



Setup Examples

NetArrays Project Program Development

NetArrays Project Program Development Example

© 2005, 2007, 2008, 2009 RTP Corporation

Not for reproduction in any printed or electronic media without express written consent from RTP Corp.

All information, data, graphics and statements in this document are proprietary intellectual property of RTP Corp. unless otherwise indicated and are to be considered RTP Corp. confidential. This intellectual property is made available solely for the direct use of potential or licensed RTP Corp. customers in their application of RTP Corp. products, and any other use or distribution is expressly prohibited. If you have received this publication in error, immediately delete, discard or return all copies to RTP Corp.

RTP Corporation
1834 SW 2nd Street
Pompano Beach, FL 33069
Phone: (954) 974-5500
Fax: (954) 975-9815
Internet: <http://www.rtpcorp.com>

File Name: NetArrays Project Example.pdf
Last Updated: 1/21/09

NetArrays Project Program Development

The RTP3000 system supports both safety and control in a single application. The example provided in this document demonstrates how easy it is to use this configuration tool. It steps you through the process of creating SIF's, adding control and providing a secure location for HMI/operator access.


In a matter of minutes you are able to create hundreds of points for programming. We will configure a traditional TMR system for safety and add another chassis for critical I/O to do control. The safety system will consist of 64 Triple redundant analog inputs and 16 dual redundant digital outputs. The DCS will consist of 32 digital inputs and 12 digital outputs.

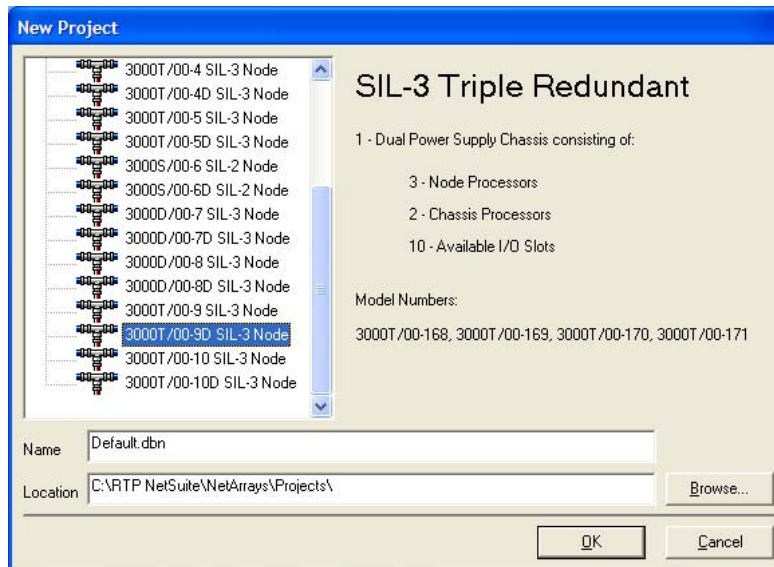
The only equipment required is a PC or laptop with NetSuite installed. To verify the project program we will run on the NetArrays built-in Simulator.

Note that the NetArrays program developed here will be used by the Setup Examples "RTPView Project Program" and "Redundant Alarm Monitoring Configuration."

Start NetArrays


Start NetArrays

- On your PC, press  **Start** and select **Programs ▶ RTP NetSuite ▶ NetArrays**.
- Click File, then New Project
- Select the 3000T/00-9D SIL-3 Node as shown below
- Click OK to save the default.dbn file



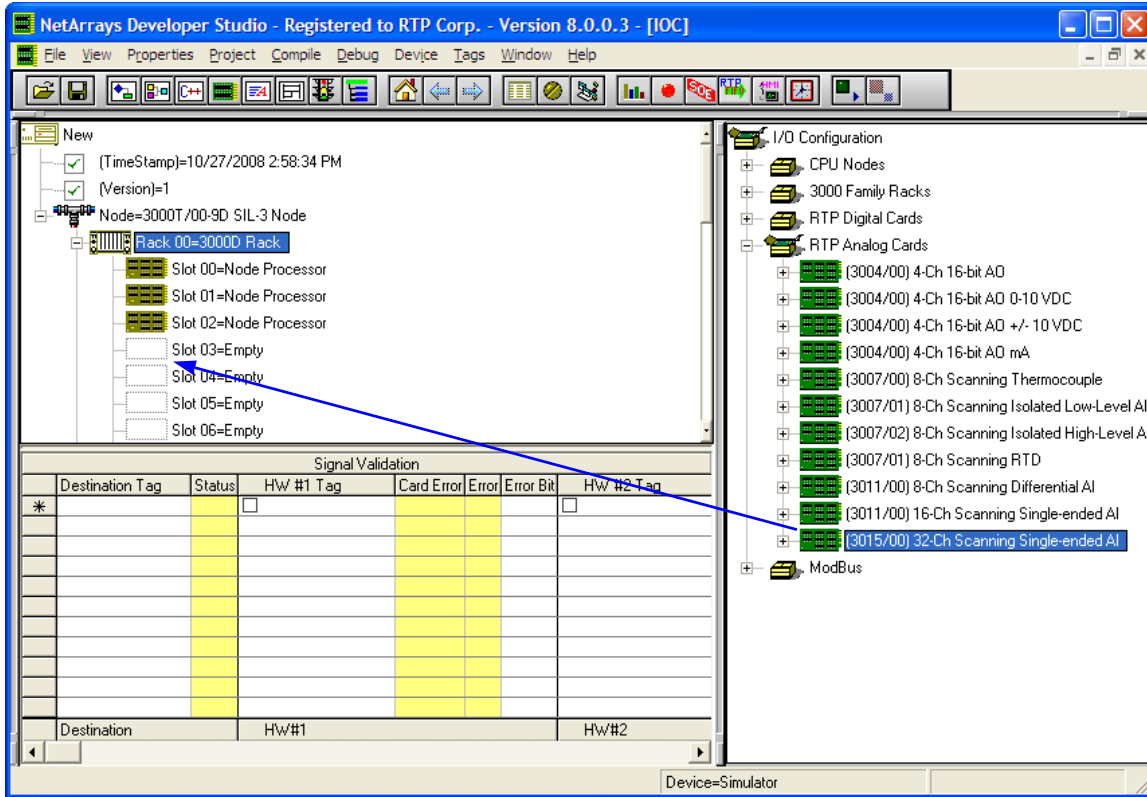
I/O Configuration

Add I/O Cards to the I/O Configuration Form

- Click on the I/O Configuration Studio button in the NetArrays main toolbar .

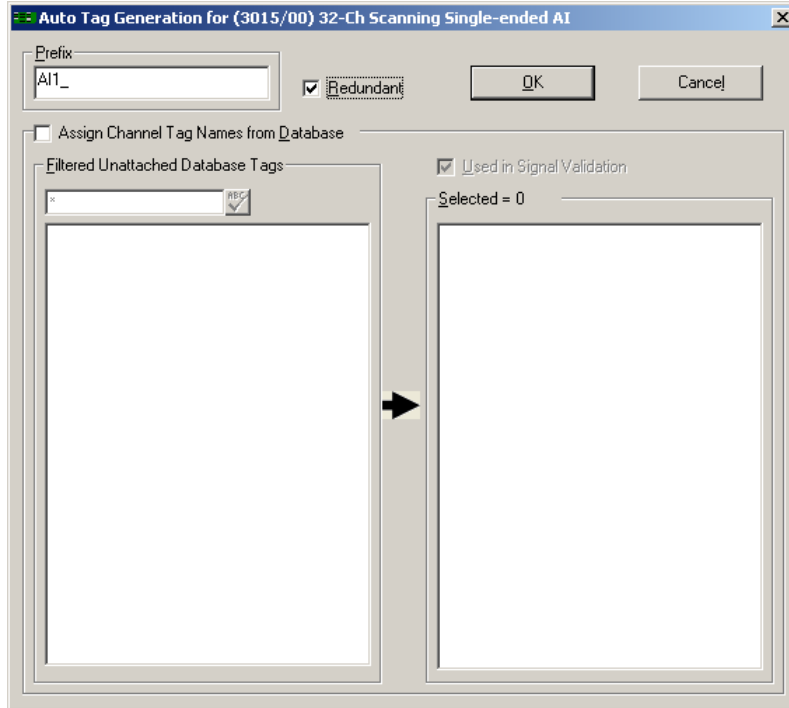
NetArrays Project Program Development Example

- In the I/O Configuration form, expand the 3000T Rack branch by clicking on the \oplus . Open the box of RTP Analog Cards in the I/O Configuration Toolbox by clicking on the \oplus .
- Drag and drop an 3015/00 32- Ch Scanning Single-ended AI icon from the I/O Configuration Toolbox into slot 3 of the TMR chassis.



- In the Auto Tag Generation window, enter the **Prefix** of AI1_. Select **Redundant**. Then click the **OK** button.

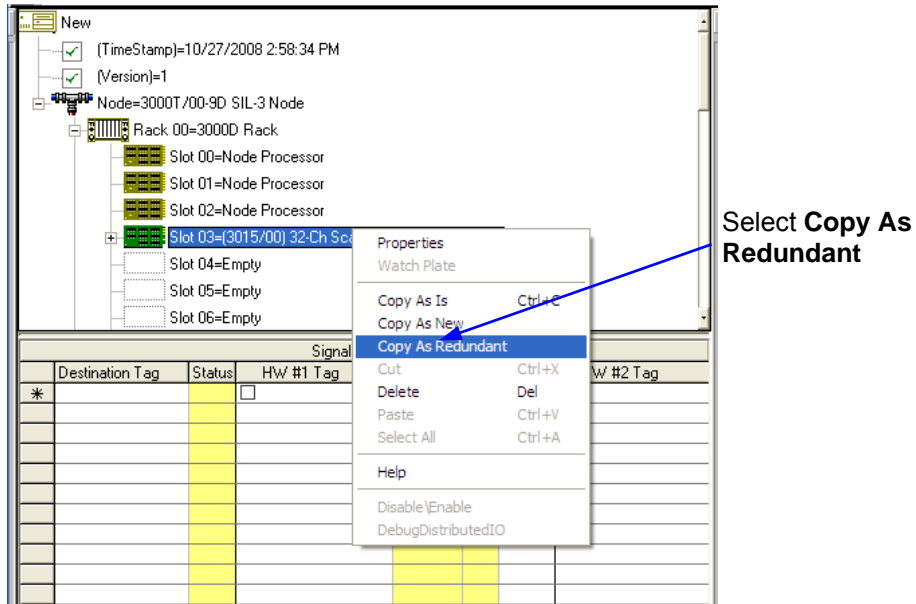
NetArrays Project Program Development Example



Note- Redundant Inputs are voted using the signal validation table. Please refer to signal validation tutorial for setup.

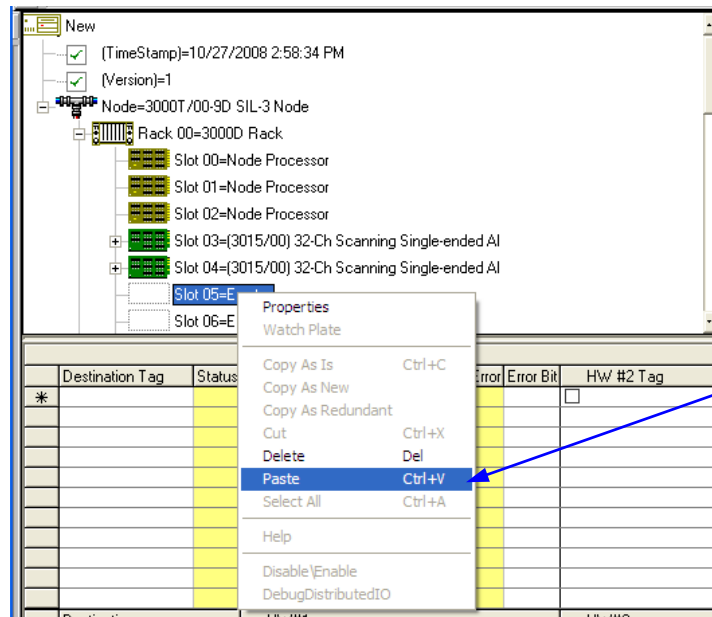
Add Two Redundant 32 Channel AI Cards

- Position the cursor over the AI card in the I/O Configuration Form and click the right mouse button.
- Select **Copy As Redundant** from the menu.

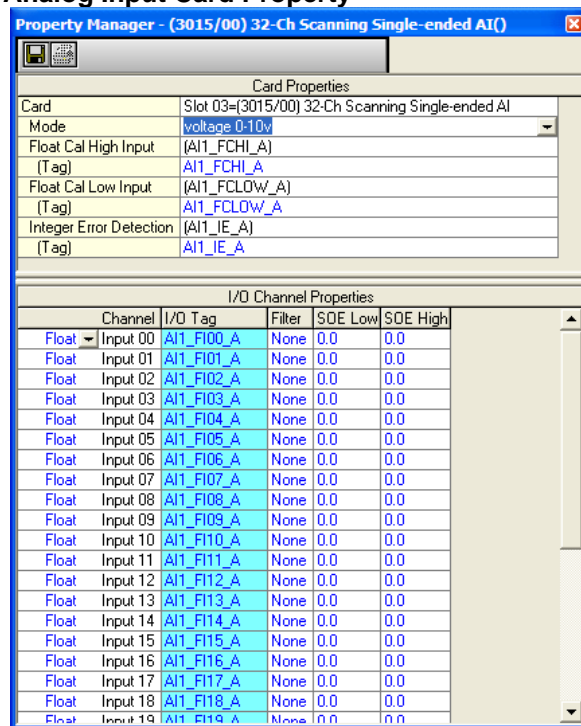



- Move the cursor to click on the **Slot 04=Empty** position. Right click and select **Paste**. Select the **Slot 05=Empty** position then right click and paste again to complete the triple redundant set of I/O cards.

NetArrays Project Program Development Example



First Analog Input Card Property



- Right-click on the Analog Input card's icon in Slot 03 to display the menu and select **Properties** to open the card's Property Manager window.
- Observe that the Postfix A has been added for each input parameter's tagname. The I/O card properties including diagnostics are listed at the top and the I/O channel information is listed below.
- Close the Property Manager window by clicking on .

NetArrays Project Program Development Example

- Redundant Analog Input cards in slots 04 and 05 have the Postfix B and C respectively.
- Drag and drop another 3015/00 Scanning Single-ended AI icon from the I/O Configuration Toolbox into slot 6 of the TMR chassis.
- In the Auto Tag Generation window, enter the Prefix AI2_ and select **Redundant**. Click **OK**.
- Select the analog input card, right mouse click and copy as redundant.
- Paste the redundant analog input card into slots 7 and 8.
- Open the box of RTP Digital Cards in the I/O Configuration Toolbox by clicking on the **+**.
- Drag and drop the 8 ch supervised DO 3005/08 card in slot 9
- In the Auto Tag Generation window, enter the Prefix SDO1_ and select **Redundant**. Click **OK**.
- Copy the card as redundant and paste into slot 10.
- Drag and drop another 8 ch supervised DO 3005/08 card into slot 11.
- In the Auto Tag Generation window, enter the Prefix SDO2_ and select **Redundant**. Click **OK**.
- Copy the card as redundant and paste into slot 12.

First 3005/08 Supervised Digital Output card property

The screenshot shows a 'Property Manager' window for a '(3005/08) 8-Ch Supervised OIDO OUT()' card. It is divided into two main sections: 'Card Properties' and 'I/O Channel Properties'.




Card Properties:

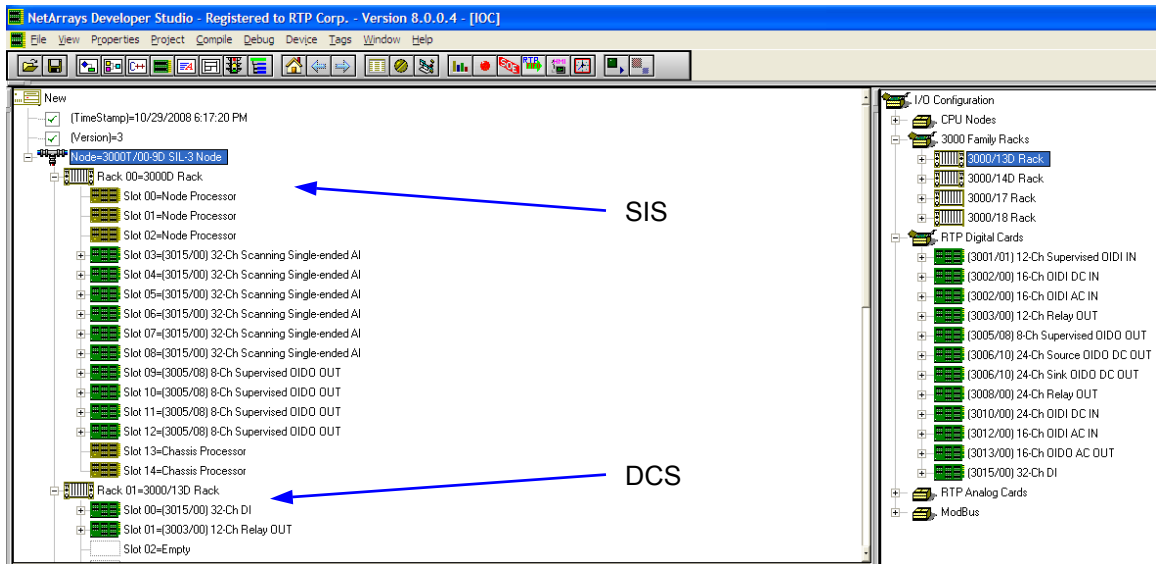
Card	Slot 09=(3005/08) 8-Ch Supervised
Redundant Card ID	1
Watchdog Timer	Enabled
Integer Error Detection (Tag)	(SD01_IE_A) SDO1_IE_A
Integer Readback Status (Tag)	(SD01_IRB_A) SDO1_IRB_A
Integer Switch Fault Status (Tag)	(SD01_ISF_A) SDO1_ISF_A
Integer Safety Switch Fault Status (Tag)	(SD01_ISSF_A) SDO1_ISSF_A
Integer Load Fault Status (Tag)	(SD01_ILF_A) SDO1_ILF_A

I/O Channel Properties:

Channel	I/O Tag
Bool Output 00	SD01_B000_A
Bool Output 01	SD01_B001_A
Bool Output 02	SD01_B002_A
Bool Output 03	SD01_B003_A
Bool Output 04	SD01_B004_A
Bool Output 05	SD01_B005_A
Bool Output 06	SD01_B006_A
Bool Output 07	SD01_B007_A

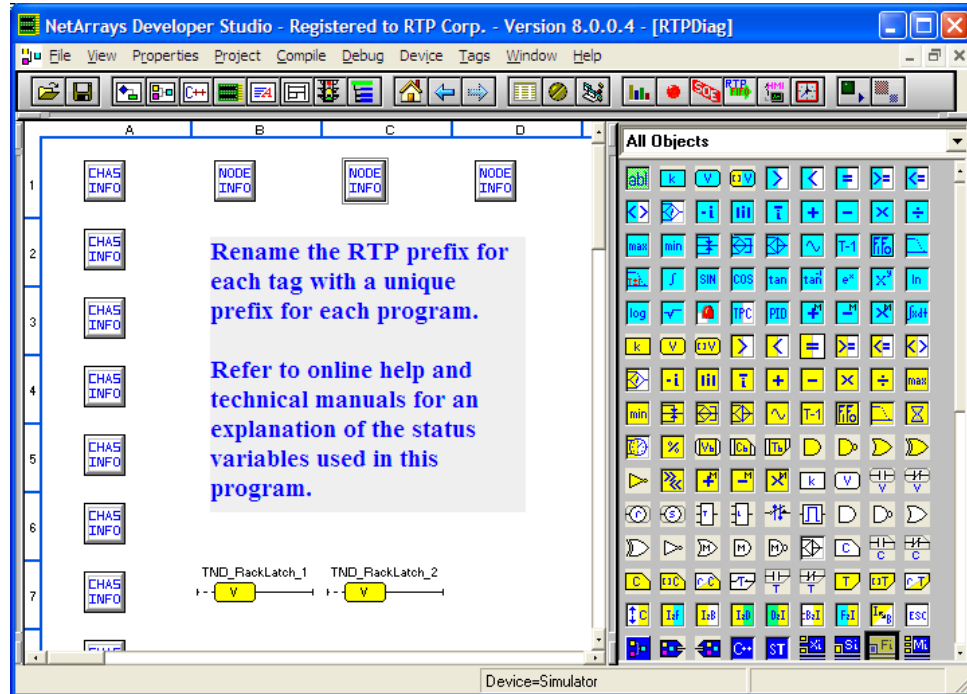
NetArrays Project Program Development Example

- Open the 3000 Family Racks box in the I/O Configuration Toolbox by clicking on the .
- Drag and drop the 3000/13D chassis into the Rack 1 location.
- Open Rack 1 by clicking on the .
- Open the box of RTP Digital Cards in the I/O Configuration Toolbox by clicking on the .
- Drag and drop the 3015/00 32-Ch DI card into slot 00.
- In the Auto Tag Generation window, enter the Prefix DI1_. Click **OK**.
- Drag and drop the 3003/00 12-Ch Relay Out card into slot 01.
- In the Auto Tag Generation window, enter the Prefix RO1_. Click **OK**.



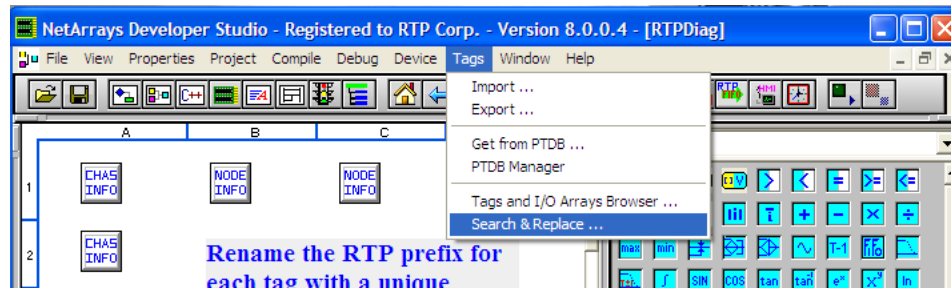
RTP Diagnostics

- Display the RTP_Diag Form by clicking on the  button in the Main Toolbar, then double clicking on the **RTP_Diag**.



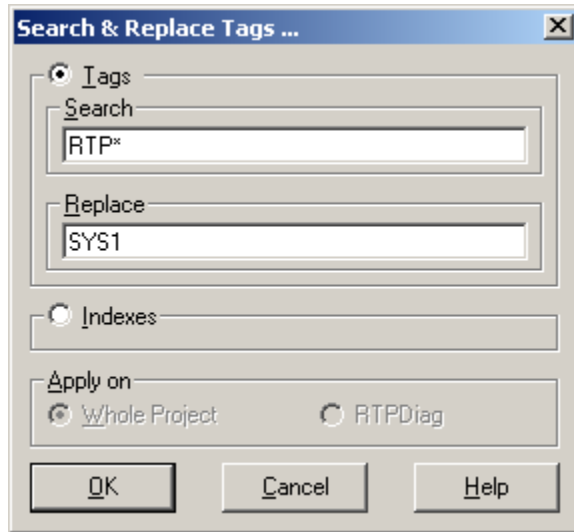
Diagnostics variables are pre-configured for each of the redundant nodes and each chassis. Simply select and delete any Node Info objects or Chassis Info objects that are not needed. This can be accomplished by using the mouse. Left click and draw a box around the un-used Node Info or Chassis Info boxes and then press the delete key.

- Double click the chassis info box to observe chassis diagnostics.
- Double click the node info box to observe node processor diagnostics.
- Delete objects from cell A3 to A16 as they are un-used.
- Using the tags search and replace feature customizes diagnostics variables. On the menu bar, click on Tags, search and replace.

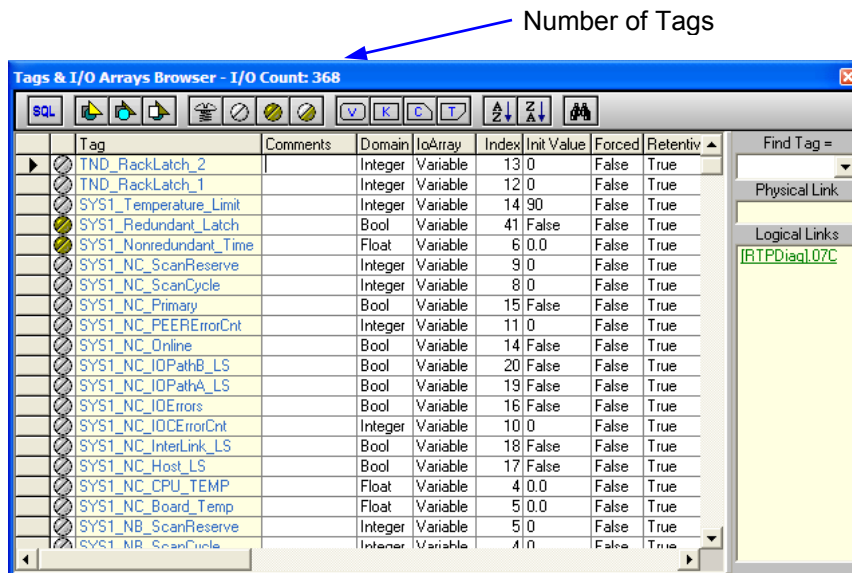


- In Search & Replace Tags... window, enter RTP* in the Search field and SYS1 in the Replace field.

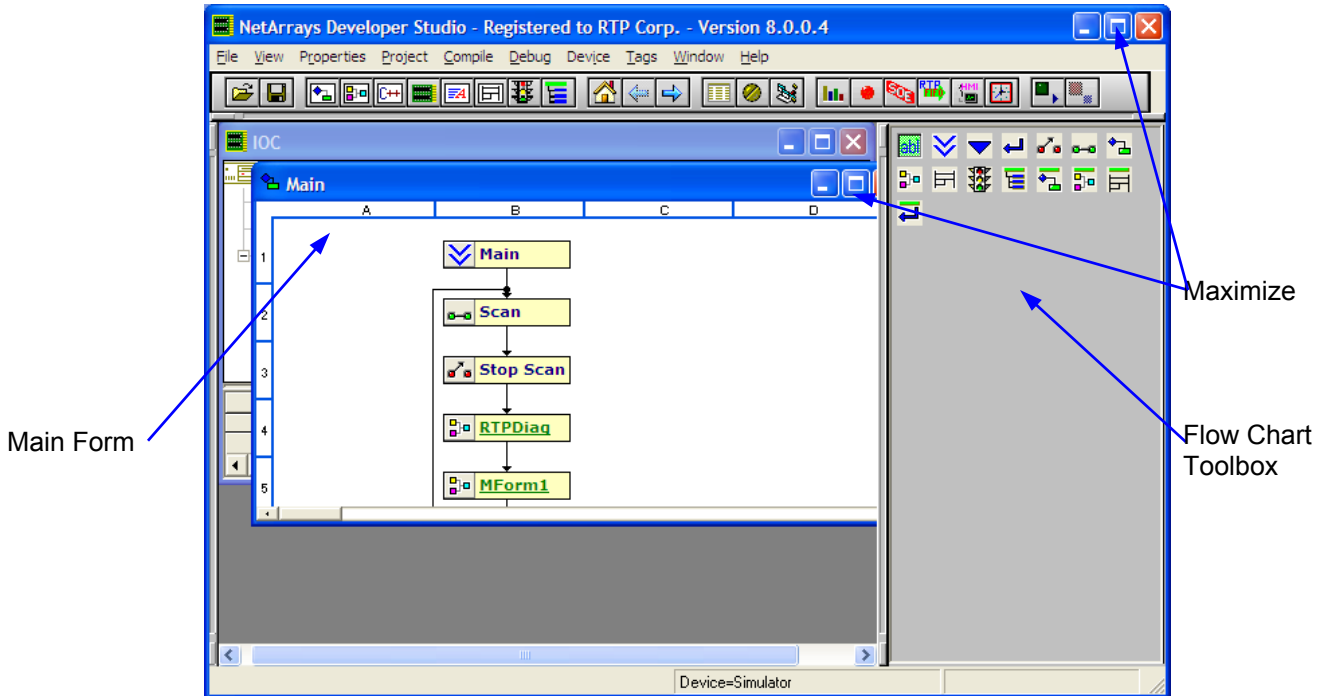
NetArrays Project Program Development Example



- Observe 62 diagnostic variables have been renamed.
- Select the Tags menu again and click on Tags and I/O Arrays Browser.
- Select the tag column and sort Z-A.
- Observe you have the SYS1 diagnostic variables and 368 points generated.



NetArrays Main Form



- Click on the Maximize buttons in the NetArrays Developer Studio and the banner of the Main Form. If the Maximize button is not visible on the Main Form, double click on the Main banner. The Main Form is a 16x16 grid. Columns A – P and Rows of 1 – 16.

Next we will demonstrate how safety and control is easily developed in the same project. The SIF's will be developed in one sequential flow chart while the controls for the DCS will be done in another.

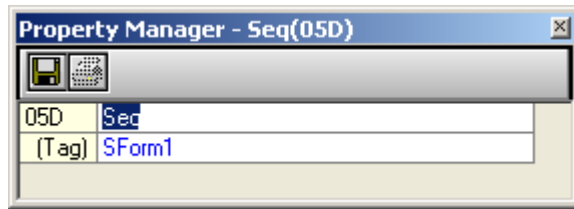
The module forms that are part of a SIF may only contain SIL I/O points. All the variables inside the SIF forms will be marked as "read only". Therefore there is protection provided against any HMI or operator having access to alter or interfere with the safety integrity function. The HMI is provided read only access.

Where necessary, the engineer programmatically may provide writable access for the HMI to selected variables, outside of the SIF's.

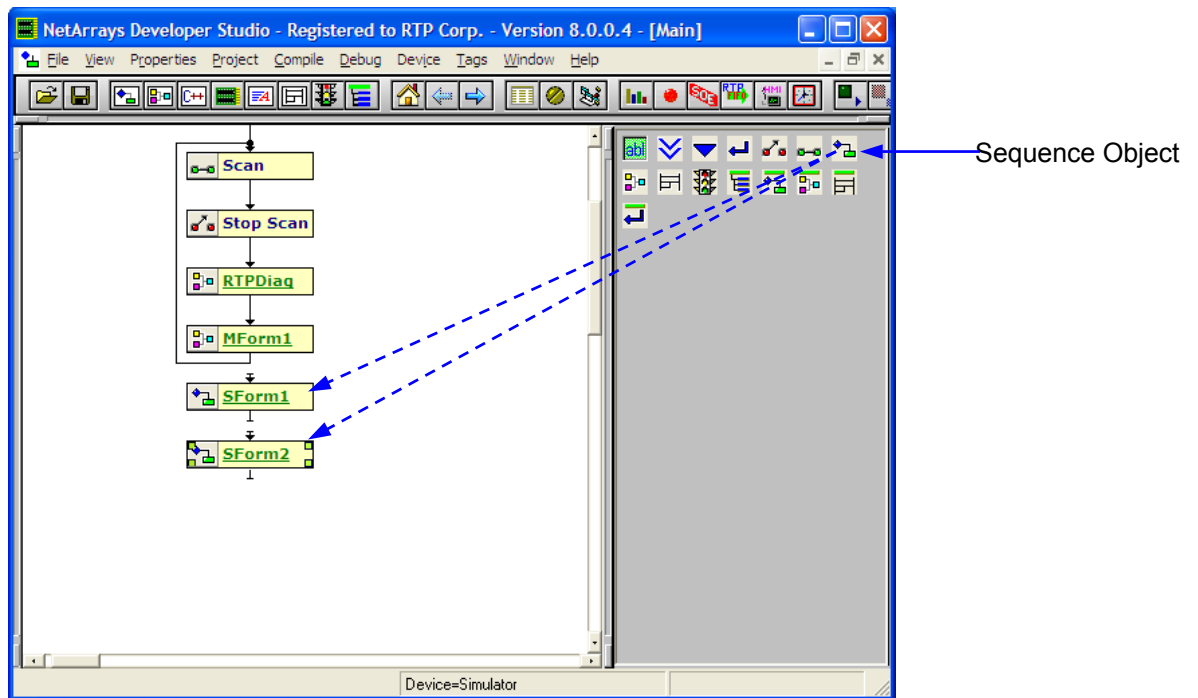
Control will be done in a separate sequence. All the variables defined in these modules will be accessible to the HMI to perform both read and write instructions.

Developing Safety Instrumented Functions

- Drag a Sequence Object icon from the Toolbox and onto cells B6 and B7. The auto tag generator window will appear. You can change the form name now or later. We will change the names later. This will be true for all forms added.

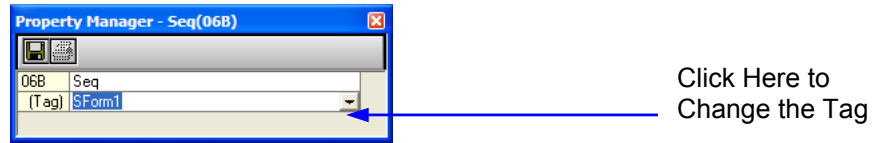
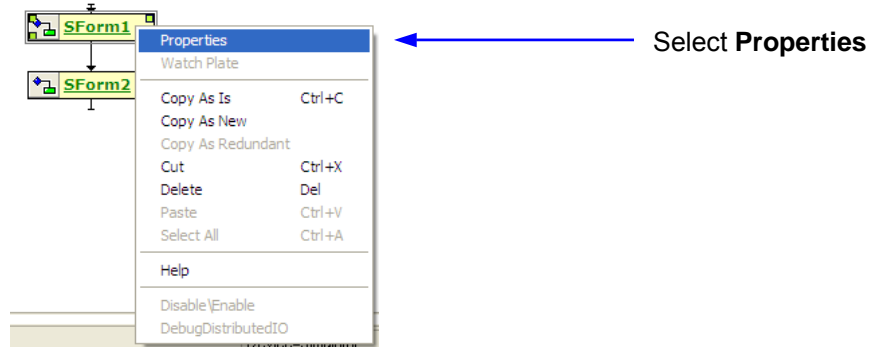


- Note that NetArrays automatically assigns them Tags **SForm1** and **SForm2**. After the two Sequence Forms are added, Main should appear as shown in the following figure.



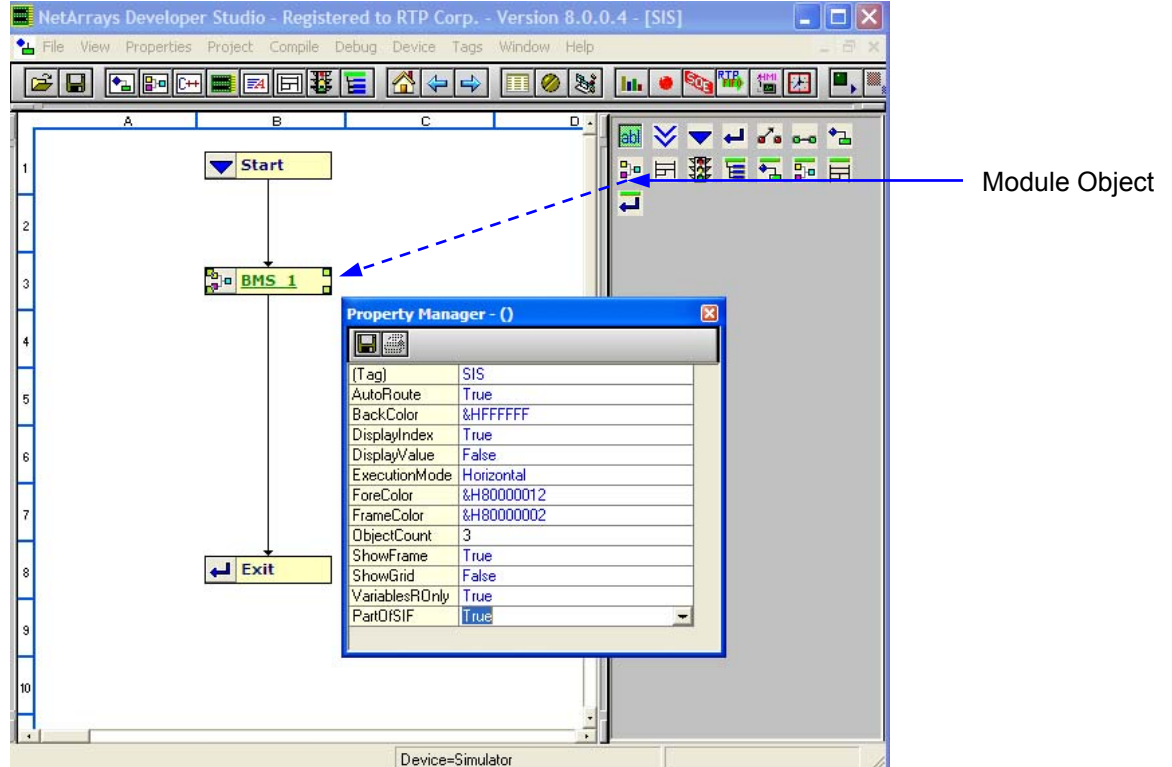
- Right mouse click on the SForm1. Click on “Properties”. Enter the tag name “**SIS**” and then Enter.
- Right mouse click on the SForm2. Click on “Properties”. Enter the tag name “**DCS**” and then Enter.

NetArrays Project Program Development Example



- Double click the SIS sequence form. Select yes to create the SIS flow chart.
- Right mouse click on the page and select Properties.
- In the Property Manager window, verify a True setting for the property “VariablesROnly” and “PartOfSIF”.


From here down, every module added to this sequence flow chart will be Part of SIF with all variables marked read only.

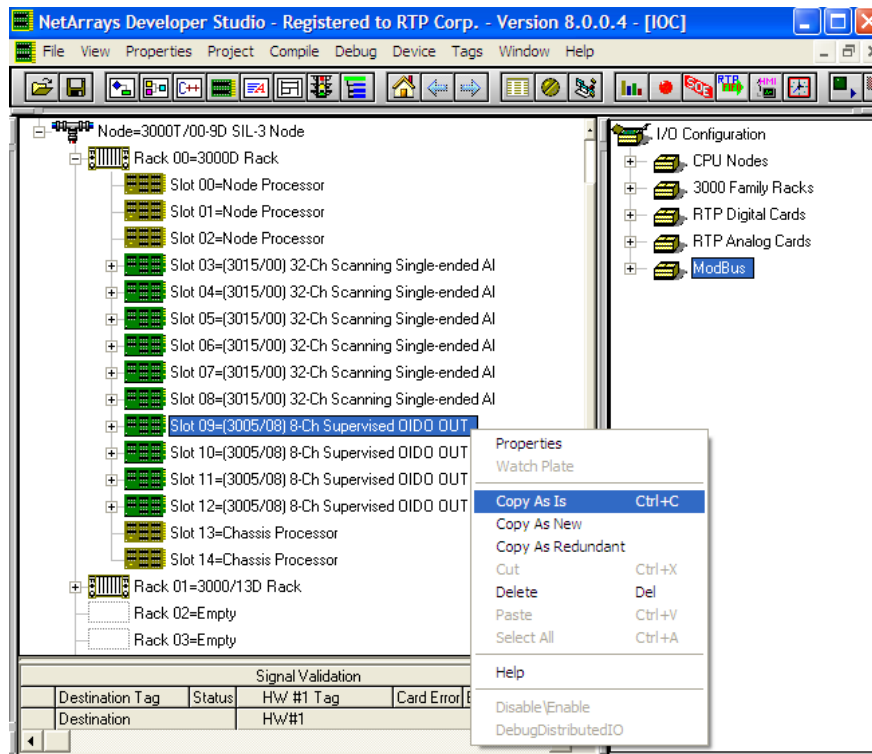


NetArrays Project Program Development Example

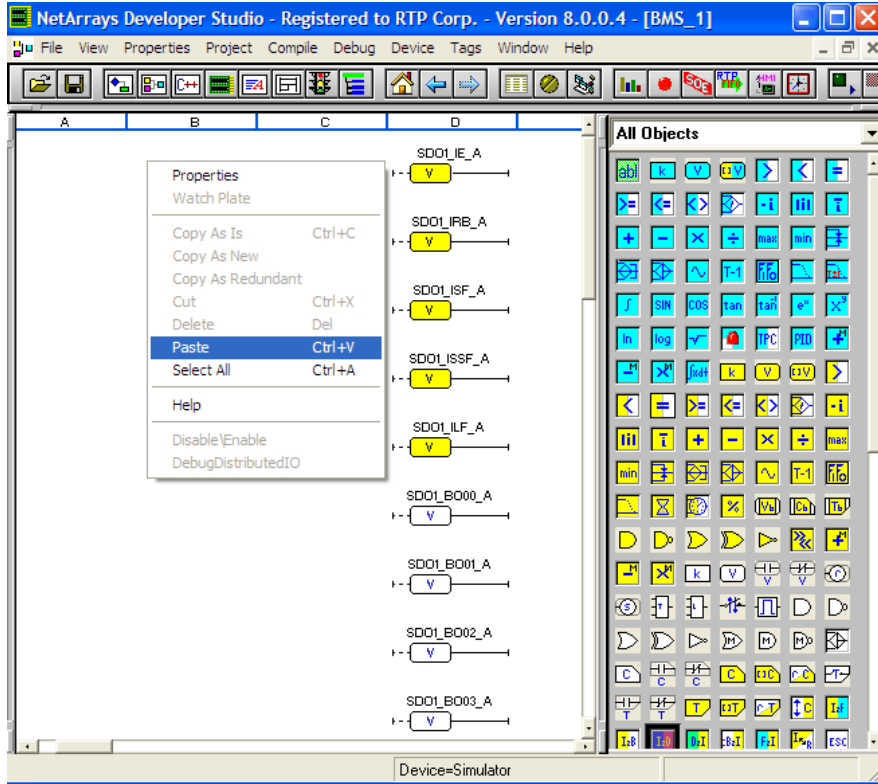
- Drag a Module Object icon from the Toolbox and onto cells B3, Note that NetArrays automatically assigns the Tag **MForm2**.
- Select Mform 2, right mouse click and enter the tag name **BMS_1**.
- Connect output of Start sequence to input of **BMS_1**. Left mouse click, hold the mouse button down and drag the cursor to the input of the **BMS_1**, then release.
- Repeat to connect **BMS_1** to the Exit sequence.
- Double click **BMS_1** and click yes to create the page. The page property here will indicate that it is Part of SIF.

Adding SIL I/O points to SIF's


- Click on the I/O Configuration Studio button in the NetArrays main toolbar .
- Open the first chassis defined for SIS and select the **3005/08 DO** card in slot 9.
- Right mouse click and select **Copy As Is**.
- Click on the menu item Window and select the BMS_1 form
- Right mouse click to **Paste** the SIL I/O points in to the BMS_1 SIF form.

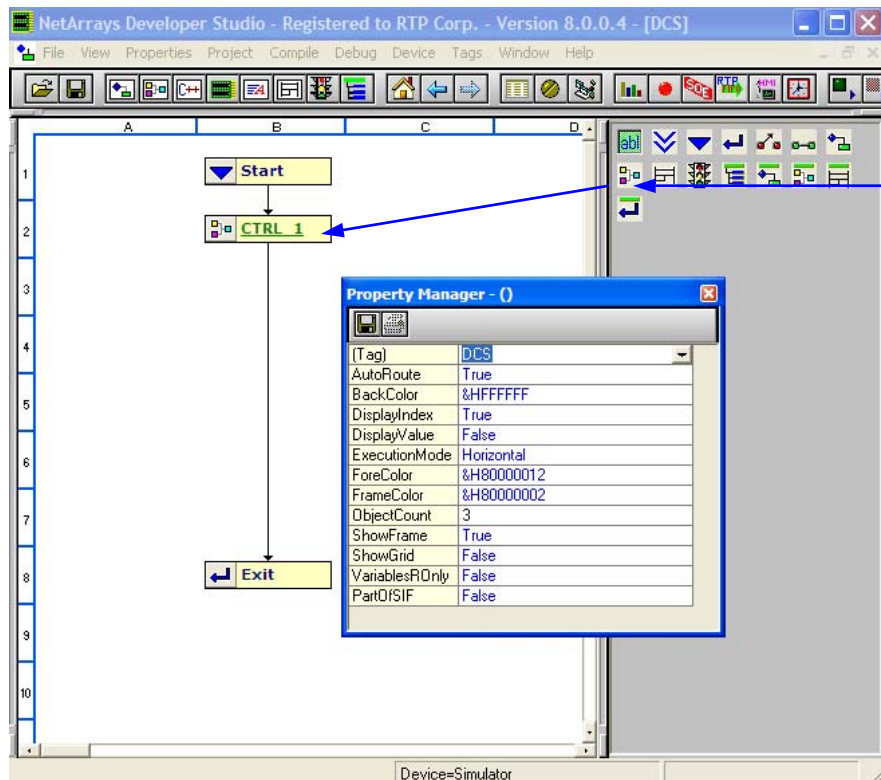


NetArrays Project Program Development Example



Developing DCS Modules

- Return to the Main Form by clicking on the  button in the Main Toolbar.
- Double click on DCS to create the control sequence flow chart.
- Right mouse click on the page and select Properties. PartOfSIF and VariablesROOnly should be set to False as indicated in diagram below.
- Drag a Module Object icon from the Toolbox and onto cell B2, Note that NetArrays automatically assigns the Tag **MForm2**.
- Select Mform 2, right mouse click and enter the tag name **CTRL_1**.
- Connect output of Start sequence to input of **CTRL_1**. Left mouse click, hold the mouse button down and drag the cursor to the input of the **CTRL_1**, then release.
- Repeat to connect **CTRL_1** to the Exit sequence.
- Double click **CTRL_1** and click yes to create the page. The page property will indicate that it is not Part of SIF.




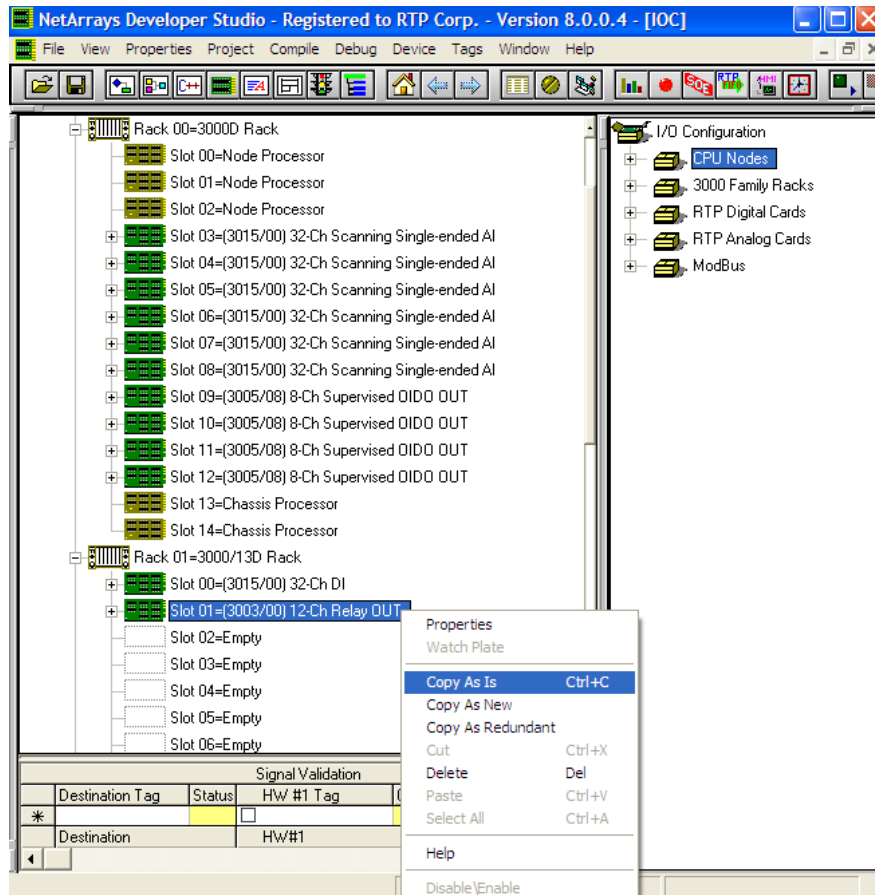
The screenshot shows the NetArrays Developer Studio interface. The main workspace displays a control sequence flow chart with three steps: 'Start', 'CTRL_1', and 'Exit'. The 'Start' step is connected to the 'CTRL_1' step, which is in turn connected to the 'Exit' step. A blue arrow points from the 'Module Object' label to the 'CTRL_1' step. A 'Property Manager' dialog box is open over the 'CTRL_1' step, showing the following properties:

Property	Value
(Tag)	DCS
AutoRoute	True
BackColor	&HFFFFFF
DisplayIndex	True
DisplayValue	False
ExecutionMode	Horizontal
ForeColor	&H80000012
FrameColor	&H80000002
ObjectCount	3
ShowFrame	True
ShowGrid	False
VariablesROOnly	False
PartOfSIF	False

The status bar at the bottom indicates 'Device=Simulator'.

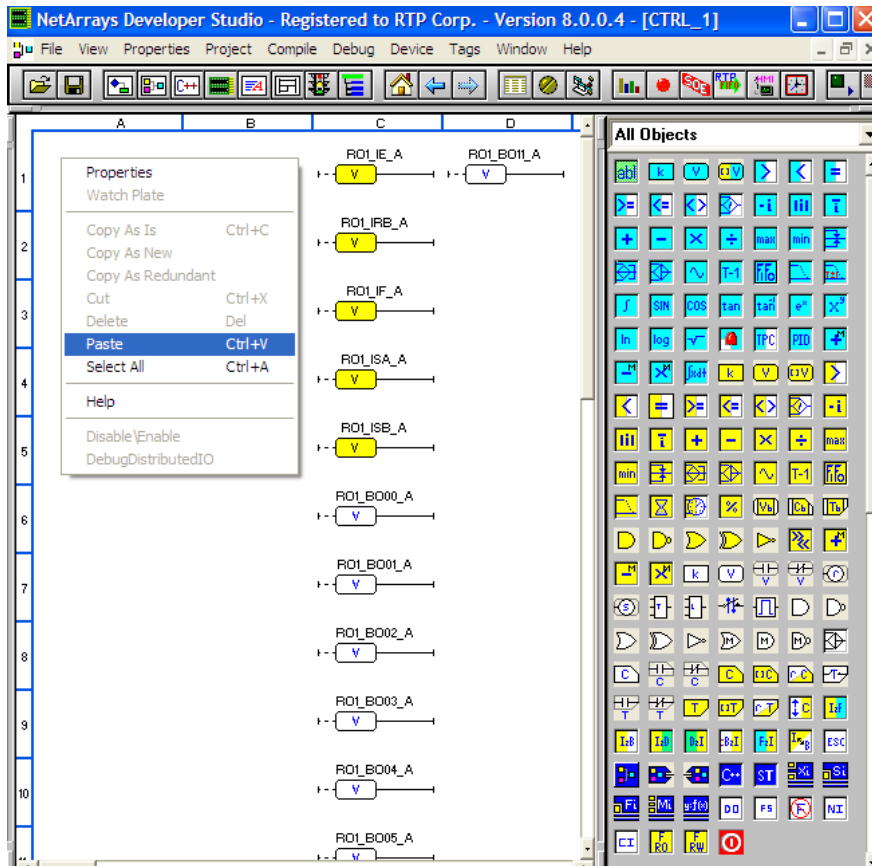
Adding Non-Interfering I/O points to DCS Control Module

- Click on the I/O Configuration Studio button in the NetArrays main toolbar .
- Open the second chassis defined for DCS and select the 3003/00 RO card in slot 1.




- Right mouse click and select **Copy As Is**.
- Click on the menu item Window and select the **CTRL_1** form.
- Right mouse click to **Paste** the DCS I/O points in to the **CTRL_1** form.

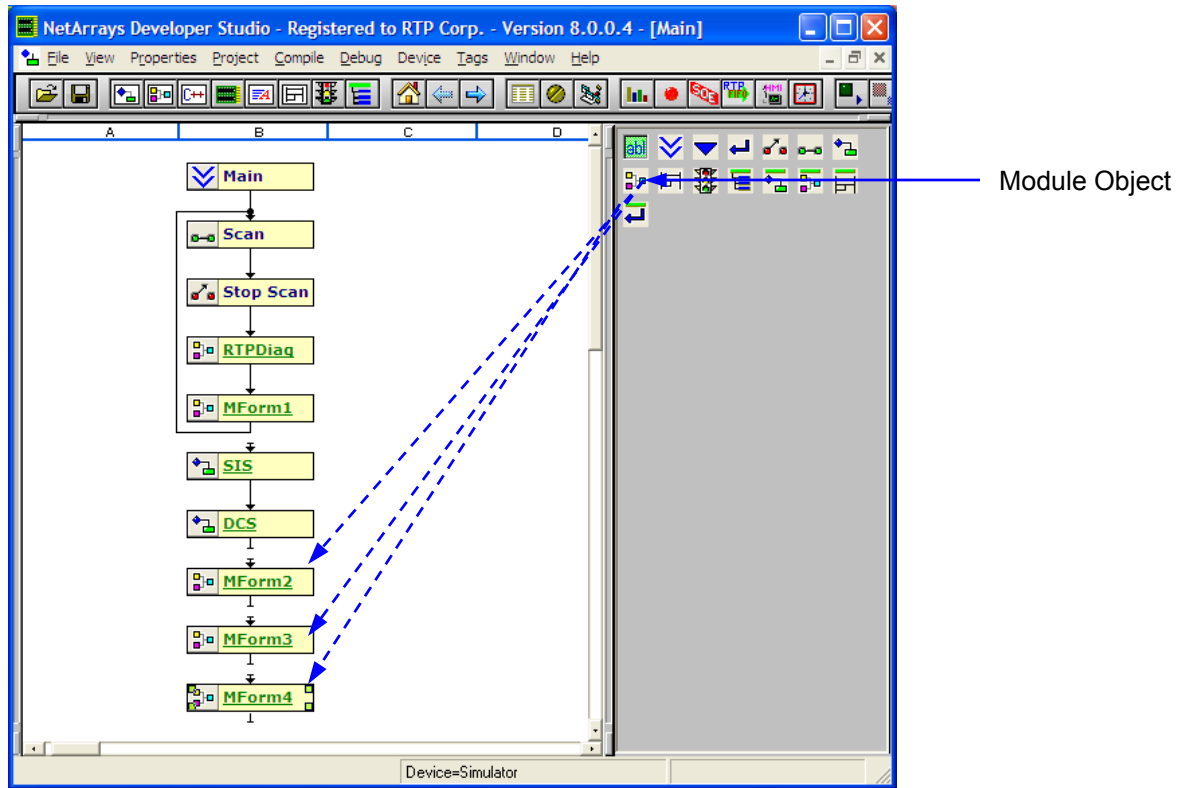
NetArrays Project Program Development Example




Objects in the right pane are SIL certified for use in SIF's and to do control.

NetArrays Logic Development

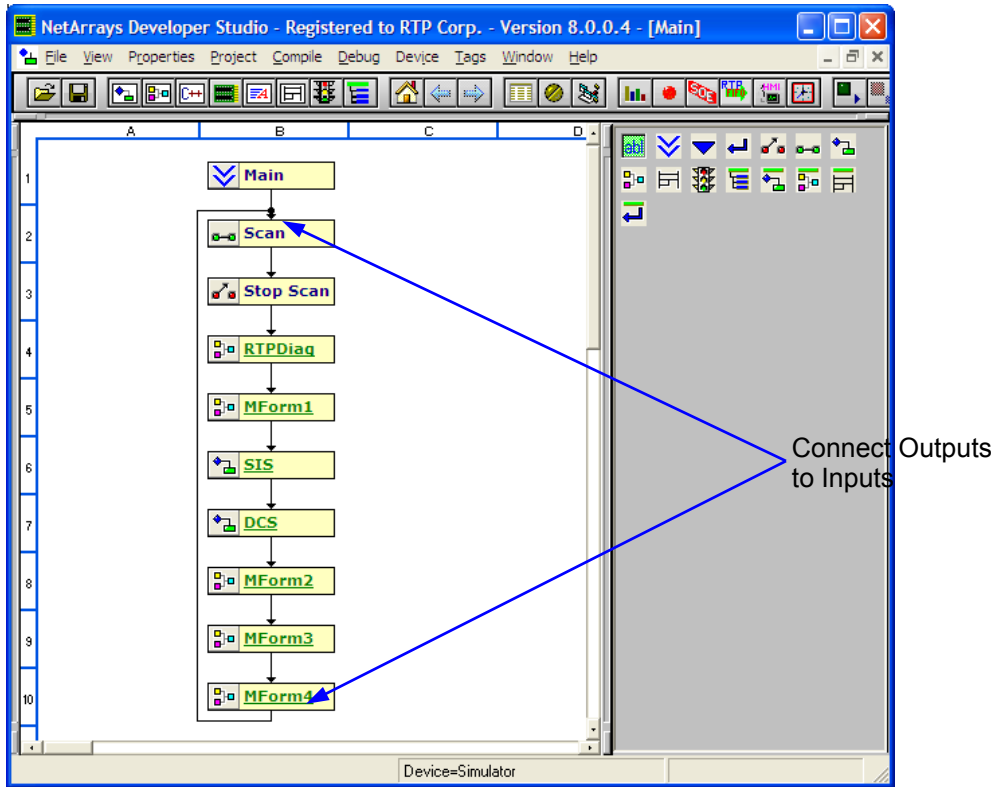
- Return to the Main Form by clicking on the  button in the Main Toolbar.
- Drag a Module Object icon from the Toolbox and onto cells B8, B9, and B10. Note that NetArrays automatically assigns them Tags **MForm2**, **MForm3**, and **MForm4**. After the three Module Forms are added, Main should appear as shown in the following figure.



The Main Form must be constructed as one continuous program loop. To include the new Module Form object into the loop, you must rearrange the connections between the objects.

- Move the cursor to the output of **MForm1**. You will know that the cursor is in the right spot when the small square  appears on the output connector. Then click on the small square to disconnect the connection between the **MForm1** object and the **Scan** object.
- Move the cursor to the output of **MForm1** again. Left mouse click, hold the mouse button down and drag the cursor to the input of the **SIS** object. Release the mouse button to complete the connection. Repeat the operation until all Forms are interconnected. The Main Form should now look like the following figure.

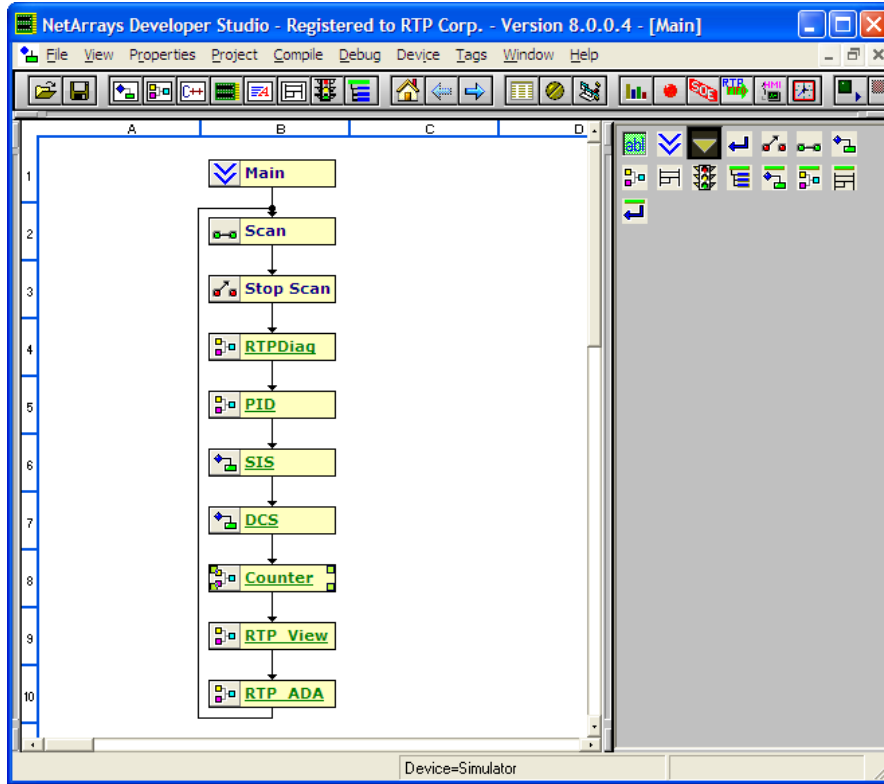
NetArrays Project Program Development Example



Note: The **Scan** and **Stop Scan** objects are required in the Main Form, and they must remain within the program's main loop as indicated.

- Right-click on the **MForm1** object to display the menu and select **Properties** to open the object's Property Manager window.
- Click on the **(Tag)** field and type in the name **PID** followed by a Return. The Tag **PID** will appear on the object in the Main Form and in the Project Explorer. Close the Property Manager window.
- Make similar changes to all Module Forms as indicated following figure.

NetArrays Project Program Development Example



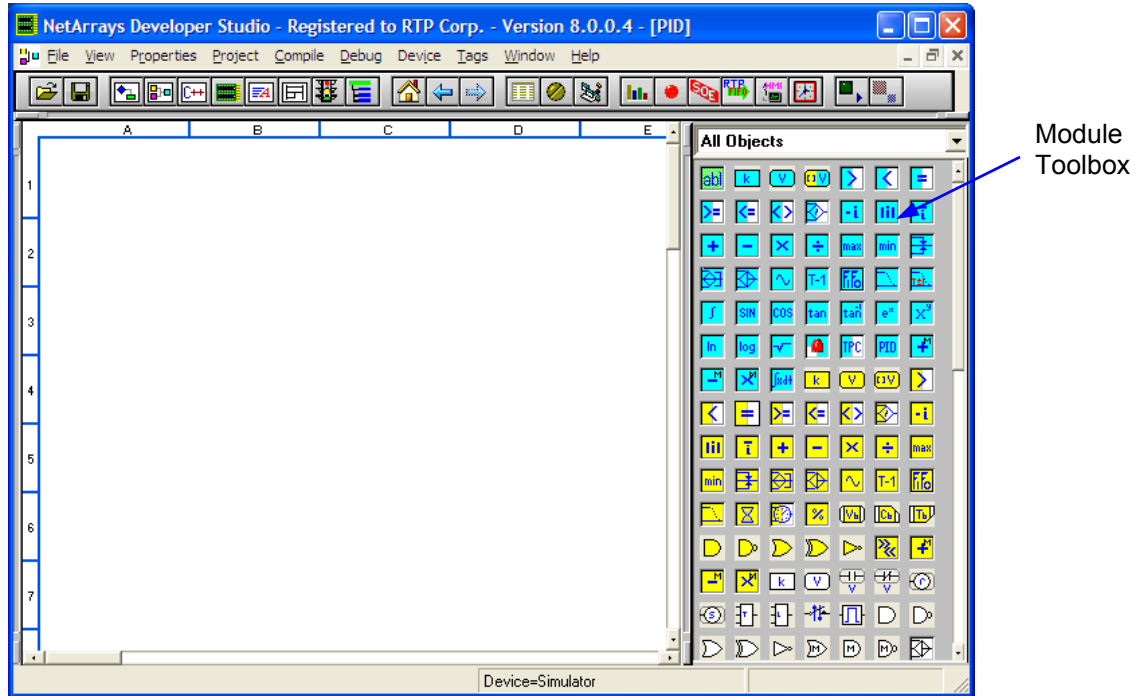
The next step is to define the logic within the Module Forms. Module Forms open as a 16x16 grid. Columns A – P and Rows 1 – 16.

Module Form “PID”

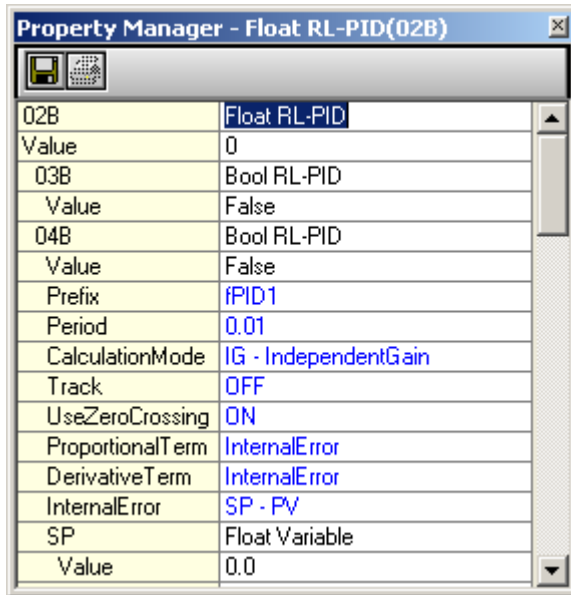
Add Objects to the Module Form to Simulate a PID Control Loop

Objects are taken from the Toolbox and placed onto the Module Form and connected to perform logic functions. This Module Form will contain a PID object, a few Variable objects, and a Filter object to create a control loop.

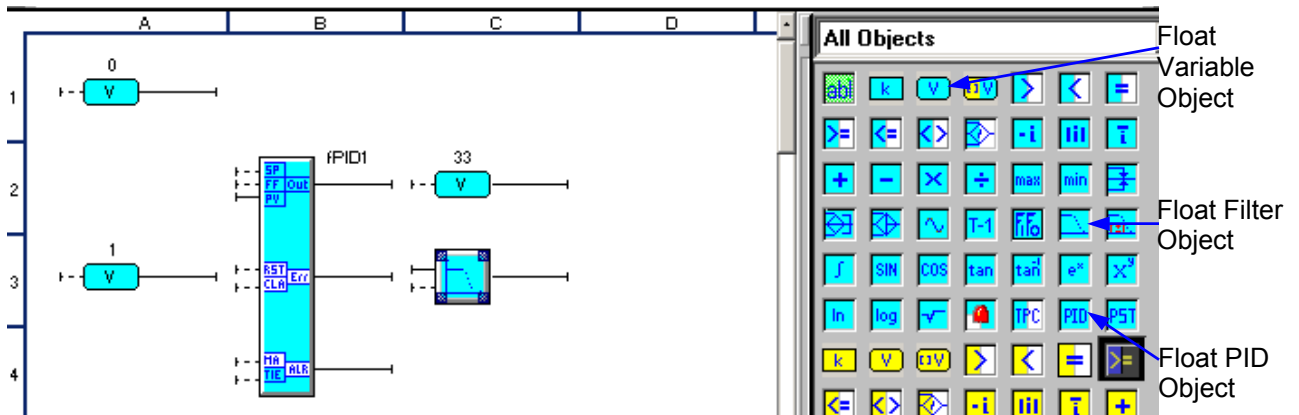
- Double-click on the **PID** form in the Main Form to display the empty Module Form. Note that the Toolbox changes to a collection of Module Objects.
- The Module Objects are color-coded to designate float, integer or bool. Float objects are blue. Integer objects are yellow. Bool objects are white.



- Drag a Float PID object icon from the Toolbox and drop it onto cell B2. Note that NetArrays assigns it the Tag **fPID1**. Please note online help is available on each object by pressing the F1 key when that object is selected. When the PID object is added, the **Property Manager** automatically pops up. You can change object properties when you first add them or later. This will be true for all objects added. We will change the tag names later.

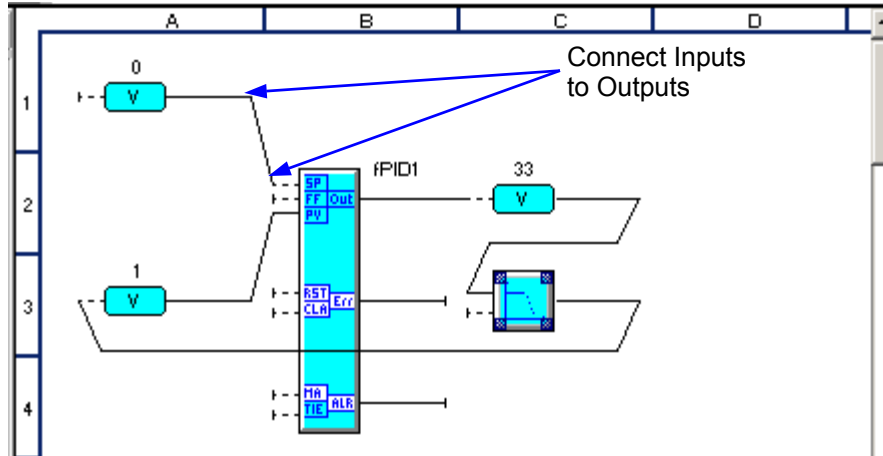


- Drag three Float Variable objects from the Toolbox, place one in cell A1, one in cell A3, and one in cell C2. Index numbers are automatically assigned based on lowest available address. Note index numbers assigned may differ in your project.
- Drag a Float Filter object icon from the Toolbox and drop it onto cell C3.

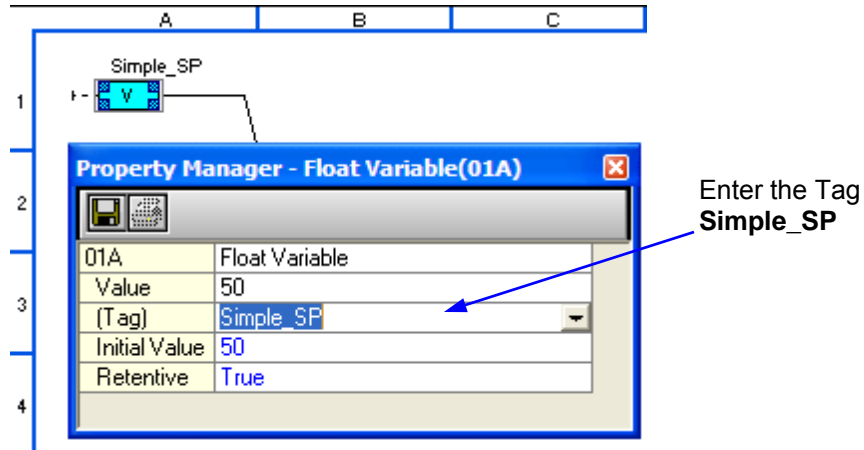


- Move the cursor to the **SP** input of the PID. You will know that the cursor is in the right spot when the small square \square appears on the input connector. Then hold the mouse button down and move the cursor to the output of the Float Variable in cell A1. Release the mouse button when the small square \square appears at the output of the Float Variable object to complete the connection.
- In the same fashion, connect the **PV** input of the PID to the output of the Float Variable in cell A3. Connect the input of the Float Variable in cell C2 to the **Out** output of the PID. Connect the input of the Float Filter to the output of the Float Variable in cell C2. Connect the input of the Float Variable in cell A3 to the output of the Float Filter.

NetArrays Project Program Development Example



- Right-click on the Variable object in cell A1 to display the menu and select **Properties** to open the object's Property Manager window. (Note that you can also access the Property Manager by double-clicking on any object in a Module Form.)
- Click on the **(Tag)** field and type in the name **Simple_SP** followed by a Return. Close the Property Manager window. The tag name Simple_SP will appear over the Float Variable in the Module Form.



- Double-click on the Variable object in cell A3. Click on the **(Tag)** field in the Property Manager and type in the name **Simple_PV** followed by a Return. Close the Property Manager window.
- Double-click on the Variable object in cell C2. Click on the **(Tag)** field in the Property Manager and type in the name **Output** followed by a Return. Close the Property Manager window.
- Double-click on the **PID** object to configure the proportional and integral constants, and the setpoint and output range parameters. Enter the values into the Property Manager indicated in the figure below. Remember to follow each entry with a Return. Close the Property Manager window when done.

NetArrays Project Program Development Example

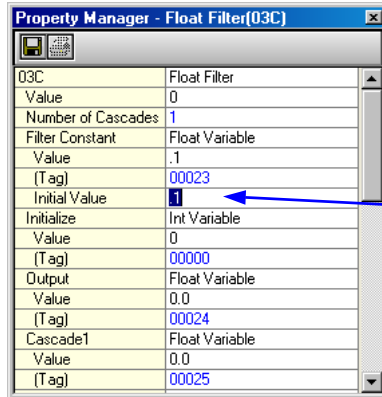
Parameter	Data Type	Initial Value
Initial Value		0.0
Kp:Kc	Float Variable	1
(Tag)		fPID1.[01]Kp:Kc
Initial Value		1
Ki:Ti	Float Variable	.1
(Tag)		fPID1.[02]Ki:Ti
Initial Value		.1
Kd:Td	Float Variable	0.0
(Tag)		fPID1.[03]Kd:Td
Initial Value		0.0
DB	Float Variable	0.0
(Tag)		fPID1.[04]DB
Initial Value		0.0
DrFlt	Float Variable	0.0
(Tag)		fPID1.[05]DrFlt
Initial Value		0.0
SPLo	Float Variable	0
(Tag)		fPID1.[06]SPLo
Initial Value		0
SPHi	Float Variable	100
(Tag)		fPID1.[07]SPHi
Initial Value		100
OutLo	Float Variable	0
(Tag)		fPID1.[08]OutLo
Initial Value		0
OutHi	Float Variable	100
(Tag)		fPID1.[09]OutHi
Initial Value		100
ROCOuTA	Float Variable	

Annotations:

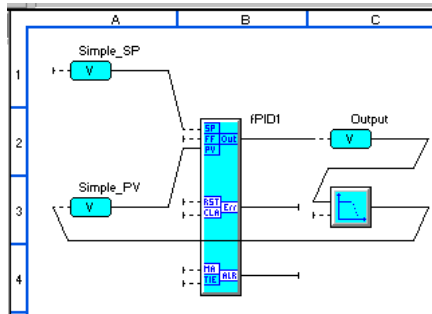
- Enter **1** for the Kp:Kc Initial Value
- Enter **0.1** for the Ki:Ti Initial Value
- Enter **0** for the SPLo Initial Value
- Enter **100** for the SPHi Initial Value
- Enter **0** for the OutLo Initial Value
- Enter **100** for the OutHi Initial Value

NetArrays Project Program Development Example

- Double-click on the Float Filter object. Click on the Filter Constant **Initial Value** field in the Property Manager and enter **0.1** followed by a Return. Close the Property Manager window.



- The Module Form PID should now look like the example shown below.




This completes the PID Module Form. The next step is to define the logic within the Counter Module Form.

Module Form “Counter”

Add Objects to a New Module Form to Demonstrate Counters and Timers

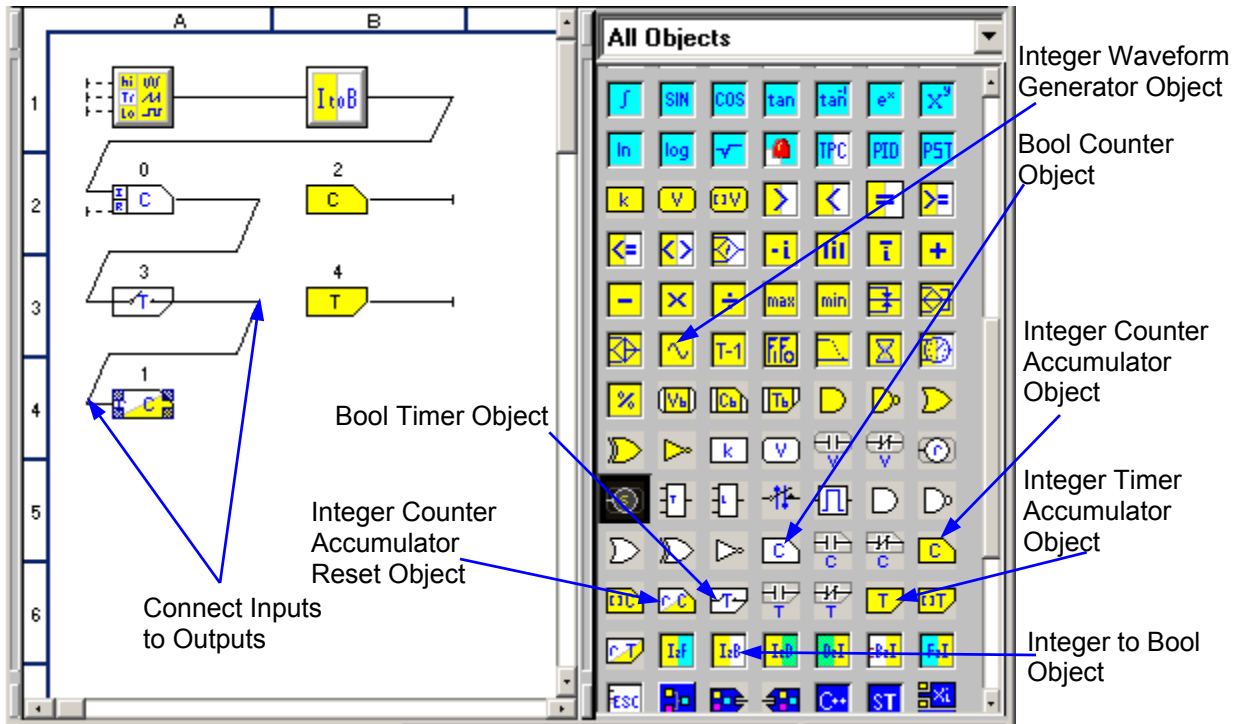
You will now create a new Module Form and place objects onto it from the Toolbox. This Module Form illustrates how NetArrays counters and timers function.

- Return to the Main Form by clicking on the  button in the Main Toolbar.
- Double-click on the **Counter** object in the Main Form. The following prompt will appear on your screen.

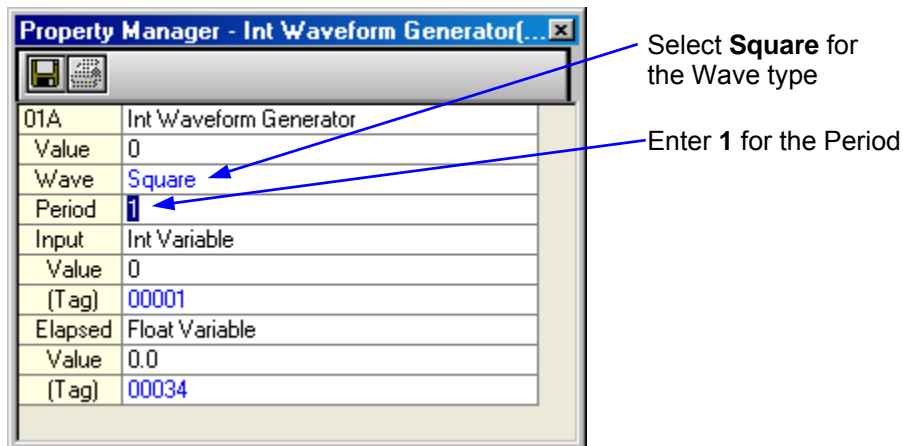


- Click on the **Yes** button to create a new Module Form with the Tag **Counter**.
- Drag an Integer Waveform Generator object icon from the Toolbox and drop it onto cell A1. This object will be the source of the pulses we will be counting.
- Drag a Bool Integer to Bool object icon from the Toolbox and drop it onto cell B1. This object converts integer values into Boolean values.
- Drag a Bool Counter object icon from the Toolbox and drop it onto cell A2. This object counts a preset number of input pulses.
- Drag an Integer Counter Accumulator object icon from the Toolbox and drop it onto cell B2. This object stores the number of pulses counted.
- Drag a Bool Timer object icon from the Toolbox and drop it onto cell A3. This object times a preset time period.
- Drag an Integer Timer Accumulator object icon from the Toolbox and drop it onto cell B3. This object stores the time period counted.

NetArrays Project Program Development Example

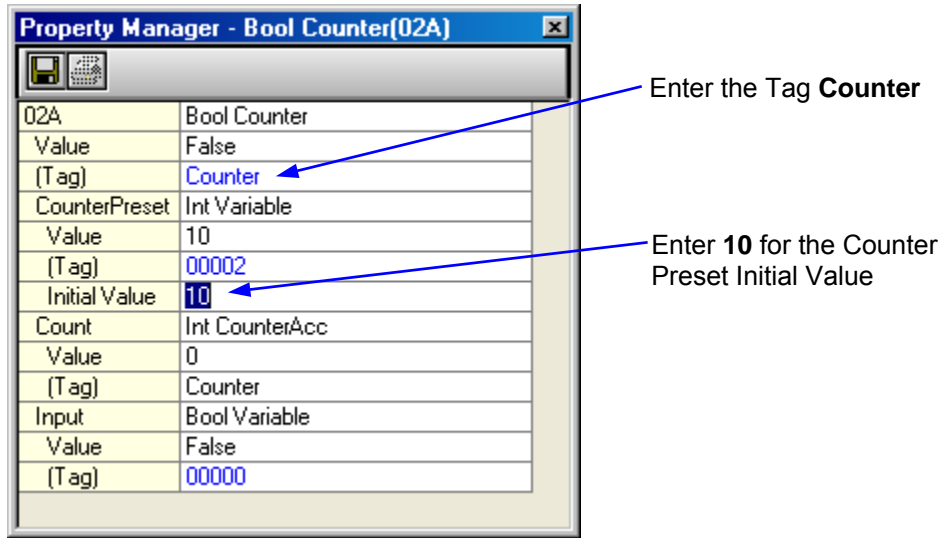


- Drag an Integer Counter Accumulator Reset object icon from the Toolbox and drop it onto cell A4. This object resets the Counter.
- Connect the input of the Integer Counter Accumulator Reset object to the output of the Bool Timer object. Connect the input of the Bool Timer object to the output of the Bool Counter object. Connect the input of the Bool Counter object to the output of the Integer To Bool object. Connect the input of the Integer To Bool object to the output of the Integer Waveform Generator.
- Double-click on the Integer Waveform Generator object. Click on the **Wave** field in the Property Manager and select **Square** from the list. Click on the **Period** field and enter **1** followed by a Return. Close the Property Manager window.

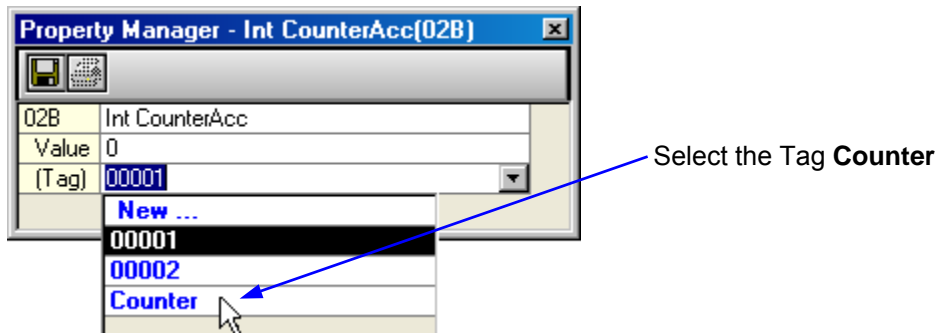


- Double-click on the Bool Counter object. Click on the **(Tag)** field in the Property Manager and enter the name **Counter**, followed by a Return. Click on the Counter Preset **Initial Value** and enter the value **10** followed by a Return. (The Counter will

count 10 pulses before it switches its output to True.) Close the Property Manager window.

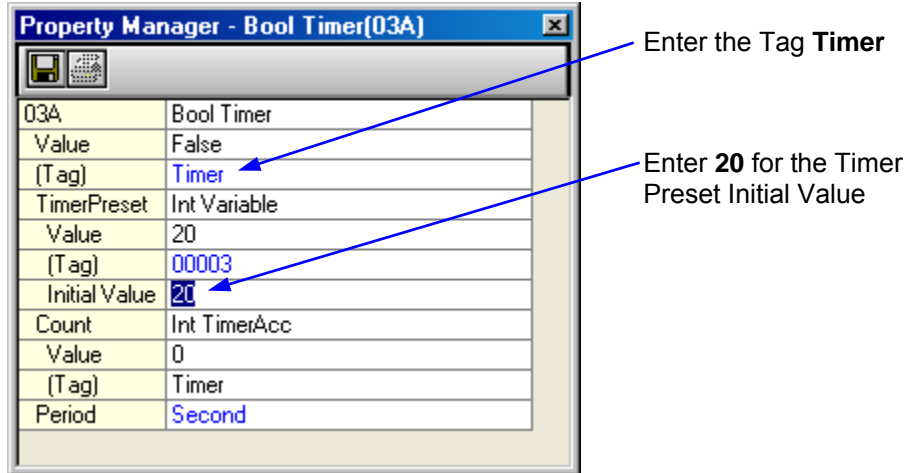


- Double-click on the Integer Counter Accumulator object. Click on the **(Tag)** field in the Property Manager and select **Counter** from the list of Tags. (The counter object and its accumulator must have the same Tag.) Close the Property Manager window.

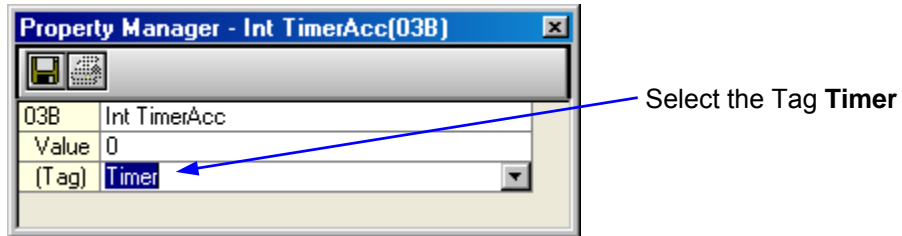


- Double-click on the Bool Timer object. Click on the **(Tag)** field in the Property Manager and enter the name **Timer**, followed by a Return. Click on the Timer Preset **Initial Value** and enter the value **20** followed by a Return. Make sure that the **Period** field indicates **Second**. (The Timer will wait 20 seconds before it switches its output to True.) Close the Property Manager window.

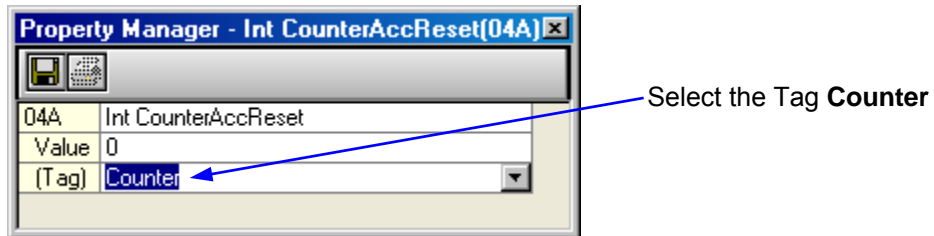
NetArrays Project Program Development Example



- Double-click on the Integer Timer Accumulator object. Click on the **(Tag)** field in the Property Manager and select **Timer** from the list of Tags. (The timer object and its accumulator must have the same Tag.) Close the Property Manager window.

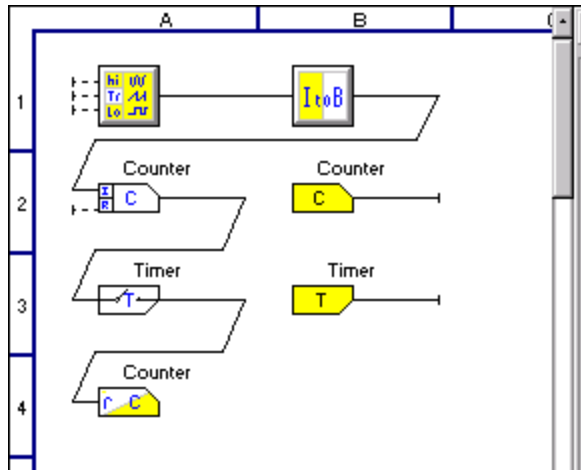


- Double-click on the Integer Counter Accumulator Reset object. Click on the **(Tag)** field in the Property Manager and select **Counter** from the list of Tags. (The reset object must have the same Tag as the counter it is resetting.) Close the Property Manager window.

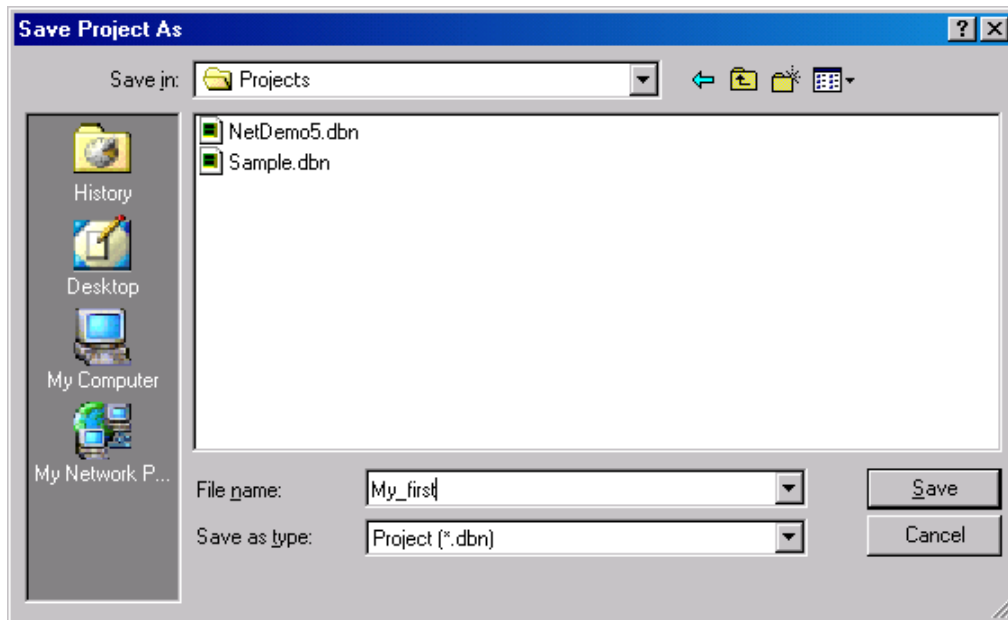


The Module Form should now look like the following example.

NetArrays Project Program Development Example



- In the NetArrays **File** menu, select **Save New Project As...** In the Save Project As dialog, enter the file name **My_first**. You can enter any name you want as long as it contains eight, or fewer, characters. Click on the **Save** button.




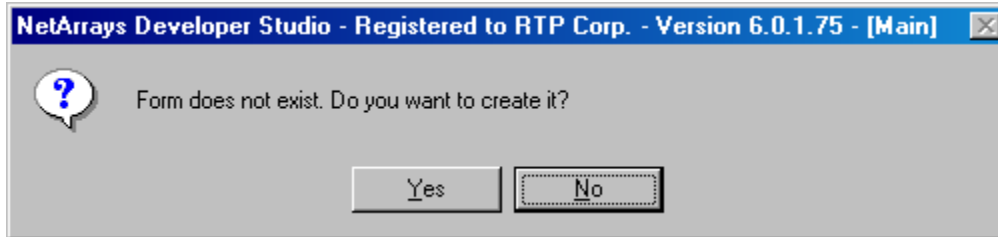
This completes the Counter Module Form. The next step is to define the logic within the RTP_View Module Form.

Module Form “RTP_View”

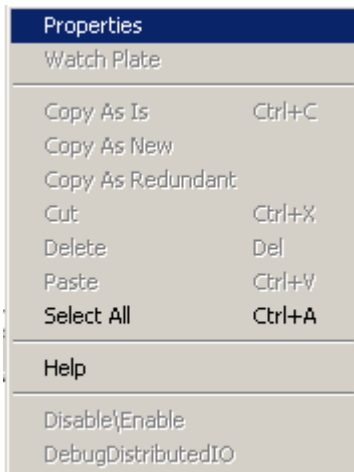
Add Objects to a New Module Form to provide signals for RTP_View Example



You will now create a new Module Form and place objects onto it from the Toolbox.

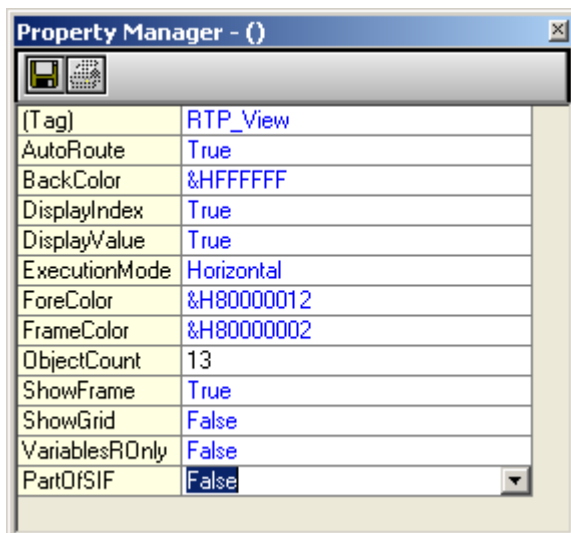
- Return to the Main Form by clicking on the  button in the Main Toolbar.
- Double-click on the **RTP_View** object in the Main Form. The following prompt will appear on your screen.




- Click on the **Yes** button to create a new Module Form with the Tag **RTP_View**.
- Right click on the page and select **Properties** to open the Project Property window.

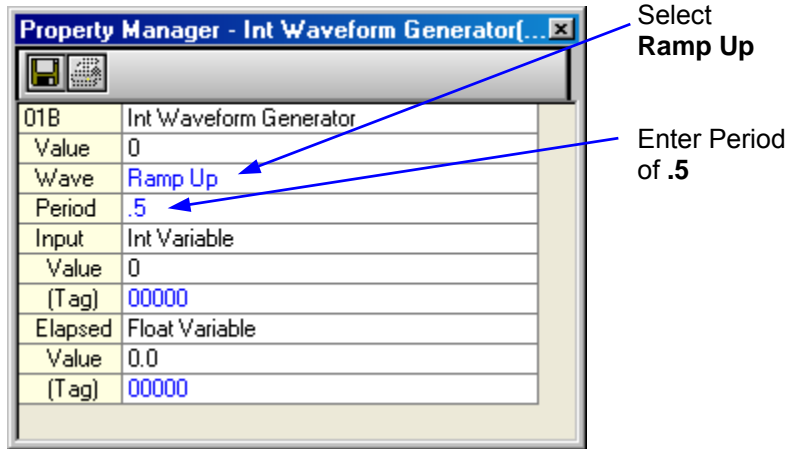



- In the Property Manager window click on **PartOFSIF**, click  and select **False**. Click on **VariablesROnly**, click  and select **False**.




NetArrays Project Program Development Example

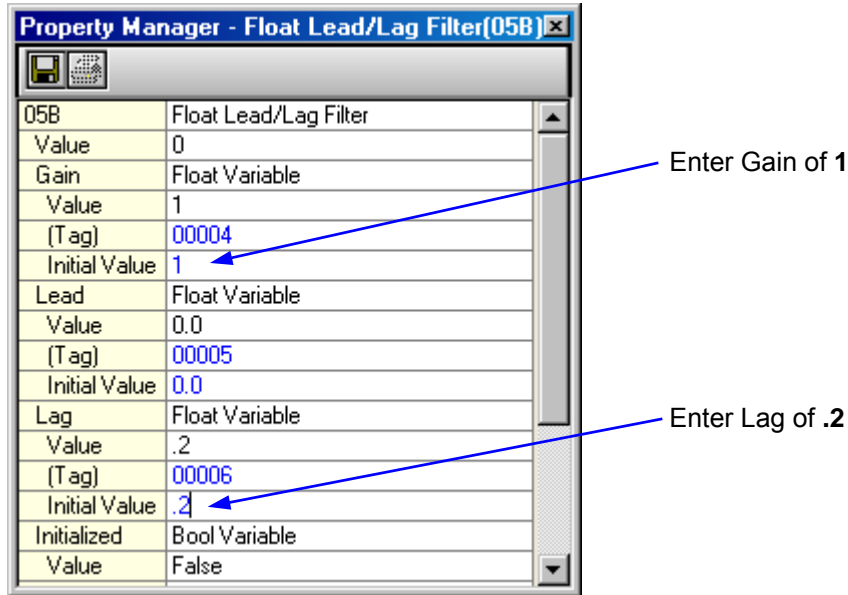
- Drag an Integer Constant object to Cell A1. Double-click on the object to display the Property Manager. Enter the Tag name **Three** and an Initial Value of **3**. Close the Property Manager window. This object will set the upper limit of a Waveform Generator.
- Drag another Integer Constant object to Cell A2. Double-click on the object to display the Property Manager. Enter the Tag name **Zero**. Close the Property Manager window. This object will set the lower limit of a Waveform Generator.
- Drag a Bool Variable object to Cell A3. Double-click on the object to display the Property Manager. Enter the Tag name **Key**. Close the Property Manager window. The output of this object will be controlled by a Key Switch object in the RTPView project.
- Drag an Integer Waveform Generator object  to Cell B2. Double-click on the object to display the Property Manager. Select **Ramp Up** for the **Wave** function. Enter **0.5** for the **Period**. Close the Property Manager window. Connect the High Limit input of the Waveform Generator to the output of Integer Variable **Three**. Connect the Low Limit input of the Waveform Generator to the output of Integer Variable **Zero**. The output of the Waveform Generator will run the Mixer animation in the RTPView project.




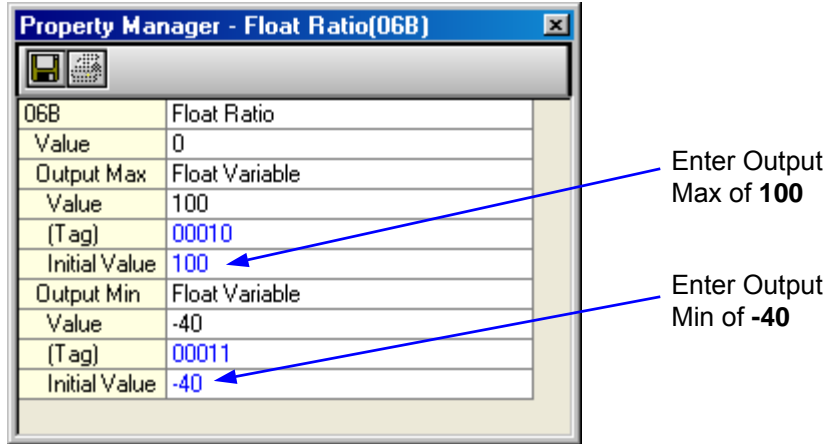
- Drag an Integer Gate object  to Cell B3. Connect the Input 1 input of the Gate object to the output of the Waveform Generator. Connect the Control input of the Integer Gate object to the output of Bool Variable **Key**. Connect the Input 2 input of the Integer Gate object to the output of Integer Variable **Zero**. This object activates the Mixer object when the Key Switch is turned on.
- Drag an Integer Variable object to Cell C3. Double-click on the object to display the Property Manager. Enter the Tag name **Mixer**. Close the Property Manager window. Connect the object's input to the output of the Integer Gate. This object will pass the output of the Waveform Generator, or zero, to operate the Mixer object in the RTPView project.
- Drag a Float Variable object to Cell A5. Double-click on the object to display the Property Manager. Enter the Tag name **Fader**. Close the Property Manager window. The Fader object in the RTPView project will control this variable.
- Drag a Float Constant object to Cell A6. Double-click on the object to display the Property Manager. Enter the Tag name **Hundred** and an Initial Value of **100**. Close the Property Manager window. This object will set the high limit of a Float Ratio object.

NetArrays Project Program Development Example

- Drag another Float Constant object to Cell A7. Double-click on the object to display the Property Manager. Enter the Tag name **FloatZero** and an Initial Value of **0.0**. Close the Property Manager window. This object will set the low limit of a Float Ratio object.
- Drag a Float Lead/Lag object  to Cell B5. Double-click on the object to display the Property Manager. Under the Gain heading, enter an Initial Value of **1**. Under the Lag heading, enter an Initial Value of **0.2**. Leave the Lead Initial Value at the default value (**0**). Close the Property Manager window. Connect the Input of the Lead/Lag Filter to the output of the Float Variable **Fader**. This object drives the fill level of a Tank in the RTPView project.



- Drag a Float Ratio object  to Cell B6. Double-click on the object to display the Property Manager. Under the Output Max heading, enter an Initial Value of **100**. Under the Output Min heading, enter an Initial Value of **-40**. Close the Property Manager window. Connect the High Limit input of the Ratio object to the output of the Float Constant **Hundred**. Connect the Input of the Ratio object to the output of the Float Variable **Fader**. Connect the Low Limit input of the Ratio object to the output of the Float Constant **FloatZero**. The Ratio object converts the range of the Fader object input (0 to 100) to a temperature range of -40 to 100 for the Temperature Gauge object in the RTPView project.

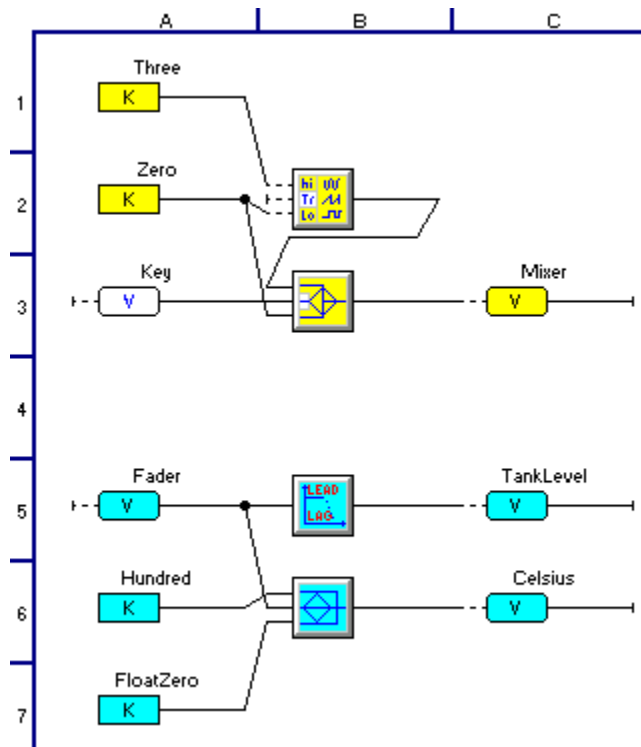


- Drag a Float Variable object to Cell C5. Double-click on the object to display the Property Manager. Enter the Tag name **TankLevel**. Close the Property Manager window. Connect the input to the output of the Lead/Lag Filter object. This variable will control the level displayed in the Tank object in the RTPView project.
- Drag a Float Variable object to Cell C6. Double-click on the object to display the Property Manager. Enter the Tag name **Celsius**. Close the Property Manager window. Connect the input to the output of the Float Ratio object. This variable will simulate a temperature input to the Temperature Gauge in the RTPView project.
- In the NetArrays **File** menu, select **Save My_first.dbn** to save your project. The RTP_View Module Form should look like the following illustration. The table provides a summary of the objects for checking your project program.

Summary of Module Form Objects

Cell	Object	Properties
A1	Int Constant	(Tag) = Three; Initial Value = 3
A2	Int Constant	(Tag) = Zero; Initial Value = 0
A3	Bool Variable	(Tag) = Key
B2	Int Waveform Generator	Wave = Ramp Up; Period = 0.5
B3	Int Gate	
C3	Int Variable	(Tag) = Mixer
A5	Float Variable	(Tag) = Fader
A6	Float Constant	(Tag) = Hundred; Initial Value = 100
A7	Float Constant	(Tag) = FloatZero; Initial Value = 0.0
B5	Float Lead/Lag Filter	Gain Initial Value = 1; Lead Initial Value = 0; Lag Initial Value = 0.2
B6	Float Ratio	Output Max Initial Value = 100; Output Min Initial Value = -40
C5	Float Variable	(Tag) = TankLevel
C6	Float Variable	(Tag) = Celsius

NetArrays Project Program Development Example




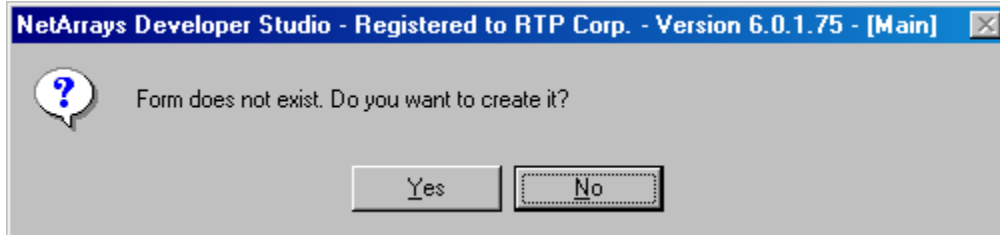
- In the NetArrays **Device** menu, select **Select ▶** and choose Simulator. In the NetArrays **Device** menu, select **Download Project** to download and run your project on the selected device.
- Click “Yes” to overwrite to the current Target Node. Examine the Output Window to confirm that the download was successful.

Module Form “RTP_ADA”

Add Objects to a New Module Form to provide signals for RTPADA Example.

You will now create a new Module Form and place objects onto it from the Toolbox.

- Return to the Main Form by clicking on the  button in the Main Toolbar.
- Double-click on the **RTP_ADA** object in the Main Form. The following prompt will appear on your screen.

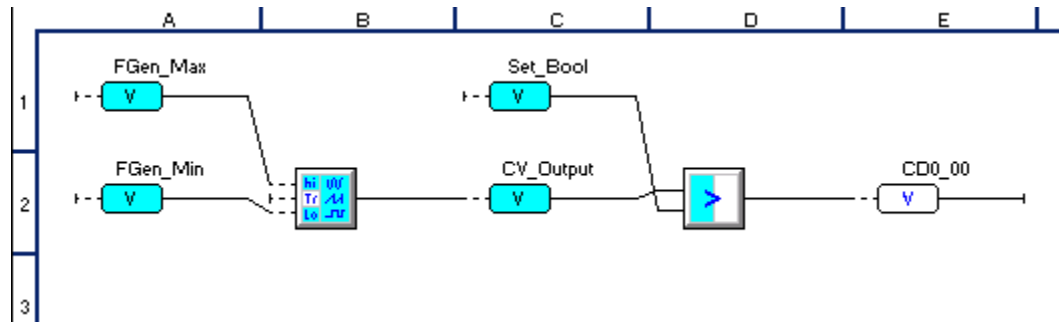


- Click on the **Yes** button to create a new Module Form with the Tag **RTP_ADA**.
- Drag a Float Variable object icon from the Toolbox and drop it onto cell A1. This object will be high value for the Waveform Generator.
- Drag a Float Variable object icon from the Toolbox and drop it onto cell A2. This object will be the low value for the Waveform Generator.
- Drag a Float Waveform Generator object icon from the Toolbox and drop it onto cell B2. This will create a sine wave.
- Drag a Float Variable object icon from the Toolbox and drop it onto cell C1. This object will be used to set the limit to trigger the Boolean Variable in Cell E2.
- Drag a Float Variable object icon from the Toolbox and drop it onto cell C2. This object will be the multi-level alarm used in the RTP ADA example.
- Drag a Float Greater Than object icon from the Toolbox and drop it onto cell D2. This object triggers the output in cell E2.
- Drag a Boolean Variable object icon from the Toolbox and drop it onto cell E2. This object will be the Boolean alarm used in the RTP ADA example.
- Interconnect the objects as shown in the figure below. Configure the object properties as shown in the table below. Leave all un-referenced at their default setting.

Cell	Object	Parameters
A1	Float Variable	Tag = "FGen_Max", Initial Value = "150"
A2	Float Variable	Tag = "FGen_Min", Initial Value = "-25"
B2	Float Waveform Generator	Wave = "Sine", Period = "120"
C1	Float Variable	Tag = "Set_Bool", Initial Value = "95"
C2	Float Variable	Tag = "CV_Output"
D2	Float Greater Than	
E2	Integer Variable	Tag = "CD0_00"

The finished page will appear as follows:

NetArrays Project Program Development Example



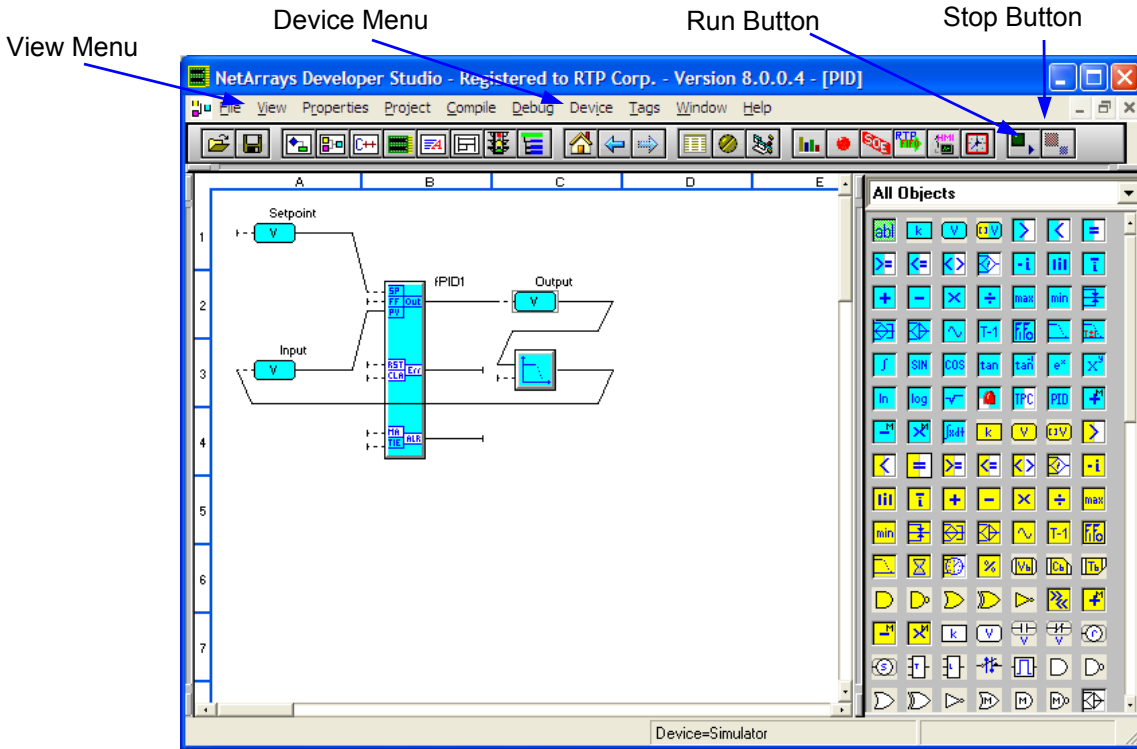
In the NetArrays **File** menu, select **Save My_first.dbn** to save your project.


This completes the creation of the NetArrays Project Program.

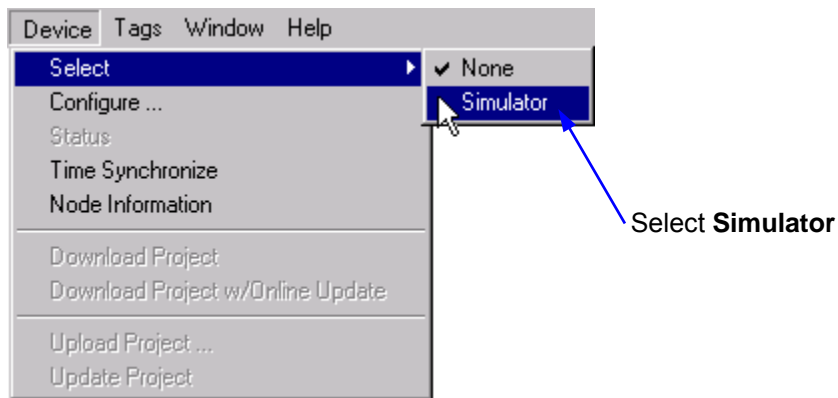
Verification

Verify Module Form PID

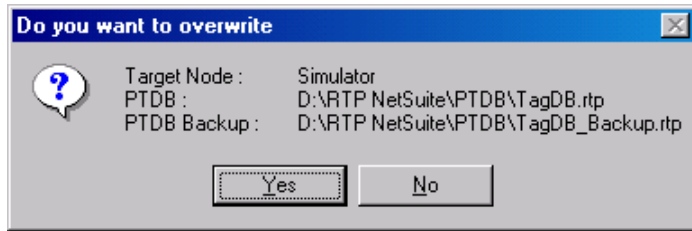
Now that the Project Program is completed, we will compile it and download it to the Simulator for verification.



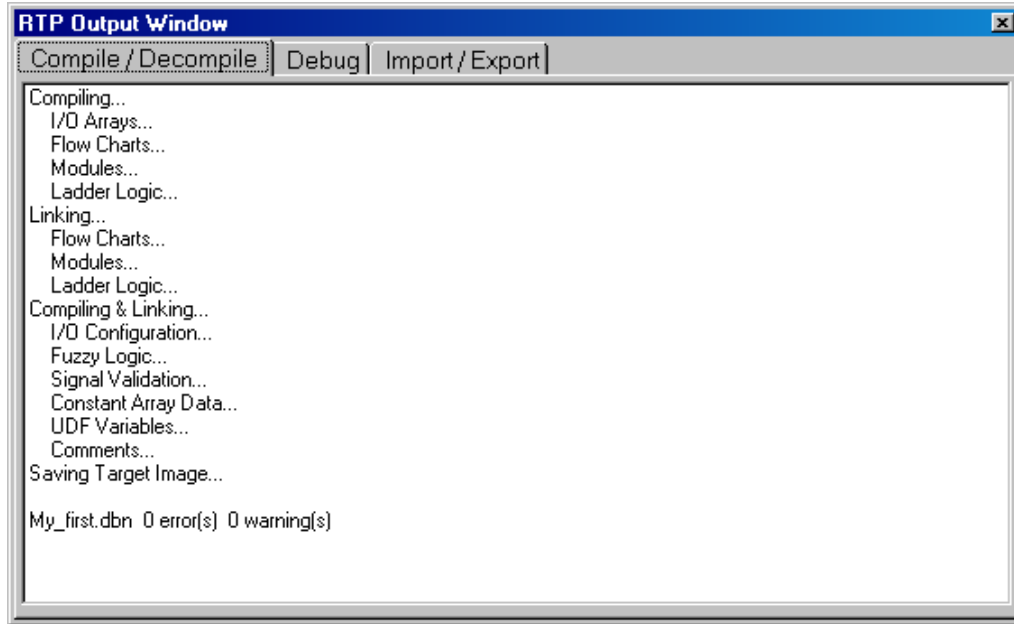
- Click on the  button in the Main Toolbar and then double-clicking on the **PID** module in the Main Form.
- In the **Device** menu, select **Select ▶ Simulator**.



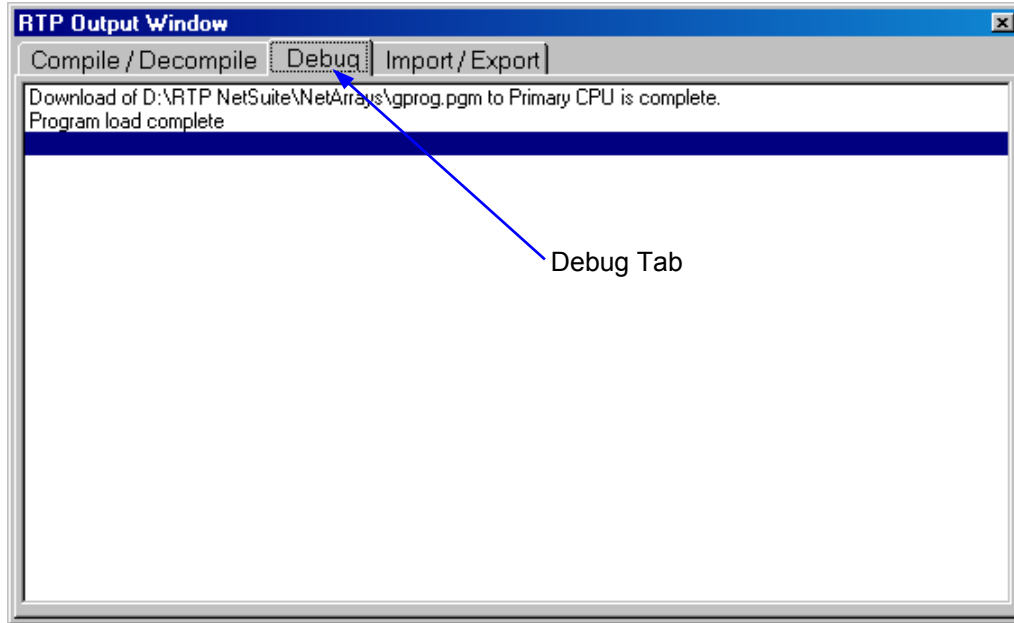
- Click on the **Run** button in the Main Toolbar. This will compile and download the Project Program to the Simulator.
- If you receive a “Do you want to overwrite” prompt, like the following figure, answer **Yes**.



- In the **View** menu, select **Output Window**. Select the Compile / Decompile Tab. Check that the compile was successful and that a 0 errors, 0 warnings message appears.



- Click on the Debug tab in the RTP Output Window. Check that the download was successful and that a program load complete message appears. Close the RTP Output Window.

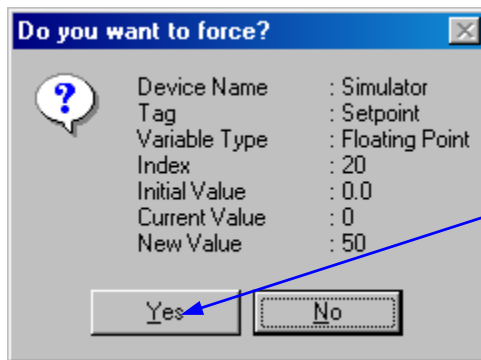


Now, we will change the Setpoint and note how the Output of the PID changes to match the Setpoint. We can change the Setpoint by "forcing" the output value of the Float Variable object.

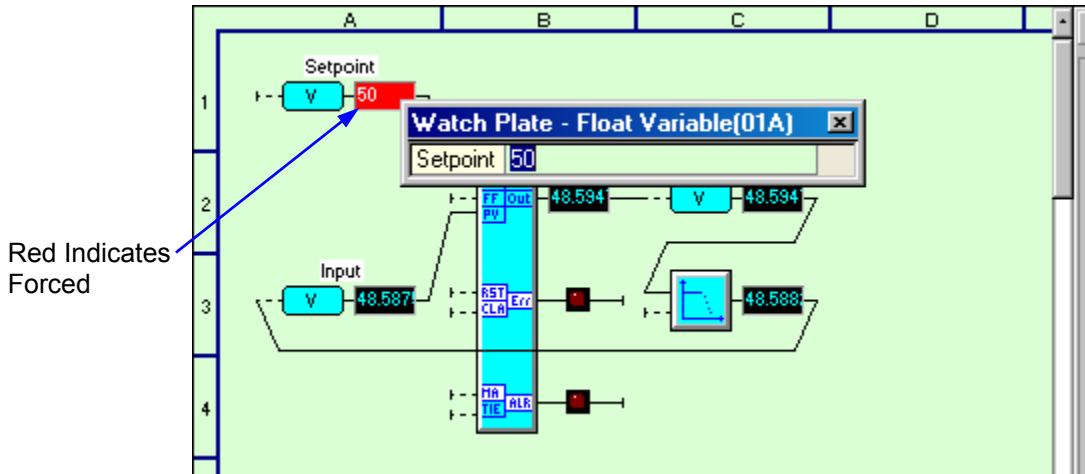
- Force the output of the **Setpoint** variable to 50. Move the cursor to the output value of the object and click the left mouse button. This displays the Watch Plate. Click on the **Setpoint** field and enter the value **50**, followed by a Return.



- Answer **Yes** to the "Do you want to force?"




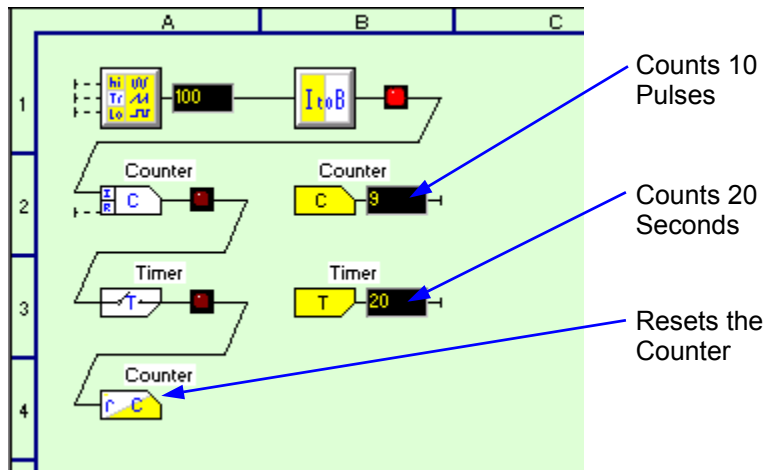
- Observe the output of the PID as it changes to the Setpoint value. Force the Setpoint to some other value and note how the output changes.



If the Output of the PID does not respond to the Setpoint change, stop the Simulator, unforce the variable, and carefully recheck the configuration of the objects within the PID Module Form.


Verify Module Form Counter

- Display the Counter Module Form by clicking on the  button in the Main Toolbar and then double-clicking on the **Counter** object in the Main Form.
- Observe that the output of the Waveform Generator alternates between 0 and 100, and that the output of the Integer to Bool converter switches between False and True every second.
- Observe that the Integer Counter Accumulator increments from 0 to 10, and when it reaches 10 the output of the Bool Counter object goes True.
- Observe that when the input to the Bool Timer goes True, the Integer Timer Accumulator ticks off 20 seconds. When the Integer Timer reaches 20 seconds, its output goes True, which resets the Integer Counter Accumulator. When the Integer Counter Accumulator gets reset, the Bool Counter will start counting input pulses again.




If the counter/timer circuit does not behave as expected, stop the Simulator and carefully recheck the configuration of the objects within the Counter Module Form.

Verify Module Form RTPView

- Display the RTP_View Module Form by clicking on the  button in the Main Toolbar and then double-clicking on the **RTP_View** object in the Main Form.
- Verify that the Integer Waveform in Cell B2 is cycling between 0 and 3, stepping approximately every half of a second.

This program will be further utilized in the Example “RTPView Project Program.”

Verify Module Form RTP_ADA

- Display the Counter Module Form by clicking on the  button in the Main Toolbar and then double-clicking on the **RTP_ADA** object in the Main Form.
- Verify that Float Variable CV_Output in Cell C2 is changing. If you observe it for a couple of minutes it will cycle between –25 and +150. The Boolean Variable in Cell E2 will be True when the value in CV_Output is greater than 95.

This program will be further utilized in the Example “Redundant Alarm Monitoring Configuration.”

Congratulations! You have successfully completed your first NetArrays Project Program and run it on the Simulator.