



**M Y N A H<sup>SM</sup>**

**Inverted Floating Point MODBUS  
Programmable Serial Interface Card**

**User Manual**

**Rev. P1.16**

**July 2008**

DeltaV is a trademark of Emerson Process Management, Inc © Emerson Process Management, Inc. 1998, 1999. All rights reserved.

Printed in the U.S.A.

While this information is presented in good faith and believed to be accurate, MYNAH Technologies does not guarantee satisfactory results from reliance upon such information. *Nothing contained herein is to be construed as a warranty or guarantee, express or implied, regarding the performance, merchantability, fitness or any other matter with respect to the products*, nor as a recommendation to use any product or process in conflict with any patent. MYNAH Technologies reserves the right, without notice, to alter or improve the designs or specifications of the products described herein.

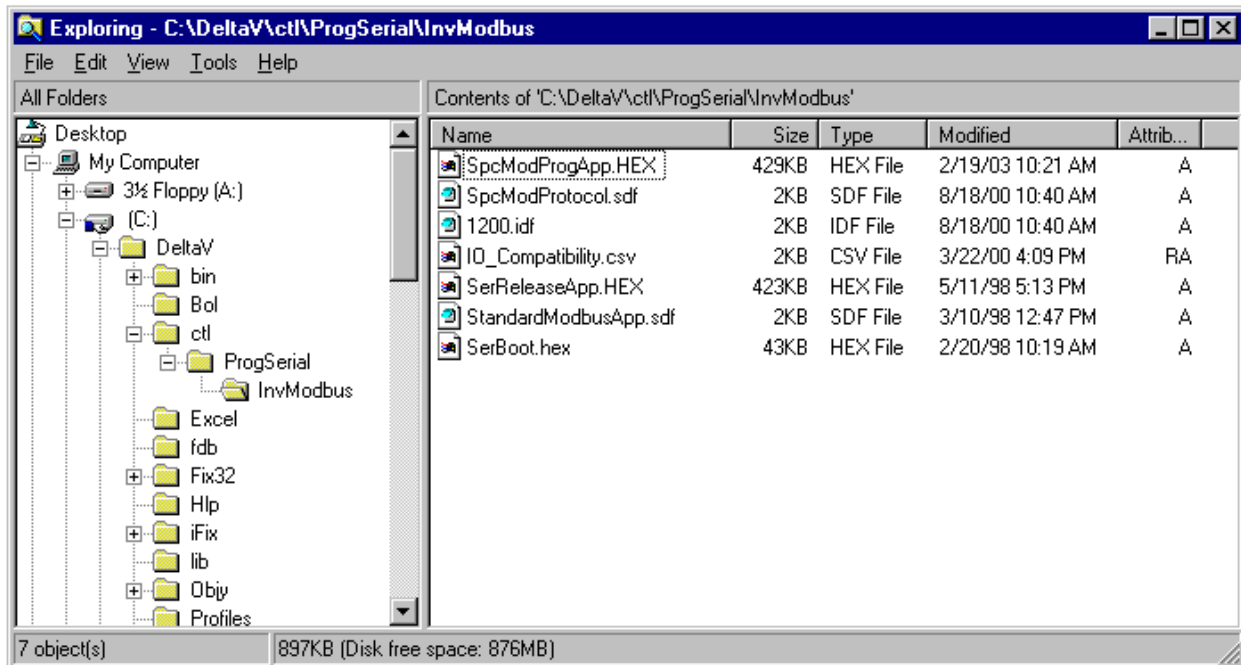


### Driver Installation

The Series 1 driver software (10 files) and the Series 2 driver software (15 files) are distributed on a CD. These required files must be copied to the DeltaV directory on your ProPlus Workstation. The path is:

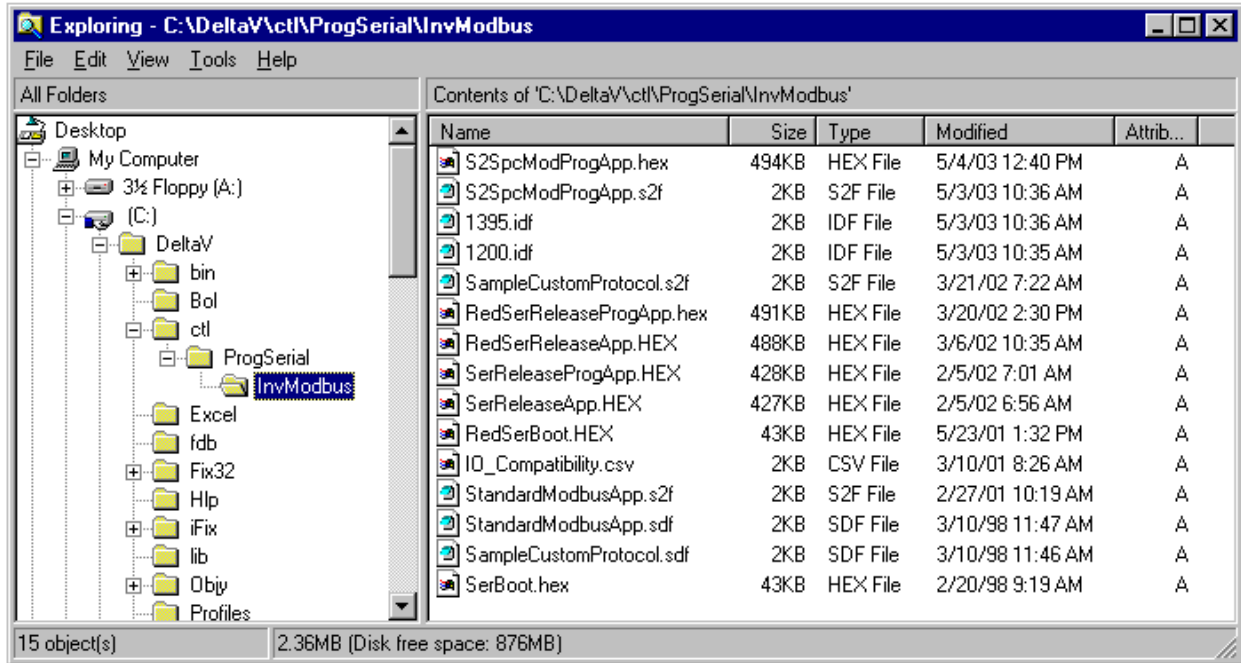
**\\DeltaV\ctl\ProgSerial\InvModbus**

The following shows a completed copy operation for Series 1:





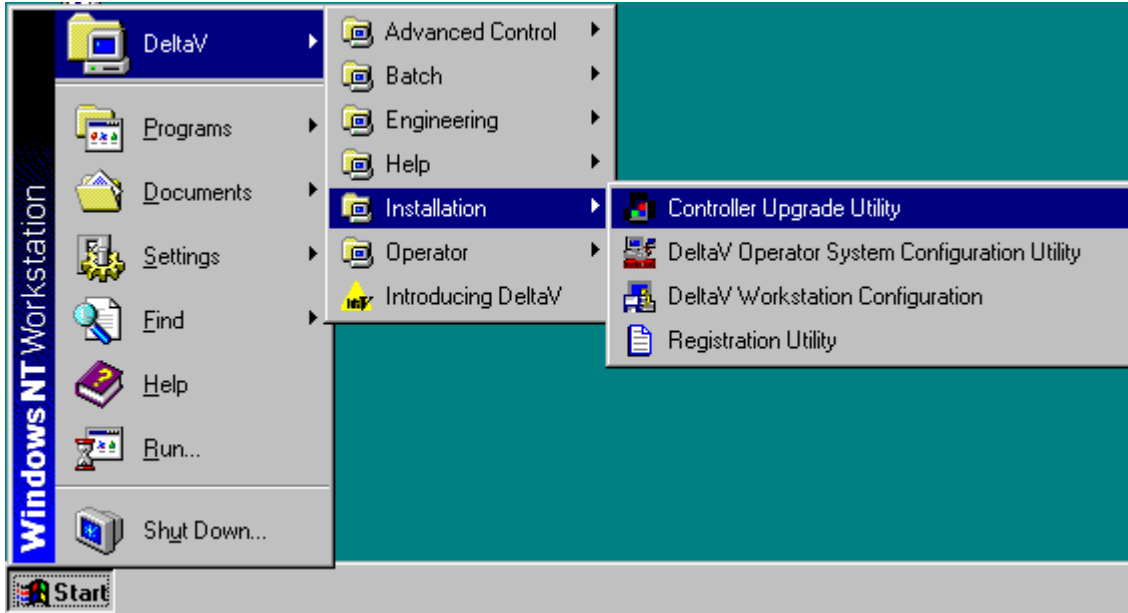
The following shows a completed copy operation for Series 2:



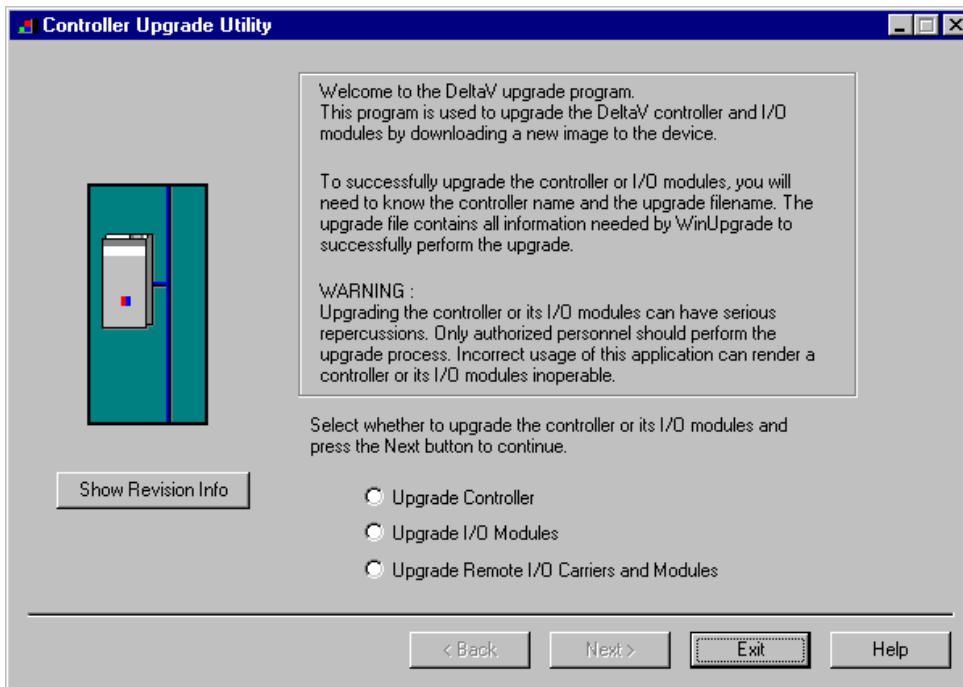
After copy completion, you are ready to program (or upgrade) the Programmable Serial Card with the supplied custom driver software. The steps are as follows:



1. Click on the Start button and select DeltaV, Installation, Controller Upgrade Utility as shown below.

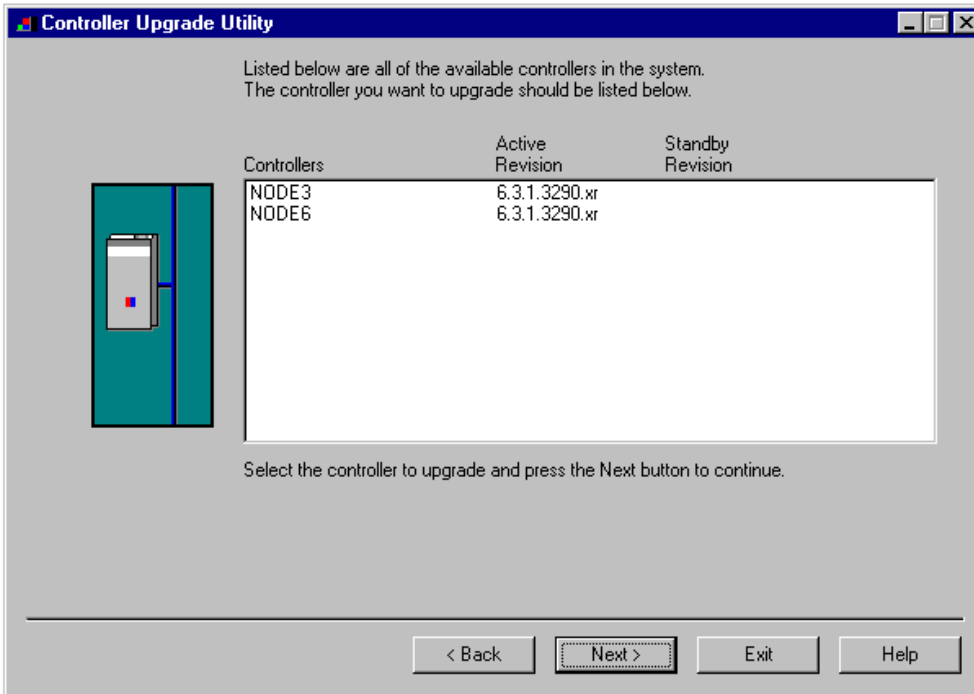
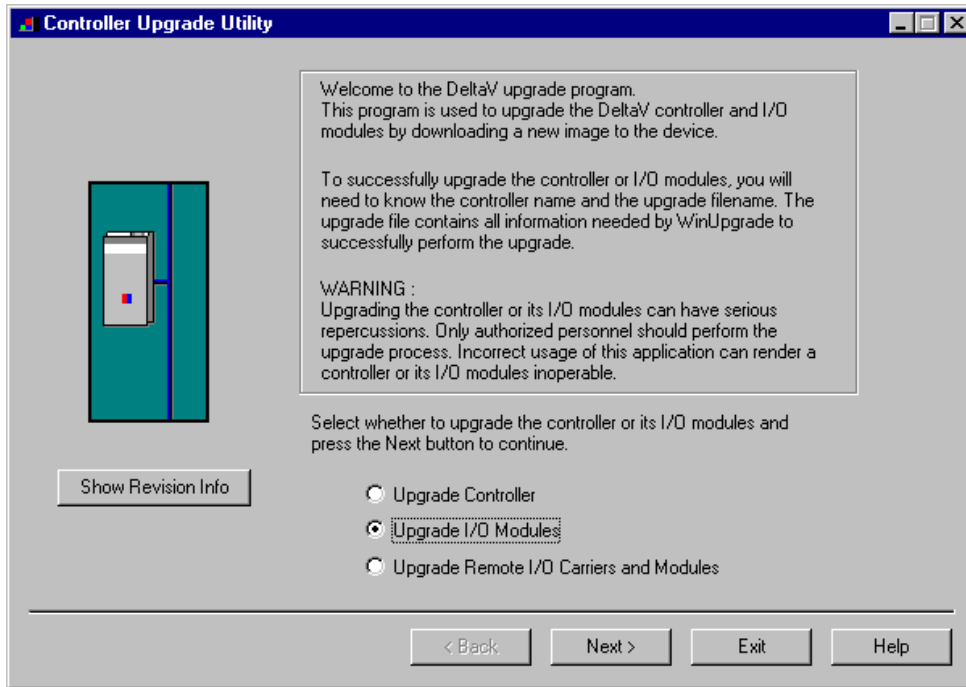


The following dialog will appear:



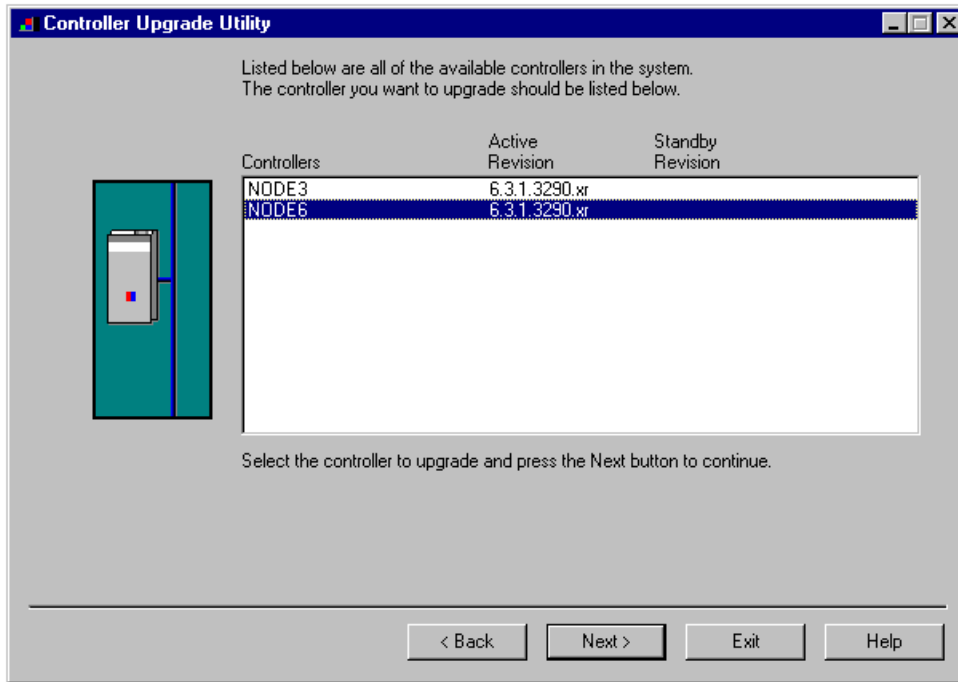


Click on the Upgrade I/O Modules radio button, and then click Next.

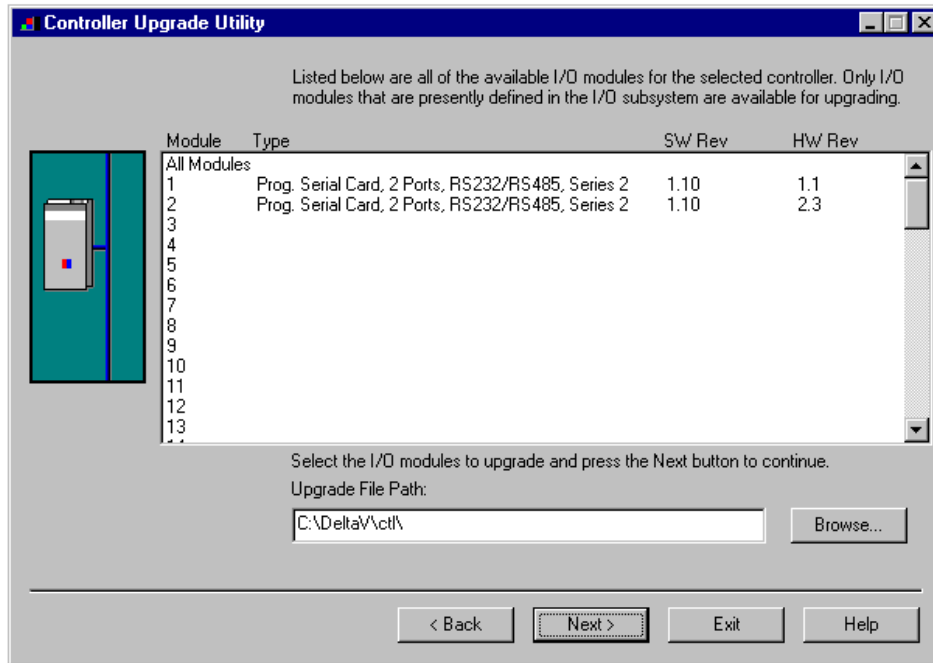




The above dialog will appear, listing all the available Controllers in your network. From this dialog, select the appropriate Controller and then Click Next as shown below.



After you Click Next, the following dialog will appear, listing all the I/O modules in your selected Controller. The shown list of I/O modules is an example only. Your list will be different.



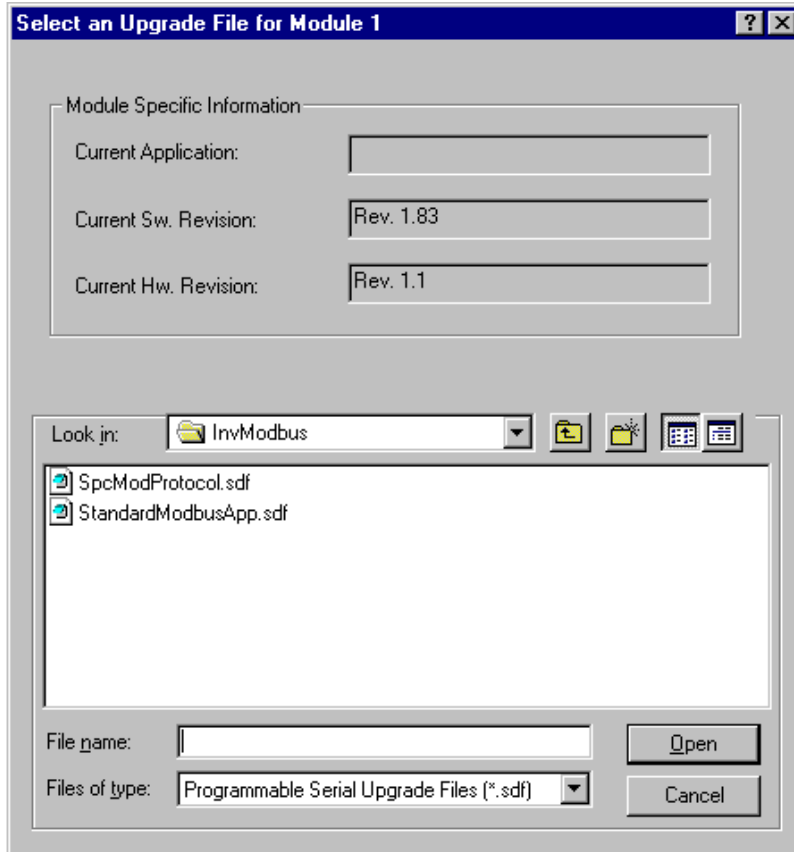
From this dialog, select the Programmable Serial Card I/O Module in the list. For example, we will select I/O Module 2. This will give you the following dialog, from which you will select the file path to where the driver software is located. This will be:

**\\Delta\ctl\ProgSerial\InvModbus**



Once you are in the specified directory, you will need to select the following file for Series 1 serial cards:

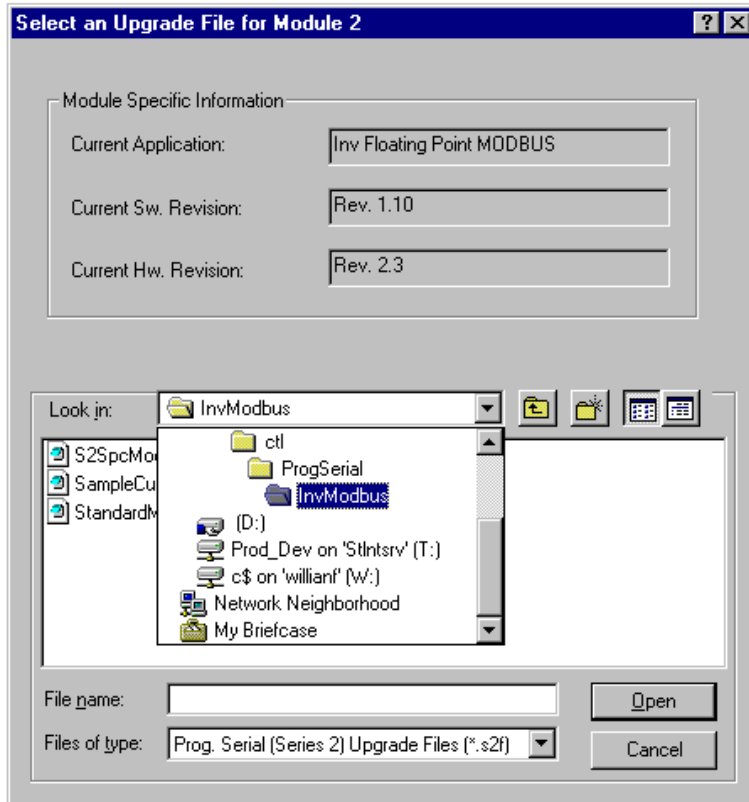
**SpcModProtocol.SDF**



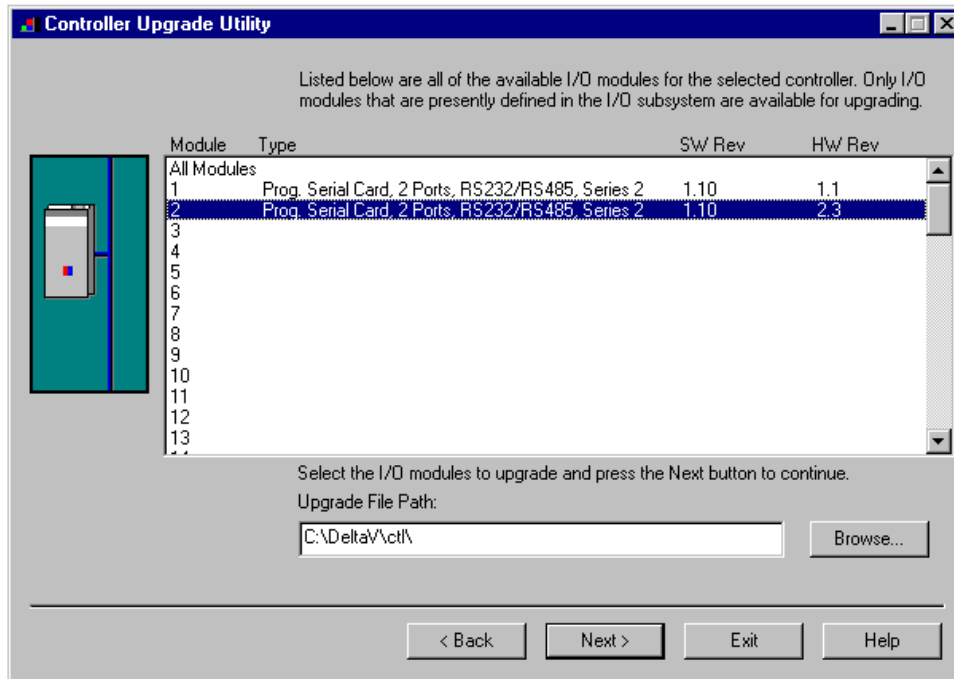


For Series 2 serial cards, select the following:

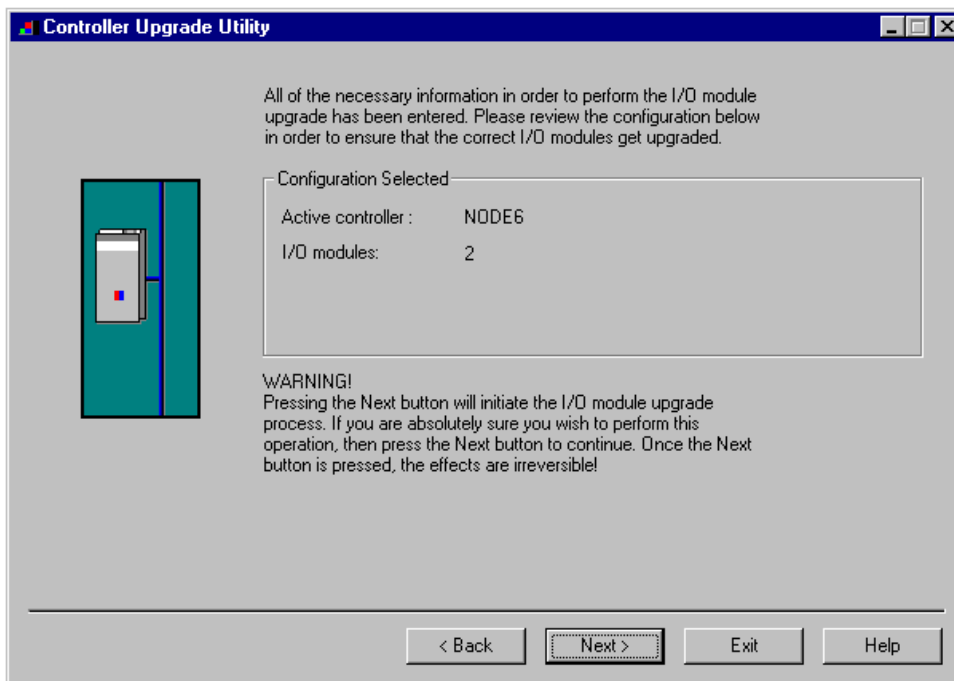
**S2SpcModProtocol.S2F**



After selecting the .SDF file (series 1), or the .S2F file (series 2), click on Open. This dialog will close and you will be back to the following:

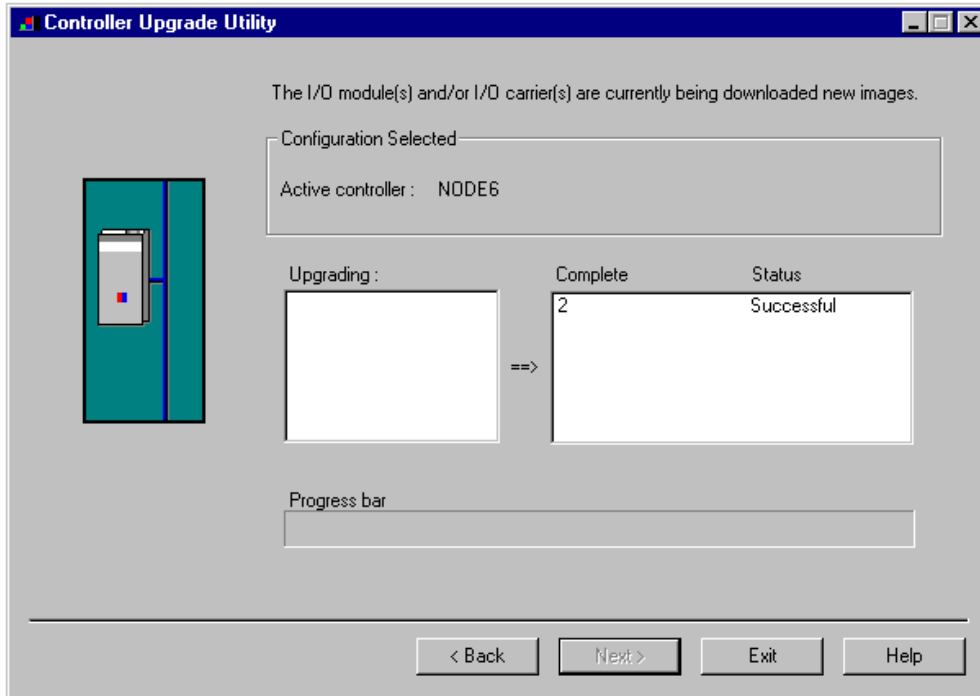


In this dialog, Click Next again. You will get the following dialog, confirming the Controller and I/O Module to program.





Click Next and the I/O Module upgrade process will begin. After completion, you will receive the following dialog, indicating success.

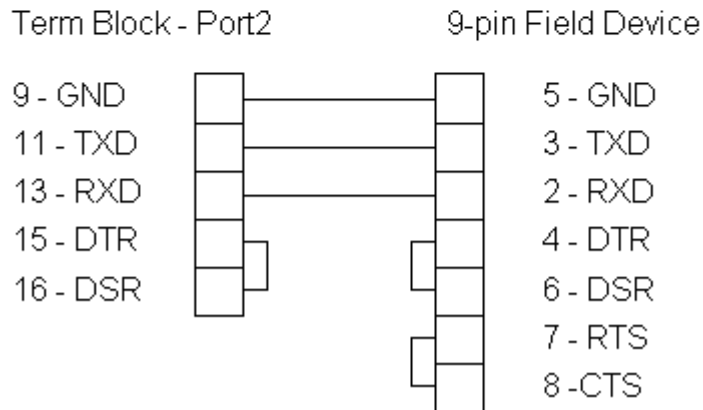
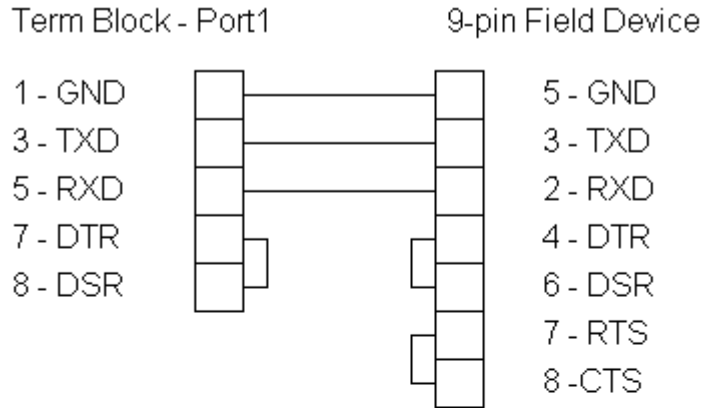


This completes the I/O Module upgrade process.



### Serial Card Hookup

The Programmable Serial Card supports both RS-232 and RS-485 communications with field devices. The following is the cable pinout to use. Note that if a NULL cable is required, swap pins 3 & 5 and 11 & 13 on the term block.





## Using the Driver

### Customization

This driver allows you to customize MODBUS communications and representation of data. Modification of data representation is typically only required when reading/writing Floating Point, or Signed/Unsigned 32-bit Integer registers. However, in some cases, you may need to use Special Data 1 for 16-bit byte swapping as well. Furthermore, this driver allows register addresses to be user defined, i.e., the user is not constrained to use 0X, 1X, 3X or 4X MODBUS addresses.

To customize data representation in a Dataset the Special data 1 and 2 registers are used as flags. This is described below.

Assume a Floating-point number 123.45, and its representation in IEEE 754 format as follows:

<b>Floating Point Number</b>	<b>Representation as 2 16-bit words</b>	<b>Representation as 4 bytes</b>
123.45	58982, 17142	230, 102, 66, 246

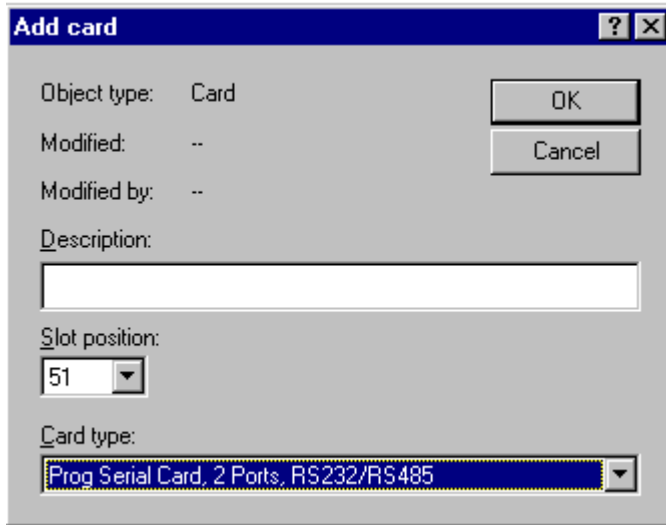
<b>Special Data</b>	<b>Value and Description</b>
1	<p>0 – Default is no customization – Floating Point and 32-bit data is represented as received. The transmitted byte order is 230, 102, 66, 246</p> <p>1 – The transmitted byte order is 66, 246, 230, 102</p> <p>2 – The transmitted byte order is 246, 66, 102, 230</p> <p>3 – The transmitted byte order is 102, 230, 246, 66</p>
2	<p>0 – Default setting where 2 MODBUS 16-bit registers are equivalent to 1 Floating Point or 32-bit value in DeltaV. There is a 1-2 correspondence between DeltaV value and read values.</p> <p>1 – Data is not read as 2 MODBUS 16-bit registers but as individual Floating Point or 32-bit values. There is a 1-1 correspondence between DeltaV value and read value.</p>
3	Not Used
4	Not Used
5	Custom DTR on/off delay used only for UDS Radio Modems. Specify in first configured dataset only.



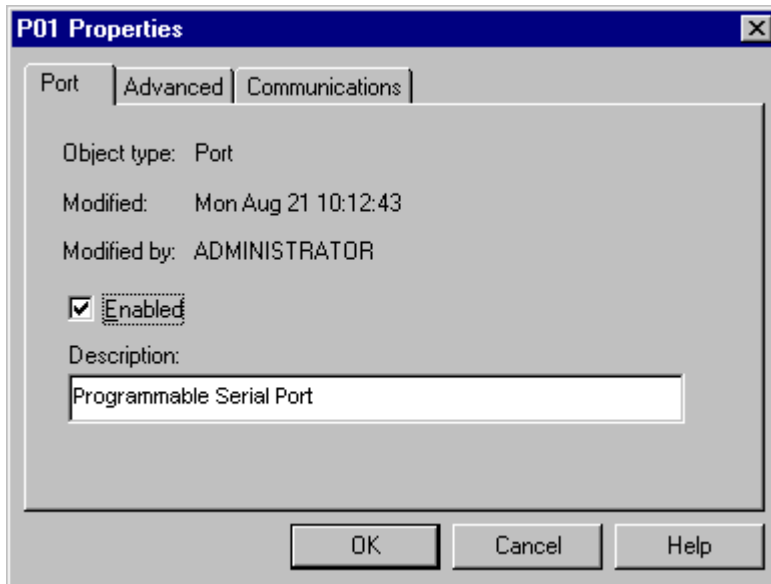
### Configuration

To have the Programmable Serial Card communicate with the MODBUS device, follow these steps:

1. In DeltaV, configure the serial card. This will create a Programmable Serial Card and define 2 ports under it, P01 and P02.

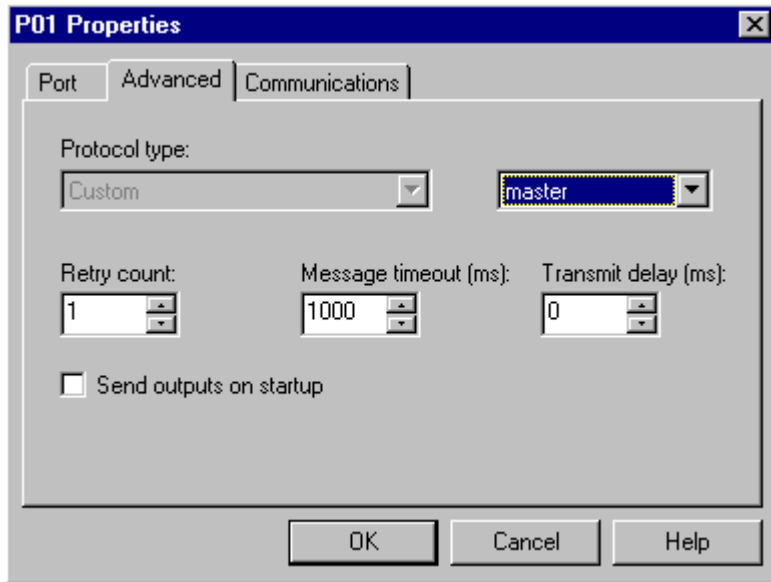


2. Right mouse click on Port 1. The following dialog will appear.

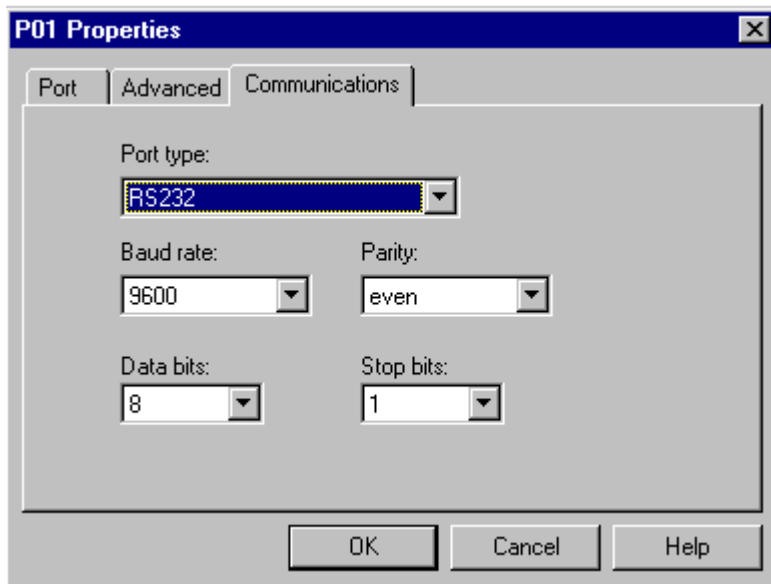




Click of the Enabled checkbox to enable the Port. Next select the Advanced tab. The following dialog will appear.



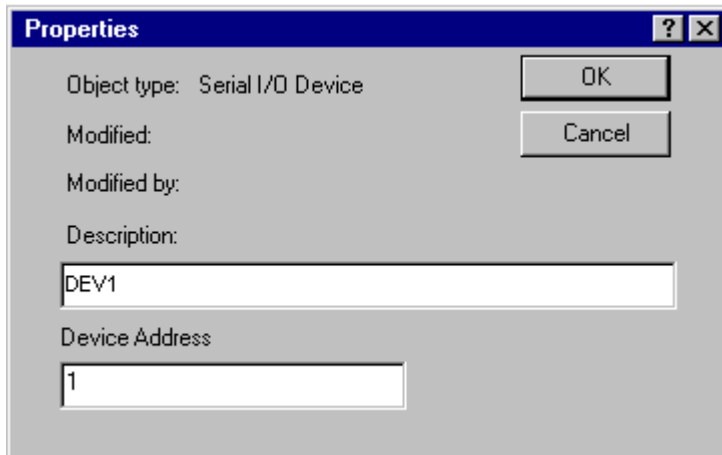
In this dialog, select whether the Serial card will behave as a Master or Slave. Also select the message time parameters. Next click the Communications tab. The following dialog will appear.



Specify the required baud rate parameters and click OK.

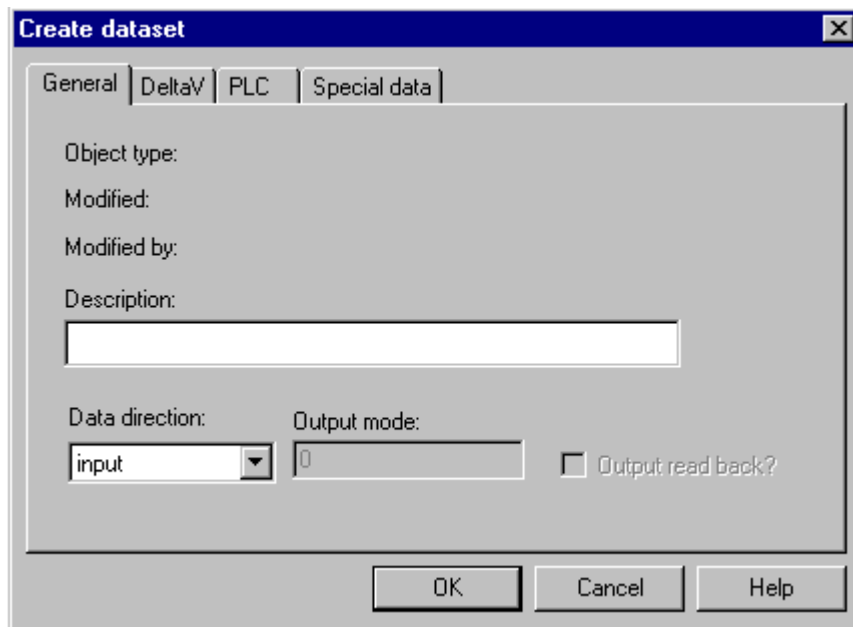


3. Configure a Serial Device under the Port by doing a Right Mouse click and selecting New Serial Device. The following dialog will appear:



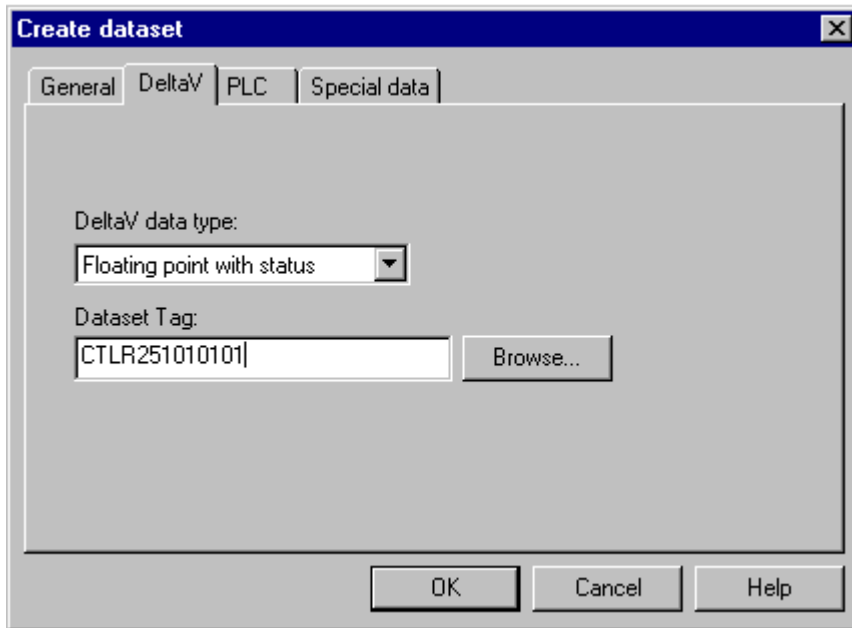
Specify the device address and description. Then click OK. This will add the serial device.

4. Next, configure datasets in the Serial Device. Each Serial Device can have 16 datasets under it. Or you can have 16 devices with 1 dataset each. A dataset can be input or output. To add a new dataset, right mouse click on the Serial Device and select New Dataset. The following dialog will appear.



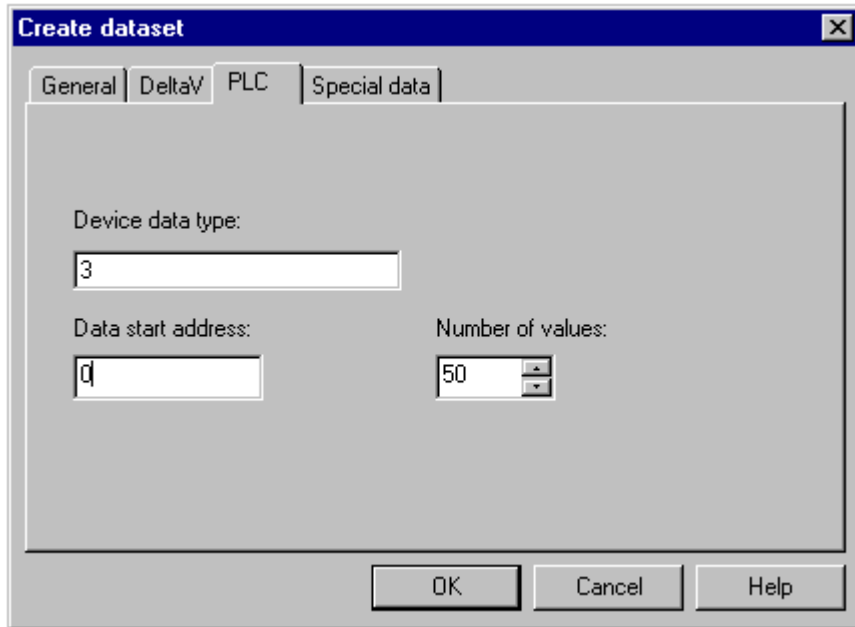


Configure the data direction to be input or output. In the above example, we are configuring an input dataset. Next click on the DeltaV tab. The following dialog will appear.



In this dialog, configure the data type needed for DeltaV. You can see the available types by clicking on the drop down list. In the above example, we are configuring the input data type to be floating point.

Next click the PLC tab. The following dialog will appear.



In this dialog, we will map DeltaV data types to PLC (or external Device) data types. PLC data type values and corresponding PLC registers are:

Device Data Type	Device Register
0	COILS
1	INPUT STATUS
2	INPUT REGISTERS
3	HOLDING REGISTERS
4	Reserved
5	Reserved
6	8 Byte ASCII read (OMNI flow computer specific)

The Data start address specifies where in the PLC we will read the data. In this example, the starting address is 0. This can be any PLC specific address.

**Note:** Device Data Type 6 is specifically used for OMNI flow computers to read or write the 8-byte ASCII data elements. This device data type is not applicable to other, non OMNI, devices.



Each dataset has a maximum of 100 values of DeltaV data type (as configured in the previous dialog). Each DeltaV value is mapped to one or more PLC registers. If two registers are required, they must be consecutive. The following table describes the mapping:

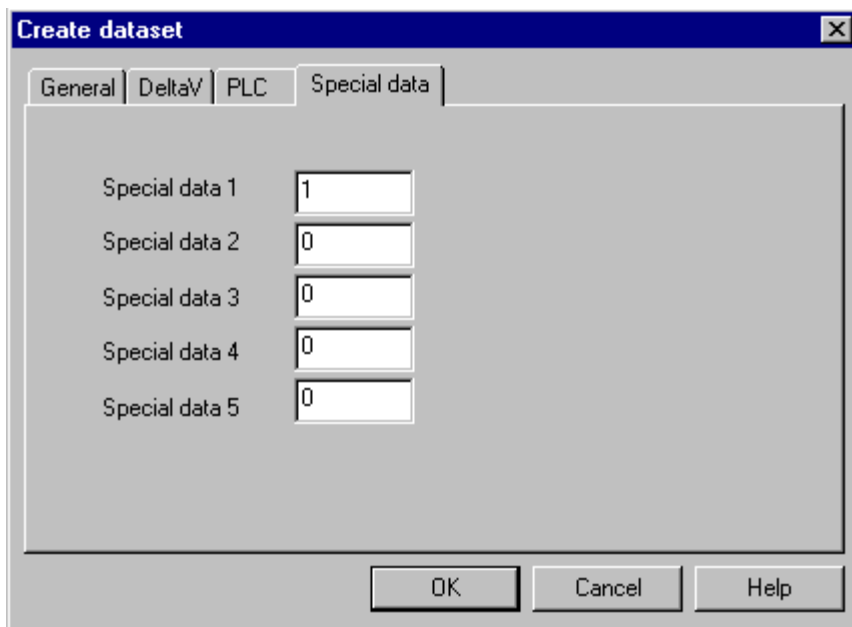
<b>DeltaV Data Type</b>	<b>PLC Register Type</b>	<b>PLC Registers Required</b>	<b>Max Number of Values</b>
Boolean	<ul style="list-style-type: none"> <li>• Coils</li> <li>• Input Status</li> </ul>	1 - 16 bit word	100
Discrete	<ul style="list-style-type: none"> <li>• Coils</li> <li>• Input Status</li> </ul>	1 - 16 bit word	100
Signed 8 bit Integer	<ul style="list-style-type: none"> <li>• Coils</li> <li>• Input Status</li> </ul>	1 - 16 bit word	100
	<ul style="list-style-type: none"> <li>• 8-Byte ASCII</li> </ul>	4 – 16 bit words	96
Signed 16 bit Integer	<ul style="list-style-type: none"> <li>• Coils</li> <li>• Input Status</li> <li>• Input Registers</li> <li>• Holding Registers</li> </ul>	1 - 16 bit word	100
Signed 32 bit Integer	<ul style="list-style-type: none"> <li>• Holding Registers</li> </ul>	2 - 16 bit words	50
Unsigned 8 bit Integer	<ul style="list-style-type: none"> <li>• Coils</li> <li>• Input Status</li> </ul>	1 - 16 bit word	100
	<ul style="list-style-type: none"> <li>• 8-Byte ASCII</li> </ul>	4 - 16-bit words	96
Unsigned 16 bit Integer	<ul style="list-style-type: none"> <li>• Coils</li> <li>• Input Status</li> <li>• Input Registers</li> <li>• Holding Registers</li> </ul>	1 - 16 bit word	100
Unsigned 32 bit Integer	<ul style="list-style-type: none"> <li>• Holding Registers</li> </ul>	2 - 16 bit word	50
Floating Point	<ul style="list-style-type: none"> <li>• Holding Registers</li> </ul>	2 - 16 bit word	50
String	<ul style="list-style-type: none"> <li>• Holding Registers</li> </ul>	1 byte	100
	<ul style="list-style-type: none"> <li>• 8-Byte ASCII</li> </ul>	4 - 16-bit words	96



In this example, the starting address is 0, the PLC data type is 3 and the maximum number of values is 50. This would result in the following DeltaV registers:

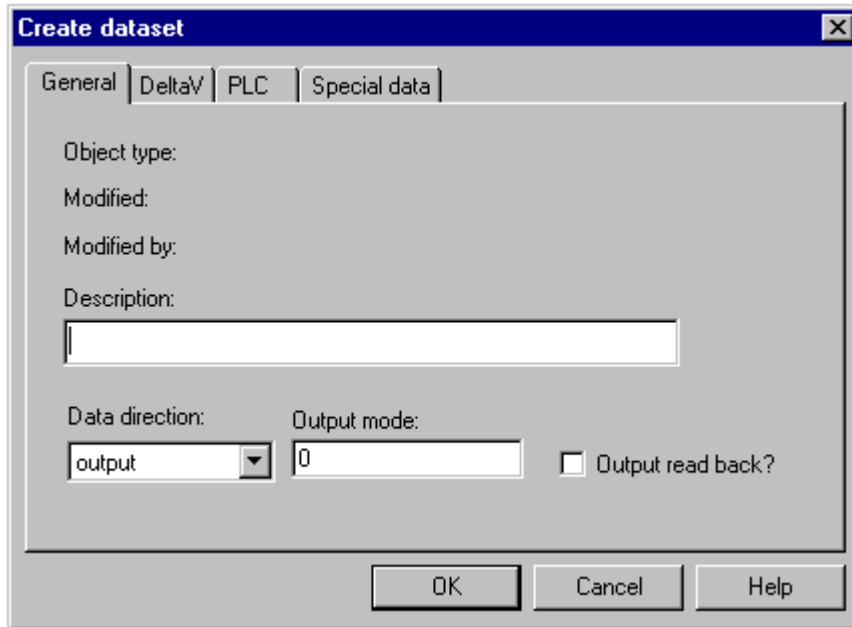
<b>DeltaV Register</b>	<b>PLC or External Device Registers</b>
R1	0 1
R2	2 3
R3	4 5
R50	99 100

5. Lastly for each dataset, click on the Special data tab. The following dialog will appear:

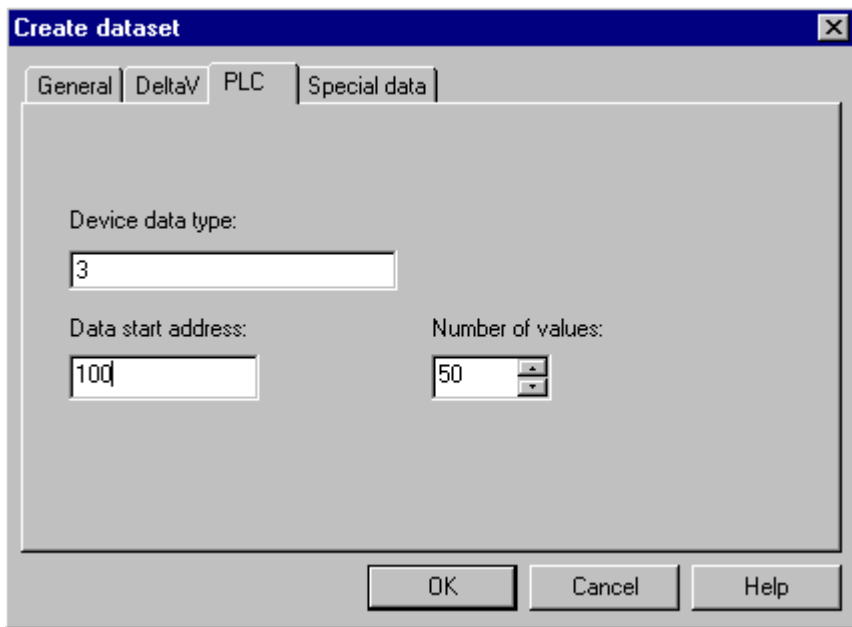


Special data 1 value is used when transferring data for Floating point, signed 32 bit integer and unsigned 32 bit integer registers. If the default value of 0 is used, the data is transferred in the received order. Please refer to Customization section above.

6. Next we configure an output dataset in the Serial Device. Repeat Step 4 above. The following dialog will appear.



Configure the data direction to be output. Next click on the DeltaV tab and select DeltaV data type as floating point. Next click the PLC tab. The following dialog will appear.



Note that the Data start address is now 100. This is because in our example, the first dataset has 50 floating point values. Consequently, this dataset starts at R101.



**M Y N A H™**

Powerful Solutions for Digital Plants

## **Technical Support**

For technical support or to report a defect, please give MYNAH Technologies a call at (636) 681-1555. If a defect is discovered, please document it in as much detail as possible and then fax your report to us at (636) 681-1660.

You can also send us your questions via e-mail. Our addresses are:  
support@mynah.com

Thank you for using DeltaV.