AUTOMATION UNIVERSITY

Continuous Process -
Advanced Process Control Instructions with Logix

Table of Contents

Continuous Process - Advanced Process Control Instructions with Logix

About Advanced Process Control Functionality

A traditional control system controls processes to fixed setpoints determined by operators. Generic control algorithms and instructions such as a PID controller are used.

Advanced Process Control (APC) is designed to automatically account for an expected (modeled, predicted, etc.) process response and calculates “optimal” control actions to minimize process variations. APC system utilizes technologies such as:

- Model Predictive control
- Multivariable control
- Fuzzy logic and control
- Adaptive control
- Inferential control
- Process modeling and simulation

What You Will Accomplish In This Lab

In this lab you will explore two different versions of APC: Internal Model Control (IMC) and Coordinated Control (CC).

Who Should Complete This Lab

This hands-on lab is intended for individuals who:

- Want to become more familiar with the Advance Process Control features of RSLogix 5000.
- Already have familiarity with RSLogix 5000 organizational layout and programming techniques.

Before You Begin

Before you begin this Hands-On Lab, please be sure to close any applications that are currently running.

To complete this lab, a general familiarity of computers, programmable controllers, I/O and automation software is recommended but not essential.

You will be using a V17 controller for this lab. Locate it on your station in the upper controller rack. The IP address of the Ethernet/IP card in the rack should be 172.16.XX.3, where XX is the number of your station

Lab Materials

For this Hands-On lab, we have provided you with the following materials that will allow you to complete the labs in this workbook.

Hardware

L63 Logix Controller.

Software

This hands-on lab uses the following software:

- RSLogix™ 5000 V17
- RSLinx Classic

Lab Files

This hands-on lab uses the following files:

- V17 Advanced Process Control AU09.doc
  - IMC_Simulation.L5X
  - SquareWave.L5X

Document Conventions

Throughout this workbook, we have used the following conventions to help guide you through the lab materials.

This style or symbol:	Indicates:
Words shown in bold italics (e.g., ***RSLogix 5000*** or ***OK***)	Any item or button that you must click on, or a menu name from which you must choose an option or command. This will be an actual name of an item that you see on your screen or in an example.
Words shown in bold italics, enclosed in single quotes (e.g., ***'Controller1'***)	An item that you must type in the specified field. This is information that you must supply based on your application (e.g., a variable). **Note:** When you type the text in the field, remember that you do not need to type the quotes; simply type the words that are contained within them (e.g., Controller1).
	The text that appears inside of this gray box is supplemental information regarding the lab materials, but not information that is required reading in order for you to complete the lab exercises. The text that follows this symbol may provide you with helpful hints that can make it easier for you to use this product. Most often, authors use this “Tip Text” style for important information they want their students to see.

Note: If the mouse button is not specified in the text, you should click on the left mouse button.

Lab 1: Using the IMC Instruction (40 minutes)

About This Lab

In this lab you will explore the functionality of the Internal Model Control (IMC) instruction. This instruction controls a single process variable by manipulating a single output. The instruction compares the actual process signal to an internal model. The internal model can be manually entered, or the built-in autotuner can be used. This instruction is most suitable for processes with long deadtimes which are difficult to control with standard PID instructions. The setup and configuration parameters, as well as the autotune function, are very similar to the PIDE instruction.

Note: The PIDE instruction uses an autotune to generate a process model which is then used to generate P, I and D gain values. Once the gains are calculated the model is not utilized in the control loop. The IMC instruction uses the process model directly and continuously compares the process response (PV) to the modeled response. This results in tighter control to the setpoint for processes with long deadtimes, when compared with PIDE performance.

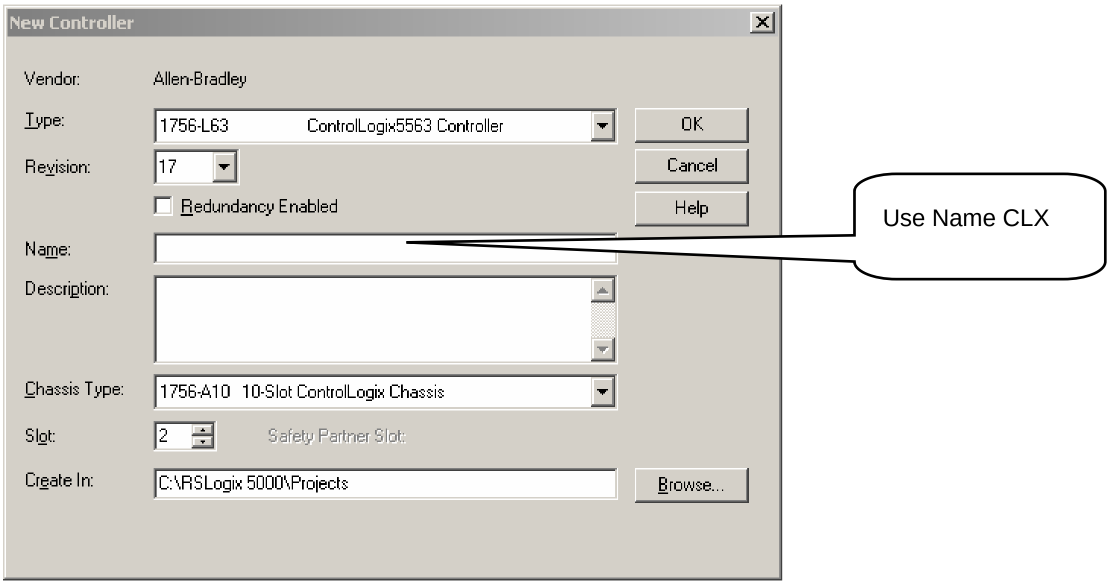
Comparing PIDE Loop Control to IMC Loop Control

1. Open ***RSLogix 5000*** 

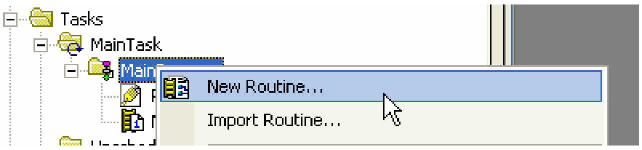
.

2. Select ***File*** > ***New***

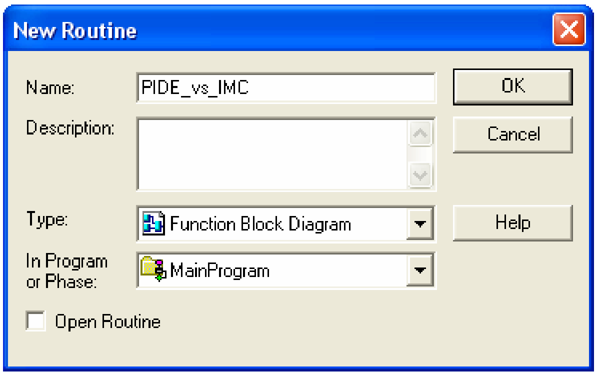
3. Configure your controller as shown below adding Name: **CLX** and then click ***OK***.



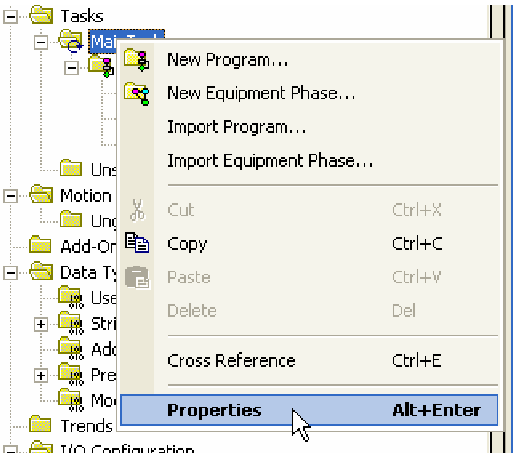
4. Expand the MainProgram by clicking on the '**+**'. Right click on ***MainProgram*** and select ***New Routine.***



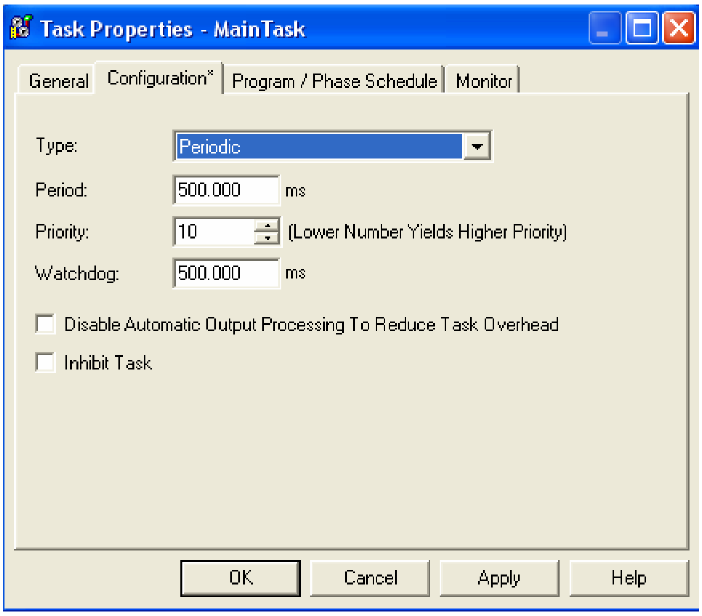
5. Configure the new routine as shown below and click ***OK***.



6. Right click on ***MainTask*** and select ***Properties***.

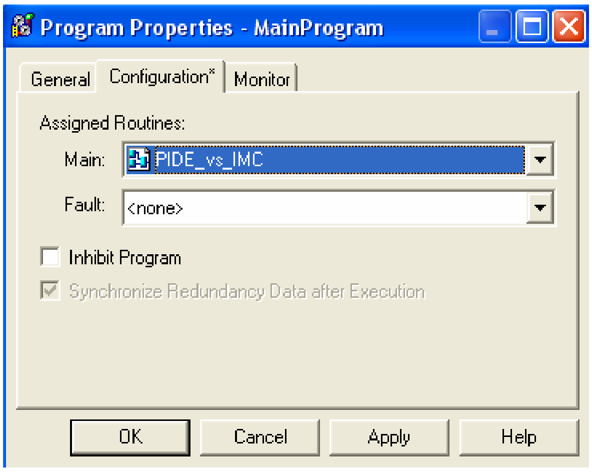


7. Go to the ***Configuration*** tab and change the ***Type*** to ***Periodic*** and set the ***Period*** to ***'500'***.

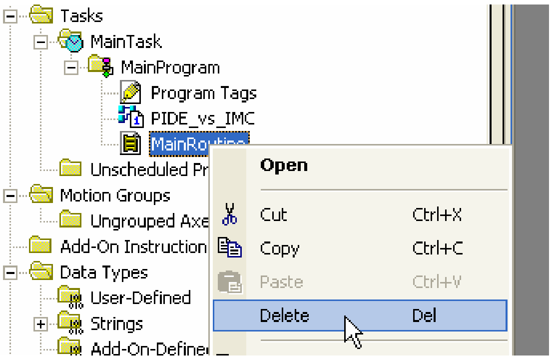


8. Click ***OK*** when done.

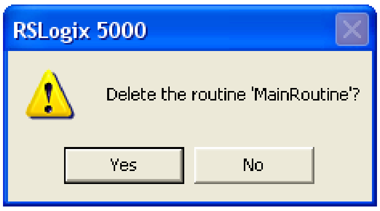
9. Right click on ***MainProgram*** and select ***Properties***. In the ***Configuration*** tab, change ***Assigned Routines Main*** to ***'PIDE_vs_IMC'*** and click ***OK***.



10. Right click on ***MainRoutine*** and select ***Delete***.

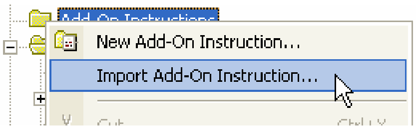


11. Click ***Yes*** to confirm the deletion.



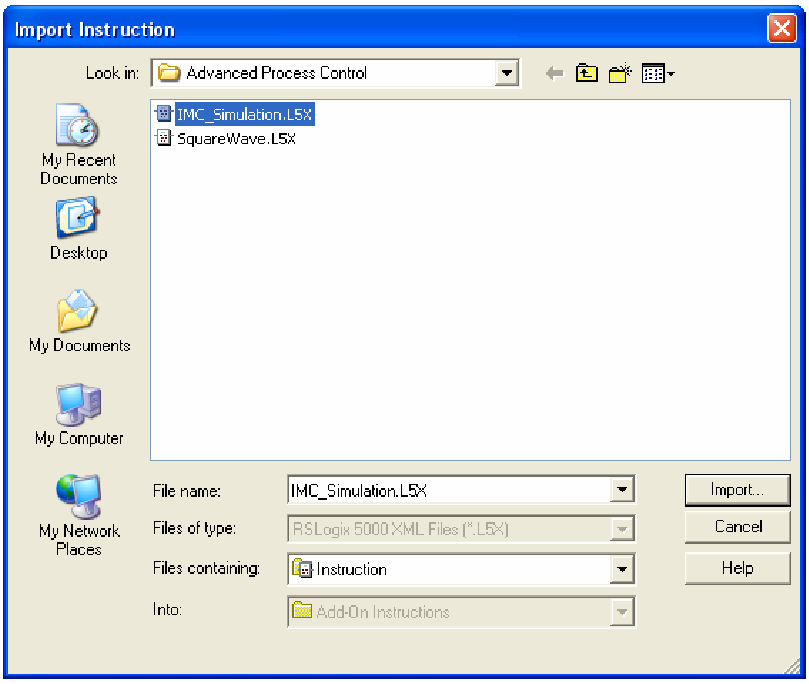
Why did we delete this? We could have just used a JSR to call the PIDE_vs_IMC routine. It was deleted to totally eliminate any ladder logic from the project. Some "Process" people would see ladder logic and say "this is a PLC – and PLCs aren't good for process control. It really just shows that ControlLogix controller is in no way dependent on Ladder logic.

12. Right click on the ***Add-On Instructions*** folder and select ***Import Add-On Instruction***.



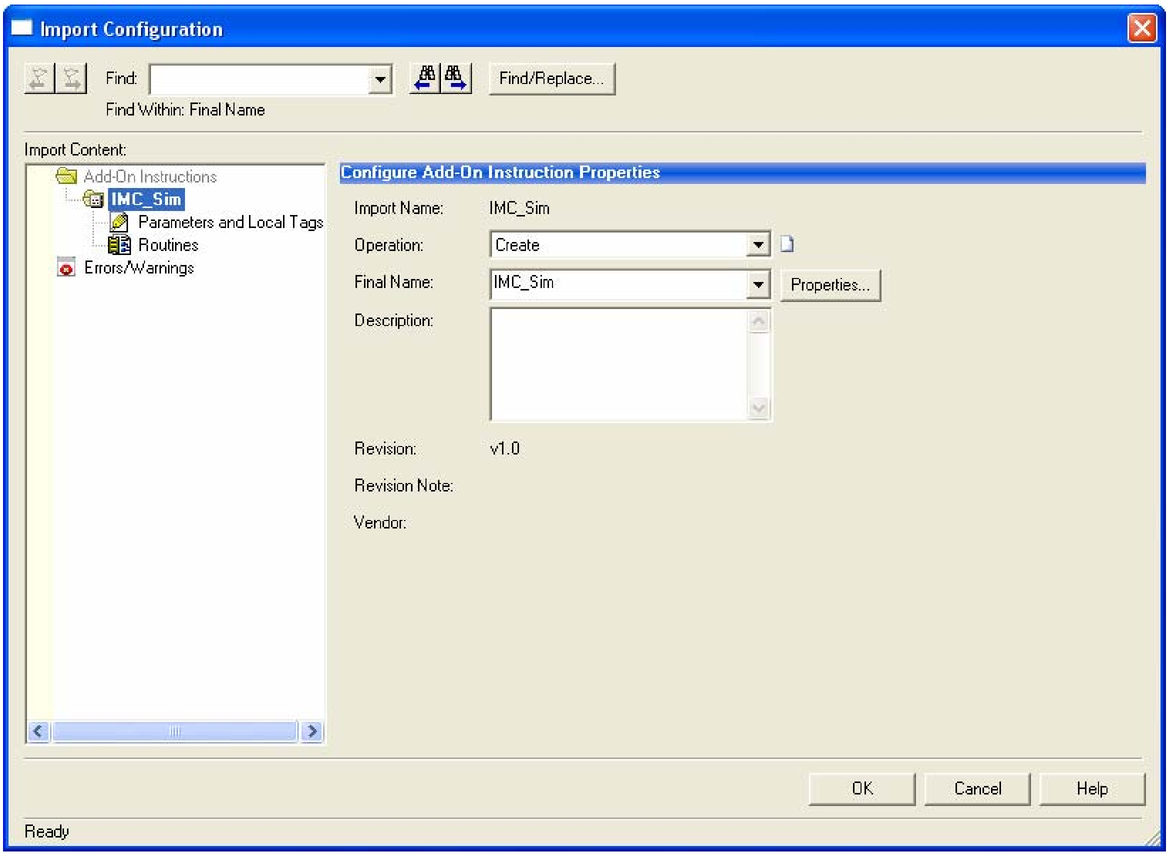
13. Go to the ***Advanced Process Control*** folder and select ***IMC_Simulation.L5K***.

This block will be used to simulate the controlled system.



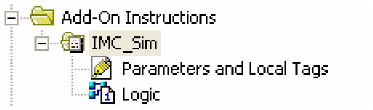
14. Click ***Import***.

The following window opens:

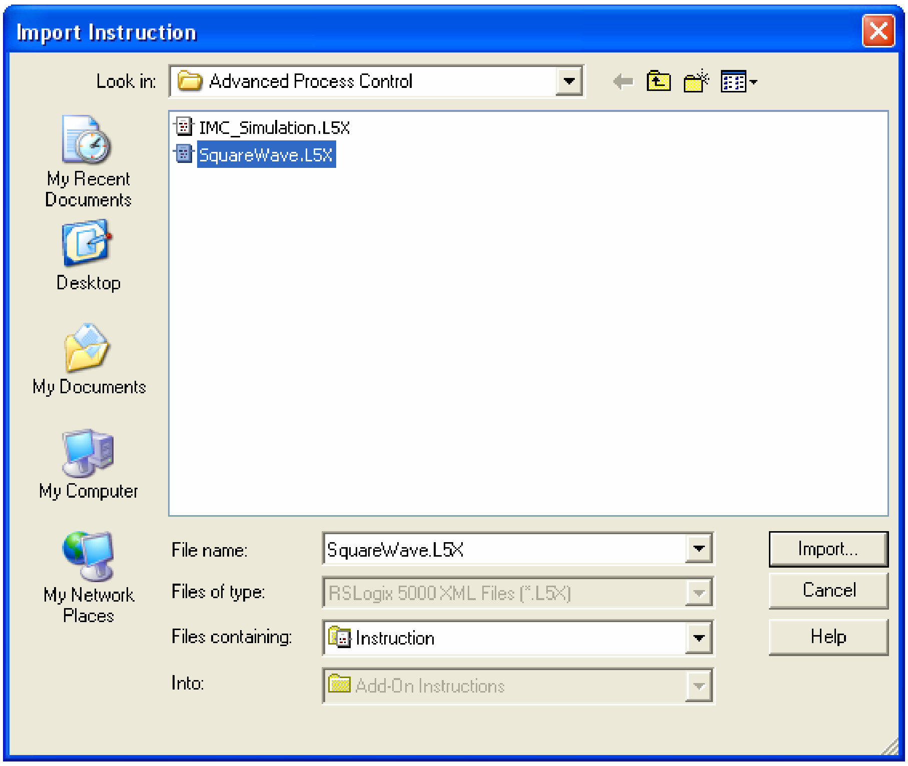


15. Click ***OK*** to complete the import.

The Add-On Instruction should now appear in your project.

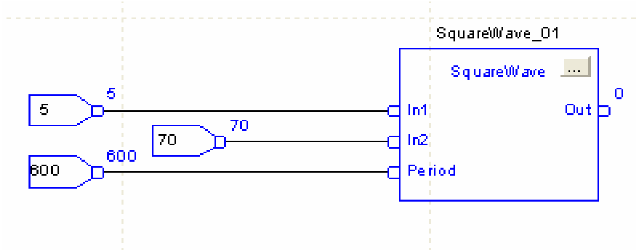


16. Import the ***SquareWave.L5K*** Add-On Instruction.

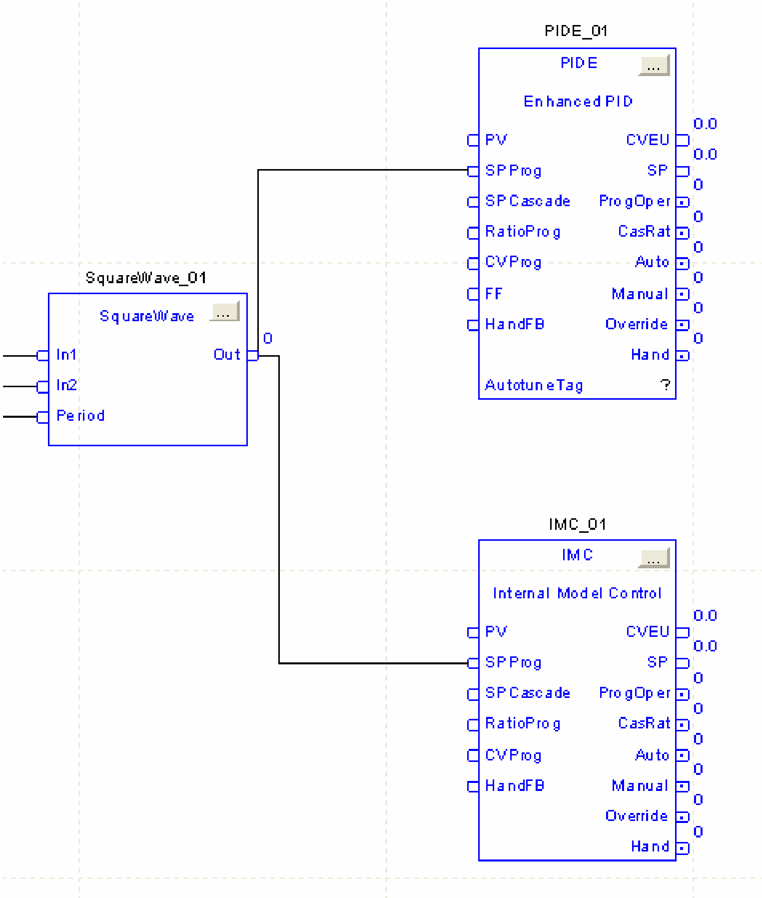


17. Double click on the **PIDE_vs_IMC routine** and configure the logic shown below.

Note: If you are unfamiliar with configuring logic using function blocks just ask your instructor for assistance.



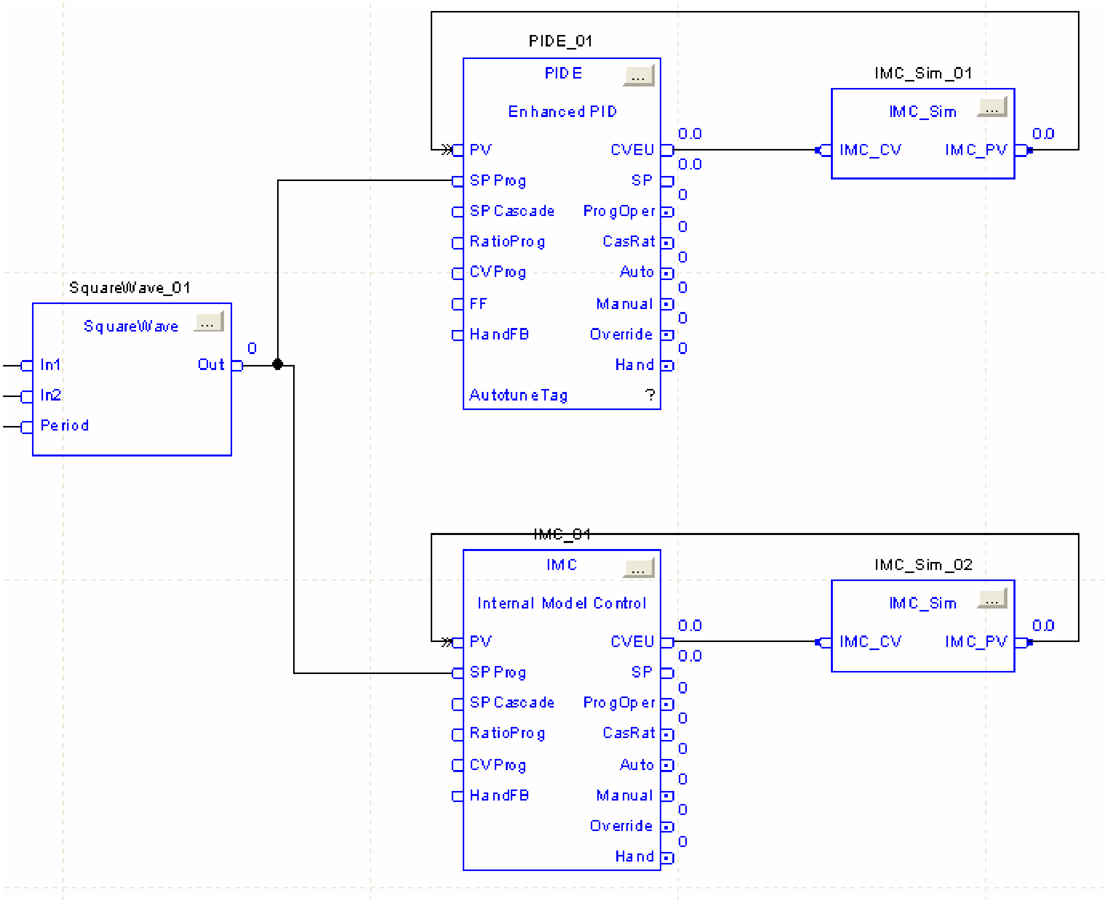
18. Add the PIDE and IMC blocks as shown below.



Note: In order to reduce the size of the function blocks several parameters (which are by default visible) have been hidden. You may leave them visible, or go into the function block parameters and uncheck the box in the first column (do this on the PIDE block only). If you are unfamiliar with this ask your instructor for assistance.

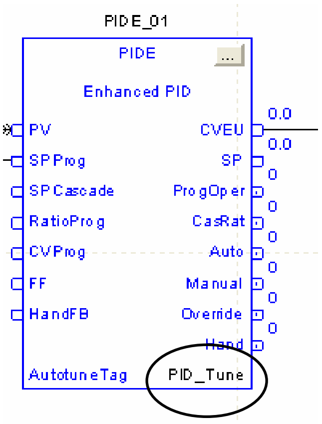
Note: You may have to increase the sheet size. Just right click on the page, select Properties and then Sheet Layout and select whatever size sheet you want.

19. Next we need to add process simulation. Add the Add-On Instruction IMC_Simulation as shown below.

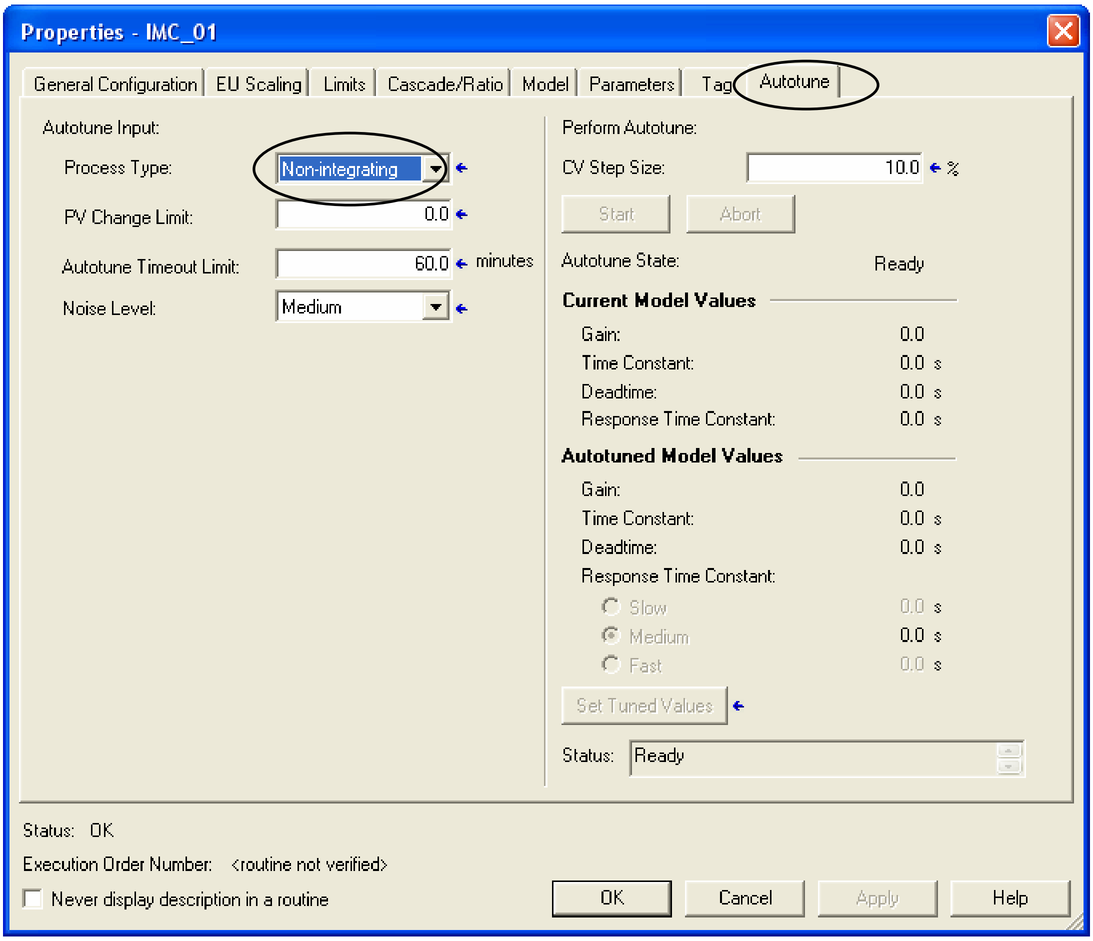


Note: Be sure to add 'Assume Data Available' to the feedback from the simulation to the PIDE and IMC blocks. Again, if you are not familiar with this concept or process ask your instructor.

20. Add a ***'PID_Tune'*** tag to the PIDE block and right click to define the New 'PID_Tune' tag.

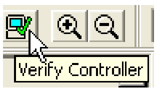


21. Open the parameters for the IMC block (...) and on the Autotune tab for Process Type select ***Non-Integrating***.



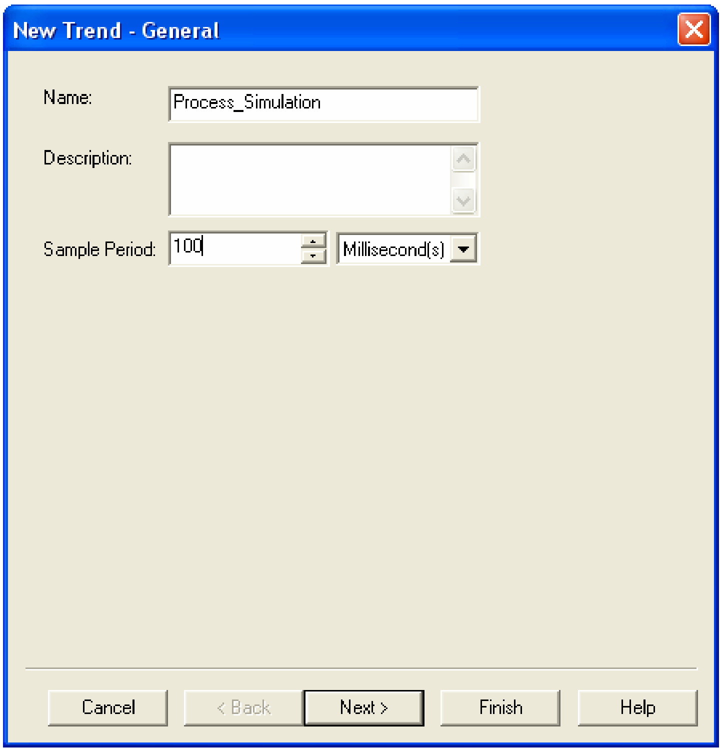
22. Click ***OK*** to accept your change.

23. Verify your controller. There should be no errors.

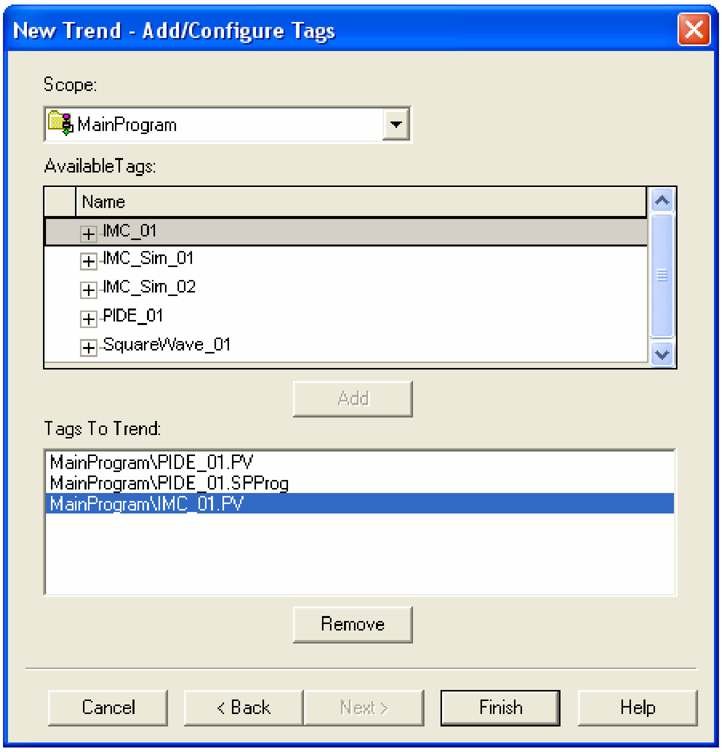


24. Create a new trend by right clicking on the ***Trend*** folder and selecting ***New Trend***.

25. Configure the new trend as shown below. Name Process_Simulation, Sample Period 100ms.

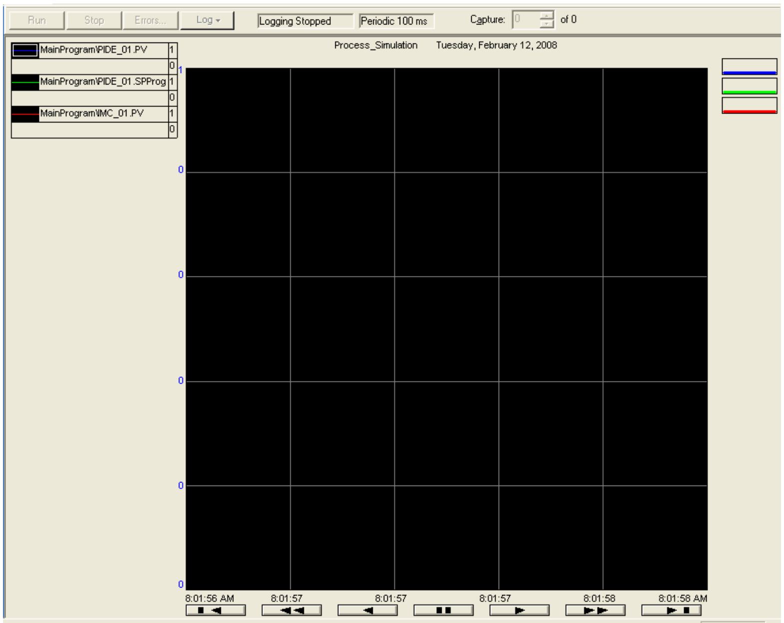


26. Click ***Next*** and add the tags listed below.

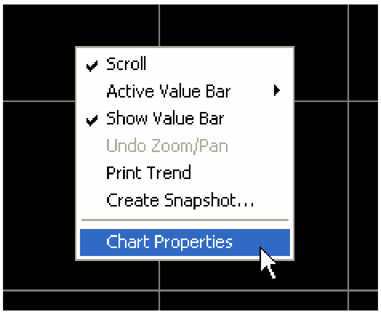


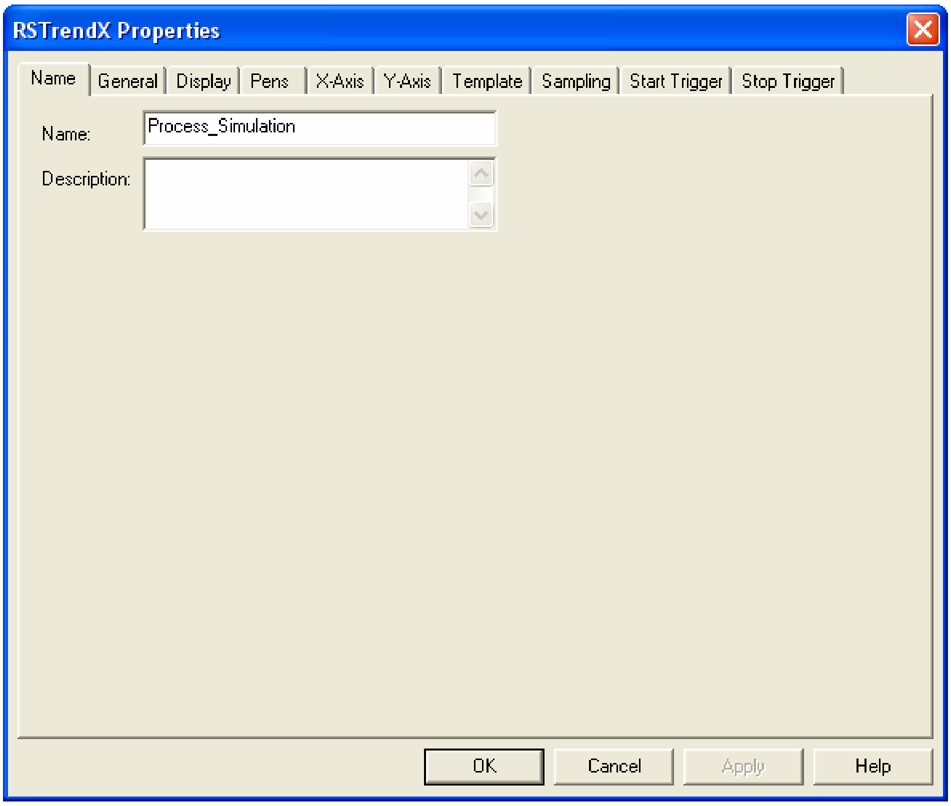
27. Click ***Finish***.

The chart will appear.

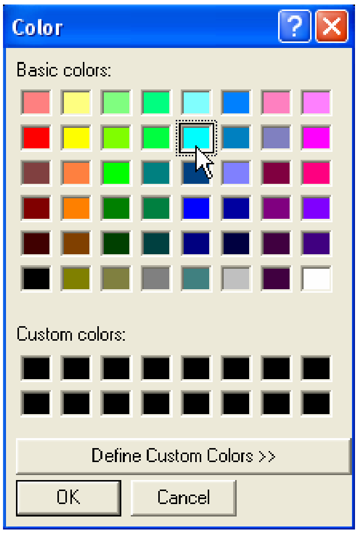


28. Right click on the chart and select ***Chart Properties***.

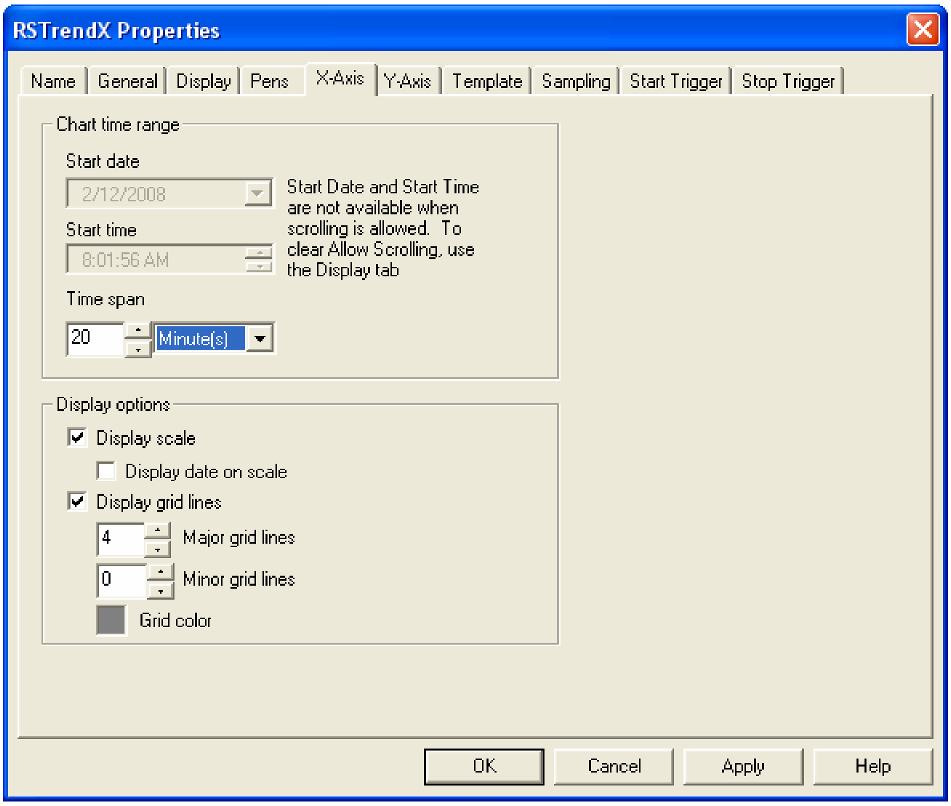




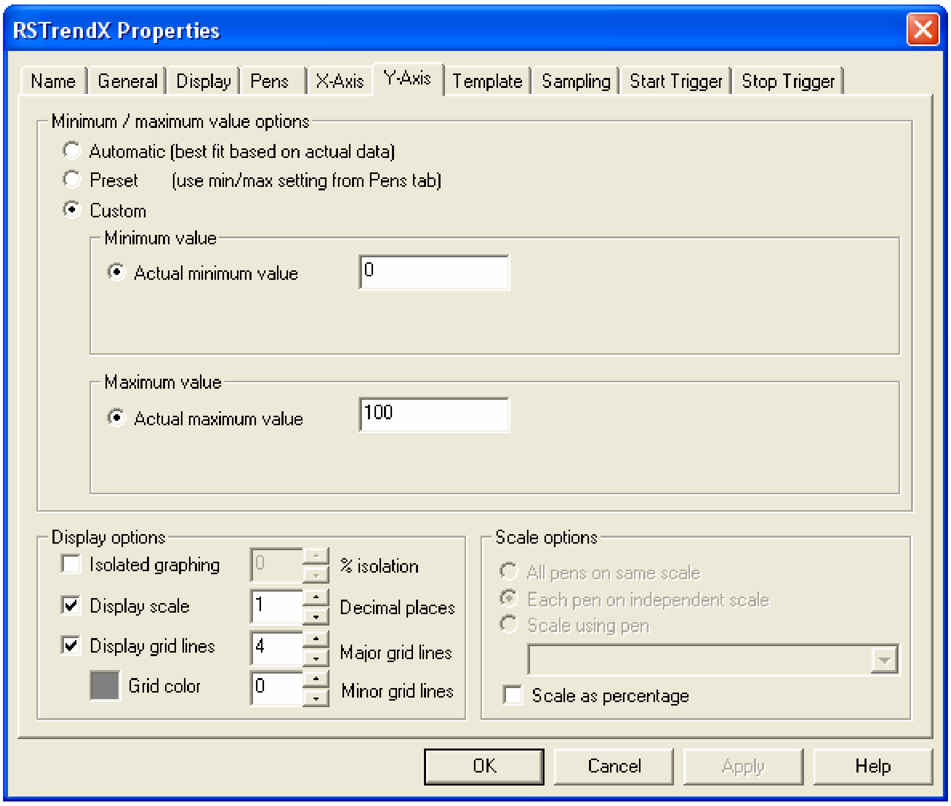
29. Click on the ***Pens*** tab and adjust the pen colors as you wish. The first pen is usually dark purple which doesn't show-up very well.



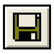
30. Click on the ***X-Axis*** tab and adjust the X-axis time span as shown below.



31. Click on the ***Y-Axis*** tab and adjust the Y-axis as shown below.



32. Click ***OK*** to finish the trend.

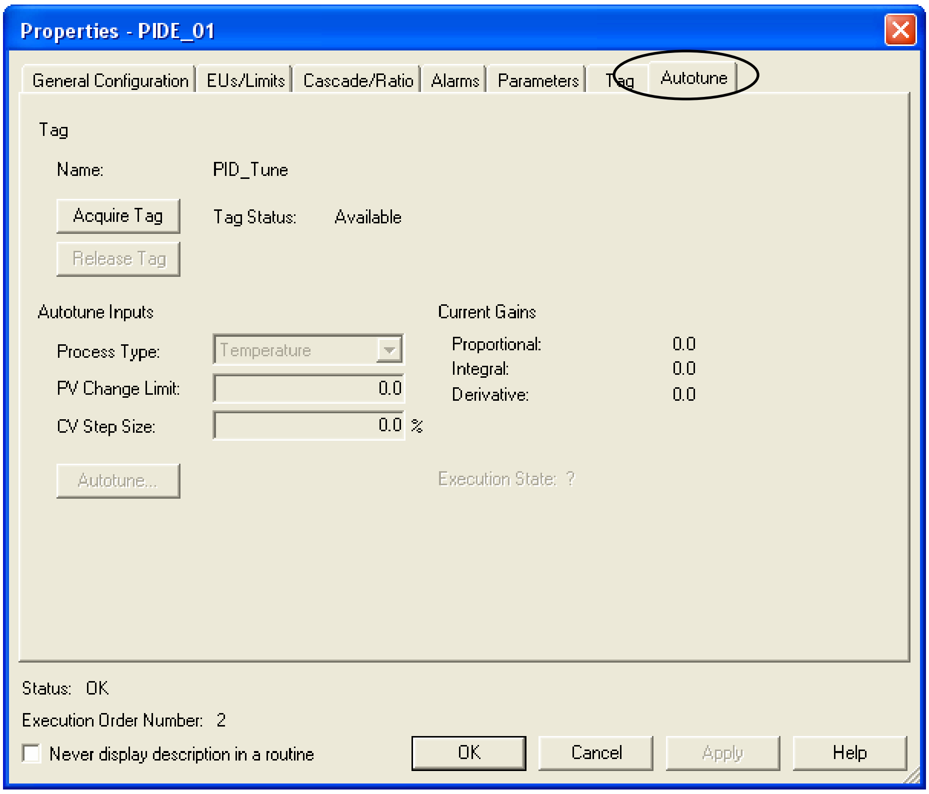
33. ***Verify***  and ***Save***  your project.

34. Download the project to your controller and put the controller into ***Run*** mode.

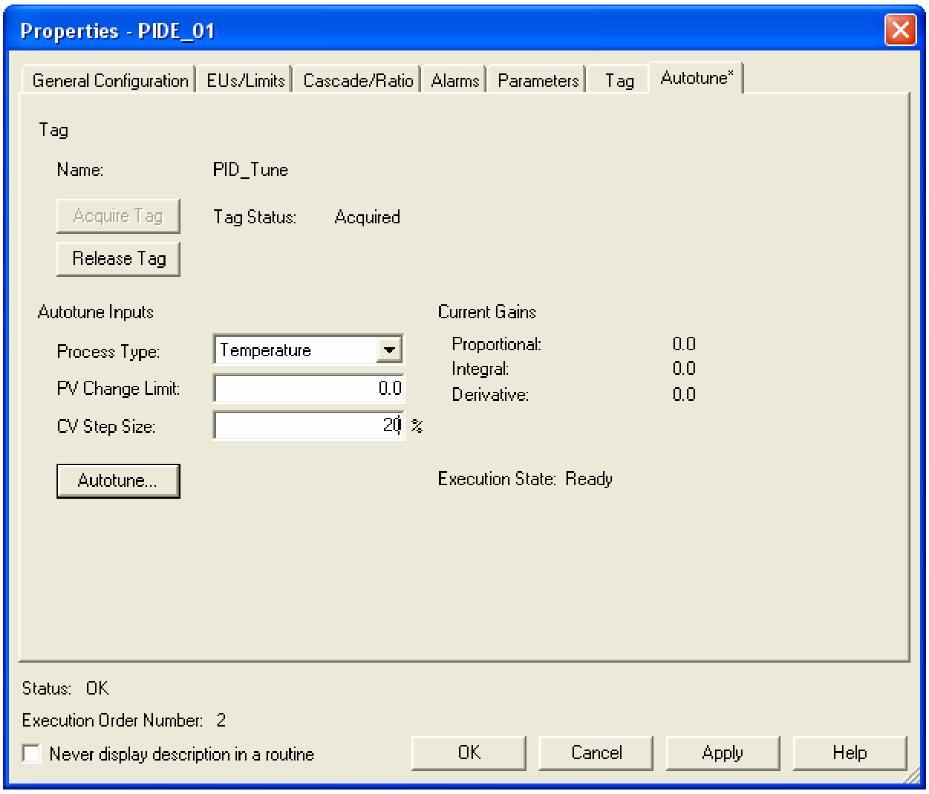
35. Double click on the ***PID_vs_IMC routine*** to view the logic.

36. Open the PIDE parameters by clicking on the  button.

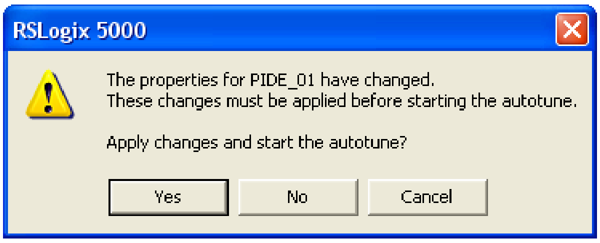
37. Select the ***Autotune*** tab.



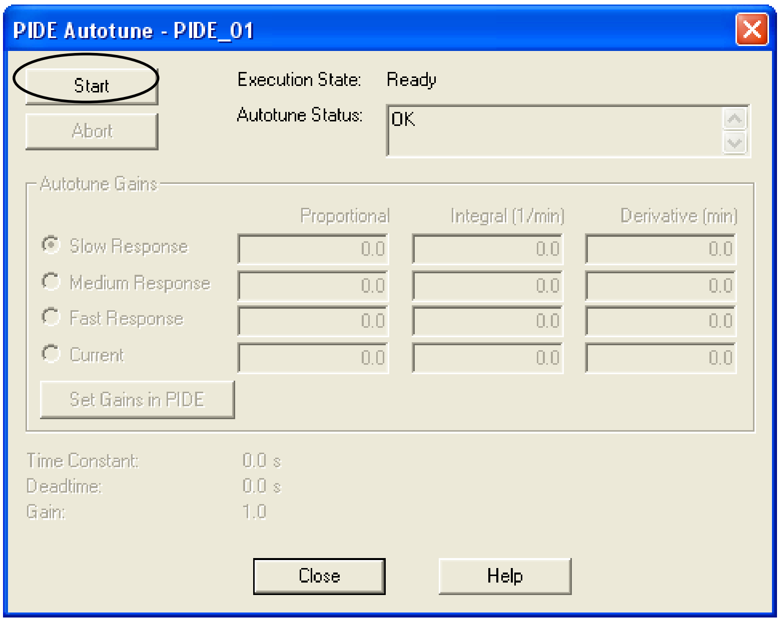
38. Click on the ***Acquire Tag*** button Acquire Tag and adjust the CV Step Size to ***'20'***.



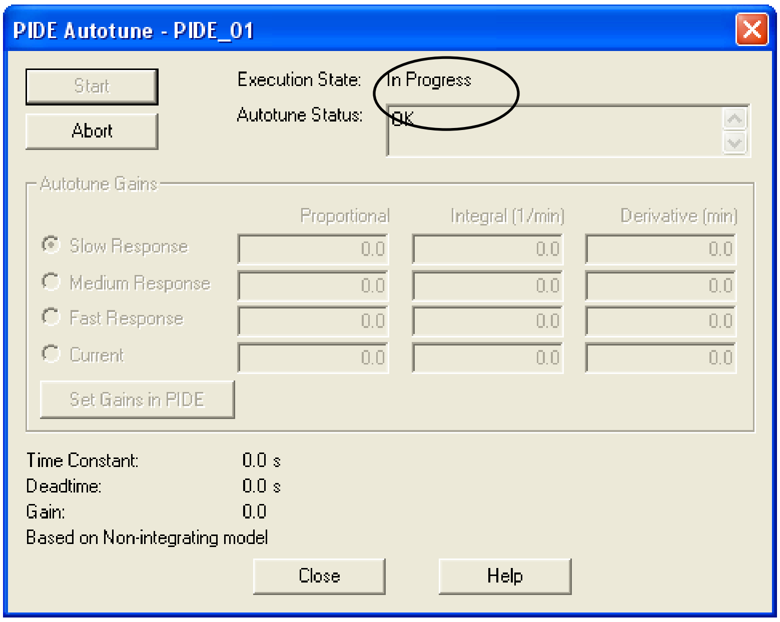
39. Click on the ***Autotune*** button  and click ***Yes*** to accept the changes before starting the autotune.



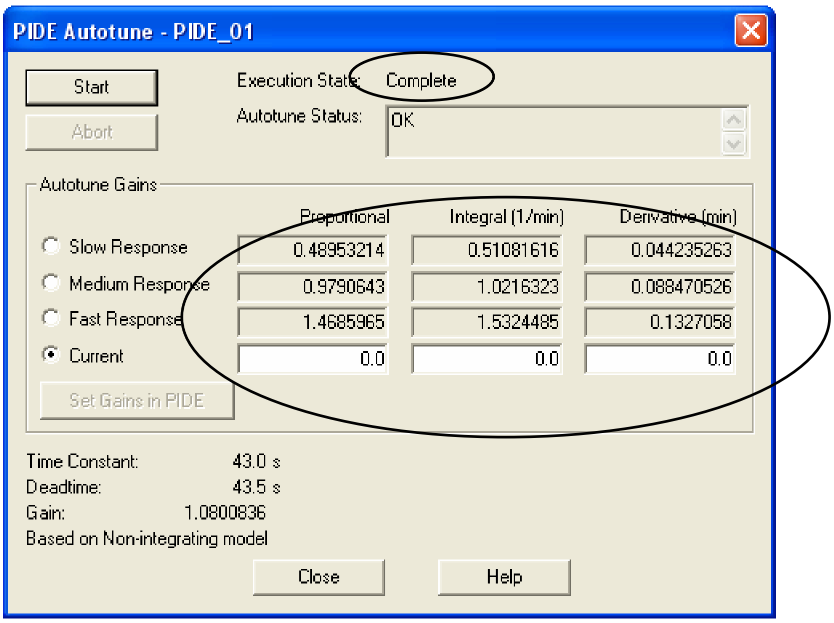
40. Click on the ***Start*** button  to begin the autotune.



41. The Execution State will change from Ready to In Progress.

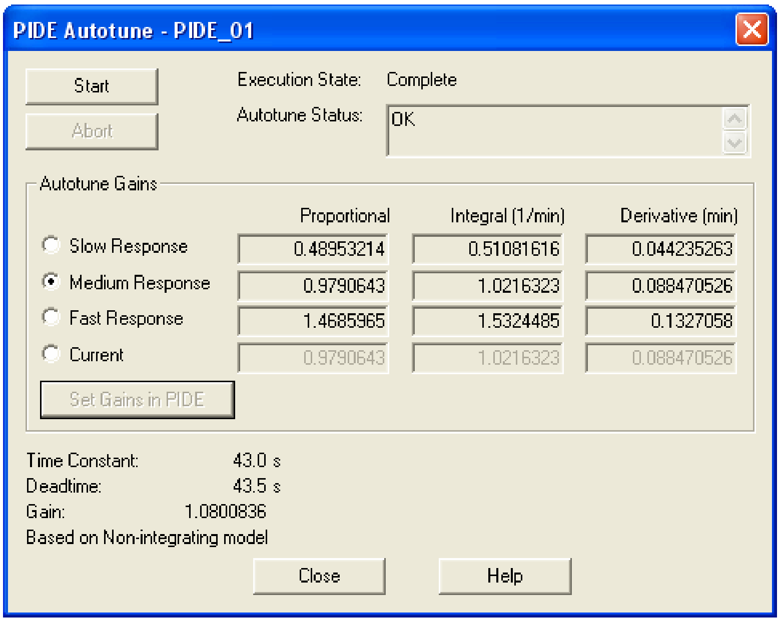


42. When the autotune is complete (this will take several minutes) the Execution State will change to Complete and Autotune Gains will be displayed.



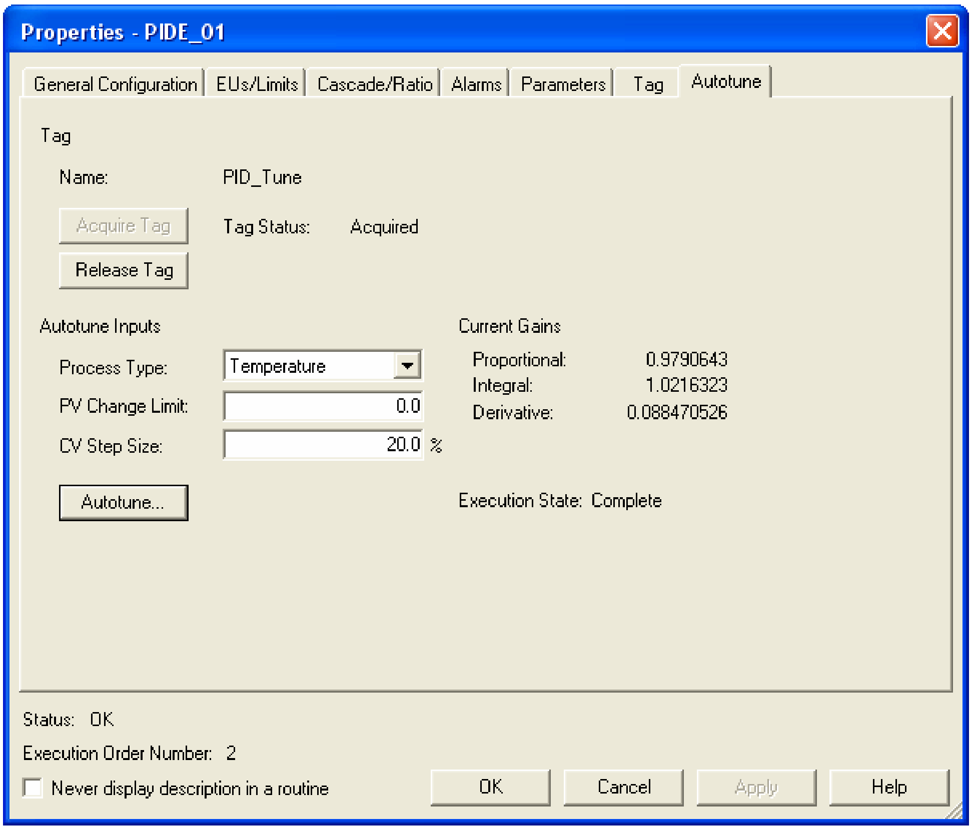
43. Select ***Medium Response*** and click on ***Set Gains in PIDE*** 

.



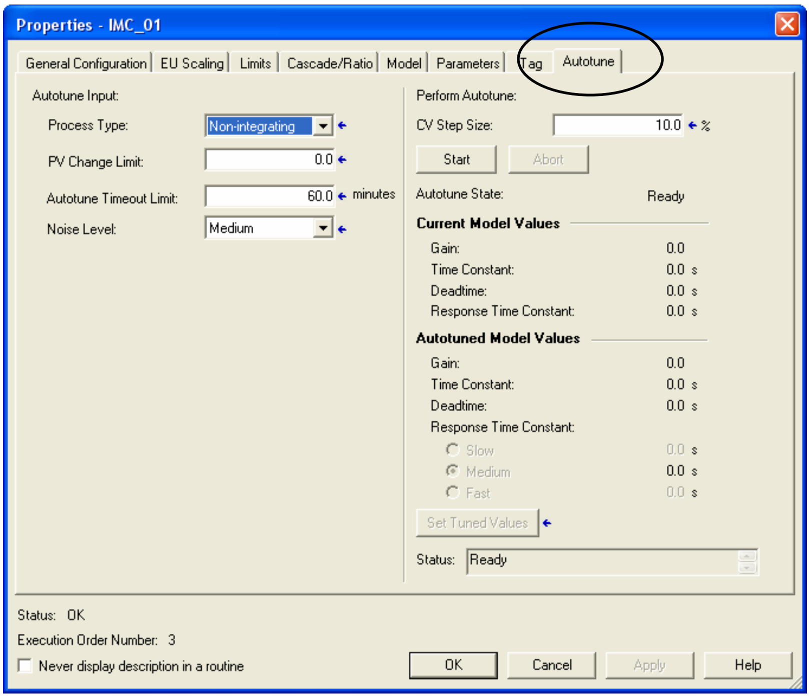
Note that gains have now been loaded into the ‘Current’ line.

44. Click ***Close***. The Autotune tab shows the results of the autotune and the current loaded gains.

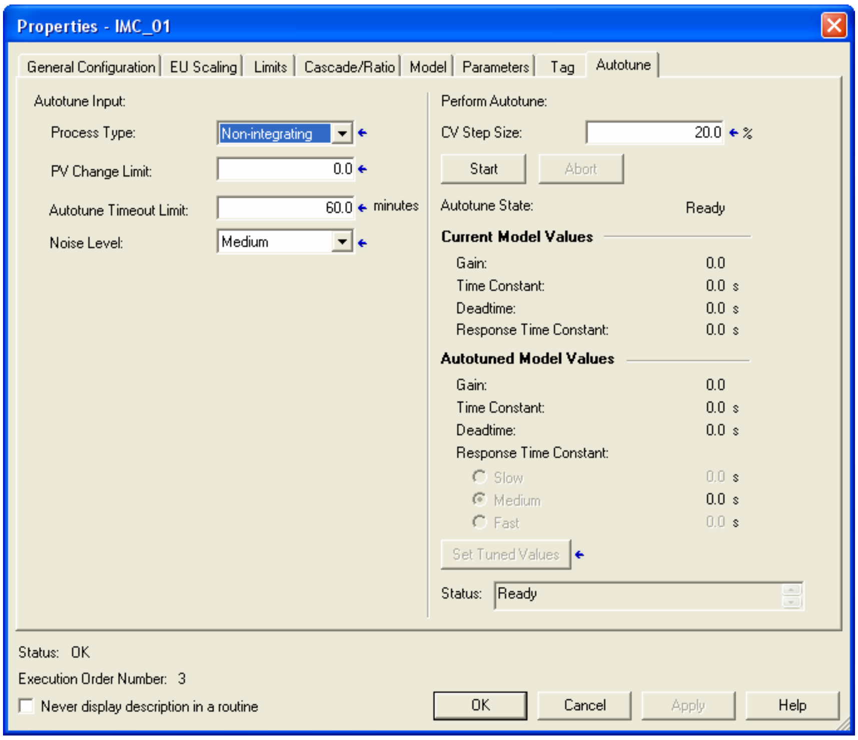


45. Click ***OK***.

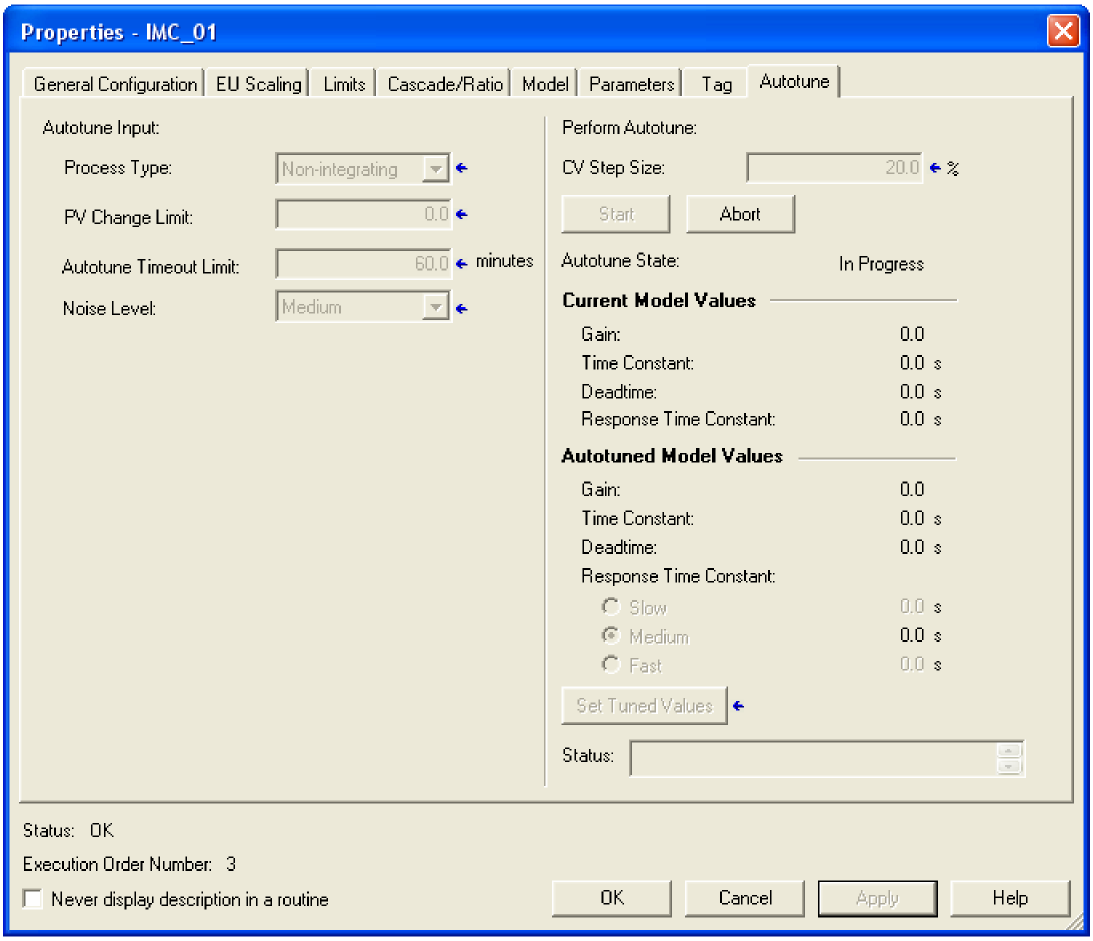
46. Click on the Properties button  for the IMC block and then select the ***Autotune*** tab.



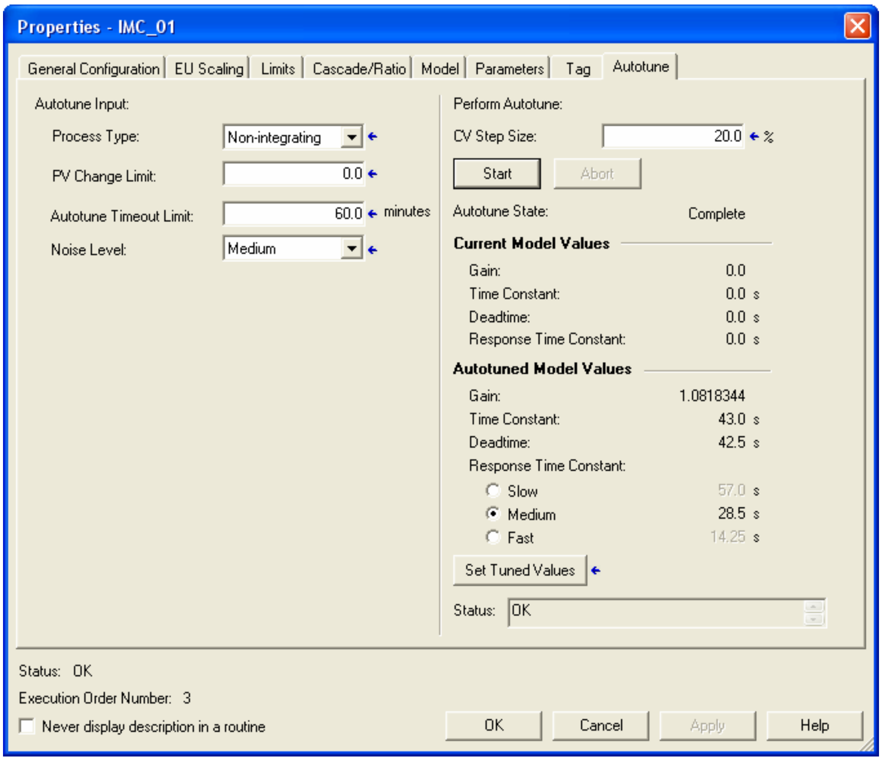
47. Change the CV Step size to ***'20'*** and then click ***Apply*** .



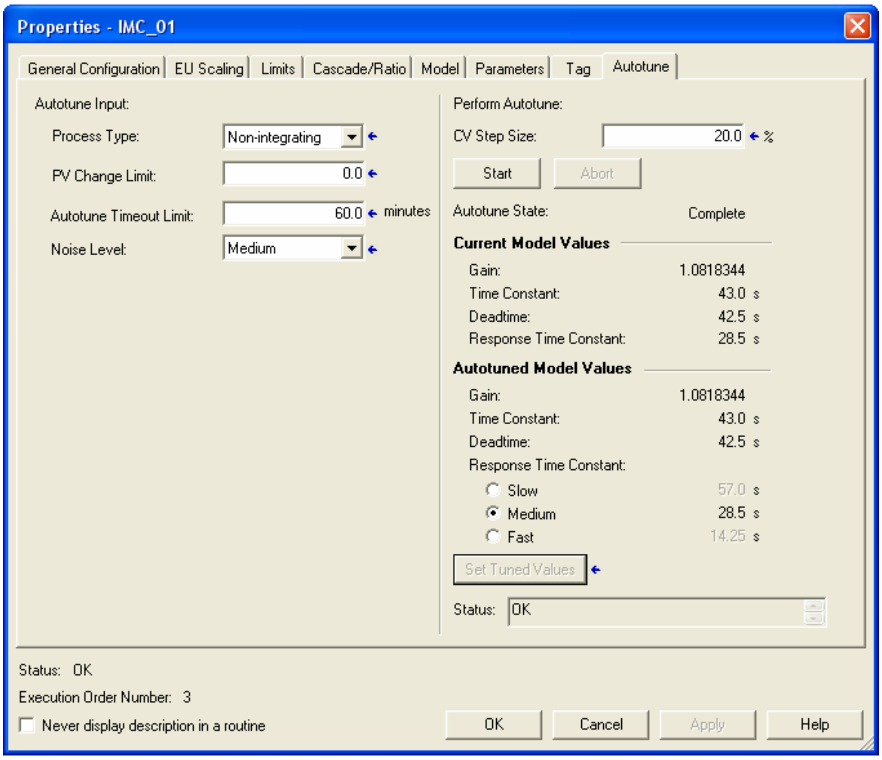
48. Click on the ***Start*** button (Start) to perform the autotune.



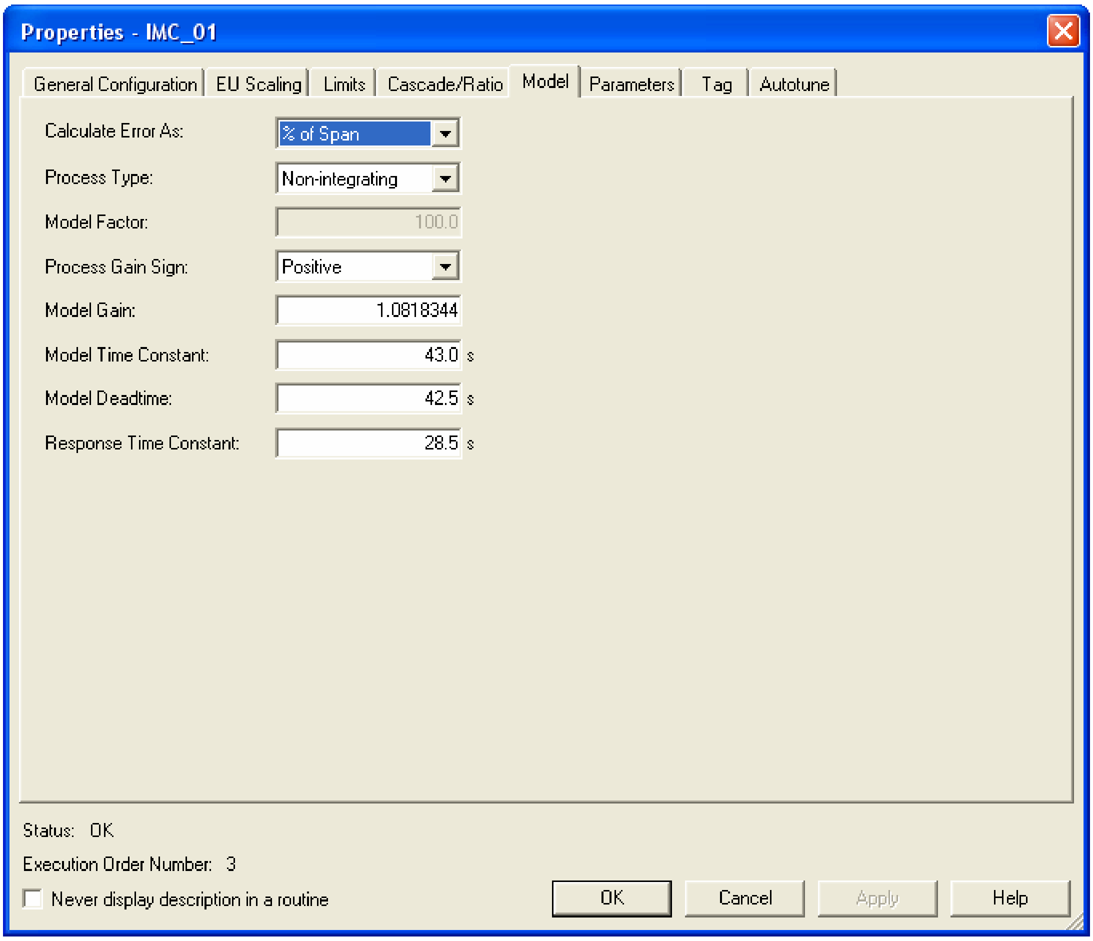
49. After completion the Autotuned Model Values will be populated.



50. ***Medium*** is the default Response Time Constant selected, click on the ***Set Tuned Values*** button Set Tuned Values.



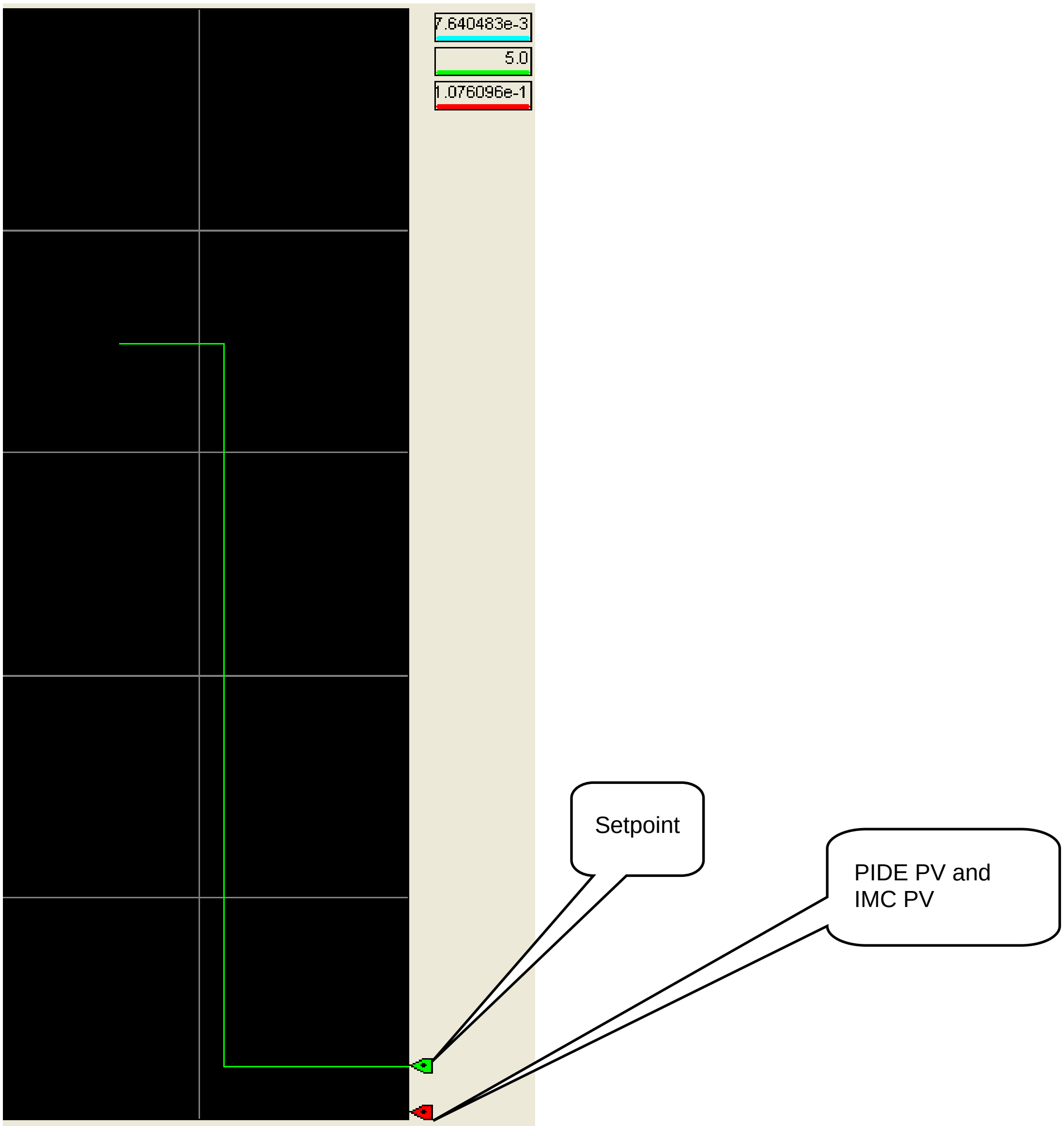
51. Click on the ***Model*** tab. This displays the internal model which will be used by the instruction.



52. Click ***OK*** to close the IMC Parameters window.

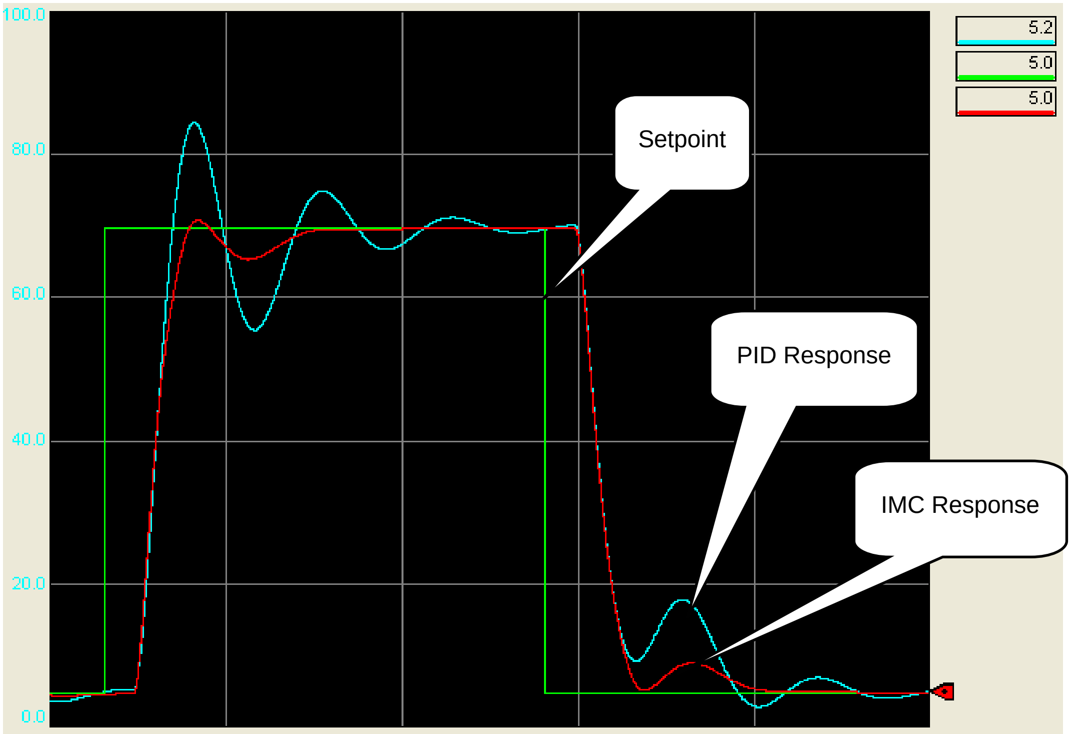
53. Double click on the ***Process_Simulation*** trend and then click on the ***Run*** button 

(top left corner) to start the chart. Depending on the moment that you start trending, the next screen may look different. The setpoint is cycled from 5 to 70 by the simulation every 10 min.



Note that the SPProg values changes (square wave) but that the PIDE.PV and IMC.PV values are steady state, that is because both the PIDE and IMC blocks are in Operator Manual mode and their CE outputs to the process simulation are at zero.

54. Change both the PIDE and IMC blocks to Program Auto mode by setting the ***ProgProgReq*** and ***ProgAutoReq*** parameters to ***'1'***. If you are not familiar with this process ask your instructor.



Observe the trend for at least one cycle.

Notice that the PID loop overshoots when the setpoint jumps up and undershoots when it drops low. The PID loop eventually gets to the setpoint, but the IMC loop does a much better job for this particular process.

If you want to see more details, add the CVEU pens to watch the control actions of the PIDE and IMC instructions.

This concludes lab 1.

Lab 2: Coordinated Control (20 Minutes)

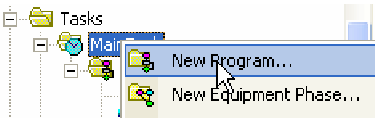
About This Lab

In this lab you will explore the functionality of the Coordinated Control (CC) instruction. This instruction controls a single process variable by manipulating as many as three different outputs. Target values and priorities for each of the outputs are used to optimize your process control. The instruction compares the actual process signal to an internal model for each output. Each output may be in manual control or automatically controlled. The internal models can be manually entered, or the built-in autotuners for each control variable can be used. The setup and configuration parameters, as well as the autotune function, are very similar to the PIDE instruction.

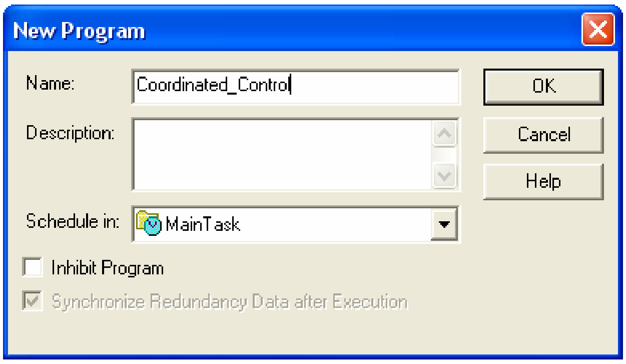
Exploring the Coordinated Control Instruction

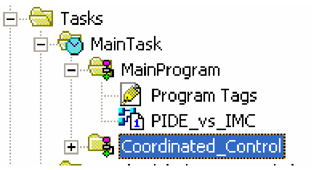
1. Go Offline with your RSLogix 5000 project ***Advanced Process Control***

2. Right click on the ***MainTask*** and select ***New Program***.

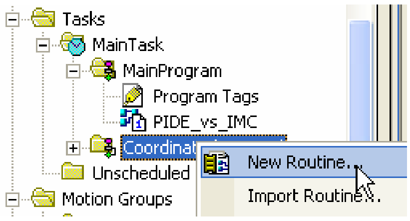


3. Name the new program ***'Coordinated_Control'*** and click ***OK.***

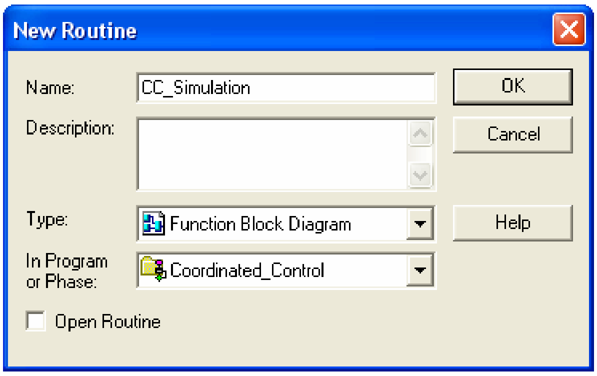


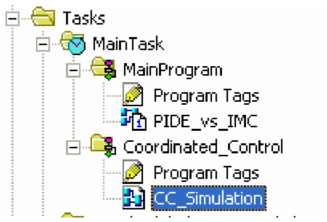


4. Right click on ***Coordinated_Control*** and select ***New Routine***.



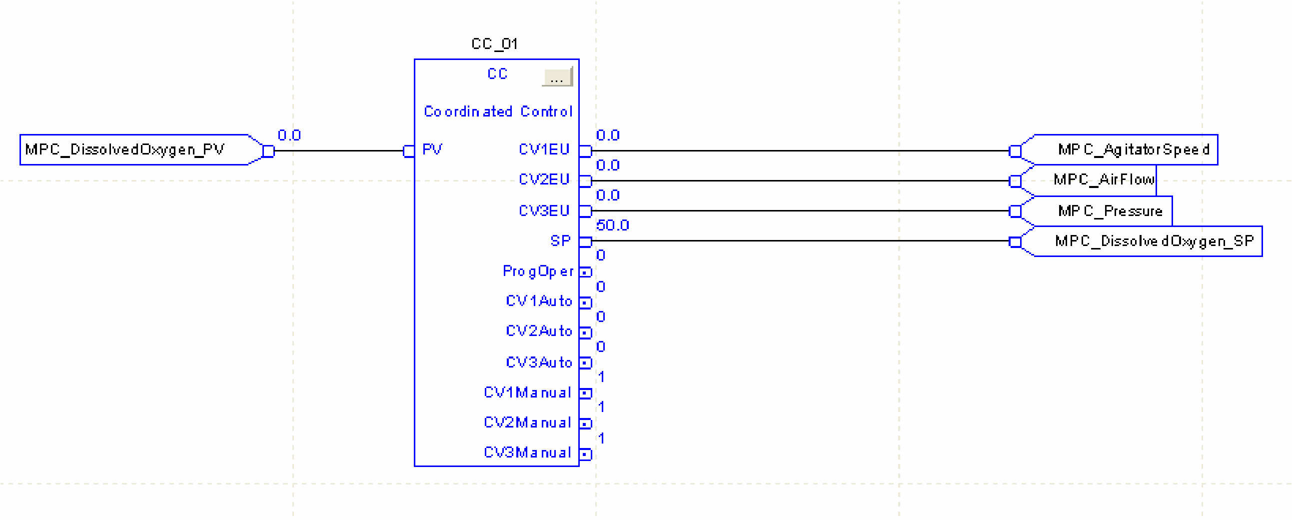
5. Name the new routine ***'CC_Simulation' ,*** select ***Function Block Diagram*** as the Type and click ***OK***.





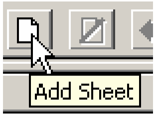
6. Double click on ***CC_Simulation*** and create the following logic:

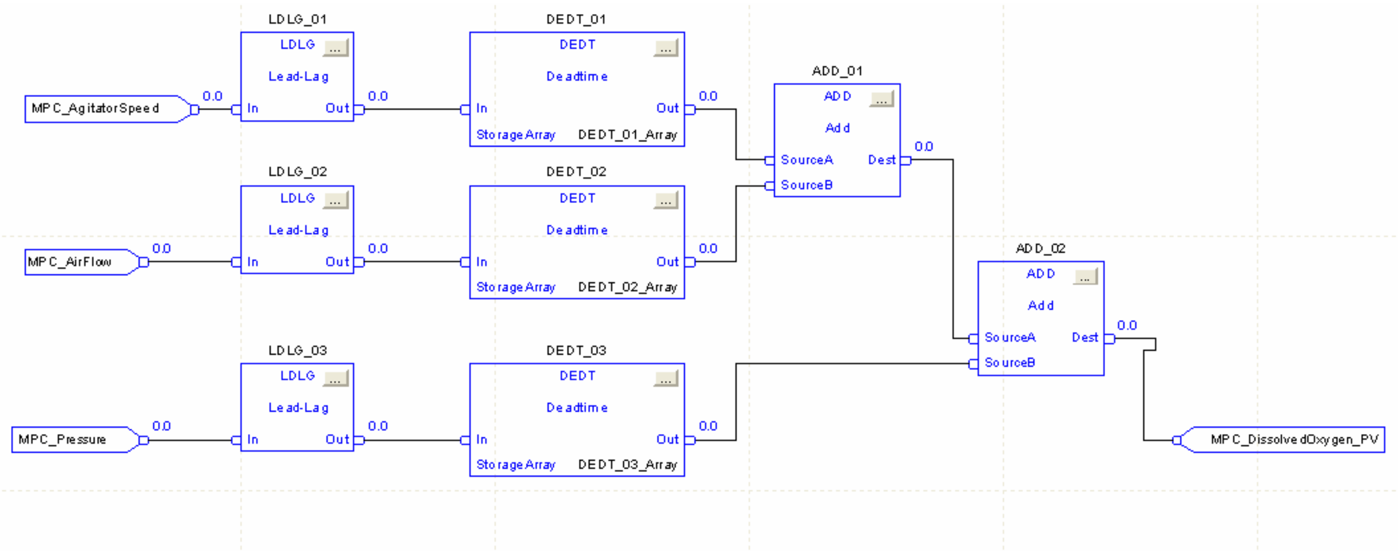
Note. Create all the tags shown as
Data type: **REAL**, Scope: **Coordinated_Control** and Style: **Float**



Scope: Coordinated_Co[Show... Show All

Name	Alias For	Base Tag	Data Type	Style
+-CC_01			CC	
MPC_AgitatorSpeed			REAL	Float
MPC_AirFlow			REAL	Float
MPC_DissolvedOxygen_PV			REAL	Float
MPC_DissolvedOxygen_SP			REAL	Float
MPC_Pressure			REAL	Float

Create a second sheet of logic as shown. 



Details: Input and Output reference tags used on sheet 2 were created on sheet 1

Create the following blocks and arrange as shown above:

LDLG01 – 15 sec Lag

LDLG02 – 30 sec Lag

LDLG03 – 45 sec Lag

DEDT01 – 5 sec Deadtime

DEDT02 – 10 sec Deadtime

DEDT03 – 15 sec Deadtime

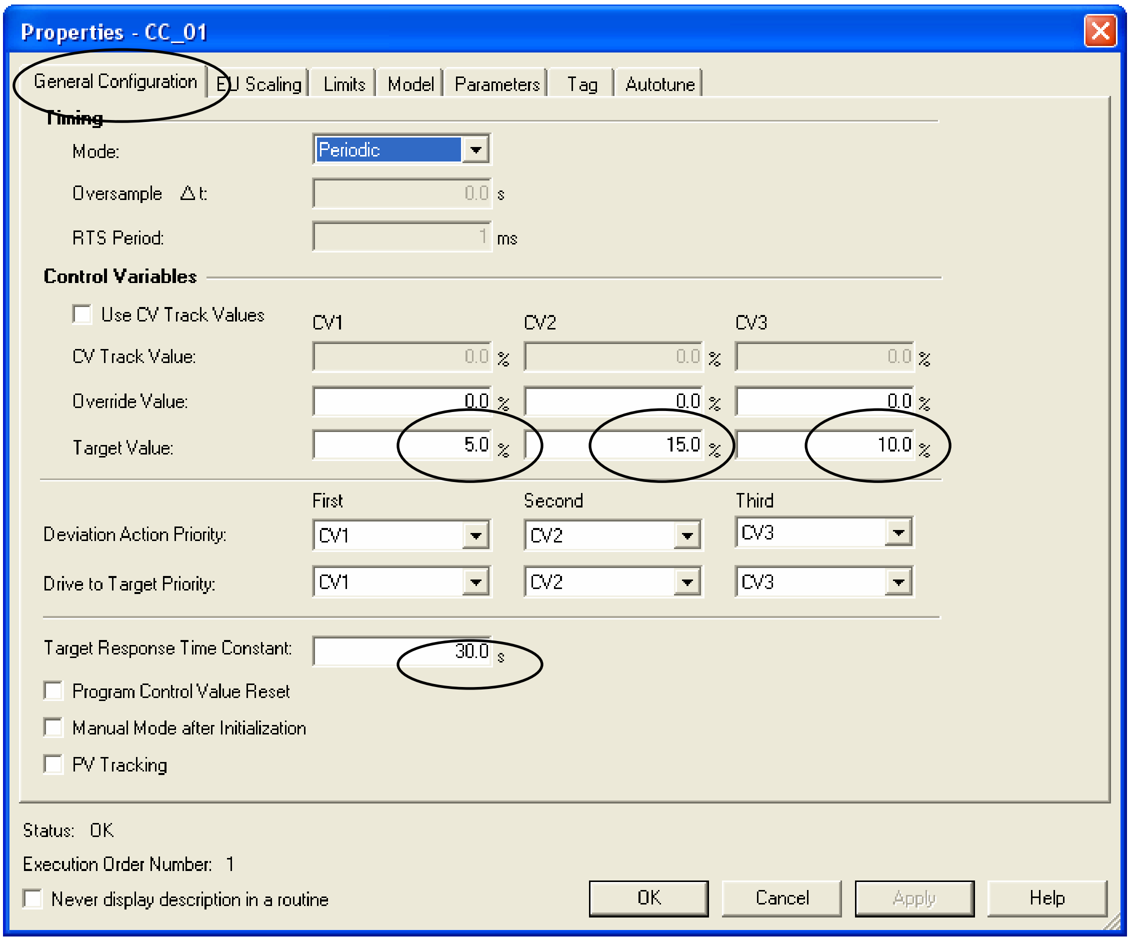
In the storage array tab in the properties of the deadtime blocks, create the following new tags with Data type: REAL (and change Dim0 value to 1000), Scope: Coordinated_Control and Style: Float

DEDT_01_Array REAL[1000]

DEDT_02_Array REAL[1000]

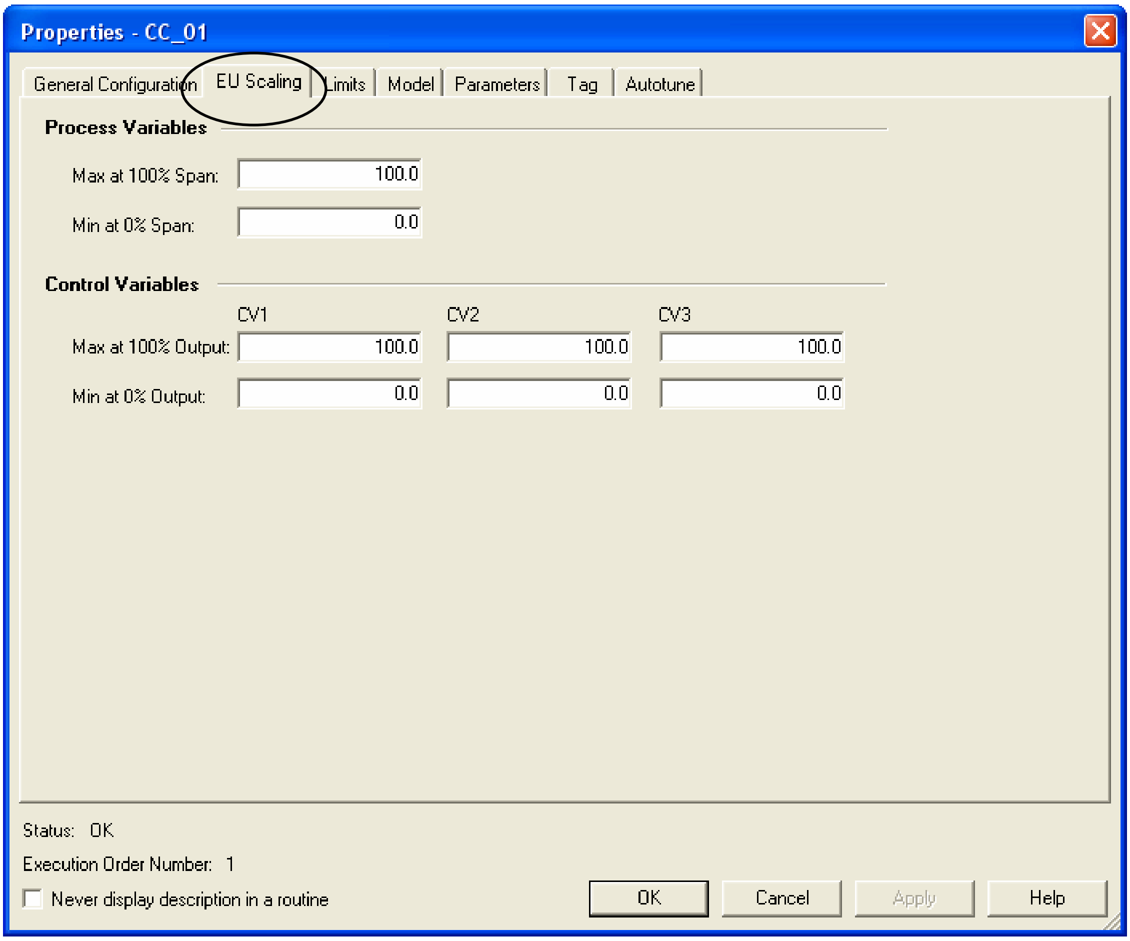
DEDT_03_Array REAL[1000]

7. Set the CC block parameters on the various tabs as shown

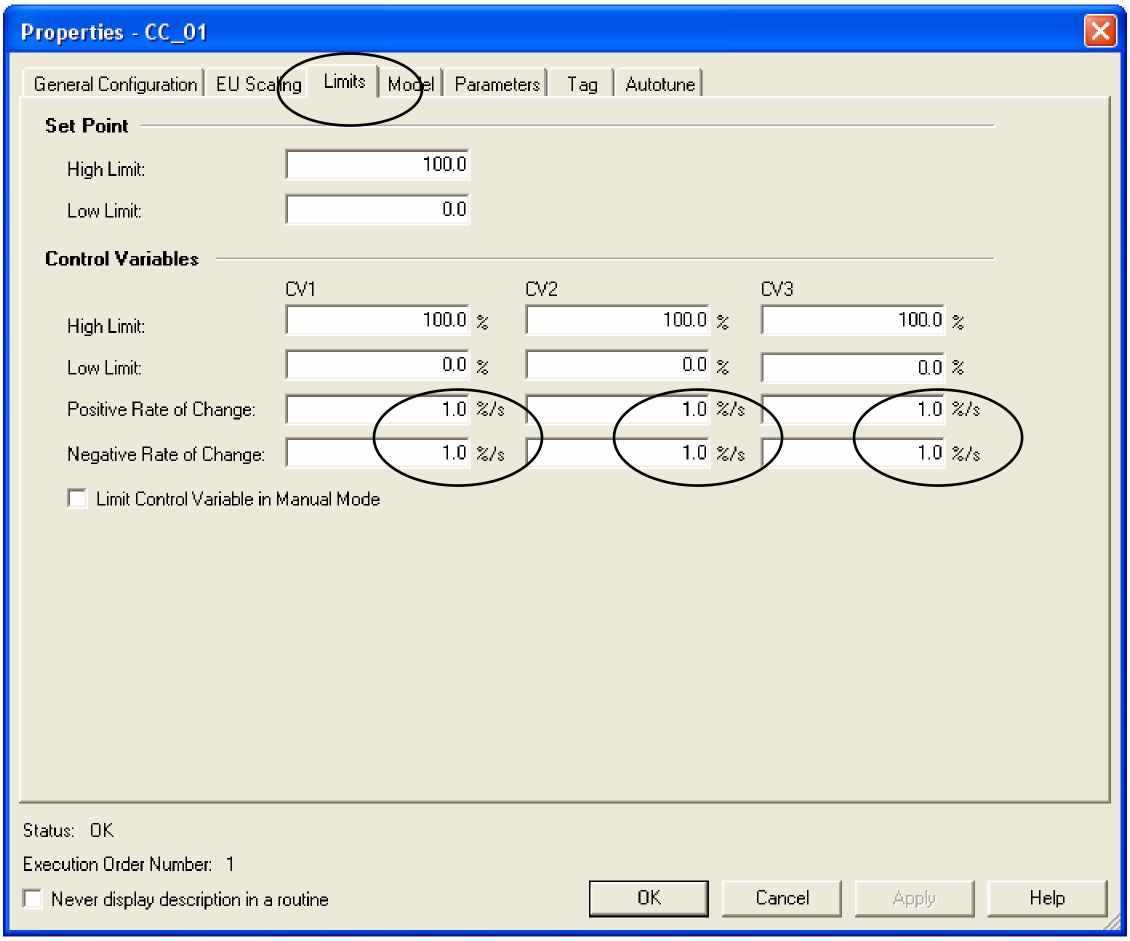


The target values are the desired values for CV1, CV2 and CV3.

Click ***APPLY***

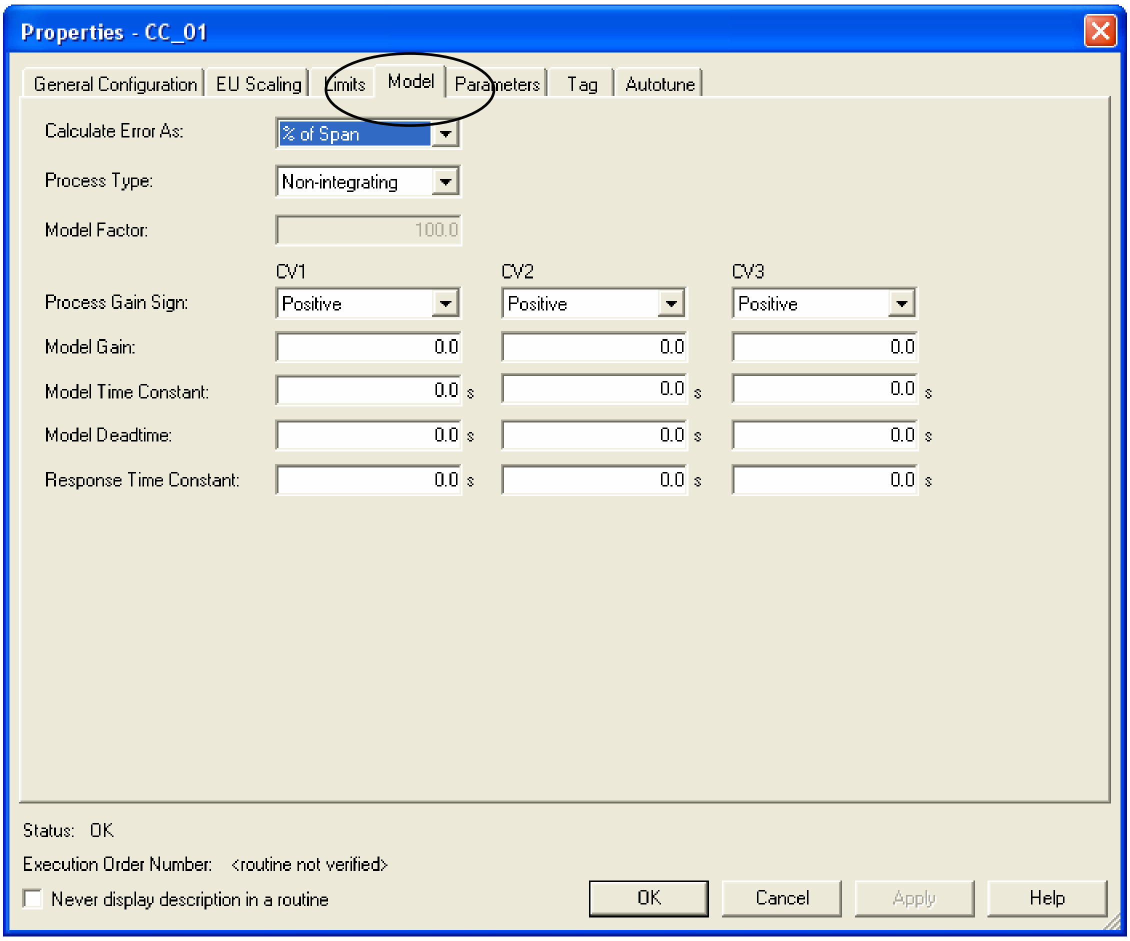


Note: These are to be left at the default settings.

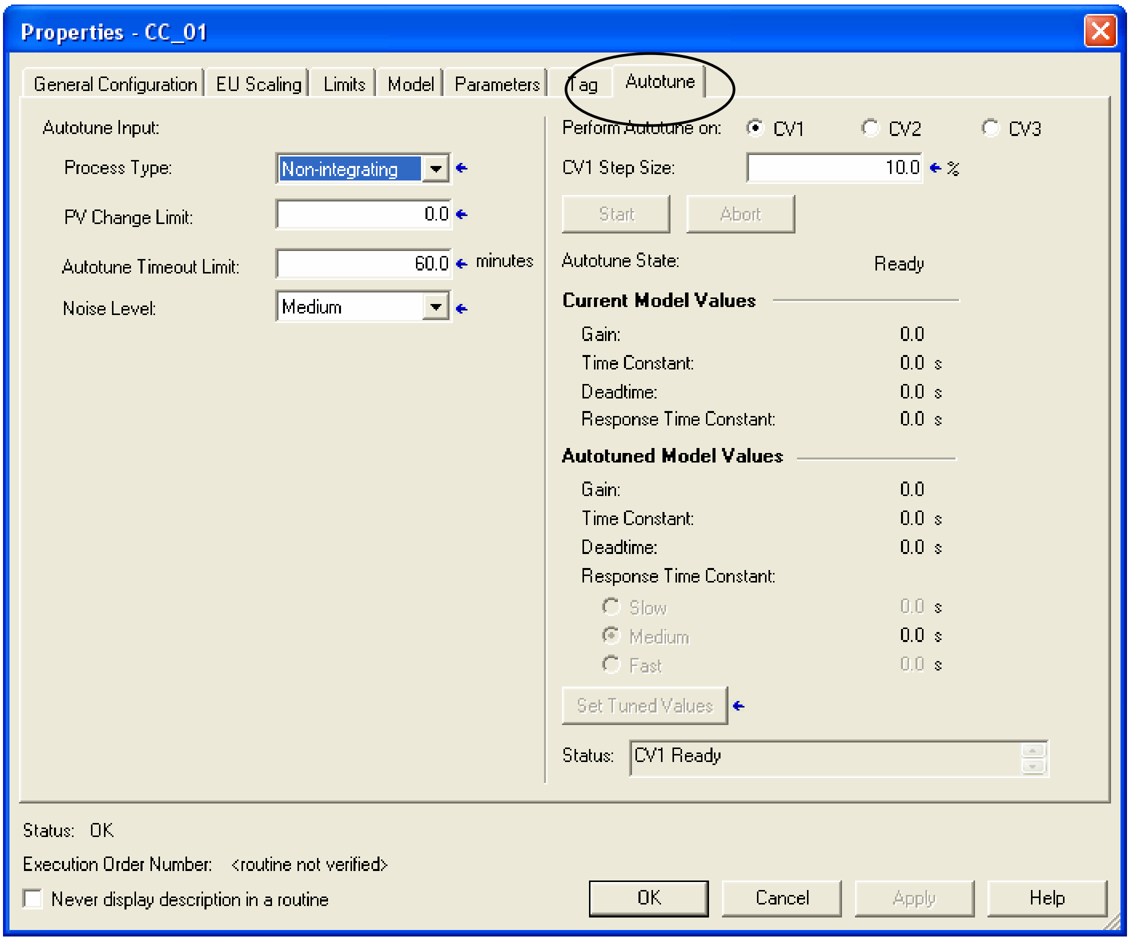


Note: This tab allows you to set limits on Set point and the control variables.

Click ***APPLY***



Note: These are to be left at the default settings. They will be automatically populated after we run the autotune feature. If models were developed using some other method they could be manually entered here.



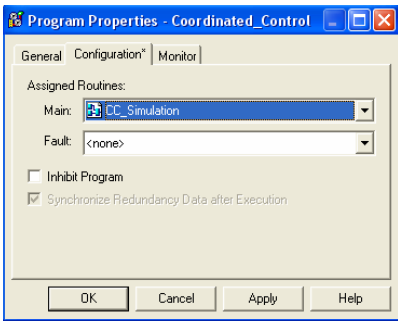
These are the default settings for this tab.

8. Click ***OK*** and then ***Verify*** controller

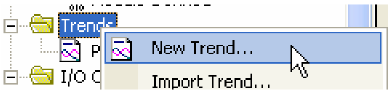
You should get the following Error:

Warning: Coordinated_Control: Program doesn't have associated main routine.

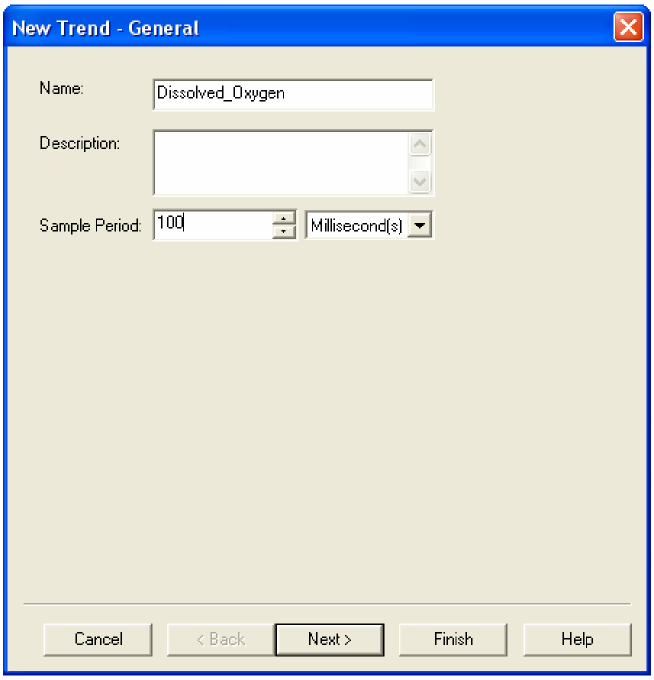
9. Right click on the Coordinated_Control program, select ***Properties, Configuration*** and then set the Main routine to ***CC_Simulation***.



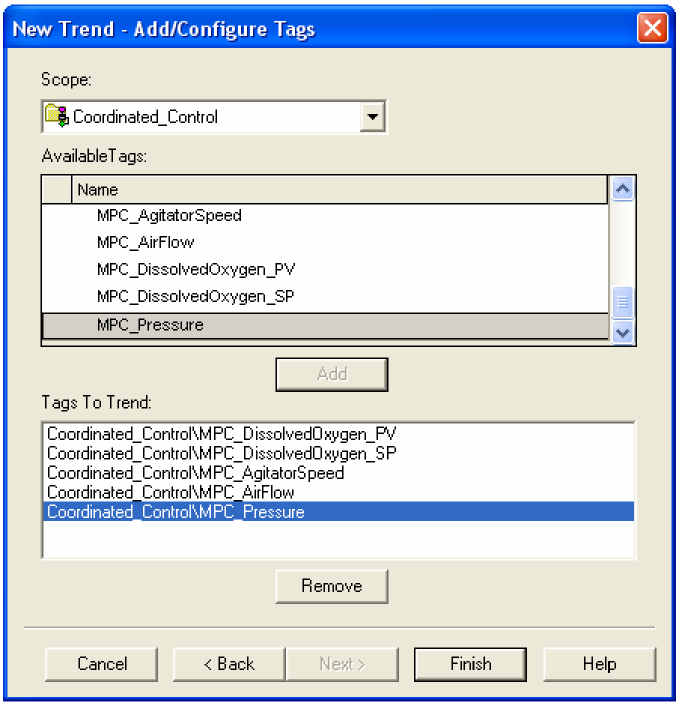
10. Create another trend as follows:

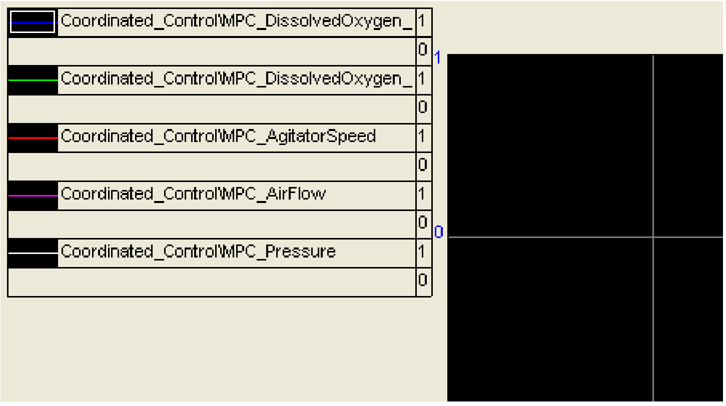


11. Add ***Name*** and ***Sample Period*** and click ***Next.***

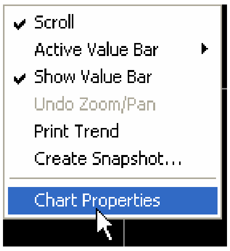


12. Select ***Tags*** as shown and click ***Finish.***

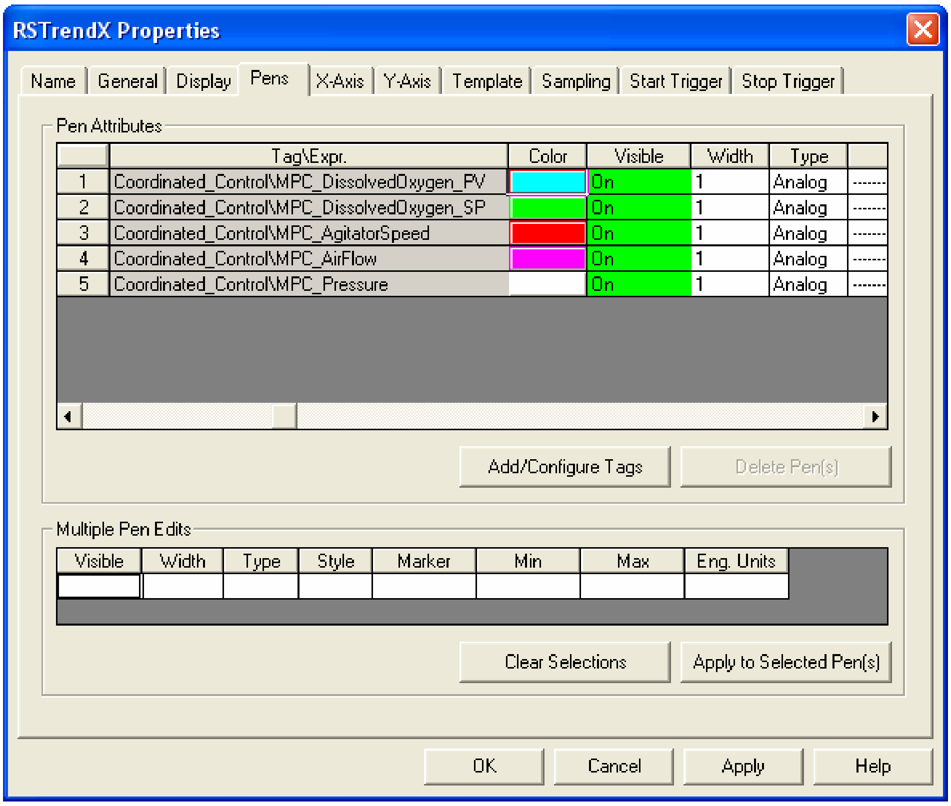




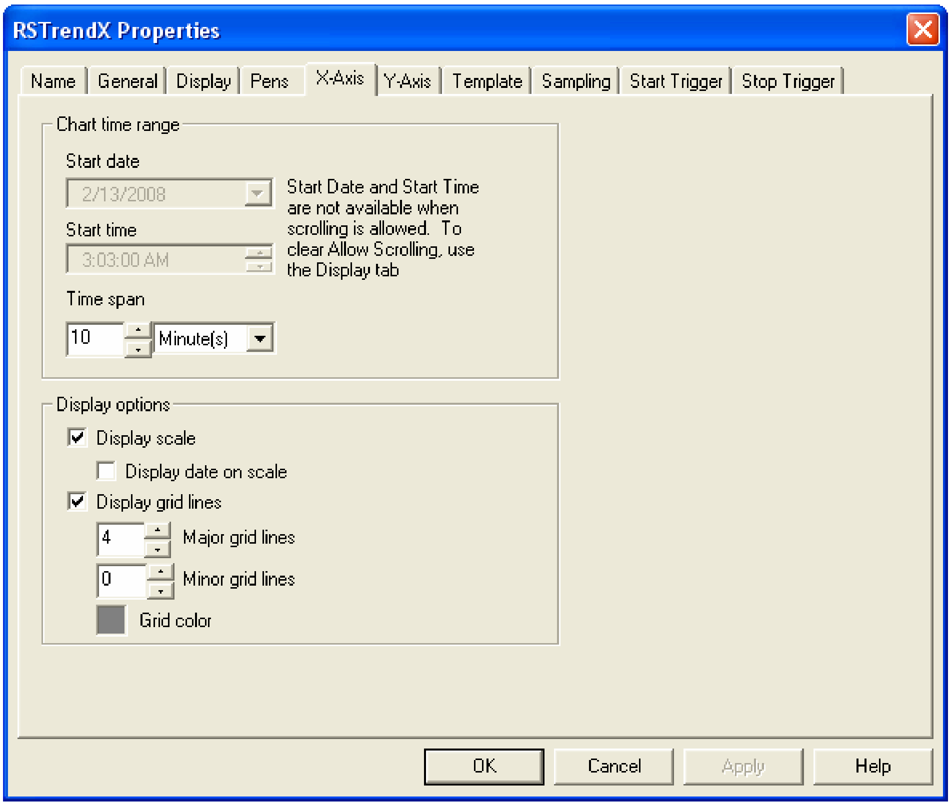
13. Right click on chart to select ***Properties.***



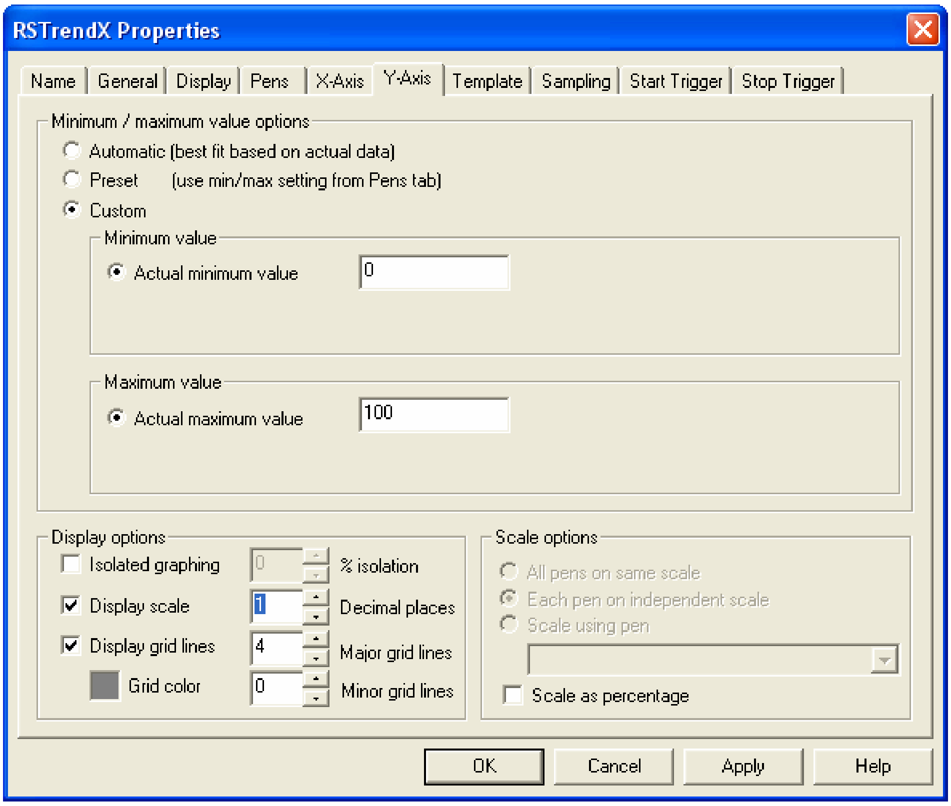
14. Adjust pen colors to your preference



15. Adjust ***Time span*** on the ***X-Axis.***

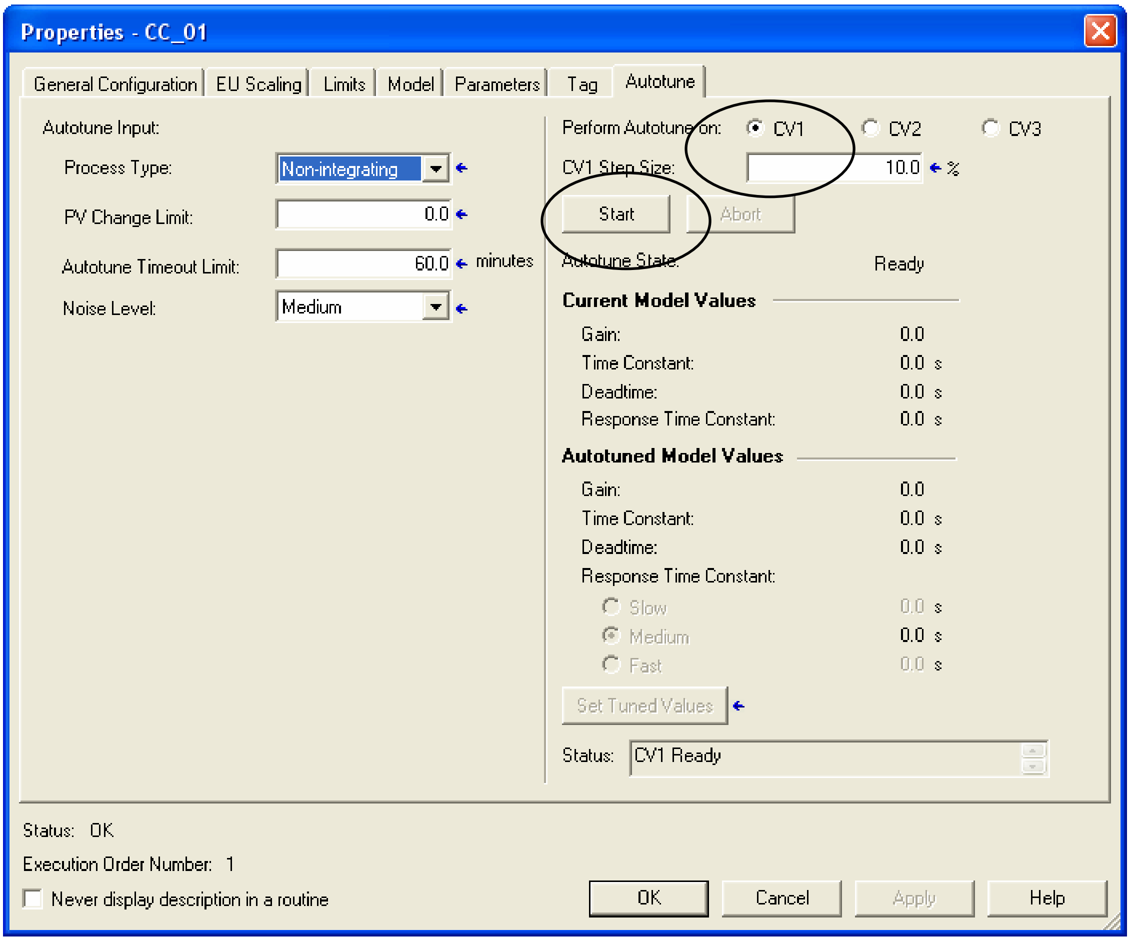


16. Adjust ***Y-Axis*** to ***Custom*** and Display Decimal Places to ***1*** and click ***OK.***

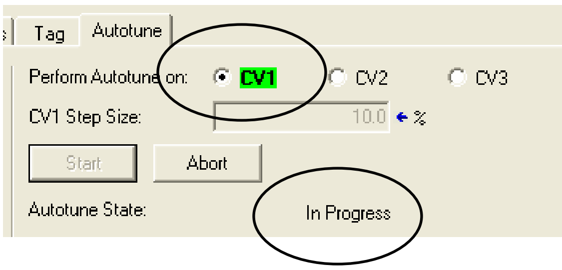


17. Download to the controller and place in ***Run*** mode.

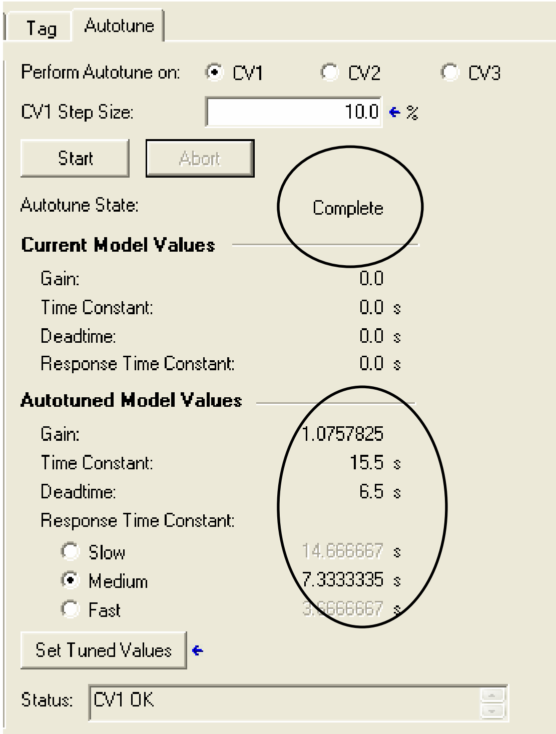
18. Open the CC_01 properties, go to the Autotune tab and click ***Start*** to run the Autotune for CV1



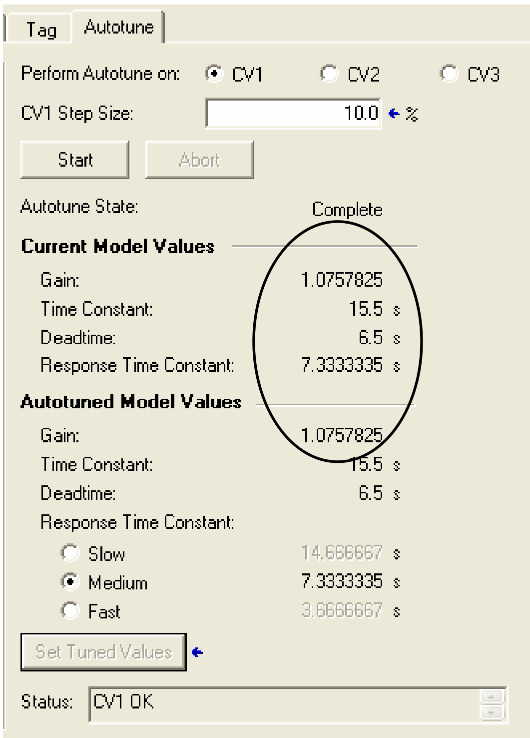
CV1 should be highlighted in green and the Autotune Status should show ***In Progress***



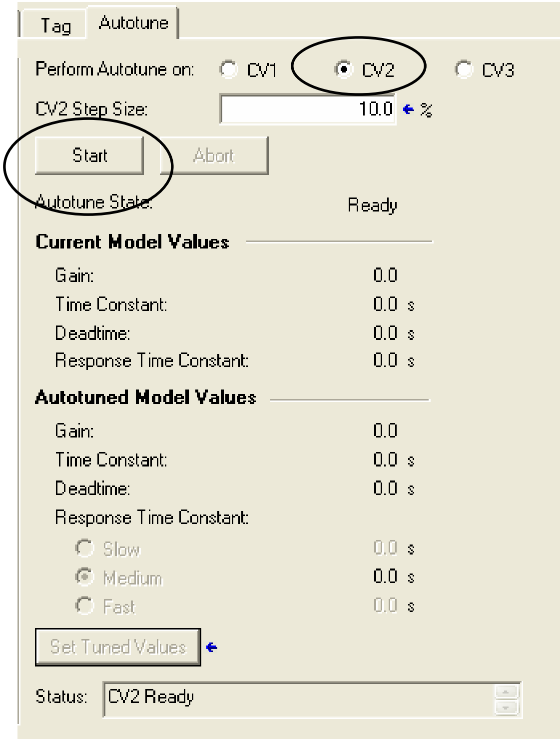
19. When the Autotune is completed the Autotune Status will change to ***Completed*** and the ***Autotuned Model Values*** will update



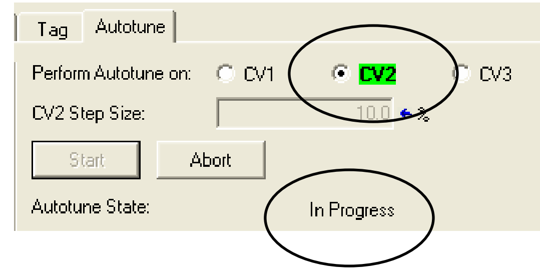
20. The ***Medium*** Response Time Constant is preselected, click ***Set Tuned Values*** ().



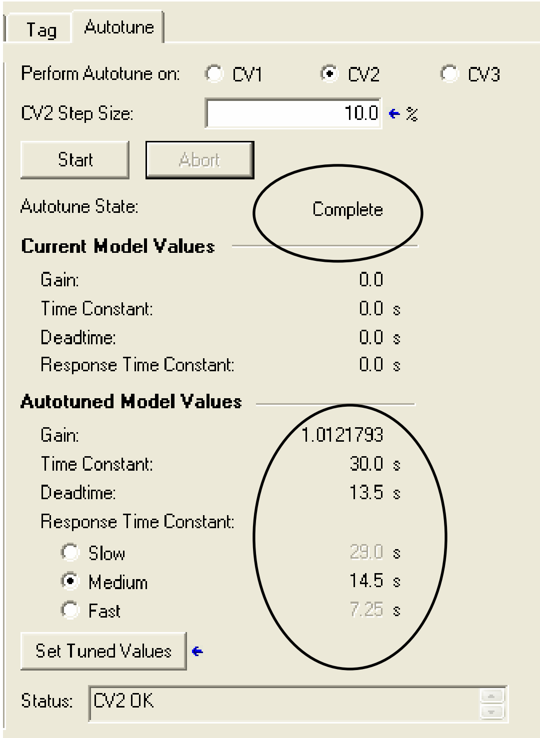
21. Run the autotune for CV2 by selecting Perform Autotune on ***CV2*** and clicking ***Start***.



CV2 will turn green and the status will show ***In Progress***.

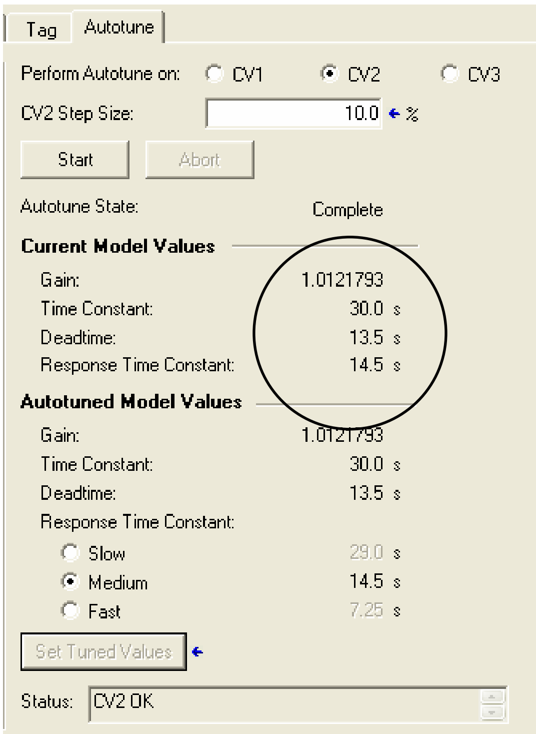


22. When the Autotune is completed the Autotune Status will change to ***Completed*** and the ***Autotuned Model Values*** will update

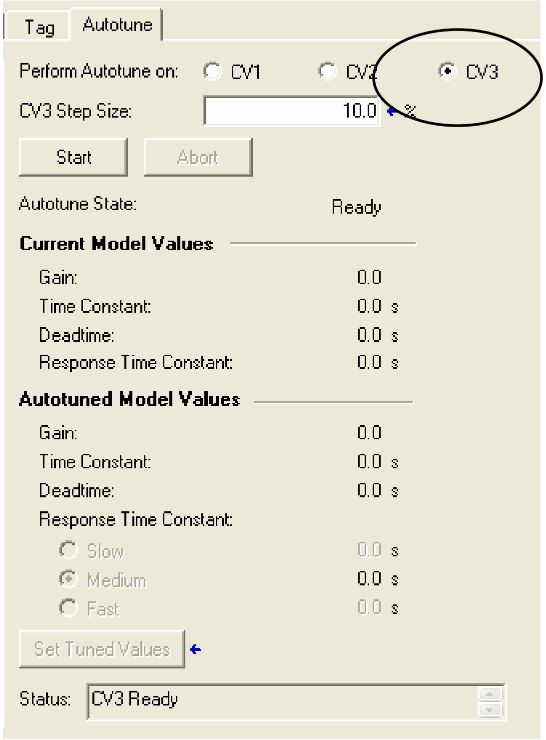


23. The ***Medium*** Response Time Constant is preselected, click ***Set Tuned Values*** (

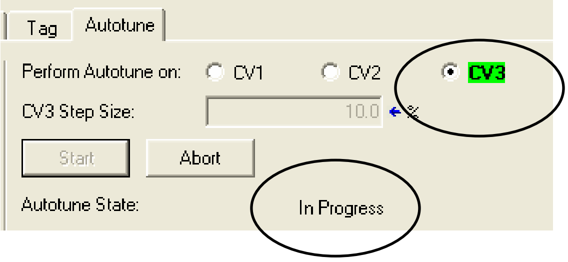
)



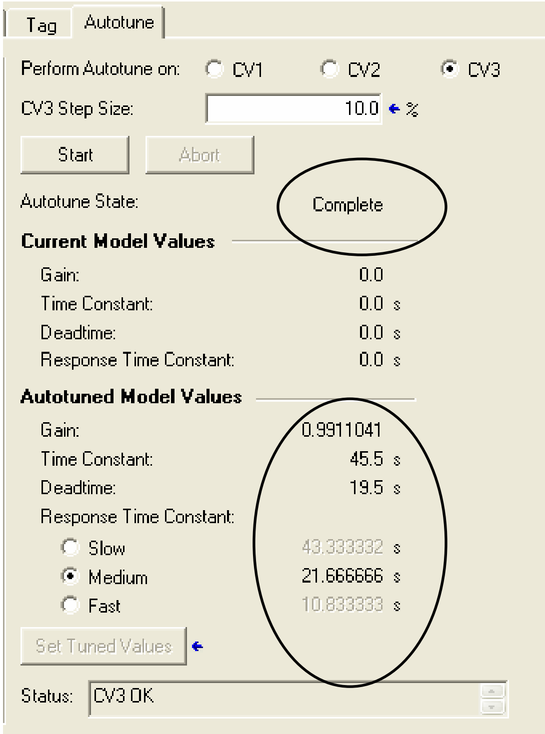
24. Run the autotune for CV3 by selecting Perform Autotune on ***CV3*** and clicking ***Start.***



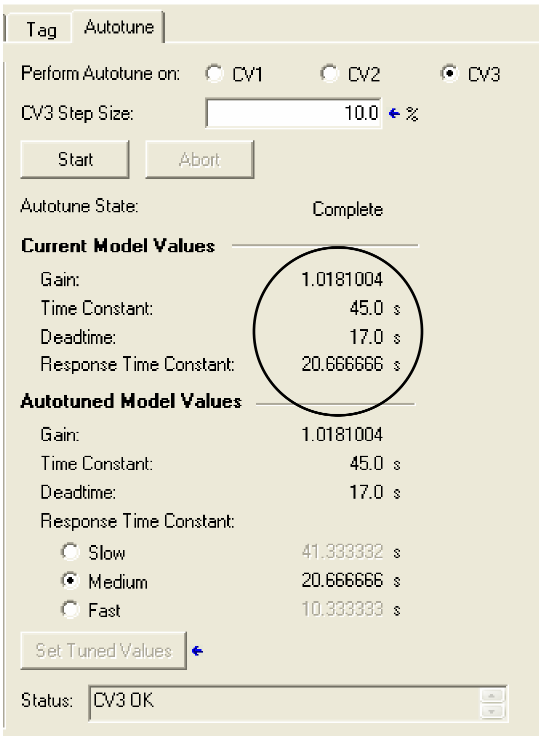
CV3 will turn green and the status will show ***In Progress***.



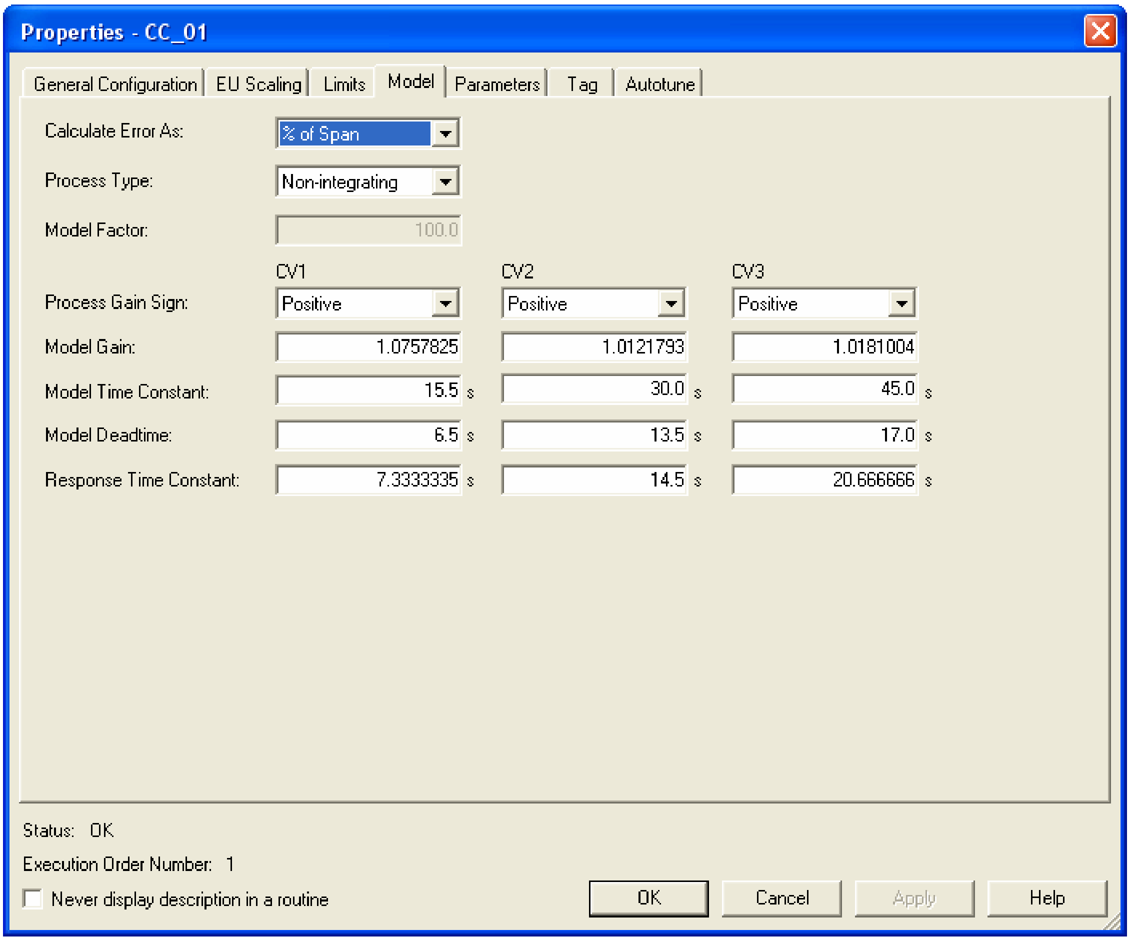
25. When the Autotune is completed the Autotune Status will change to ***Completed*** and the ***Autotuned Model Values*** will update.



26. The ***Medium*** Response Time Constant is preselected, click ***Set Tuned Values*** (Set Tuned Values).



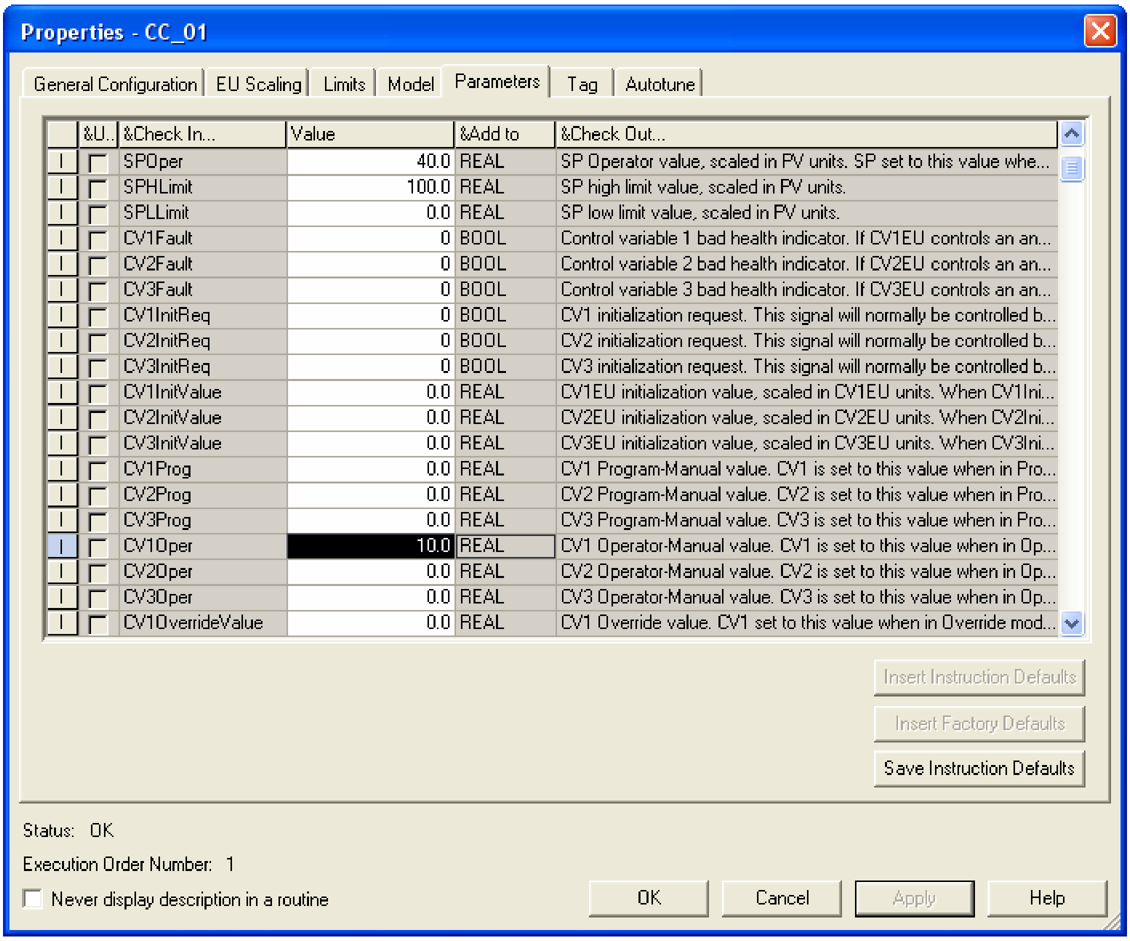
27. Click on the ***Model*** tab



The internal models being used for each of the outputs is shown here.

28. Open the CC_Simulation trend and click ***Run*** to begin trending.

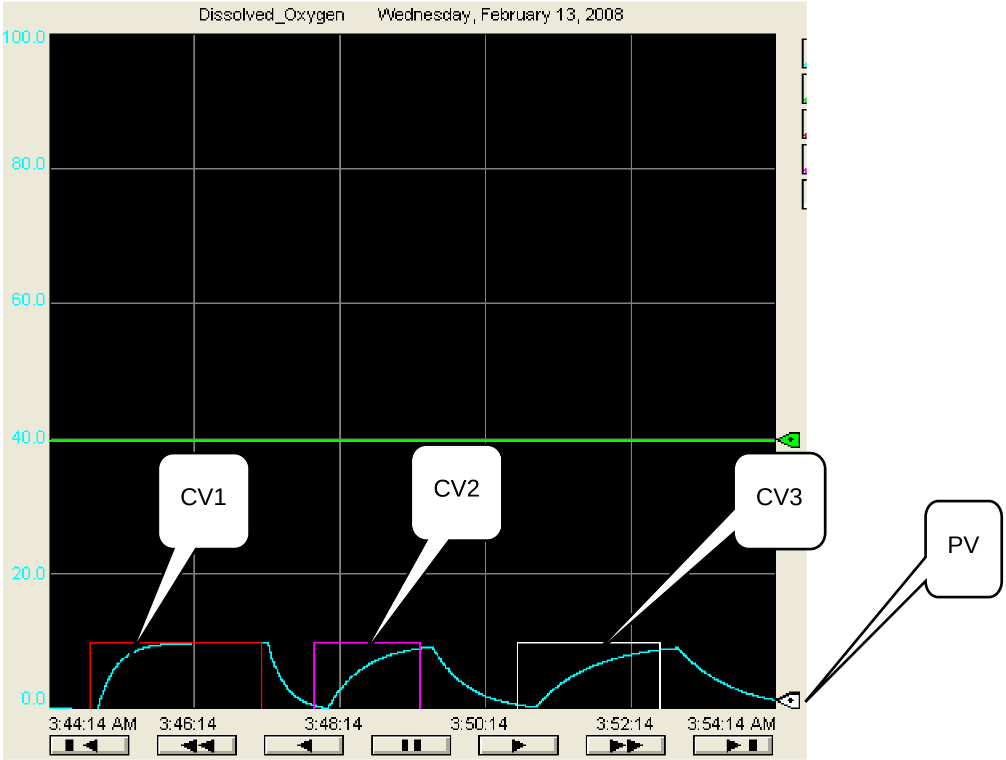
29. The state of the CC control is Operator – Manual. Change SPOper to ***'40'*** and CV1Oper to ***'10',*** click ***OK*** and observe the trend.



Always click ***Apply*** after each change.

Note: The trend will show a response to the CV1Oper change since the CC control is in manual. The SPOper change will not affect anything at this time since we are not in Auto control.

30. Put CV1Oper back to zero, let the PV settle back to zero and then make CV2Oper 10. Repeat for CV3Oper. Observe the trends.

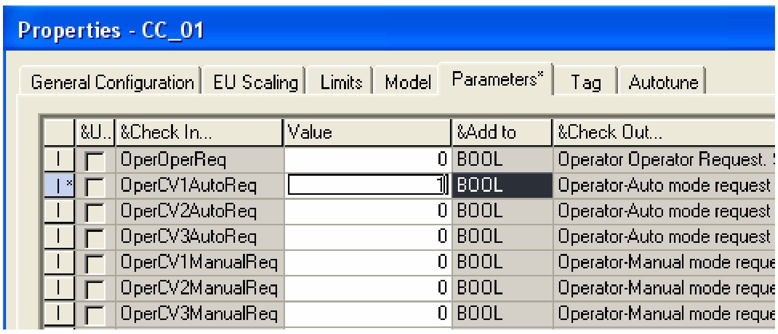


Note that the dissolved oxygen levels respond to each of the outputs being moved from 0 to 10. The response is slightly different for each of the CV actions.

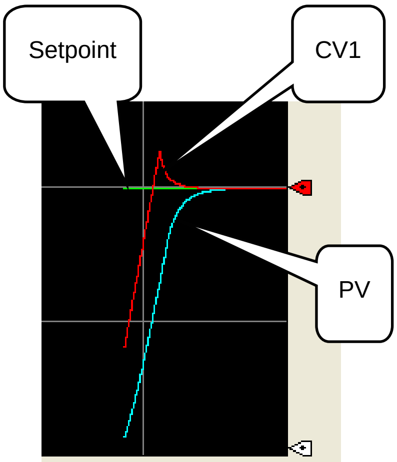
Put CV3Oper value back to zero and click ***Apply***

31. Now put CV1 into auto by entering a ***'1'*** into OperCV1AutoReq and then clicking ***Apply***.

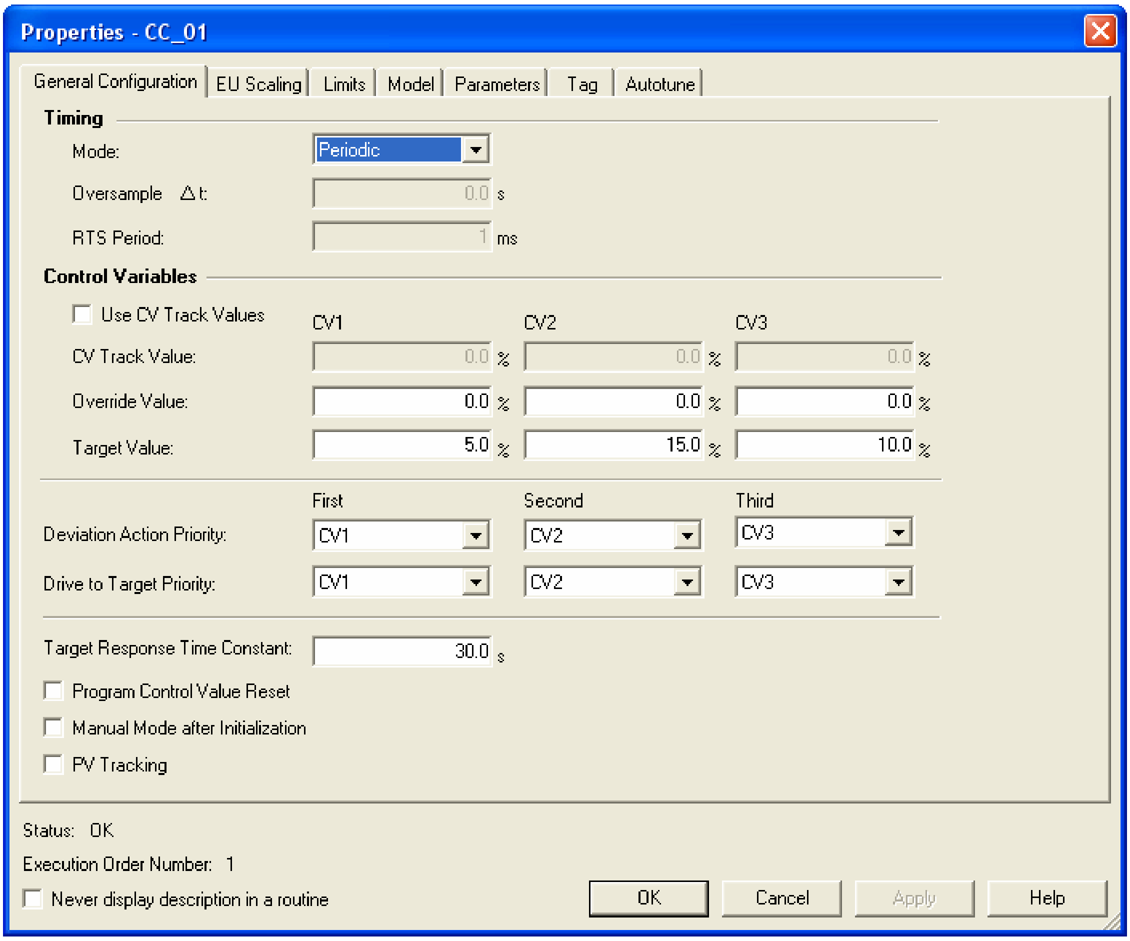
IMPORTANT: the OperCV1AutoReq is a front edge request, meaning as soon as you Apply it triggers back to 0



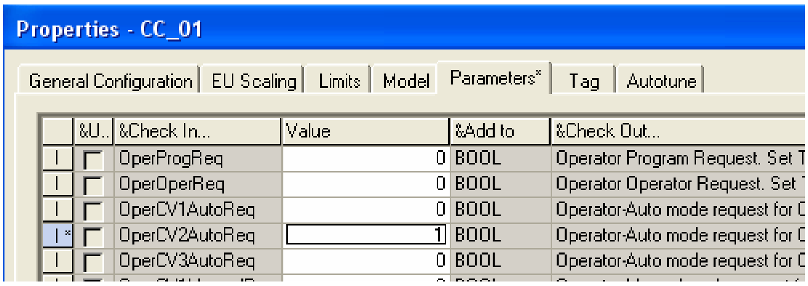
Note: The '1' is a one-shot entry and will return to a '0'.



Note that AgitatorSpeed (CV1EU) has been adjusted to control the DissolvedOxygen to the setpoint of 40. The Target value for CV1 is 5, but since the only control available is CV1 (CV2 and CV3 are still in manual), CV1 is adjusted as needed to reach the setpoint.

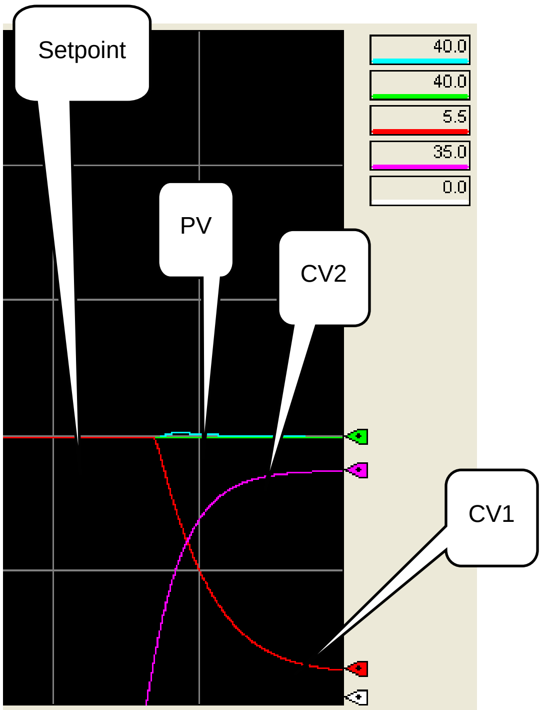


32. Put ***CV2*** into auto too.

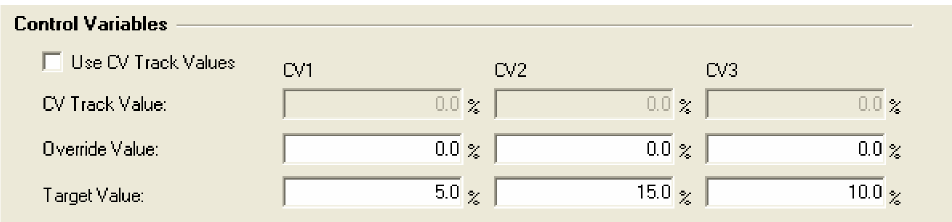

Properties - CC_01

General Configuration | EU Scaling | Limits | Model | Parameters* | Tag | Autotune

	&U..	&Check In...	Value	&Add to	&Check Out...
I	☐	OperProgReq	0	BOOL	Operator Program Request. Set T
I	☐	OperOperReq	0	BOOL	Operator Operator Request. Set
I	☐	OperCV1AutoReq	0	BOOL	Operator-Auto mode request for C
I *	☐	OperCV2AutoReq	1	BOOL	Operator-Auto mode request for C
I	☐	OperCV3AutoReq	0	BOOL	Operator-Auto mode request for C



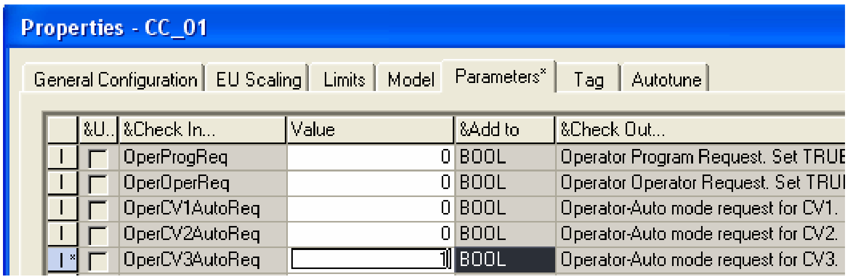
Note that AgitatorSpeed (CV1EU) has been reduced to its target value (5) and AirFlow (CV2EU) adjusted to control the DissolvedOxygen to the setpoint of 40. The Target value for CV2 is 15, but since the only control available is CV1 and CV2 (CV3 are still in manual), and CV1 is the first priority to stay at target value, CV2 is adjusted as needed to reach the setpoint.

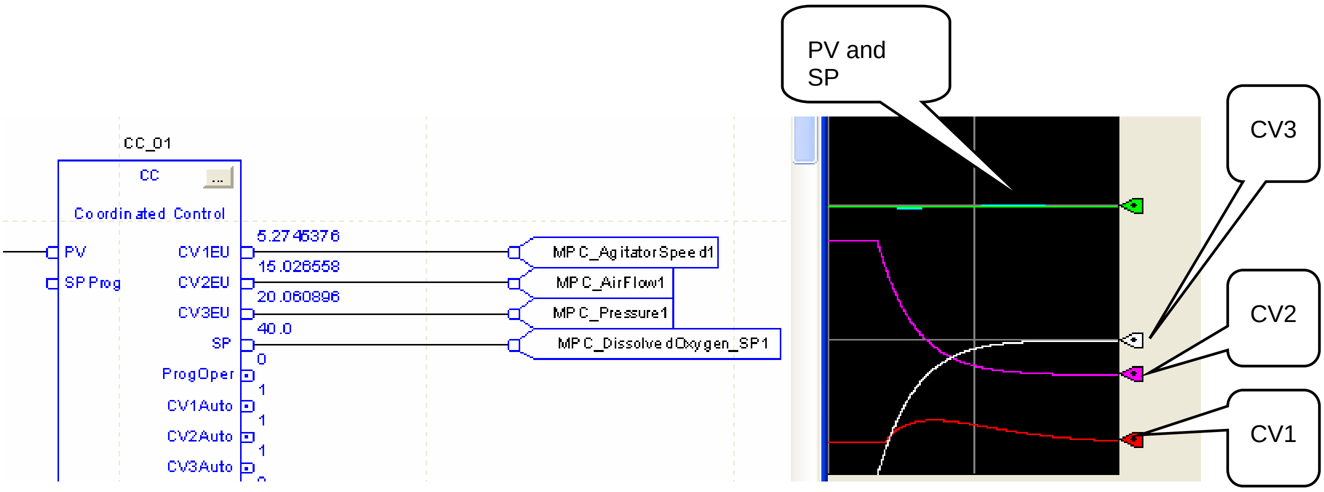

Control Variables

☐ Use CV Track Values

	CV1	CV2	CV3
CV Track Value:	0.0 %	0.0 %	0.0 %
Override Value:	0.0 %	0.0 %	0.0 %
Target Value:	5.0 %	15.0 %	10.0 %

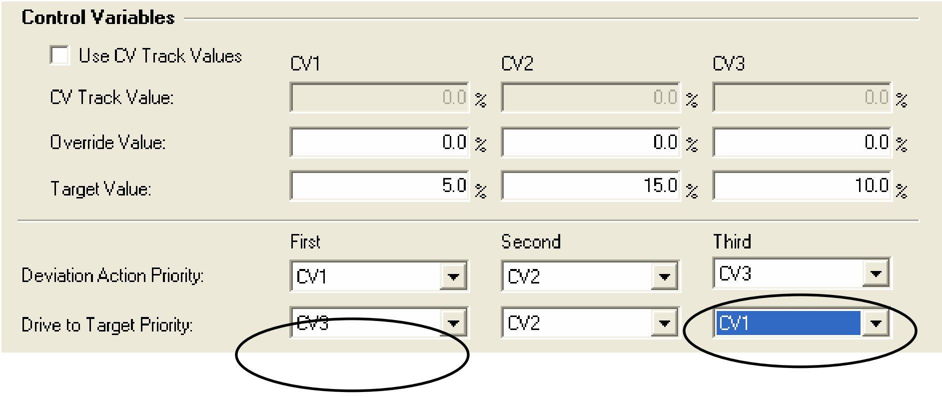
33. Put ***CV3*** into Auto.

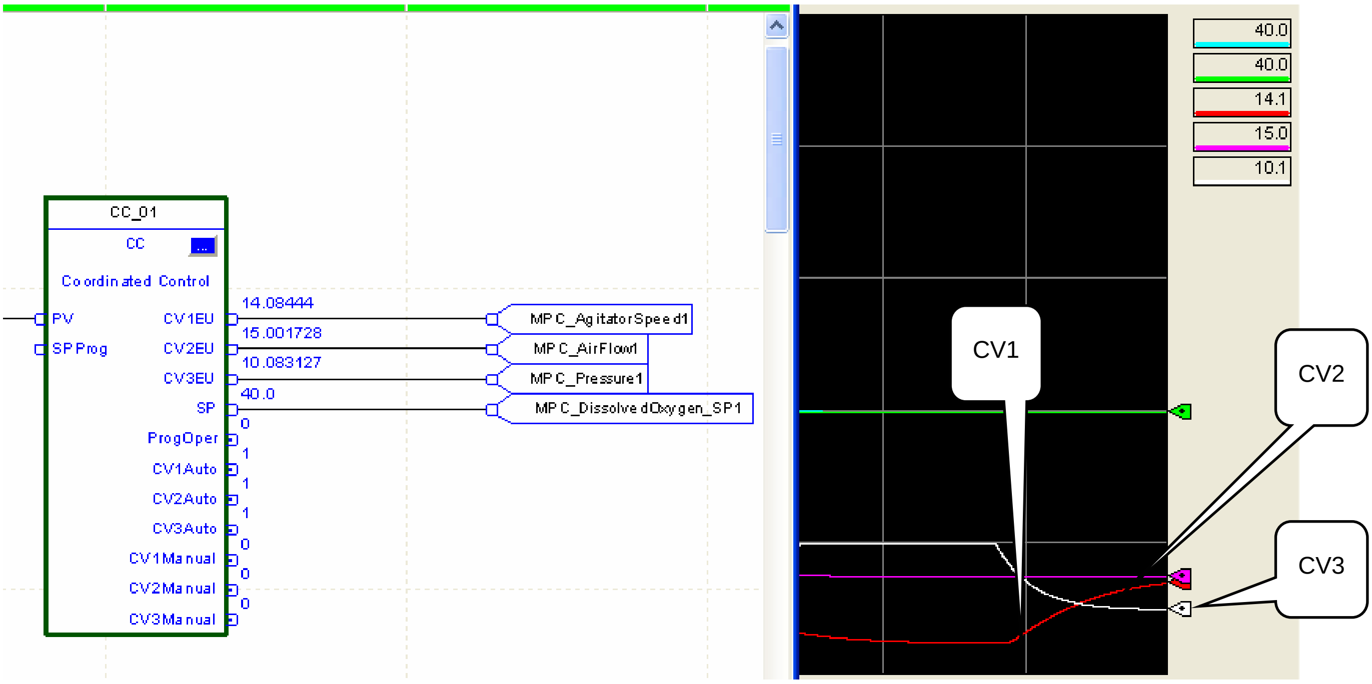




Note that AgitatorSpeed (CV1EU) has been reduced to its target value (5), AirFlow (CV2EU) has been reduced to its target value (15) and Pressure has been adjusted to control the DissolvedOxygen to the setpoint of 40. The Target value for CV3 is 10, but since CV3 is the lowest priority it is adjusted as needed to reach the setpoint.

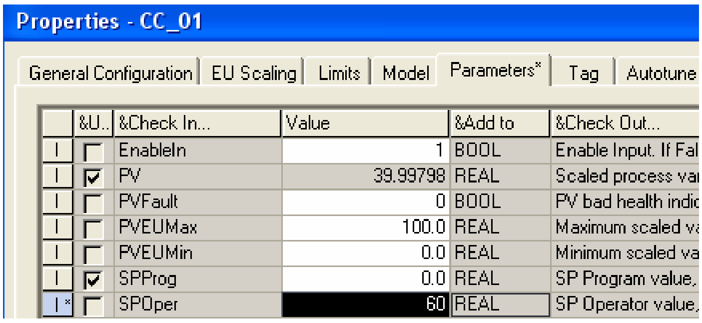
34. In the General Tab change the ***Drive to Target Priorities*** and observe the changes in the three outputs.





Notice that CV2 and CV3 (now having the highest priorities) are being brought at their target values and CV1 is being adjusted to meet the setpoint.

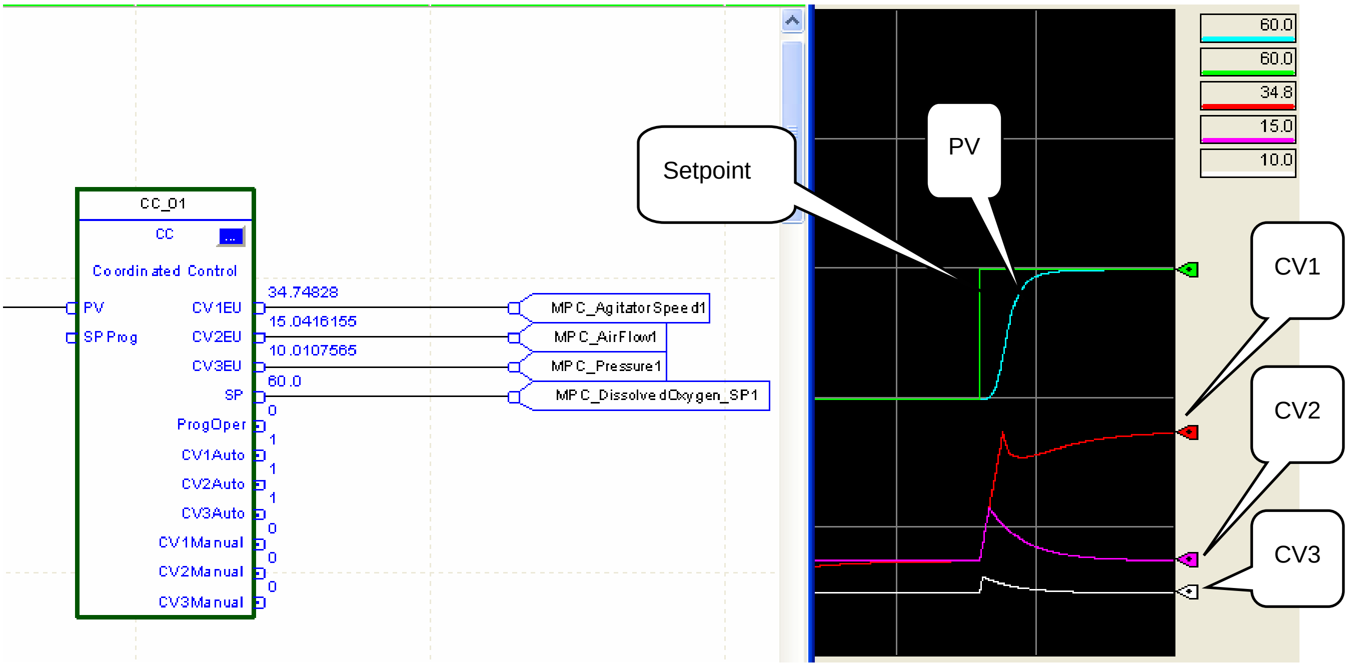
35. Leaving our priority setting as they were, change the Operator SP to 60.



Properties - CC_01

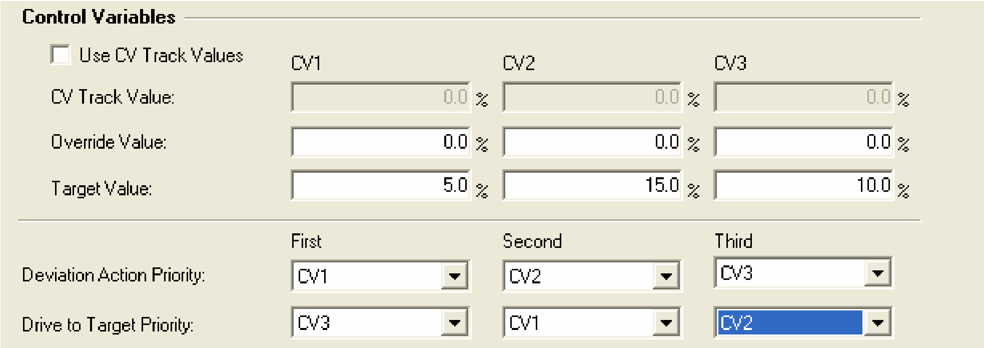
General Configuration | EU Scaling | Limits | Model | Parameters* | Tag | Autotune

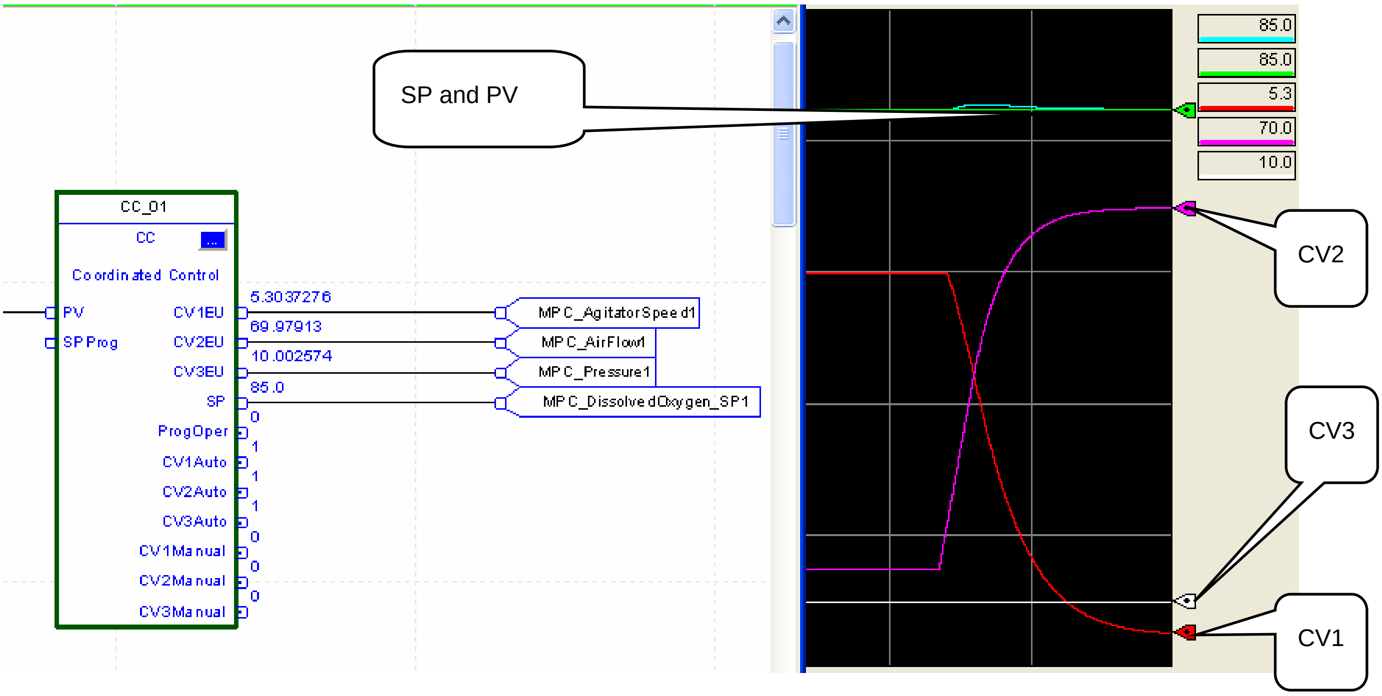
	&U..	&Check In...	Value	&Add to	&Check Out...
I	☐	EnableIn	1	BOOL	Enable Input. If Fal
I	☑	PV	39.99798	REAL	Scaled process val
I	☐	PVFault	0	BOOL	PV bad health indic
I	☐	PVEUMax	100.0	REAL	Maximum scaled va
I	☐	PVEUMin	0.0	REAL	Minimum scaled va
I	☑	SPProg	0.0	REAL	SP Program value,
I *	☐	SPOper	60	REAL	SP Operator value,



CV1 is adjusted to meet the setpoint. CV2 or CV3 did initially move from their target values, but eventually went back to their target values.

36. Changes the setpoint to 85 and priorities as shown. Observe the reaction of the control.





End of lab 2

CONGRATULATIONS!

YOU HAVE COMPLETED THE ADVANCED PROCESS CONTROL HANDS-ON LAB!

You can find this workbook on www.rockwellautomation.com/events/au