

Productivity 3000™ Monthly Instruction Showcase

May 21, 2010

PID Instruction Box

This month's Productivity 3000 Instruction Showcase will feature the versatile and time-saving PID Loop (PID) instruction box. I will cover the basics of PID loops and where to enter the pertinent information in order to set up your loop. The biggest advantage that the Productivity Software Suite has over the DirectSOFT software package is its ability to use tags and do away with cumbersome v-memory locations. This is especially helpful when setting up and tuning PID loops. No longer do you have to keep track of tables for each loop and ramp/soak. You can choose whatever tag you wish as long as the correct data type is used.

The screenshot shows the Productivity 3000 software interface. On the left is a list of instructions categorized into Logical Words, Drum Sequencers, Math Functions, Program Control, String Functions, and System Functions. The 'PID' instruction is highlighted in the list. A callout box shows a detailed view of the 'PID' instruction box with the following configuration parameters:

PID	
Enable	Loop Name
Manual/Auto	Set Point
	Process Variable
	Input Range Max
	Input Range Min
	Gain (Proportional)
	Reset (Integral) Time
	Rate (Derivative) Time
	Output Type
	Process Output
	Output Range Max
	Output Range Min

At the top right, a block diagram illustrates the PID control loop. It shows a Setpoint entering a summing junction (Σ). The error signal is fed into three parallel paths: a Proportional (P) block with gain K_p , an Integral (I) block with gain K_i and integration time τ , and a Derivative (D) block with gain K_d and derivative time τ_d . The outputs of these three blocks are summed at another junction (Σ) and fed into a Process block, which produces the final Output.

At the bottom right, a photograph shows the physical hardware rack containing the Productivity 3000 modules.

PID Basics:

- PID stands for Proportional-Integral-Derivative and is defined as a control loop algorithm useful for precisely controlling a system such as a tank, heater, or refrigeration device.
- PID loops are used to correct the error between a measured variable (PV or Process Variable) and the desired value (Setpoint). The CPU uses this error to output a corrective action that can adjust the process rapidly to minimize the error.

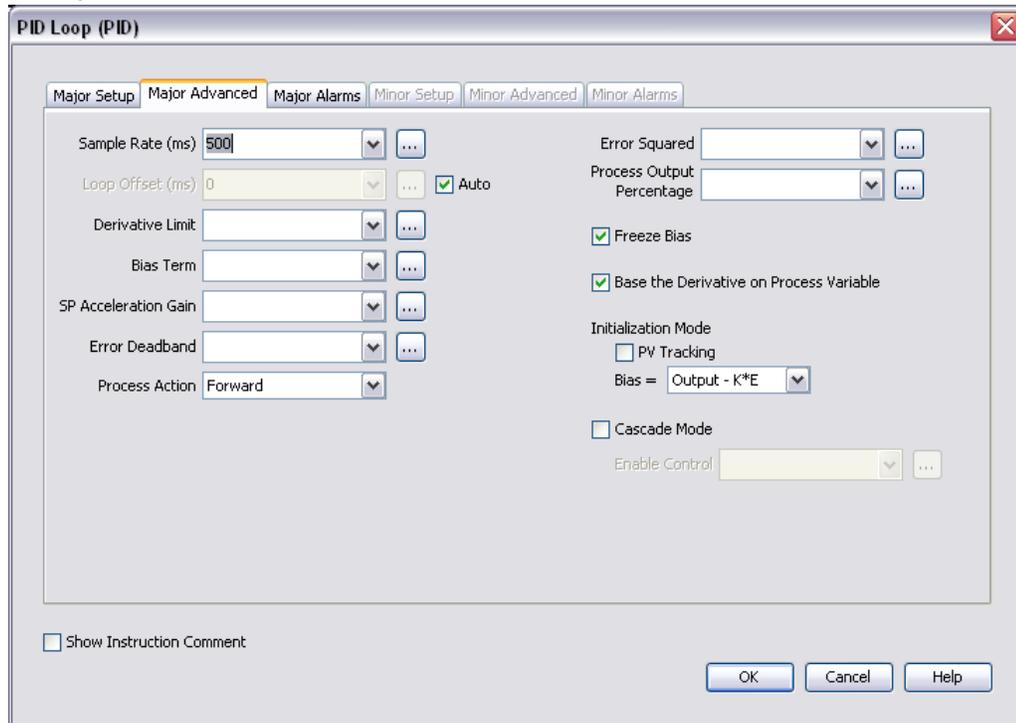
Key Features:

- **Saves Time!** You never have to look up confusing memory tables to find where your variables are. This feature is especially helpful if you have multiple loops and/or cascading loops.
- Productivity Suite PID instruction setup is only 3 pages versus the 7 pages of configuration necessary in DirectSOFT
- **Unlimited number of PID Loops!** You are only limited by the 50 megabytes of CPU memory
- The Ramp/Soak (RPS) feature is now a separate instruction box. Instead of looking through the manual to find which V-memory location each point is stored you can use the same fill-in-the-blanks style form to program your profile.

Tab 1. Major Setup

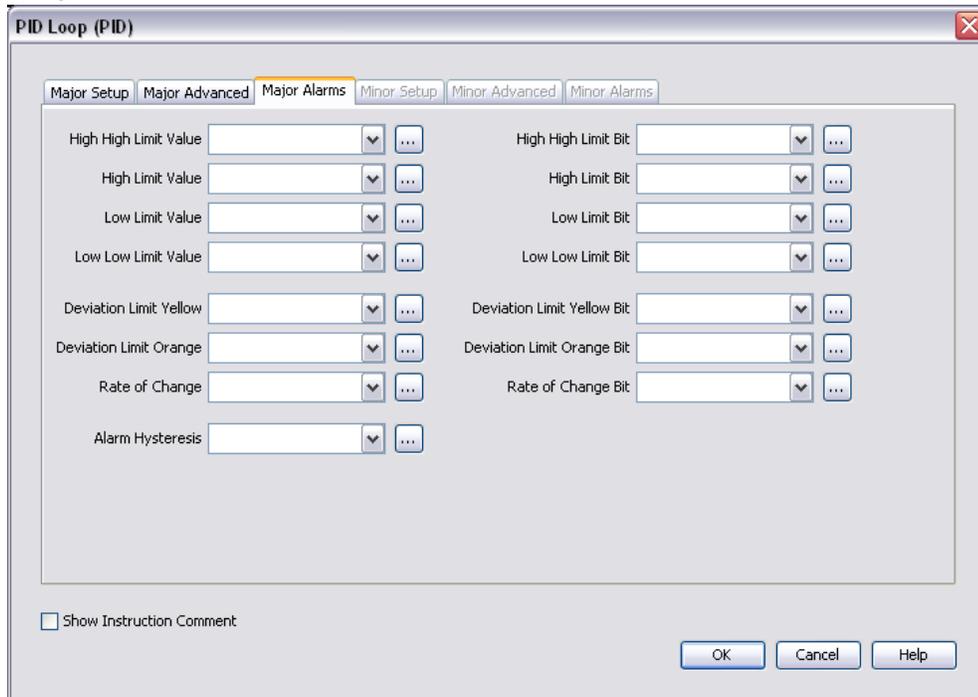
- Use this page to set up the basic tags needed for a single PID loop. All blanks can be constants or variable tags besides the Process Output and the process control inputs (gain, integral time, and derivative time)
- Easily scale inputs and outputs by using the Range Max and Range min blanks.
- Output limits can be conveniently set here to prevent costly hardware damage that may occur if your output goes out of range.

Tab 2. Major Advanced



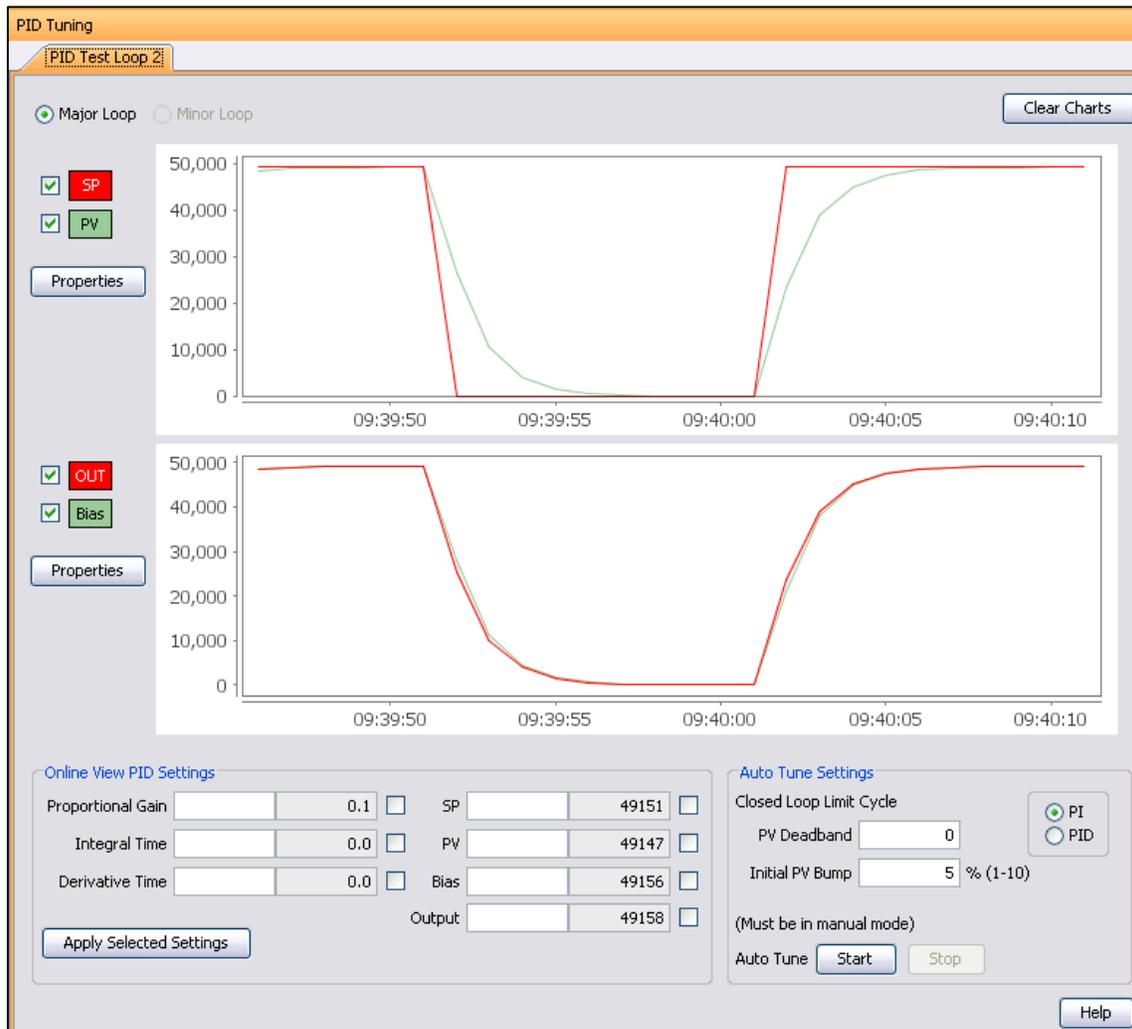
- Here you can configure the advanced settings that control how the loop will perform.
- By checking the Cascade Mode box you can easily add a cascade loop to your system. This will automatically enable the three gray tabs for the cascade PID loop: Minor Setup, Minor Advanced, and Minor Alarms.
- The Productivity Programming Software allows you to choose the state of the process variable and bias term at startup. You can choose from output, output-K*E, or current
- Another feature that can be very helpful for tuning is the ability to base the Derivative on the Process Variable versus the changes in the Error term

Tab 3. Major Alarms



- Not only can you set the values at which the limit alarms trigger but you can also use variable tags as the limit values. This allows you to change the limit value easily from an HMI or your computer.
- You can also specify the bits that will trigger when an alarm goes off with the Limit Bit tags. This is applicable to all alarms in the setup window

PID Tuning



- **Simple PID tuning:** The tuning window is now much cleaner and easier to use than its DirectSOFT counterpart
- **Save screen space:** No DataView needed to edit and monitor your process control inputs, setpoint, process variable, bias or output
- Fast processing times equals better results due to a higher sampling rate.
- Control the Process Variable Deadband and the Initial PV Bump to get more precise tuning

Application Example

Example of a Process Using PID to Control the Oven Temperature

