outputs. The on-board programming engine includes data processing and analysis routines. LoggerNET, PC200 and PC400 software provide program generation, editing, data retrieval and real-time monitoring.

3. Digiquartz® Intelligent Products

Paroscientific Intelligent Transmitters consist of a Digiquartz® Pressure Transducer and a digital interface board in an integral package. Commands and data requests are sent via two-way RS-232 or RS-485 serial interfaces. Digital outputs are provided directly in engineering units with typical accuracy of 0.01% over a wide temperature range. Output pressure is fully thermally compensated using a quartz

crystal temperature signal. All intelligent transmitters are preprogrammed with calibration coefficients to provide temperature compensated pressure output.

Intelligent Transmitters are available in 19 absolute pressure ranges from 15 psia (0.10 MPa) to 40000 psia (276 MPa), 6 gauge pressure ranges from 15 psig to 200 psig, and differential pressure ranges of 3, 6, and 18 psid. Dual RS-232 and RS-485 interfaces allow complete remote configuration and control of all operating parameters, including resolution, sample rate, choice of engineering units, integration time, and sampling requests.

Digiquartz Intelligent Products can be packaged in the form of a depth sensor, portable pressure standards and meteorological measurement systems. All Digiquartz products with Intelligent RS-232 output can be integrated with a CR1000 data logger. This paper elaborates the procedure for integrating the following Digiquartz Intelligent products to a CR1000 via RS-232 communication link.

- a. Transmitters (Series 1000, Series 6000 and Series 9000) – RS-232 and RS-485 interfaces
- b. Depth Sensors (Series 8000) – RS -232 and RS-485 interfaces
- c. Meteorological Measurement Systems (MET1, MET3 and MET3A) – RS-232 interface
- d. Portable Laboratory Standards (Model 745) – RS-232 interface

If you are interested in an integrated data logging solution with Digiquartz® accuracy, Digiquartz® Model 765 High Accuracy Pressure Standard provides turn-key solution for data logging applications. For further information on the Digiquartz Model 765s, please consult a Paroscientific application engineer.

4. Installation/Integration Steps

CR1000 data logger has 16 SE (single ended) or 8 differential channels (DIFF) and four COM ports to read data from external sensors. Digiquartz® Intelligent products can be interfaced via either of the following types of ports on the CR1000 data logger. The port selection decision is mostly dictated by your application:

- 9-pin RS-232 (not isolated) port
- CS I/O Port (Campbell Scientific Input/Output)
- COM ports (marked in blue)

In most cases, since there will be other products or accessories using the 9-pin RS-232 or the CS I/O port, this technical note will focus on integrating the Digiquartz Intelligent products via the COM ports (TX/RX lines). In this paper, we assume that COM1 is used for integration.

The connection of a Digiquartz® Intelligent instrument to a CR1000 has the following steps. It is assumed that you have already established communication between your PC and data logger and set the necessary configuration parameters in your data logger. To learn more about the data logger configuration details, please consult CR1000 user's manual of your data logger.

- Verify/establish communication with Digiquartz® Intelligent Product via COM port of your PC. Write down the instrument's communication parameters.
- Make the necessary electrical and data line connections to CR1000 data logger.
- Program the data logger to acquire data from Digiquartz® Instrument.
- Verify that the logger is acquiring and storing data.
- Locally or remotely download/view data.

4.1 Verify/establish communication with Digiquartz® Intelligent Product via RS-232 port of your PC. Write down the instrument communication parameter settings.

Load Digiquartz Interactive Software (DQI) that came with your intelligent product on your PC. Connect the intelligent product to the RS-232 port of your PC. Launch the DQI software and detect the intelligent unit automatically. Once the unit is detected and communication is established, see Figure 1. Write down the instrument ID, ComPort number, baud rate, and parity from this window.

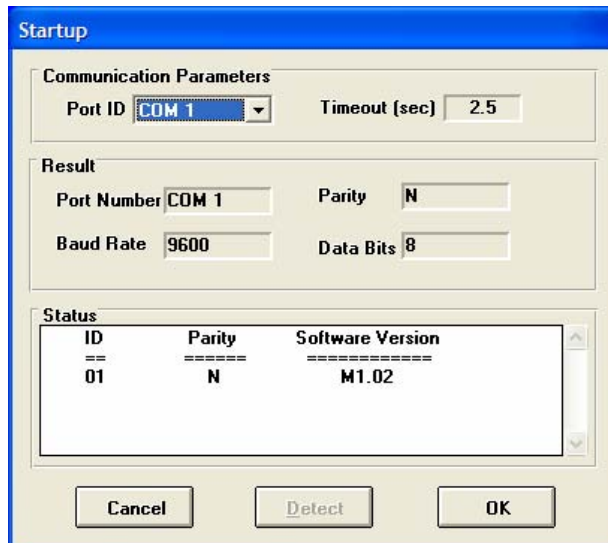


Figure 1. DQI Startup Form

Use DQI to change other parameters on the intelligent unit such as pressure unit or integration time.

4.2 Make necessary electrical and data line connections to data logger

Disconnect the intelligent instrument from the PC. Connect the intelligent instrument to the CR1000. Locate the pin numbers for the pins labeled in Table 1 below. Pin numbers can be found either on the product Specification Control Drawing (SCD) or in the product user’s manual delivered with the intelligent product. Write the pin numbers in the table below.

Pin	Signal	Connection on CR1000
	RS-232 TX	COM1 RX
	RS-232 RX	COM1 TX
	Ground	G
	Power	12V

Table 1. Pinout Table

Connect each pin in Table 1 to its location in Figure 2.

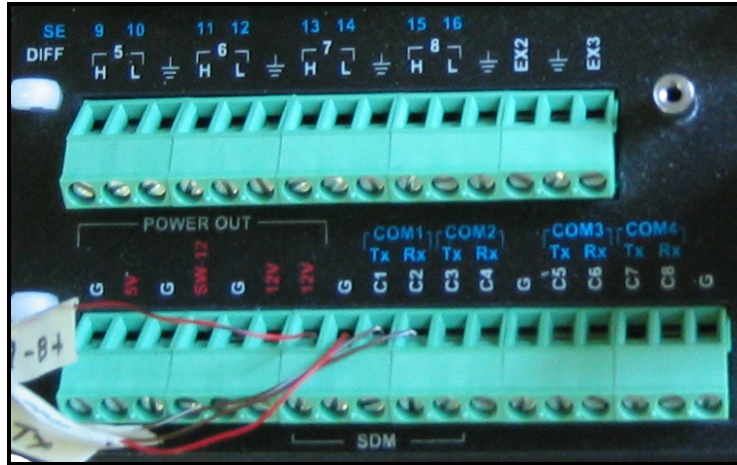


Figure 2. CR1000 COM1 Data and Electrical Connection

Connect the RS-232 TX and RX wires first. Make sure that the TX wire from the sensor is connected to RX port of COM1 and RX wire to TX port. Connect ground wire to G and power wire to 12V terminal on the CR1000. Tighten the screws and make sure that there is contact between the sensor RS-232, power and ground wires and the screws on each port.

4.3 Program the data logger to acquire data from Digiquartz® Intelligent Instrument

A program must be provided for the data logger in order to make measurements and store data. The CR1000 data logger has a total memory of 2 MB. Approximately 1400 Kbytes of memory is available for data table storage. Additional storage space can be provided by using an optional compact flash card. The memory has battery back up. When power is lost or the data logger is powered down, programs and data remain in memory. If power is lost, an active program will start running again when the power is back.

Stored data is maintained in table format. In addition to data tables, there are public and status tables. The public table holds variables and the status table holds variables that define logger status. Each data table has its own file. **Tables are erased when program is loaded and run.** The number and size of data tables is defined by the individual CR1000 program. Numerous data tables can be used to store different data on different intervals. The CR1000 can “auto allocate” its tables; so, the newest data begins to overwrite the oldest data at approximately the same time. There is a field in the **Status Table** that indicates how long it will take before the oldest data in the table is overwritten by the newest data.

There are several ways to program the CR1000. The simplest way is to use Campbell Scientific’s software called **ShortCut**. For more complex programming, the **CRBasic** editor can be used.

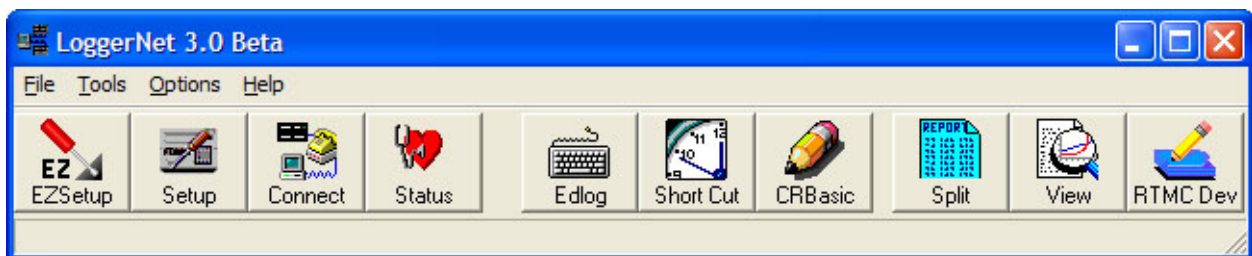


Figure 3. LoggerNet Software Interface

The best way to program and manage your CR1000 is to use Campbell Scientific's LoggerNet software. LoggerNet is a comprehensive data logger support software package that provides communications, data logger program editing, data processing and viewing. Please follow the steps below to read data from Paroscientific products.

1. Make sure that the Digiquartz® product is connected to the data logger and powered as described above. You can use a voltmeter to verify that there is power applied to any Digiquartz® product. (Measure the voltage between power and ground pins.
2. It is assumed that you have already configured CR1000. If you have not done so, consult the CR1000 user's manual to configure the data logger. The data from a Digiquartz® Intelligent product can be pressure, pressure period, temperature period, ambient temperature and humidity. A table of these parameters and corresponding commands is as in Table 2 below.

Parameter	Command	Typical response
Pressure	*0100P3<cr><lf>	*000114.57645<cr><lf>
Pressure Period	*0100P1<cr><lf>	*000128.12456<cr><lf>
Temperature	*0100Q3<cr><lf>	*000122.345<cr><lf>
Temperature Period	*0100Q1<cr><lf>	*00015.123456<cr><lf>
Ambient Temperature (MET Only)	*0100TT<cr><lf>	*000122.5<cr><lf>
Humidity (MET Only)	*0100RH<cr><lf>	*000175.9<cr><lf>

Table 2. Measurement Parameters & Commands from Digiquartz Intelligent Product

Note: <cr> represents carriage return, <lf> line feed.

3. Download the data generic logger program provided on our web site to integrate Digiquartz® Intelligent Product to the CR1000 data logger. In this program, disable the parameters you don't need. The source code for reading pressure only is available in Appendix-A. If you need to add more sensors such as wind speed and direction to the logger, incorporate the related code in the source code provided. Program to read other intelligent parameters from Digiquartz® Intelligent Sensors is in Appendix-B. If you modify the code, save the modified code on your computer and note the saved location.
4. Click on the "Connect" icon in LoggerNET to launch the program as illustrated in Figure 3.
5. Click on the "Connect" button on the program window to connect to the data logger.

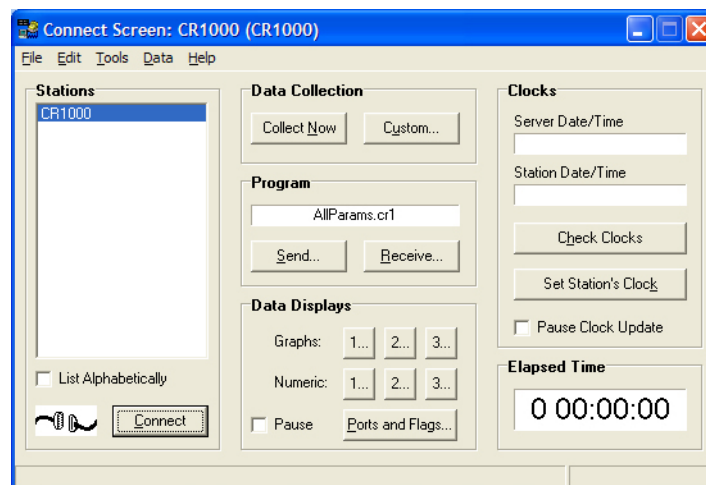


Figure 4. Connect program window

6. Communication with logger will be established and the following screen will appear (See Figure 5). Set date/time information in this window if needed.

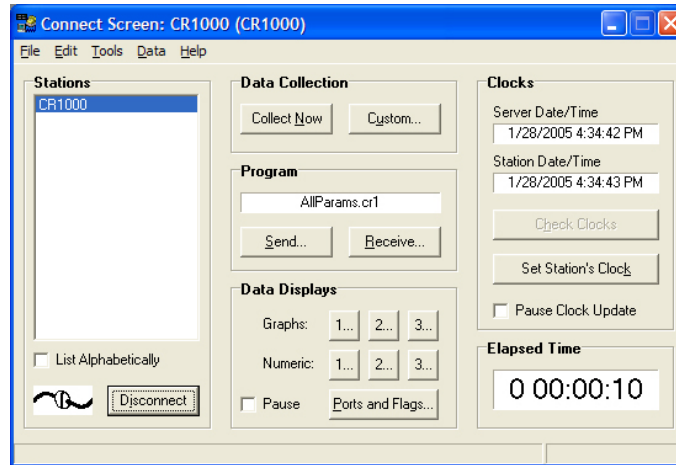


Figure 5. Connect program window displaying CR1000 Logger parameters

7. Click on the “Send” button in the “Program” group. This will bring up an “Open” dialog box. Select the program file to send to the CR1000 (“AllParams.cr1” in the example). When you click on the “OK” button the message box in Figure 6 will appear. Sending a program to the logger erases all the stored data.

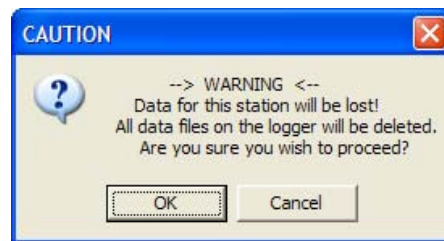


Figure 6. Program upload warning box

8. Click on “OK” to transfer and compile the program to the logger. If the program states that you successfully uploaded the program, then you are ready to collect data.
9. To verify that logger is reading data, click on the “Numeric 1...” button in the “Data Displays” group. This will open a window in Figure 7.

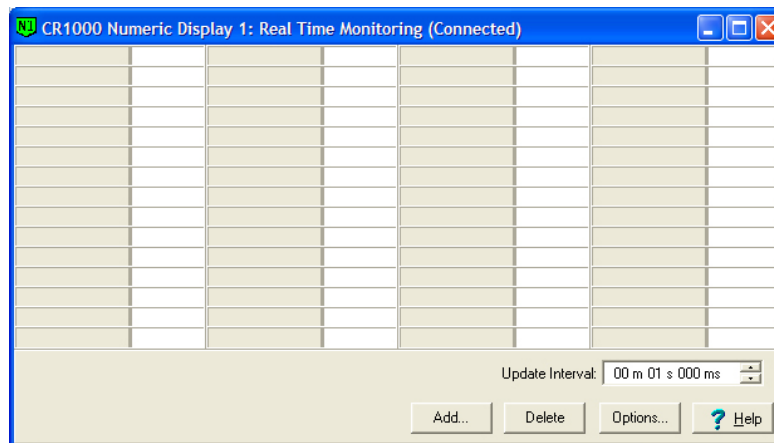


Figure 7. Program upload warning box

10. Add the parameters you would like to display in this window. Select a cell in the gray area in Figure 7. Click on the “Add” button. The window in Figure 8 will appear.

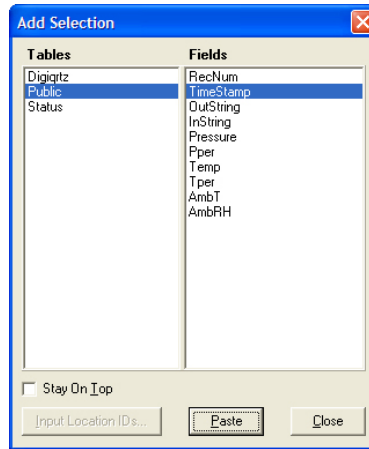


Figure 8. Tables and Parameters

11. Figure 8 shows the tables and fields in each table. The public table will show all the public parameters defined as “Public” in your program as in Appendix-A and B. The “Digiqrztz” table shows the values of parameters logged most recently. To see how the program works, add parameters from the “Public” table and the “Digiqrztz” tables as in Figure 9. In Figure 9, the parameters on the left are public variable; the ones on the right are parameters from the “Digiqrztz” table defined in the program.

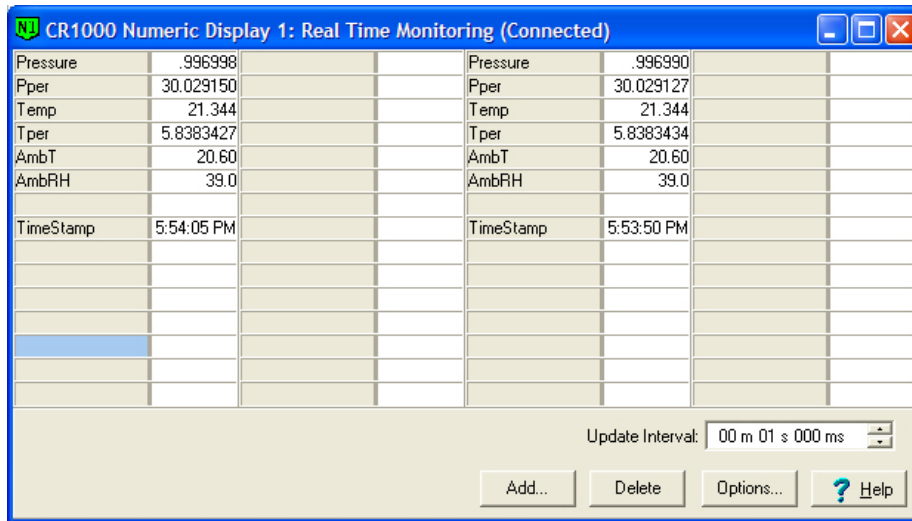


Figure 9. Verifying data logger operation real-time

12. If all of the public parameters (the group on the left above) are updating periodically, this means your program is communicating with your instrument and reading data. If the parameters from the “Digiqrztz” table are updating periodically, this means the program is also logging data.
13. At this point, you can exit from the program and let the CR1000 continue logging data. To set the logger into delayed logging mode, use the “EZSetup” software that comes with LoggerNet. See software user’s manual to explore the details of the EZSetup option.
14. To download logged data to your PC, again use the “Connect” program and click on the “Collect Now” button in the “Data Collection” group. This option allows you to download the data to a text file. This text

file can easily be imported into an MS-Excel spreadsheet. LoggerNet can also be programmed to download data periodically.

5. Other Accessories

CR1000 comes with on-site and off-site communication options. Optional communications for Ethernet, voice, satellite, and wireless are available from Campbell Scientific. Please contact Campbell Scientific at www.campbellsci.com.

Appendix –A

Source Code for CR1000 to Read Pressure from Digiquartz® Instruments

'CR1000 Series Data logger Program for Digiquartz Intelligent Sensors

'Program author: Paroscientific, Inc.

'Description: This code reads and stores pressure from Digiquartz Intelligent Products.

'The code has the P3 command to read pressure

Public OutString as string * 100

Public InString as string * 100

Public Pressure as string * 100

'Define storage area in memory. Table name is Digiqrtz.

DataTable (Digiqrtz,1,-1)

 DataInterval (0,10,Sec,10)

 Sample (1,Pressure,String) 'Store Pressure

EndTable

'Main Program

BeginProg

 'Set up communication port to send and receive data

 SerialOpen (Com1,9600,0,0,10000) 'Uses COM1. Default 9600 baud rate. Change if instrument has different baud rate.

 Scan (10,Sec,0,0) 'Read data every 10 seconds. Change if needed.

 OutString = "*0100P3"+CHR(13)+CHR(10)

 'Send Pressure string over one communication port C1 (COM1 TX).

 SerialOut (Com1,OutString,"",0,500)

 'Receive String on port C2 (COM1 RX).

 SerialIn (Instring, Com1,500,10,100) 'Receive timeout is 5 secs. Change if needed.

 SplitStr (Pressure,InString,"*0001",1,4)

 'Call Output Tables to store values

 CallTable Digiqrtz

 NextScan

EndProg

Note: This source code is available at Paroscientific web site. (File Name: DQReadPressure.cr1)

Appendix –B

Source Code for CR1000 to Read Data from Digiquartz® Instruments

'CR1000 Series Datalogger Program for Digiquartz Intelligent Sensors

'Program author: Paroscientific, Inc.

'Description: This code reads and stores data from Digiquartz Intelligent Products.

'The code has the necessary commands to read pressure (P3), pressure period (P1), sensor

'temperature (Q3), sensor temperature period (Q1), humidity (RH) and ambient temperature (TT).

'Enable or disable related code lines to read different parameters.

Public OutString as string * 100

Public InString as string * 100

Public Pressure as string * 100

Public Pper as string * 100

Public Temp as string * 100

Public Tper as string * 100

Public AmbT as string * 100

Public AmbRH as string * 100

'Define storage area in memory. Table name is Digiqrtz.

'Enable/disable the parameters below to include/exclude different parameters. Default reading is pressure

DataTable (Digiqrtz,1,-1)

 DataInterval (0,10,Sec,10)

 Sample (1,Pressure,String) 'Store Pressure

 Sample (1,Pper,String) 'Store Pressure Period

 Sample (1,Temp,String) 'Store Temperature

 Sample (1,Tper,String) 'Store Temp. Period

 Sample (1,AmbT,String) 'Store Ambient Temperature

 Sample (1,AmbRH,String) 'Store Humidity

EndTable

'Main Program

BeginProg

'Set up communication port to send and receive data

SerialOpen (Com1,9600,0,0,10000) 'Uses COM1. Default 9600 baud rate. Change if instrument has different baud rate.

Scan (10,Sec,0,0) 'Read data every 10 seconds. Change if needed.

 OutString = "*0100P3"+CHR(13)+CHR(10)

'Send Pressure string over one communication port C1 (COM1 TX).

 SerialOut (Com1,OutString,"",0,500)

'Receive String on port C2 (COM1 RX).

 SerialIn (Instring, Com1,500,10,100) 'Receive timeout is 5 secs. Change if needed.

 SplitStr (Pressure,InString,"*0001",1,4)

'Send Pper string over one communication port C1 (COM1 TX).

 OutString = "*0100P1"+CHR(13)+CHR(10)

 SerialOut (Com1,OutString,"",0,500)

'Receive String on port C2 (COM1 RX).

 SerialIn (Instring, Com1,500,10,100) 'Receive timeout is 5 secs. Change if needed.

 SplitStr (Pper,InString,"*0001",1,4)

```
'Send Temperature string over one communication port C1 (COM1 TX).  
OutString = "*0100Q3"+CHR(13)+CHR(10)  
SerialOut (Com1,OutString,"",0,500)
```

```
'Receive String on port C2 (COM1 RX).  
SerialIn (Instring, Com1,500,10,100) 'Receive timeout is 5 secs. Change if needed.
```

```
SplitStr (Temp,InString,"*0001",1,4)
```

```
'Send Tper string over one communication port C1 (COM1 TX).  
OutString = "*0100Q1"+CHR(13)+CHR(10)  
SerialOut (Com1,OutString,"",0,500)
```

```
'Receive String on port C2 (COM1 RX).  
SerialIn (Instring, Com1,500,10,100) 'Receive timeout is 5 secs. Change if needed.
```

```
SplitStr (Tper,InString,"*0001",1,4)
```

```
'Send ambient temperature string over one communication port C1 (COM1 TX).  
OutString = "*0100TT"+CHR(13)+CHR(10)  
SerialOut (Com1,OutString,"",0,500)
```

```
'Receive String on port C2 (COM1 RX).  
SerialIn (Instring, Com1,500,10,100) 'Receive timeout is 5 secs. Change if needed.
```

```
SplitStr (AmbT,InString,"*0001",1,4)
```

```
'Send humidity string over one communication port C1 (COM1 TX).  
OutString = "*0100RH"+CHR(13)+CHR(10)  
SerialOut (Com1,OutString,"",0,500)
```

```
'Receive String on port C2 (COM1 RX).  
SerialIn (Instring, Com1,500,10,100) 'Receive timeout is 5 secs. Change if needed.
```

```
SplitStr (AmbRH,InString,"*0001",1,4)
```

```
'Call Output Tables to store values  
CallTable Digiqrtz
```

```
NextScan  
EndProg
```

Note: This source code is available at Paroscientific web site. (File Name:AllParams.cr1)