HC900 Hybrid Controller

When you need more than just discrete control

HC900 Connection to Wonderware – Product Note

Table of Contents

1.0 Introduction and Scope

This document is intended to assist a systems integrator that is familiar with both Wonderware FactorySuite InTouch and with the Honeywell HC900 controller and software. This document will help to link the two together using one specific driver, using one specific addressing scheme, with emphasis on uncovering and explaining undocumented information that was discovered during a project. A demo HC900 program (HC900 Demo.cde) was used as the project program.

There are many driver candidates for communicating via Modbus Ethernet to the HC900, as any Ethernet Modbus OPC driver should allow communication. In addition, Wonderware has a native Ethernet Modicon Modbus driver and the author has successfully linked Wonderware to the HC900 demo program and verified basic operation of this driver.

The driver that was used in this project and that will be discussed in this document is a Kepware OPC driver specifically written for the HC900 and available as of January, 2003. Specifically, the name of the driver is the Honeywell HC900 Ethernet driver, and it runs within the KepwareEX OPC Server (version 4.82.223 used for this project, free demo download available).

As will be shown later, the Kepware driver will automatically generate tags within the OPC server. These tags are not required (depending on addressing scheme used), but will be browsable by most OPC clients. Because of the way the Kepware OPC server integrates with Wonderware, the tags are not browsable directly by Wonderware, at least not without using Wonderware's OPCLink. The use of OPCLink is not required in order for data exchange to occur. The points in this project will be built within InTouch very much like the points for any other native InTouch driver are built.

This document will cover only one addressing scheme, (there are multiple variations). It will show how to create tags within Wonderware InTouch that use dynamic addresses. This means that the tags will not reference the tags that may or may not be built in the OPC server; the tags will directly reference the Modbus address of the data in the HC900.

It might be noted that in working on a large project, the Wonderware database can be created by exporting the automatically created OPC server database and in turn, importing it into an InTouch project. This method would easily create InTouch tags that reference the KepserverEX tags. The details on how to import and export between Kepware and InTouch are not detailed here.

2.0 HC900 Initial Setup

The HC900 automatically assigns a Modbus address as variables, signal tags and function blocks are added to the system. The first signal tag address is assigned to 8193 (0x2000). PID loops and Set Point Program blocks have unique addresses that are always assigned the same. Getting slightly ahead, an InTouch tag would ask for 408193 to receive this address.

Setup of the HC900 is fairly straightforward. Use standard Ethernet patch cables if you are using a Hub/Switch. Use a crossover cable if you are communication directly to the HC900 top Ethernet port. The top port is used for communications to the HC900. The bottom port is dedicated to remote I/O.

The IP address of the HC900 is changed from the Utility Tab. Set the IP and subnet mask to something that will network with your current Wonderware PC configuration. It is beneficial to add the IP address to the HC900 configuration list prior to actually changing the address. To do this, at the bottom of the Utility screen, click the NETWORK button. Add the new address of the HC900 if it does not exist. Set the toggle switch on the controller to the center position to allow online mode changes. Using the SET CONTROLLER MODE button, place the controller in program mode.

While on line to the HC, ether by the Ethernet or serial port, click the SET CONTROLLER NETWORK PARAMETER button. Click the second radio button for changing Controller Name or Modbus TCP Register format and press enter. Click the check box mid-left of the window next to the "!" in the yellow diamond as shown in Figure 1. This will change the IP Address entry box from grayed to white. Change the address as required and select "NEXT". If using Ethernet communications, you will lose communications and must go back to the utility page and change the CURRENT PC TO CONTROLLER CONNECTION SETTING to the new IP address.

Go back to the SET CONTROLLER NETWORK PARAMETER screen, click the third radio button that allows changing the TCP Double Register Format and click next. The CONTROLLER IDENTIFICATION SETUP WIZARD will open. Change the Modbus TCP double register format to FB B - BIG ENDIAN as shown in Figure 1.

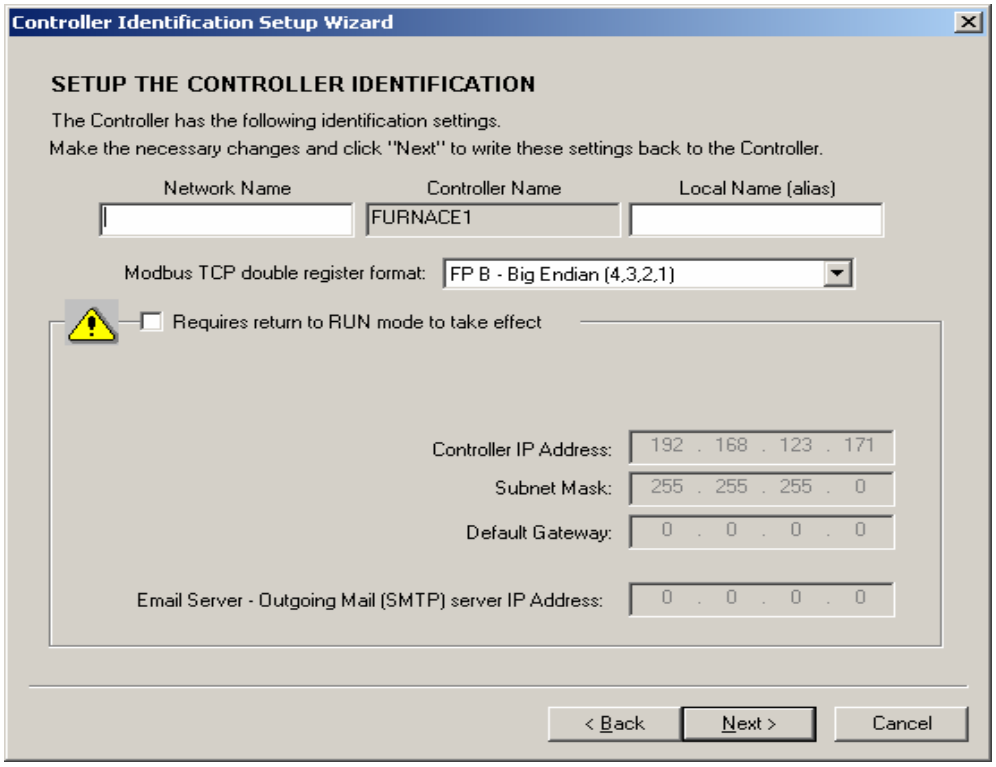

Figure 1, Setting the Floating Point Format and Network Settings

3.0 Kepware OPC Server and Driver Setup

3.1 Adding an OPC Server Channel

Install the Kepware OPC server (KEPServerEX) and driver (Honeywell HC Ethernet) software. Starting with a new server project, the first step is to add a channel. See Figure 2 below.

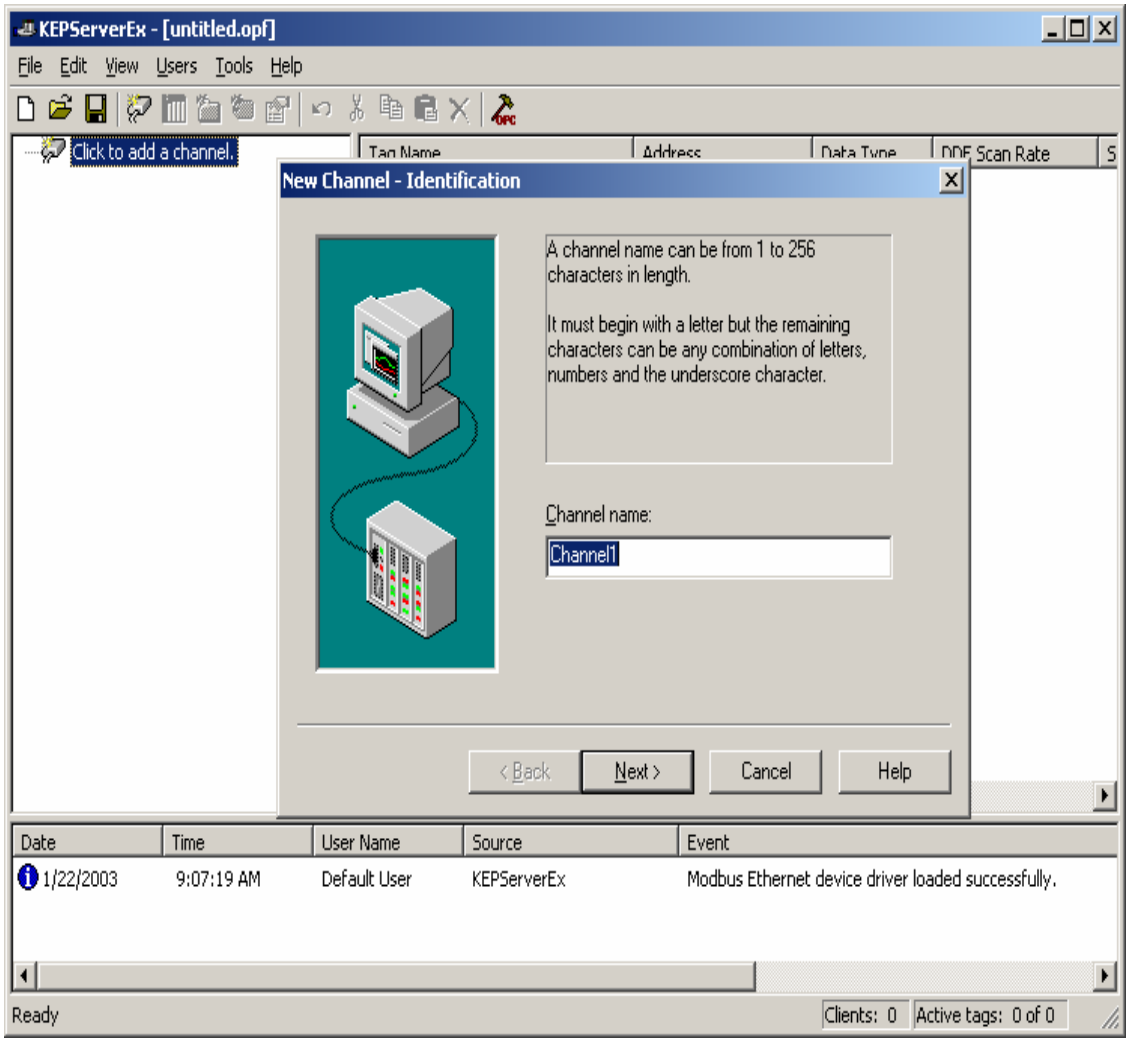

Figure 2, Kepware OPC Server, Setting up a Channel

Click on NEXT and select the Honeywell HC Ethernet driver and enable diagnostics as shown in Figure 3 below.

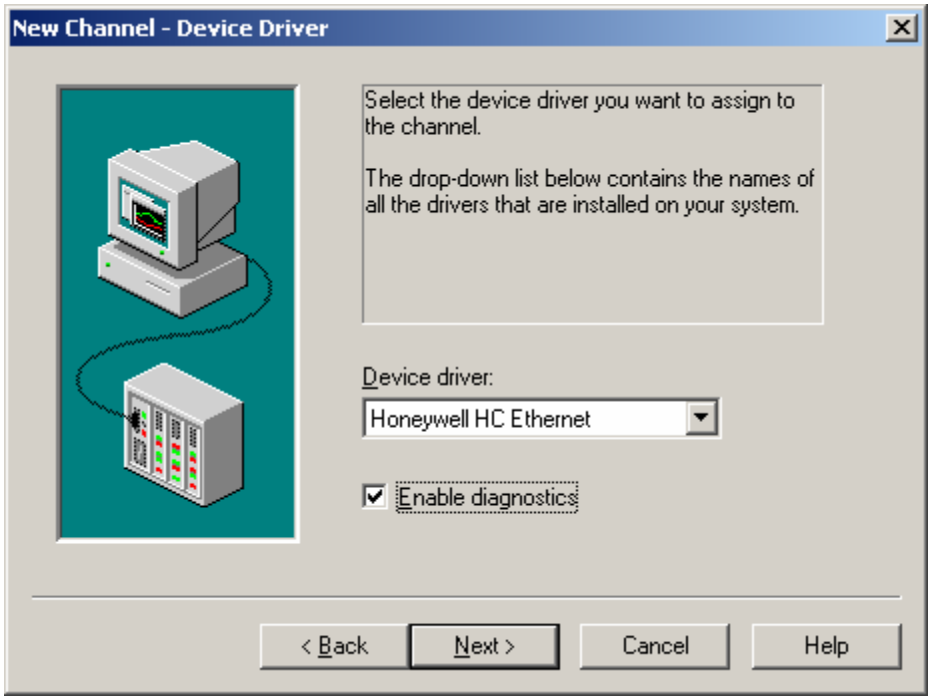

Figure 3, Selecting OPC Server Driver

Click NEXT and select the network adapter that the OPC Server will connect to. If you have only one adapter on the PC, you will not have a choice here. See Figure 4 below.

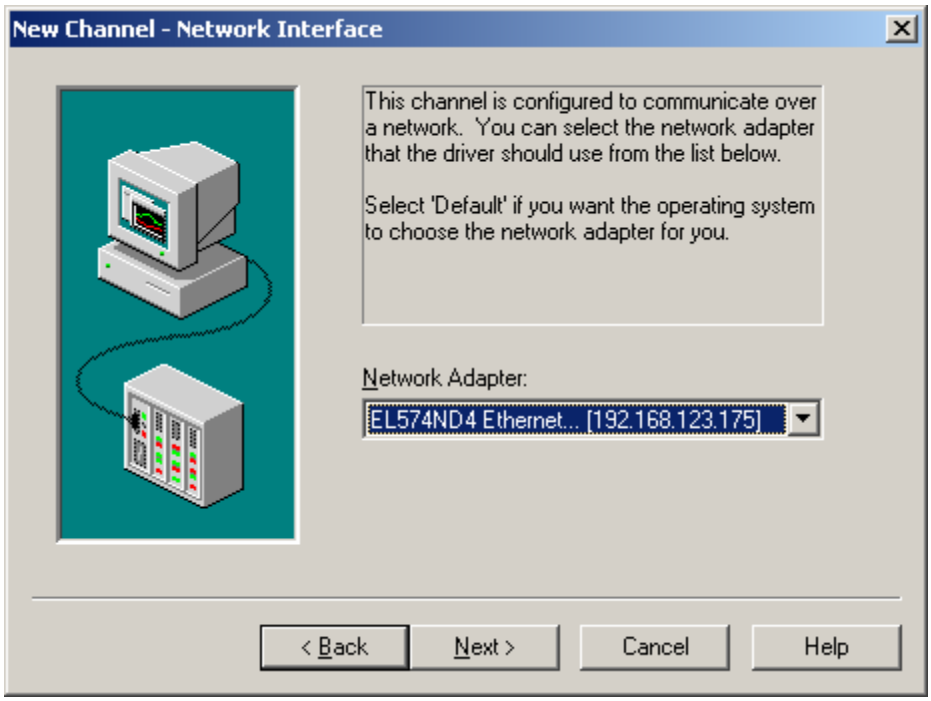

Figure 4, Selection Network Adapter for OPC Server

Click NEXT and you will see the Write Optimizations dialog box as shown below. These settings can be left in the default position.

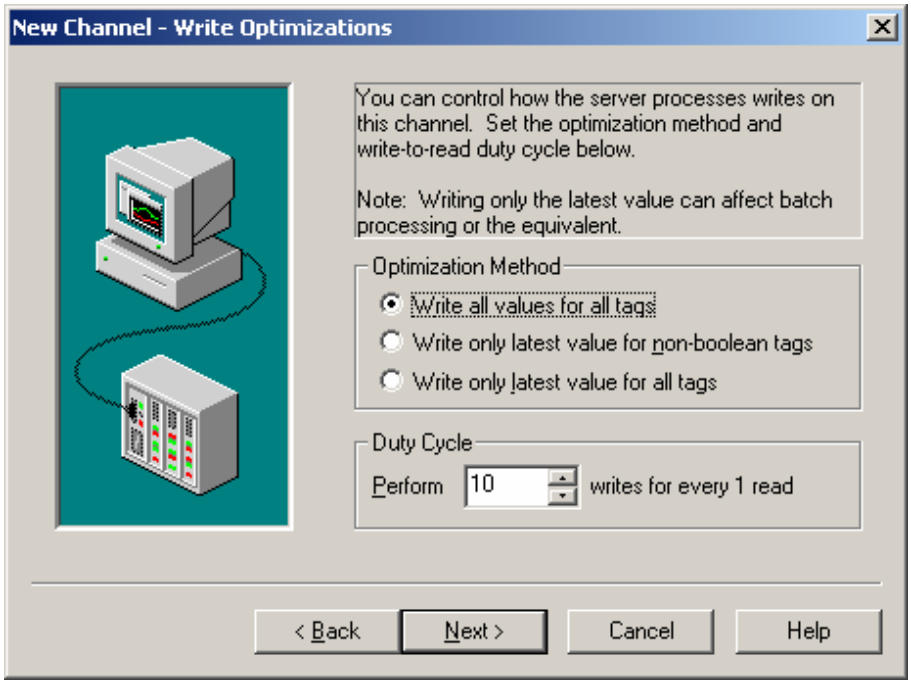

Figure 5, Write Optimizations, (leave at defaults)

3.2 Adding an OPC Server Device

Save the server configuration and continue by adding a device. For this project example the device name used will be HC900. See Figure 6 below.

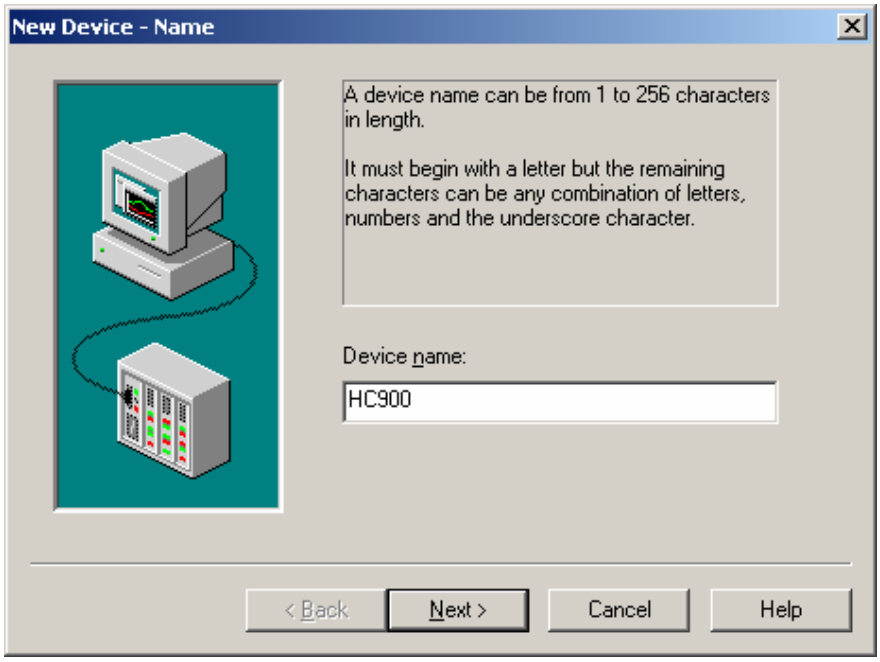

Figure 6, Assigning Device Name

Click NEXT and assign the IP address of the HC900. This is the same IP shown in Figure 7. You must use an address that is compatible with your PC configuration.

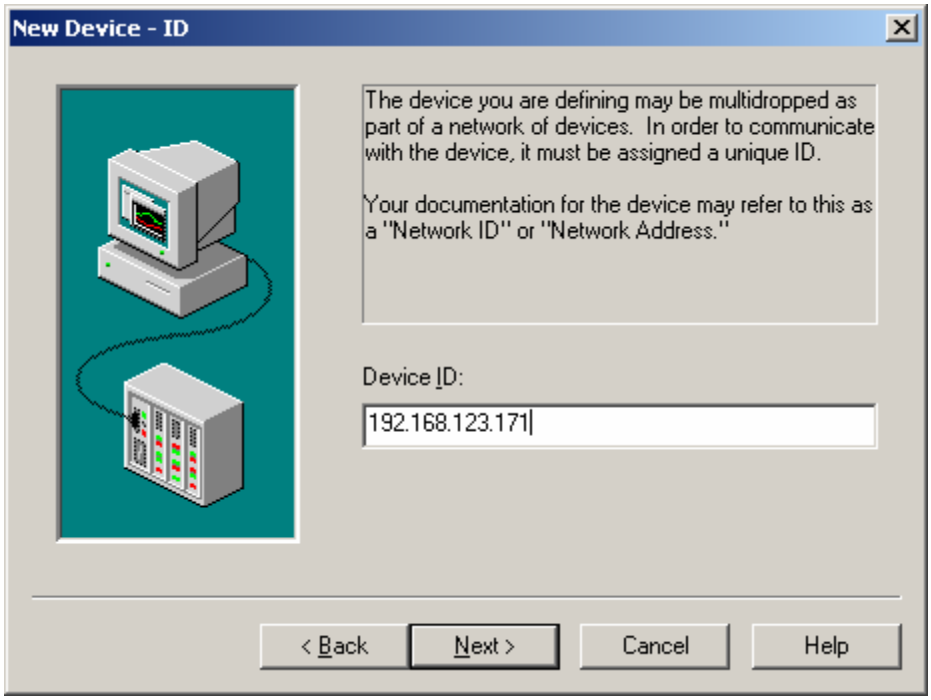

Figure 7, Assigning Device IP Address

Click NEXT and you will see a dialog box to set communication parameters. These can be left at the default settings as shown below in Figure 8. These setting can be changed later after the device has been created.

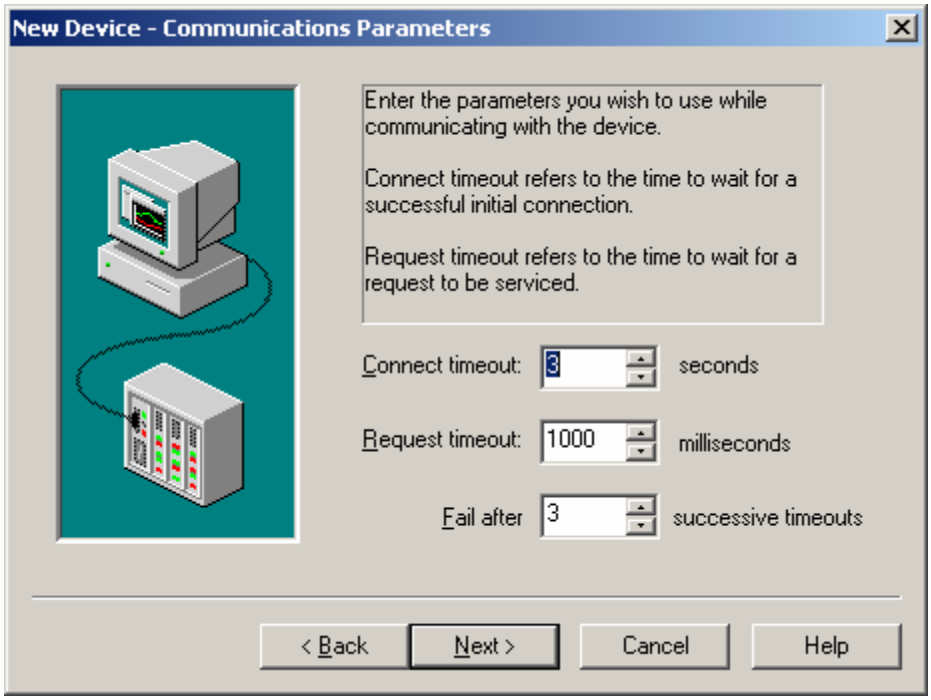

Figure 8, Communication Parameters (leave default)

Click NEXT and you will see a dialog box to set the TCP/IP port that the device uses. Leave this at the default 502 as shown below in Figure 9.

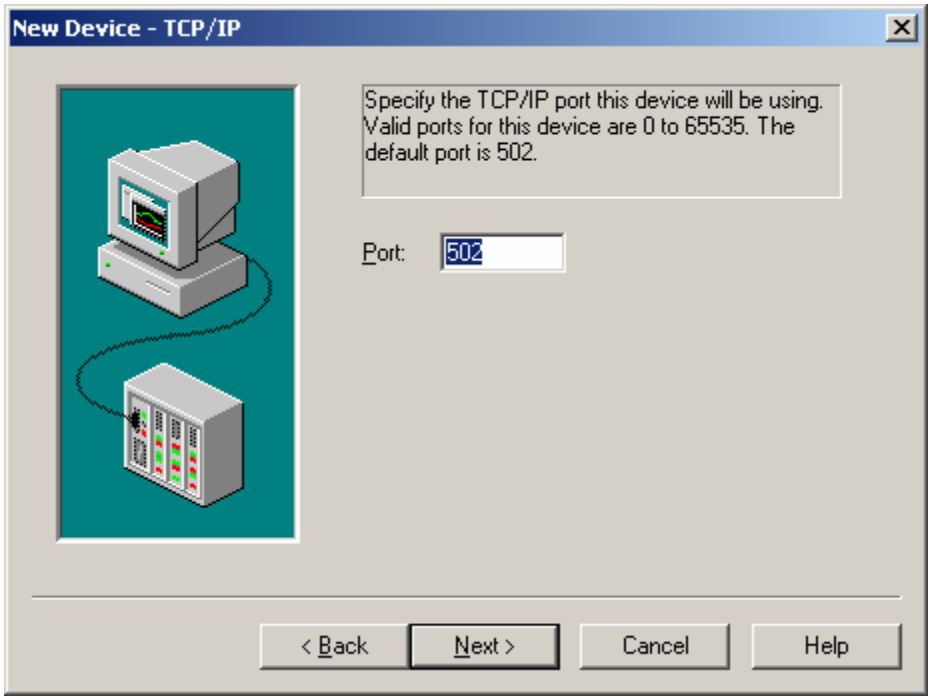

Figure 9, Device TCP/IP Port (leave default)

Click NEXT and you will see a dialog box that sets the IEEE floating point word order. Leave this box unchecked as shown below in Figure 10.

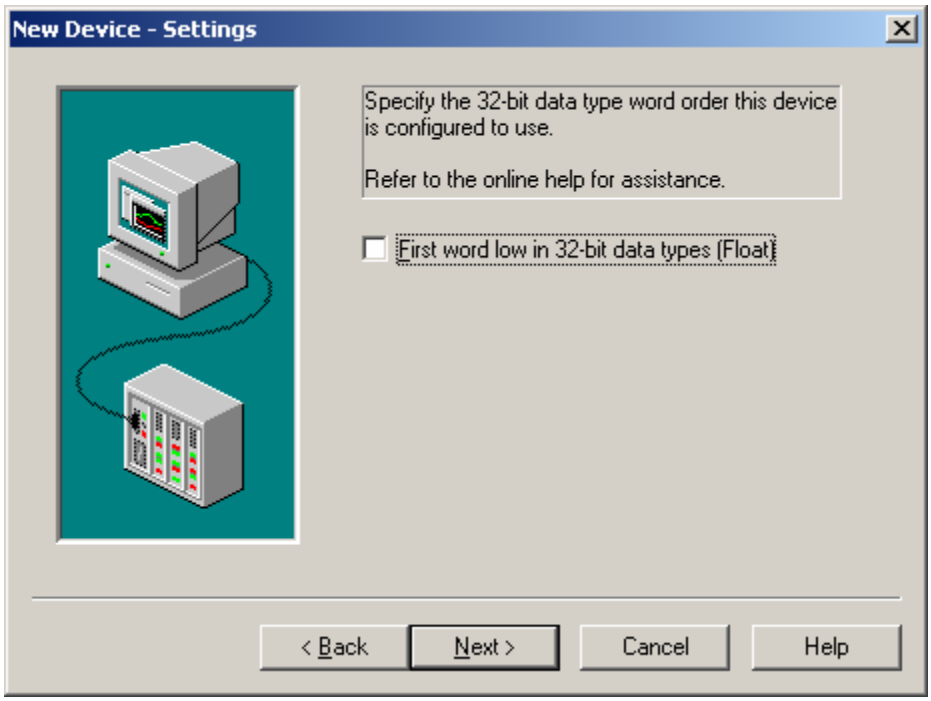

Figure 10, Setting Word Order for Floating Point Numbers

Click NEXT and you will see a dialog box used for setting the maximum block size when reading data from the device. These settings can be left as default as shown below in Figure 11. For more information on optimizing these settings see the Honeywell document HC900 Hybrid Controller Ethernet Modbus/TCP Communications User Manual, page 10.

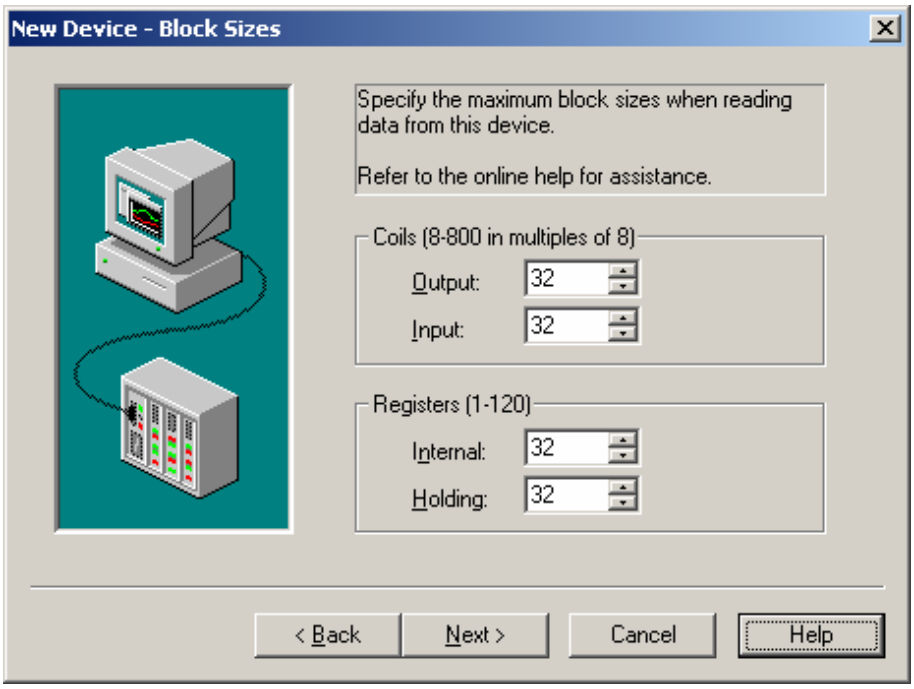

Figure 11, Specify Data Block Sizes

Click NEXT and you will see a dialog box that asks for the number of data types that are found in the HC900 for which you want OPC Server tags automatically created. As discussed earlier, OPC Server tags are not required to communicate to InTouch, but if they are created, any other OPC client can browse them. For this project OPC Server tags were created, in order to illustrate the process. A benefit of allowing the driver to create these tags is that they automatically provide the formatted Modbus address for all the HC900 variables. For the demo program, the following tags were created as shown in Figure 12 below.

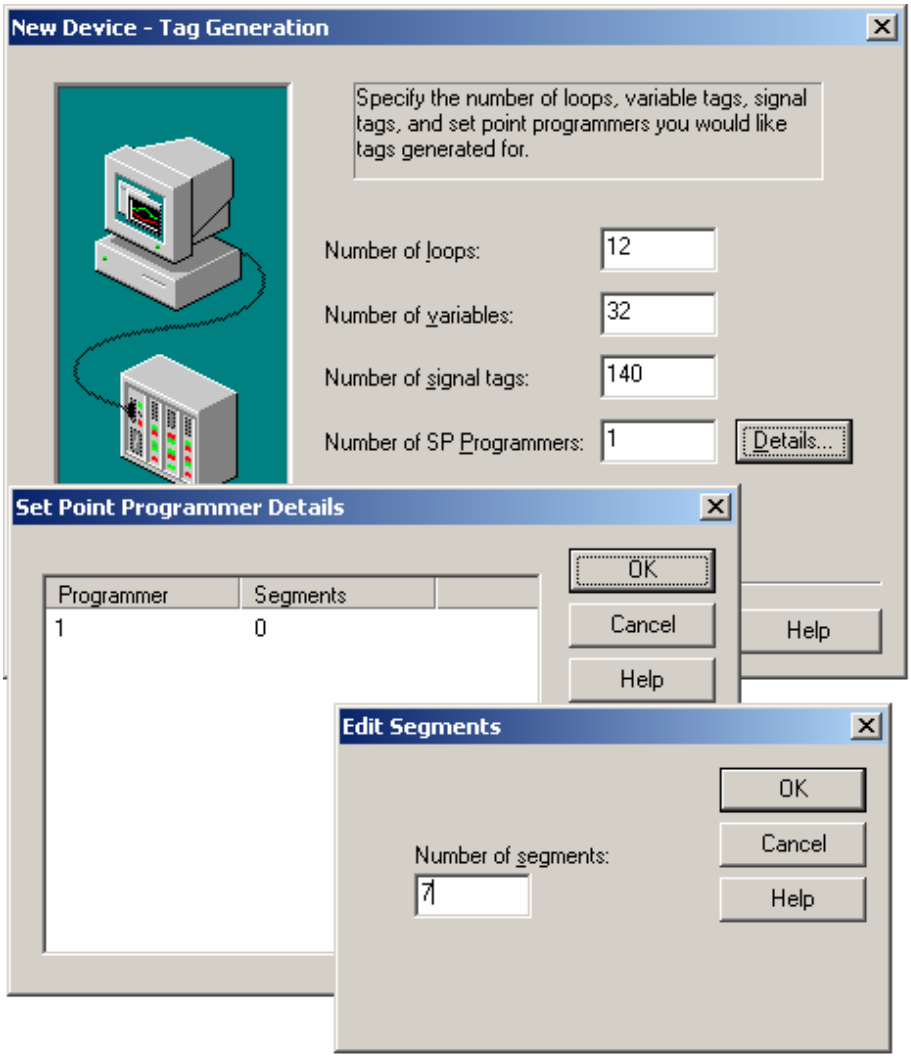

Figure 12, Automatic OPC Server Tag Generation

Click NEXT and FINISH and the application will create the tags and finish creating the device. The OPC Server is now ready to serve any client that wants to connect. If you start the Kepware QuickClient on the toolbar and look at the tags you will see live data if your HC900 is connected and everything is properly configured (see Figure 13 below). Notice that because both Signals and Variables can be used either as discrete or analog values with the HC900, the KepwareEX application creates two tags for each of these, one of each type.

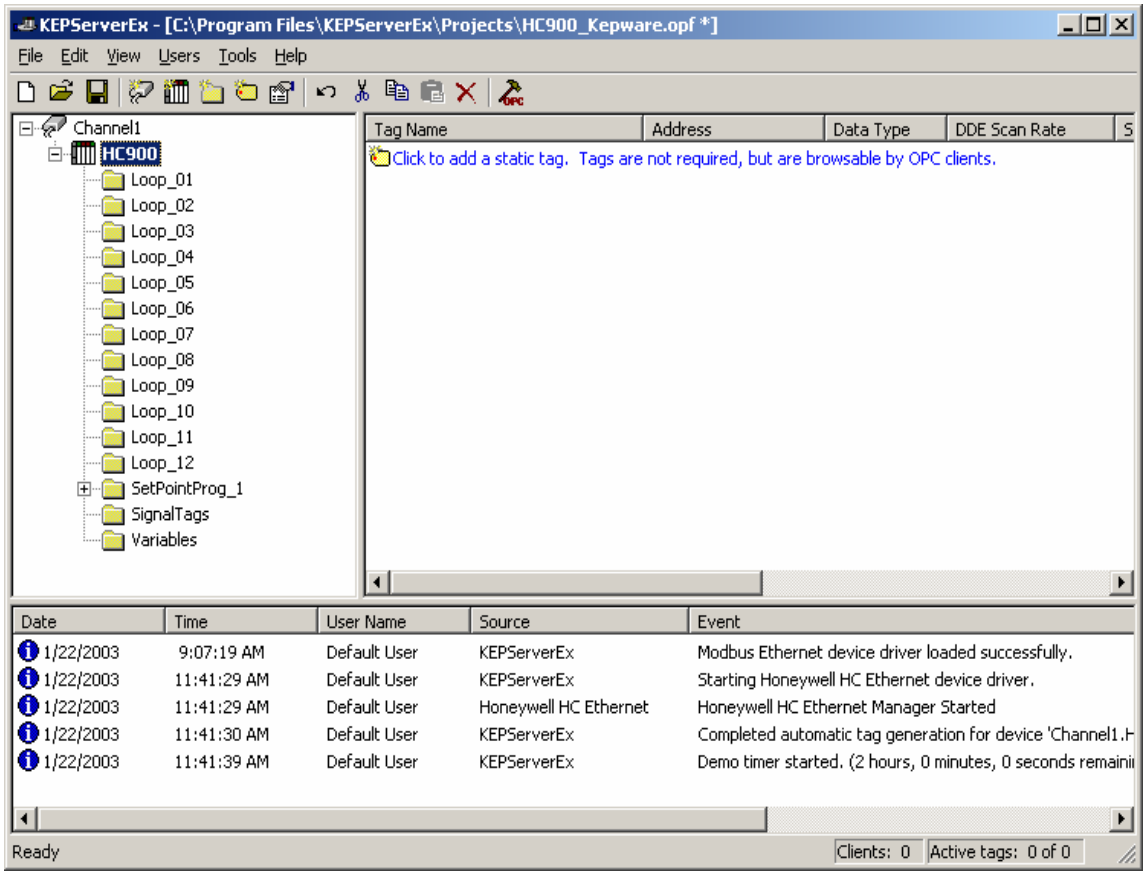

Figure 13, Completed OPC Server Configuration

Test the configuration by starting the Kepware QuickClient and you should see live data. If not, troubleshoot the configuration settings of the HC900, the Server, and the PC network settings.

4.0 Wonderware InTouch Configuration

4.1 Creating a Project and an Access Name

Starting with a new InTouch project, the first step is to create an Access Name. On the project tree double click the Access Names icon under the Configure heading and create a new Access Name as shown below in Figure 14.

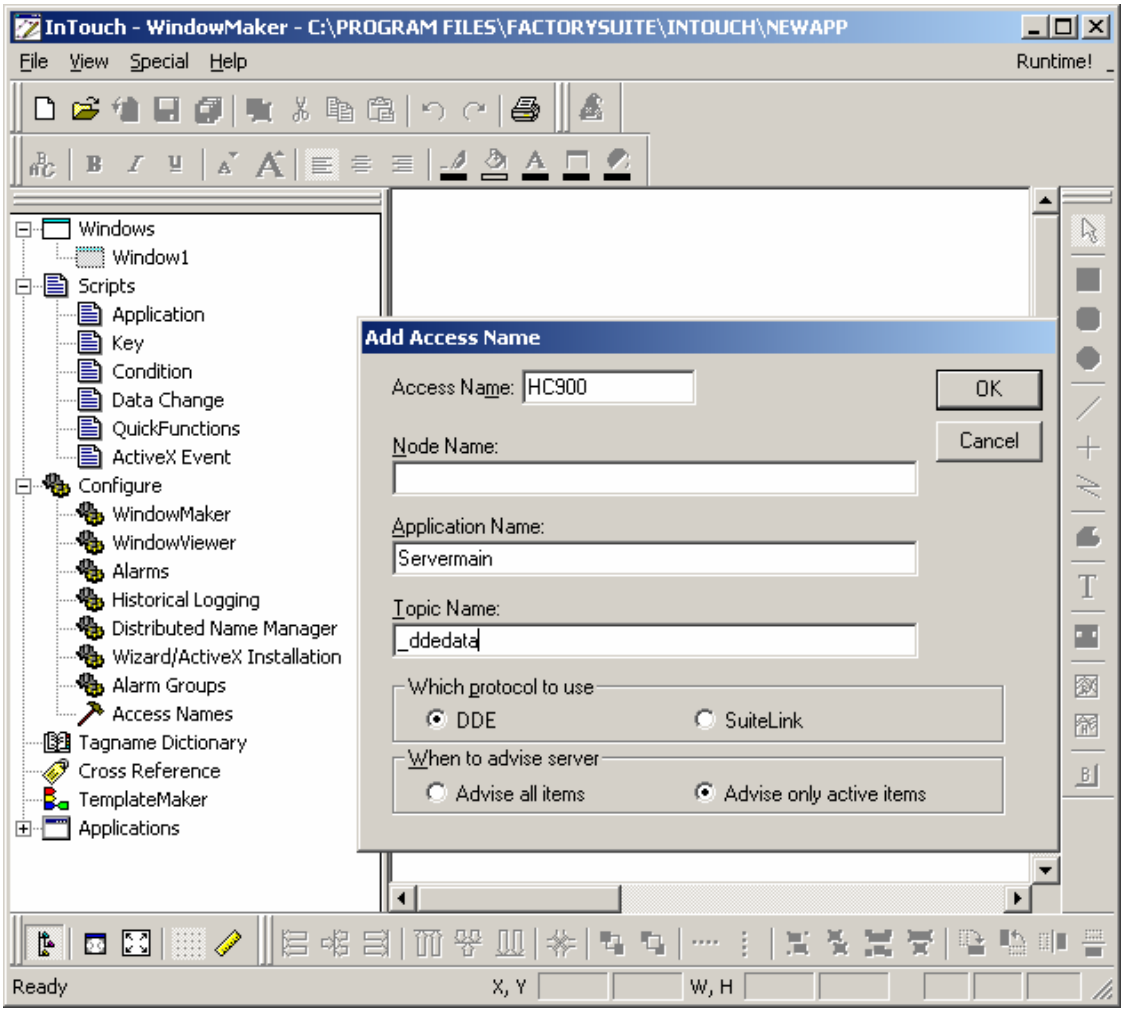

Figure 14, New InTouch Project, Adding new Access Name

4.2 Addressing an Analog Value

Tags can be created individually or imported into InTouch. To create a single analog tag, configure a new tag within the Tagname Dictionary tool as shown below in Figure 15. This example tag is for SignalTag 67, which is an analog Signal tag within the demo program in the HC900. The point is addressed within InTouch as an I/O Real data type. The Modbus address of this tag is 415333 and it is an IEEE floating point value. In addition, to address an analog input or output, this type of tag (I/O Real) would be built. As discussed earlier, the addressing shown in this document is only one of several addressing methods that will work using the KepwareEX server. Any given analog point should be configured as required for read/write settings, ranging, initial value, deadband, etc, per the standard InTouch point configuration dialog box. See Figure 15 below.

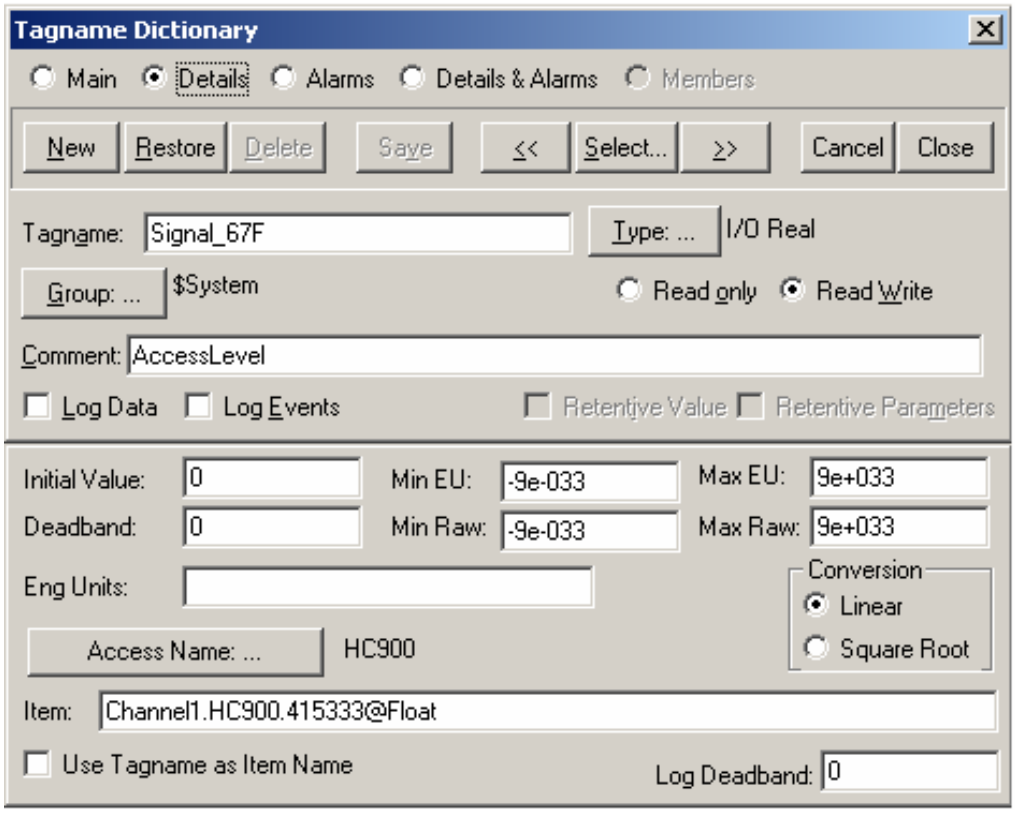

Figure 15, Analog Signal Tag Addressing, I/O Real Type, Read/Write Access

4.3 Addressing a Simple Discrete Value

To create a simple discrete tag, configure a new tag within the Tagname Dictionary tool as shown below in Figure 16. This example tag is for Signal 137, which is a discrete Signal tag within the demo program in the HC900. The Modbus address of this tag is 406399 and it is an IEEE floating point value. The point is addressed within InTouch as an I/O Discrete data type as shown below. This addressing format works for both Signal and Variable tags that are used within the HC900 as discrete variable types. In addition, to address a digital input or output, this type of tag (I/O Discrete) would be built. Any given discrete point should be configured as required for read/write settings, initial value, messages, conversion, etc, per the standard InTouch point configuration dialog box. See Figure 16 below.

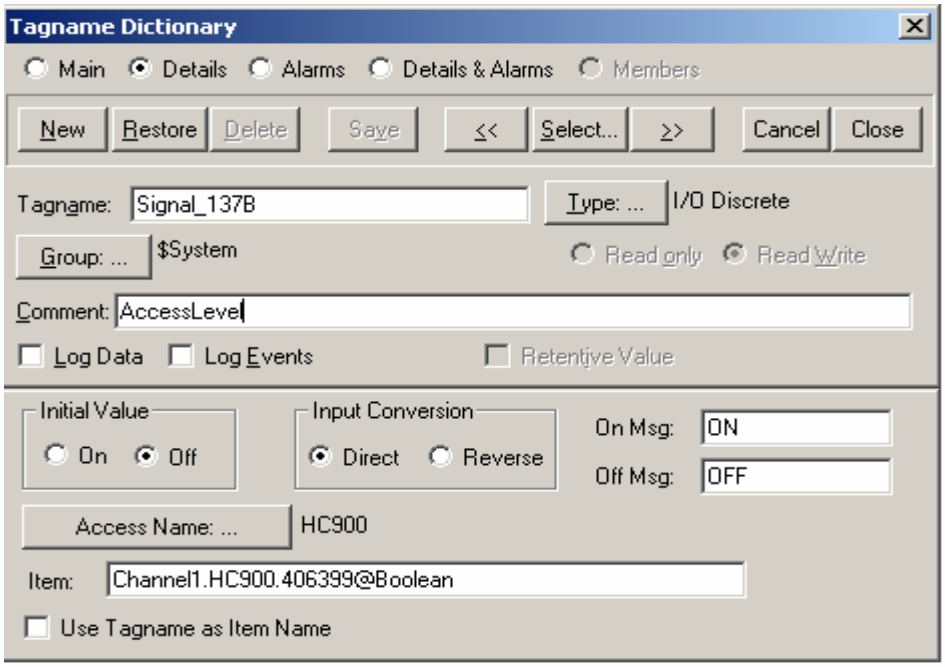

Figure 16, Discrete Signal Tag Addressing, I/O Discrete Type, Read/Write Access

4.4 Addressing a Bit-Packed Discrete Value

To create a discrete tag that looks at a bit-packed binary value (a specific bit within an integer word), configure a new tag within the Tagname Dictionary tool as shown below in Figure 17. This example tag is for the PID9 parameter AUTO/MANUAL, which is the zeroth bit within the Modbus address 402299. The point is addressed within InTouch is an I/O discrete data type as shown below. Packed bits can also be addressed as InTouch integer data types. Any given discrete point should be configured as required for read/write settings, initial value, messages, conversion, etc, per the standard InTouch point configuration dialog box. See Figure 17 below.

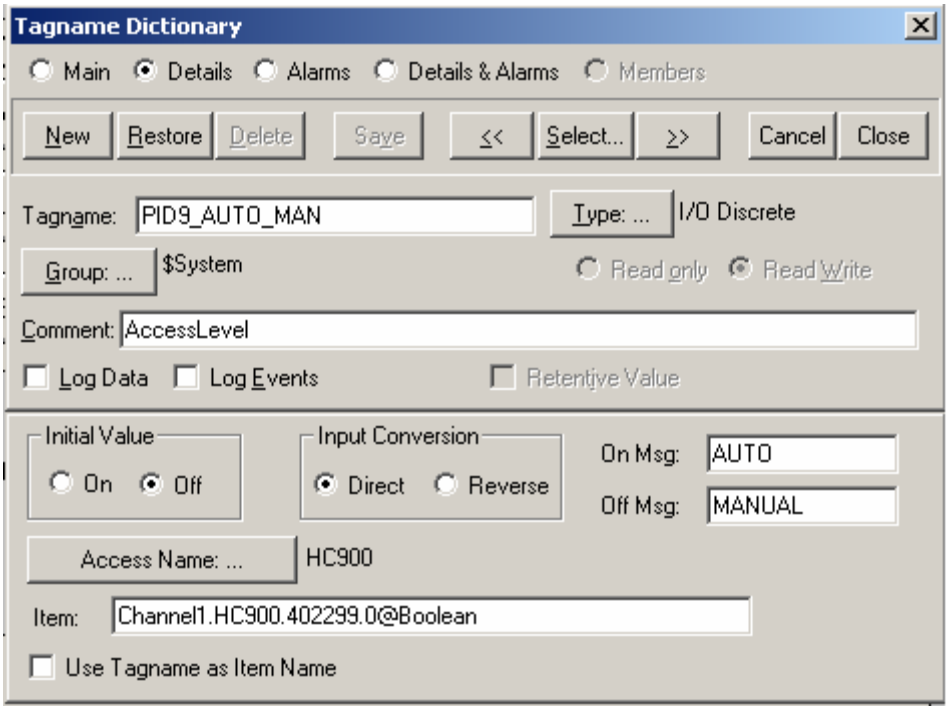

Figure 17, Discrete Packed-Bit Addressing, I/O Integer Type, Read/Write Access

5.0 The HC900 Set Point Programmer Block

The HC900 can have up to 8 SPP blocks assigned within a program. An individual tag is required in InTouch for each address within the HC900 that communicates across the driver. Because of a tag database limit of 32 tags when running InTouch in the demo mode, only the basic functionality of the SPP is configured in the InTouch demo application; it will allow starting, resetting, holding, and monitoring a setpoint programmer in the HC900 demo program.

This discussion will be for SPP #1 as assigned within the Demo program. Additional SPP blocks have a fixed offset address from the first block. The first SPP address is 407681 and runs contiguous through block 407696. Blocks 407681 through 407690 are 32-bit floating point numbers and require two words per number.

5.1 Enable, Run, Hold, and Reset of the SPP Block

The addresses assigned to Start, Hold and Reset are registers 407693, 407694, and 407696. They are two byte words and any value other than Zero sent to the address causes the command function to occur. These are declared as Write only blocks by Honeywell. These addresses within the HC900 automatically reset to Zero once a non-zero value is detected.

5.2 Monitoring and Changing Set Point Programmer (SPP) Profiles

The HC900 can store up to 99 profiles in internal memory. These profiles contain up to 50 segments or steps of ramp and soak. This profile becomes the recipe through which

the SPP block advances. Profiles may be created with Hybrid Control Designer and downloaded to the HC900 memory.

Each of the SPP blocks has a unique memory location to store a working copy of a profile. With the SPP block in Ready mode, a Profile Number is entered into the Current Program Number, address 407937 for SPP#1. The HC900 retrieves the matching recipe profile from memory and enters it as the SPP block's current working profile. The data for SPP Block #1 is then available for review starting at address 410241. While in Ready mode, the current profile can be cleared by calling profile Zero, or an existing profile may be recalled, modified with new data, and stored to the same or different profile by entering a profile number in address 407939. There is no over-write protection within the HC900, so entering a number into the New Profile Number will overwrite an existing profile. InTouch can be configured so the data entry link for entering the New Profile Number brings up a popup window that requires confirmation prior to down-loading the entered value. This would allow some protection against over-writing of existing profiles. The data link used to enter a new profile number could also be made invisible when the block is not in the Ready mode so that changes cannot be attempted unless the SPP block is in Ready mode.

Hybrid Control Designer allows a description to be assigned to each recipe. This description is NOT downloaded to the HC900. If a description of the recipe is required for display on an InTouch screen, it will need to be stored within InTouch or a PC-based database and be recalled as the recipe number is changed.

One enhancement that was not attempted but would be good practice is to develop a current segment value display using the present segment number as an offset within the segment map. The SP and Time/Rate value along with the events could be displayed for review by the operator. Once this is developed, a "next" segment display could also be created.

5.3 InTouch Screen for the Setpoint Programmer Block

The Setpoint Programmer Screen displays a small portion of the current SPP#1 Program that has been recalled from internal HC900 memory. With the SPP block in Ready mode, the recipe is recalled by entering a value into the data link on the screen. When the new recipe number is downloaded to the HC900, the recipe stored in the memory location of the entered number is copied to the working recipe for SPP#1. The first segment details are displayed on the screen. The data links for the segment are a combination of AI and DI points being read from and written to the HC900 using tags configured as shown previously in this document. The stored working recipe for SPP#1 starts at Modbus address 410241 with the first segment running through 410248, and requires eight words per segment. Each segment consists of two bit-packed integers and three floating-point words. The end address for the fiftieth segment would be 410640.

With the SPP block in Ready mode, the values for the recipe can be changed. To get to Ready mode, a running SPP block must first be put in Hold mode by clicking the HOLD pushbutton. An SPP block in Hold or Completed mode can be changed to Ready mode by clicking the RESET pushbutton. A RUN pushbutton is provided to start the SPP execution from the Ready mode.

As values are changed, the working copy of the recipe assigned to the SPP#1 Block is changed. If the SPP block is changed to Run mode, these values will be used. As stated earlier, the values can not be changed in any mode other than Ready.

If a permanent change to the recipe is desired, or the modified recipe stored to a new location, a value can be entered into the data link at the upper right section of the screen and the recipe will be stored to that recipe number. Once the recipe is stored to an assigned storage location, it could be recalled for use in SPP#2 or other block. On this display, the working recipe pertains to SPP#1 only. To display the working memory for another SPP would require a different screen or the use of tag group substitution to redirect the data links to a different SPP.

Each segment of the profile contains a 16-bit word with each bit assigned as an event within the segment. These are used to turn on and off discrete events as the segments are stepped through. On the right side is a break down of the status bits. These are used to determine the Ready, Run, Hold or End of the SPP1 active profile.

Three push buttons are included within the screen. The Run, Hold and Reset buttons write a value of "1" to the appropriate address within the Status word of the SPP1 block.

Also displayed are several of the status words showing time variables and current segment within the SPP1 block.

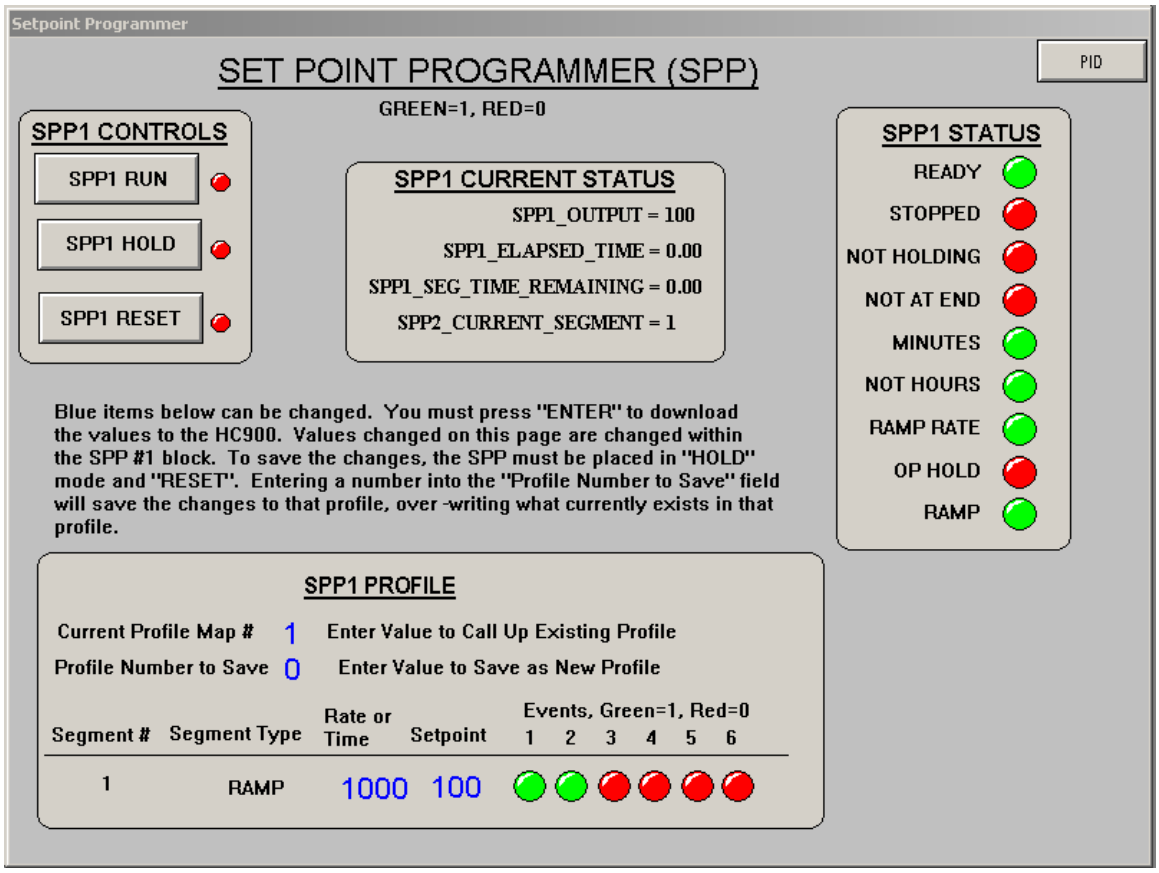

Figure 18, Setpoint Programmer Interface Through Wonderware InTouch

6.0 The PID Block

6.1 Connecting to the PID Block

The HC900 can have up to 32 loops configured per program. The first loop starts at address 400065 and has a set configuration of internal parameters that occupy 256 addresses. Loop numbers 2 through 24 are incremented by the 256 address length. Loop 25 and following have an address offset that is not contiguous with the previous loops and this should be noted from the Honeywell manuals. The InTouch demo program uses PID9 starting at address 402113.

6.2 InTouch Screen for the PID block

The PID and Data Types screen shows a basic PID faceplate that connects to PID9 in the HC900 demo program. Shown are the Setpoint, Process Variable, and Output, all with a bar graph. There are two buttons for toggling the modes for AUTO/MANUAL and REMOTE SETPOINT/LOCAL SETPOINT. When in the Manual mode there is a graphical tool for incrementing/decrementing the output value. See Figure 19 below.

7.0 Other Data Types

7.1 Signal Tags, Floating Point and Boolean

As mentioned earlier, Signal and Variable tags can be used either as floating point numbers or as Boolean variables. The PID and Data Types screen shows some examples of interfacing with Read only and Read/Write data types.

7.2 Accessing Digital Bits

The HC900 Demo program has a routine under Logic/Timing that is titled CYCLING DIGITAL OUTPUTS. This logic once started, turns on six digital outputs. As a further example of interfacing with digital bits, the PID and Data Types screen shows the status of these outputs along with a START/STOP button. InTouch writes a value of one to VARIABLE 33 to start the cycle. When the stop button is pushed, a value of zero is written to VARIABLE 33. The logic cycle does not stop until all six lights are turned on. See Figure 19 below.

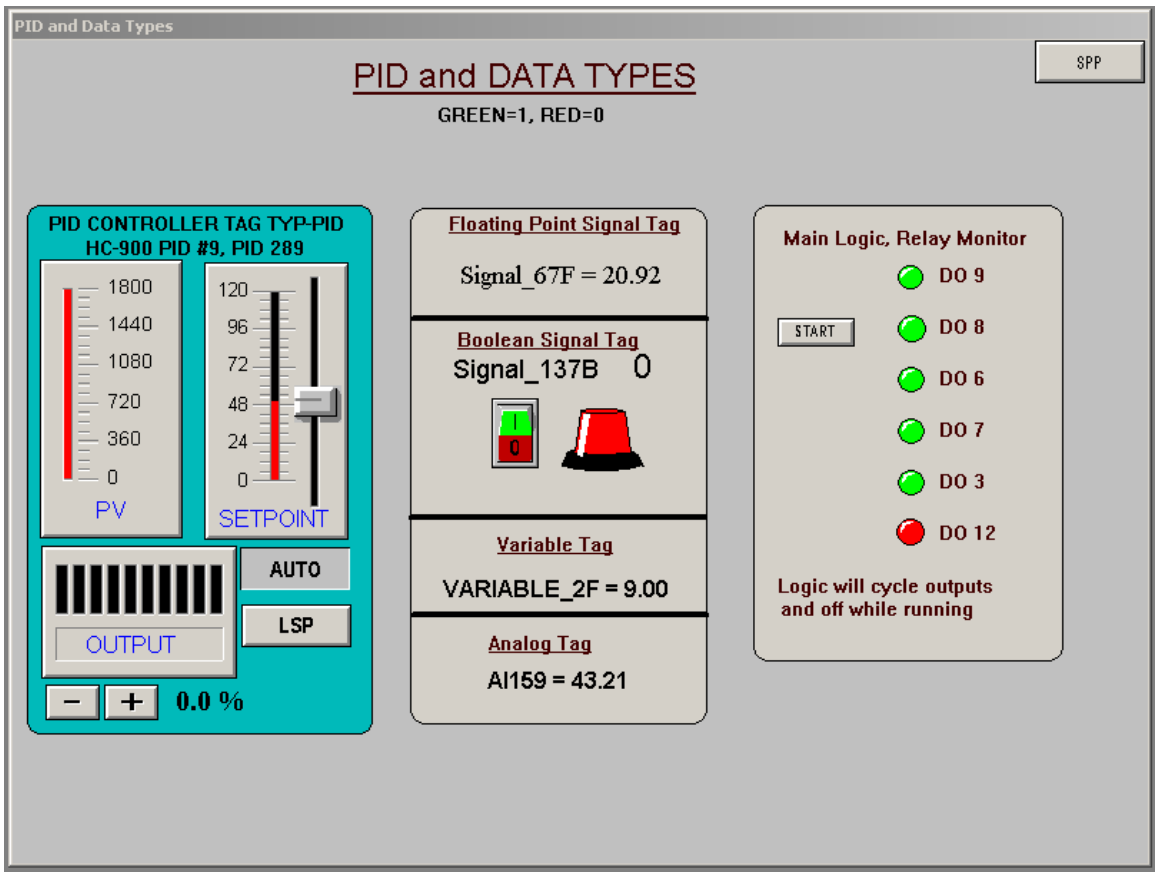

Figure 19, PID Faceplate, Example Data Types, and Relay Monitor