

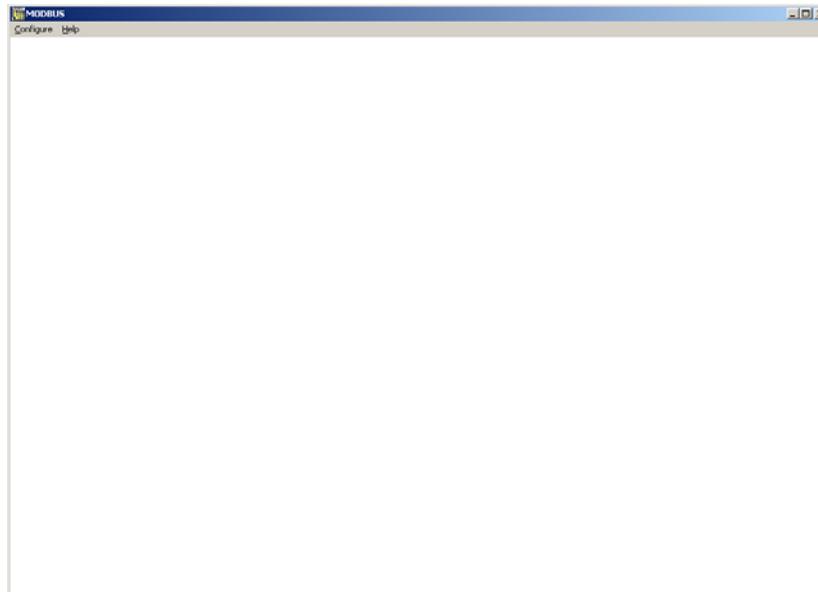
WonderWare ModBus Communication

The purpose of this White Paper is to document how to configure WonderWare for ModBus communications using a Sealevel Systems asynchronous communications card.

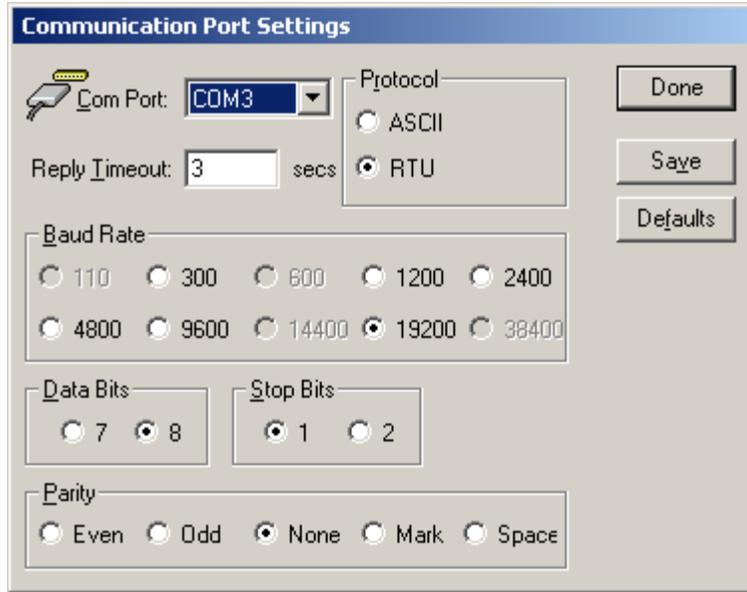
ModBus is nothing more than a messaging structure and therefore is independent of the underlying physical layer. In layman's terms, this means it can be implemented using RS-232, RS-422, or RS-485 over a variety of media. There are 2 implementations of ModBus, ModBus ASCII and ModBus RTU. ModBus ASCII transmits each 8-bit byte as 2 ASCII characters. ASCII mode allows time intervals of up to 1 second to occur between characters without causing an error. ModBus RTU transmits each 8-bit byte as 2 4-bit hexadecimal characters. RTU mode offers higher throughput and if the option exist, is the preferable choice. For a complete detailed description of the protocol as well as other pertinent information, visit www.modbus.org.

For the purpose of this paper, we will use a Sealevel Systems Ultra-COMM+2.PCI (Model 7201) asynchronous communication card. This particular Sealevel card can be field configured for RS-232, RS-422, and RS-485. We are using Port 1 configured in half-duplex RS-485 mode. It is assumed that your Sealevel communication card is already installed and its' proper operation has been verified using the supplied Sealevel utilities. It is also assumed that WonderWare is already installed and functioning on your system and that you installed the included WonderWare Modicon ModBus I/O Server.

The first step in the process is to configure the WonderWare Modicon ModBus I/O Server. Start the I/O server and the following screen should appear:

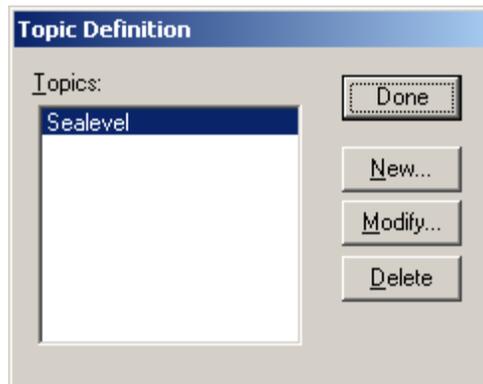


Select 'Configure' to open the configuration menu and choose 'Com Port Settings...' to open the following dialog:

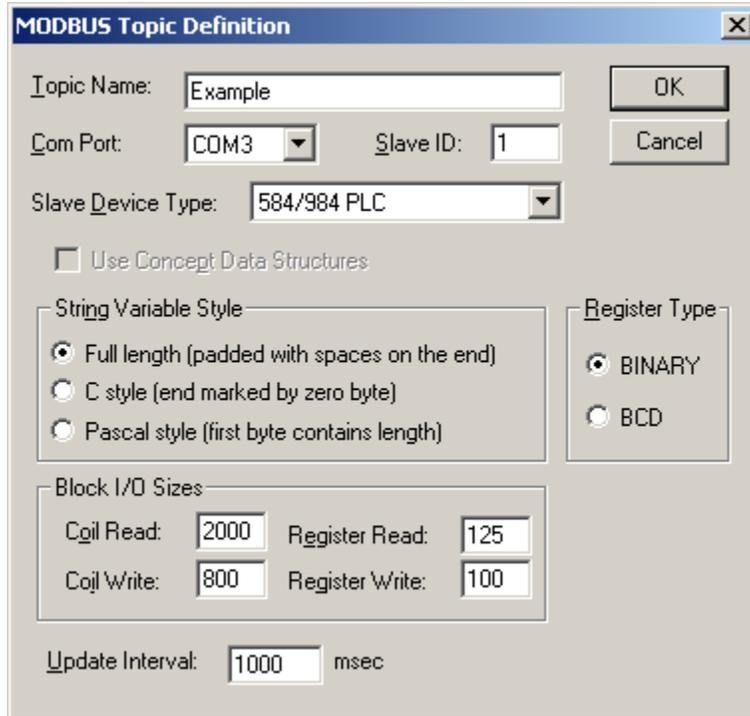


This dialog allows you to connect to and configure the settings of the COM: port the ModBus I/O server will use. The COM: identifier for Port 1 on our 7201 card is COM3 and thus we have used the drop down box to select COM3. We have selected the RTU protocol and communication parameters of 19200 baud, 8 data bits, 1 stop bit, and no parity. The ModBus protocol and port communication parameters simply need to match those of your external I/O devices. Configure your system as required and then click 'Save' and 'Done' to complete configuration.

WonderWare will exchange information with the I/O server using Dynamic Data Exchange (DDE). In order to make this work, a "Topic" must be created that can be configured and used in WonderWare. Select 'Configure' to open the configuration menu and choose 'Topic Definition...' to open the following dialog:



This dialog can be used to create new "Topics" as well as to modify or delete existing ones. To begin creating a new "Topic" click 'New...' and the following dialog will appear:

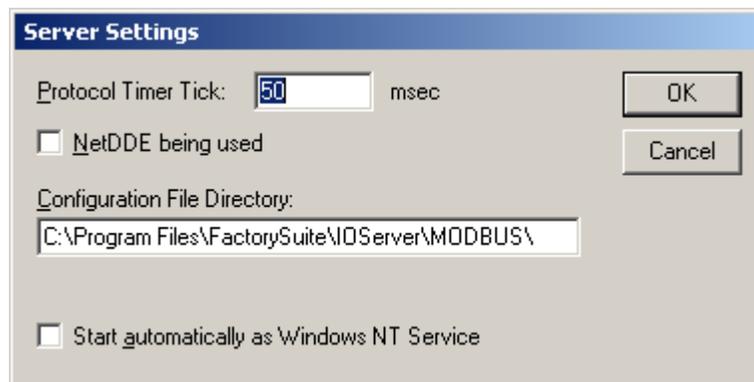


The dialog box is titled "MODBUS Topic Definition". It contains the following fields and options:

- Topic Name:** A text input field containing "Example".
- Com Port:** A dropdown menu set to "COM3".
- Slave ID:** A text input field containing "1".
- Slave Device Type:** A dropdown menu set to "584/984 PLC".
- Use Concept Data Structures**
- String Variable Style:** A group box containing three radio buttons:
 - Full length (padded with spaces on the end)
 - C style (end marked by zero byte)
 - Pascal style (first byte contains length)
- Register Type:** A group box containing two radio buttons:
 - BINARY
 - BCD
- Block I/O Sizes:** A group box containing four text input fields:
 - Coil Read:** 2000
 - Register Read:** 125
 - Coil Write:** 800
 - Register Write:** 100
- Update Interval:** A text input field containing "1000" followed by "msec".

Enter a "Topic Name" into this field. This name is user selectable and a good practice is to use something descriptive that will help to identify the type of device or devices you are communicating with. This will also be one of the required pieces of information you will need to configure WonderWare. Another point of interest is the COM number. Be sure to set this to the COM: port you are using and that it matches what was configured in the previous section. The remaining settings would need to be configured for your particular installation and type of I/O device. Click 'OK' when finished. This will return you to the "Topic Definition" dialog where you may now click 'Done' to complete configuration.

Select 'Configure' to open the configuration menu and choose 'Server Settings...' to open the following dialog:

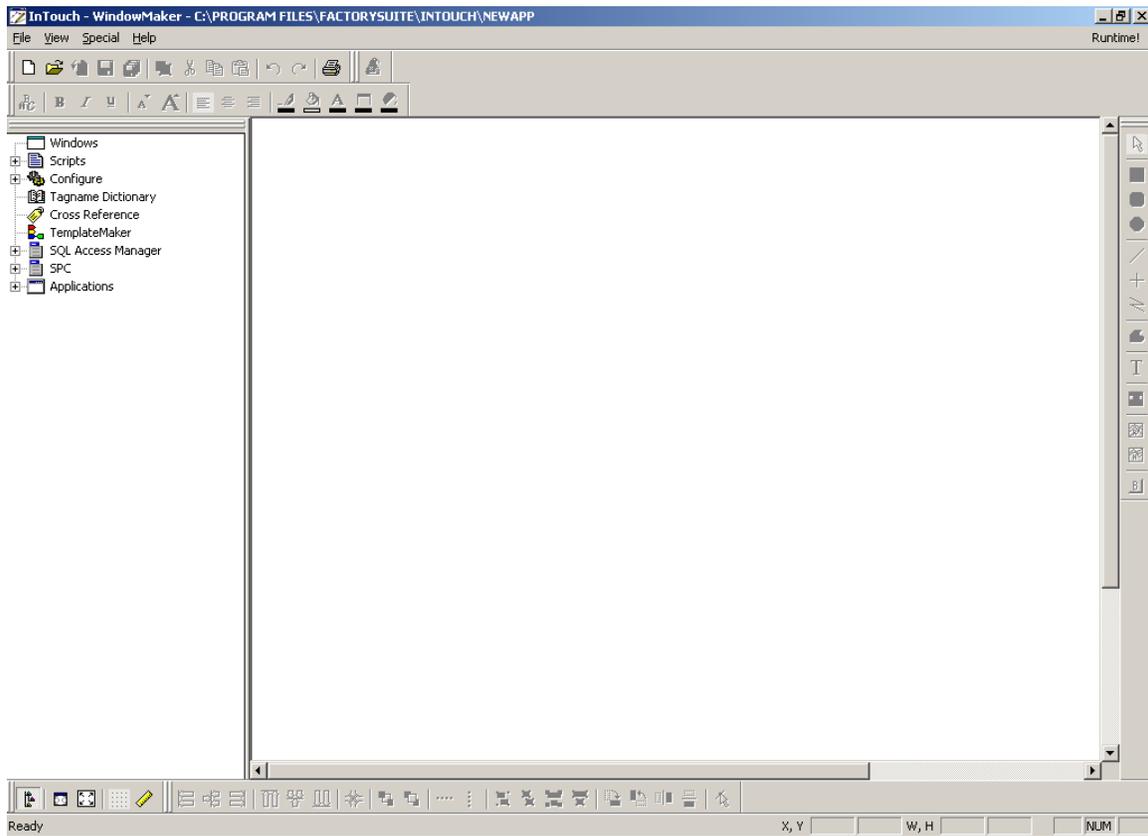


The dialog box is titled "Server Settings". It contains the following fields and options:

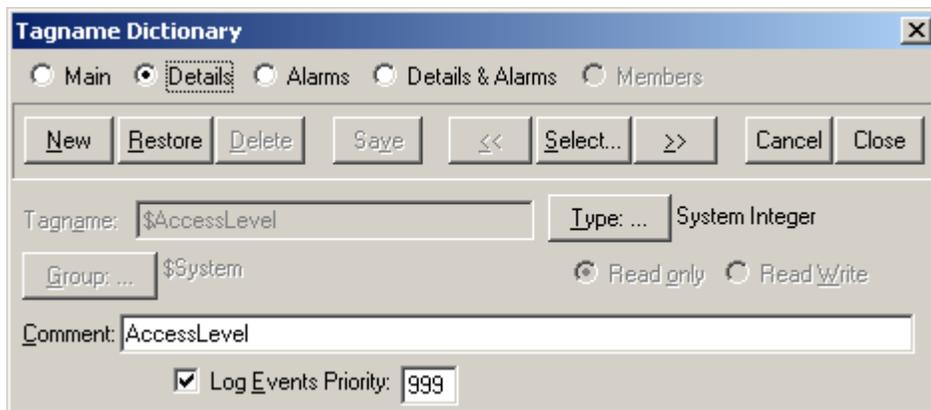
- Protocol Timer Tick:** A text input field containing "50" followed by "msec".
- NetDDE being used**
- Configuration File Directory:** A text input field containing "C:\Program Files\FactorySuite\I/O Server\MODBUS\".
- Start automatically as Windows NT Service**

The main item of interest here is the last checkbox 'Start automatically as Windows NT Service'. If checked, this will force the ModBus I/O Server to automatically start each time the computer is started. You will probably want to check this to save having to start the Server manually each time. Click 'OK' after configuration and minimize the ModBus I/O Server.

The next step in this process is to configure WonderWare. An 'Access Point' and 'Tag Names' must be created that will allow WonderWare's InTouch Process Visualization software to communicate with the ModBus I/O Server. Start 'InTouch' and create a new project. The following screen should now appear:



In the window on the left, double click 'Tagname Dictionary' and the following dialog should now open:



Click 'New' since tag names must be created for your external I/O and set the 'Tag Type' to "I/O Discrete". The dialog should now appear as below:

The screenshot shows the 'Tagname Dictionary' dialog box with the following fields and options:

- Buttons: New, Restore, Delete, Save, <<, Select..., >>, Cancel, Close
- Radio buttons: Main, **Details**, Alarms, Details & Alarms, Members
- Fields: Tagname: [], Type: ... I/O Discrete
- Fields: Group: ... \$System, Radio buttons: Read only, **Read Write**
- Field: Comment: AccessLevel
- Checkboxes: Log Data, Log Events, Retentive Value
- Initial Value: Radio buttons On, **Off**
- Input Conversion: Radio buttons **Direct**, Reverse
- Fields: On Msg: [], Off Msg: []
- Field: Access Name: ... Unassigned
- Field: Item: []
- Checkbox: Use Tagname as Item Name

Enter a "Tagname" in this field and use something descriptive about that particular I/O point. Next click 'Access Name: ...' to bring up the following dialog:

The screenshot shows the 'Access Names' dialog box with the following elements:

- Buttons: Close, Add..., Modify..., Delete
- Field: []

Click 'Add...' and edit the following dialog:

As you can see, we have already filled in the information. The 'Access Name' is arbitrary and for simplicity it should match the 'Application Name'. The 'Application Name' must be ModBus since that is the I/O Server we are using in this example. The 'Topic Name' must match the name used earlier when creating a 'Topic Definition' in the ModBus I/O Server configuration. As mentioned earlier we will exchange information with the I/O server thru DDE so be sure this is the selected protocol. After entering all settings, click 'OK'. You should now see a dialog similar to the one below:

The last piece of information that must be entered is the 'Item' name. In our example we have chosen to use the actual ModBus I/O address for the 'Item' name. At this point, click 'Save' to complete this



tagname. Create any additional tagnames by clicking 'New' and repeating this process. You won't have to define another access point unless the I/O point is accessed thru another I/O server. After creating all tagnames for your external I/O, you are now ready to create your process visualization screens. You can animate your screens by attaching an I/O points tagname to its visualization object.