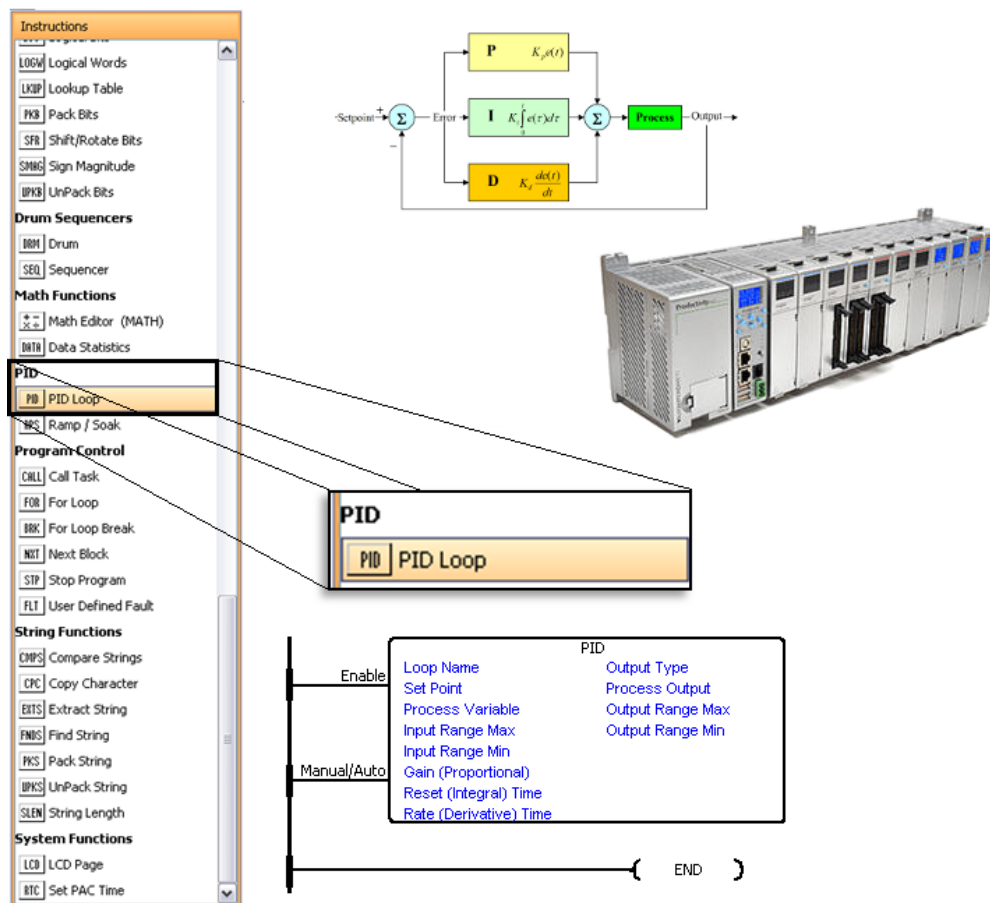


# Productivity<sup>3000</sup> Monthly Instruction Showcase

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.



### 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

**PID Loop (PID)**

Major Setup | Major Advanced | Major Alarms | Minor Setup | Minor Advanced | Minor Alarms

Loop Name:

Set Point:  ...

Process Variable:  ...

Input Range Max:  ...

Input Range Min:  ...

Process Output:  ...

Output Range Max:  ...

Output Range Min:  ...

Output Upper Limit:  ...

Output Lower Limit:  ...

Proportional (Gain):  ...

Integral (Reset) Time:  ... Sec

Derivative (Rate) Time:  ... sec

☐ Show Instruction Comment

OK Cancel Help

- 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

The screenshot shows the 'PID Loop (PID)' configuration window with the 'Major Advanced' tab selected. The window contains various input fields and checkboxes for configuring the PID loop's advanced settings.

**Major Setup:** Sample Rate (ms) 500, Loop Offset (ms) 0, Derivative Limit, Bias Term, SP Acceleration Gain, Error Deadband, Process Action Forward.

**Major Advanced:** Error Squared, Process Output Percentage, Freeze Bias (checked), Base the Derivative on Process Variable (checked), Initialization Mode (PV Tracking), Bias = Output - K\*E, Cascade Mode (unchecked), Enable Control.

**Major Alarms:** (Empty)

**Minor Setup:** (Empty)

**Minor Advanced:** (Empty)

**Minor Alarms:** (Empty)

At the bottom, there is a checkbox for 'Show Instruction Comment' and buttons for 'OK', 'Cancel', and 'Help'.

- 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

PID Loop (PID)

Major Setup | Major Advanced | **Major Alarms** | Minor Setup | Minor Advanced | Minor Alarms

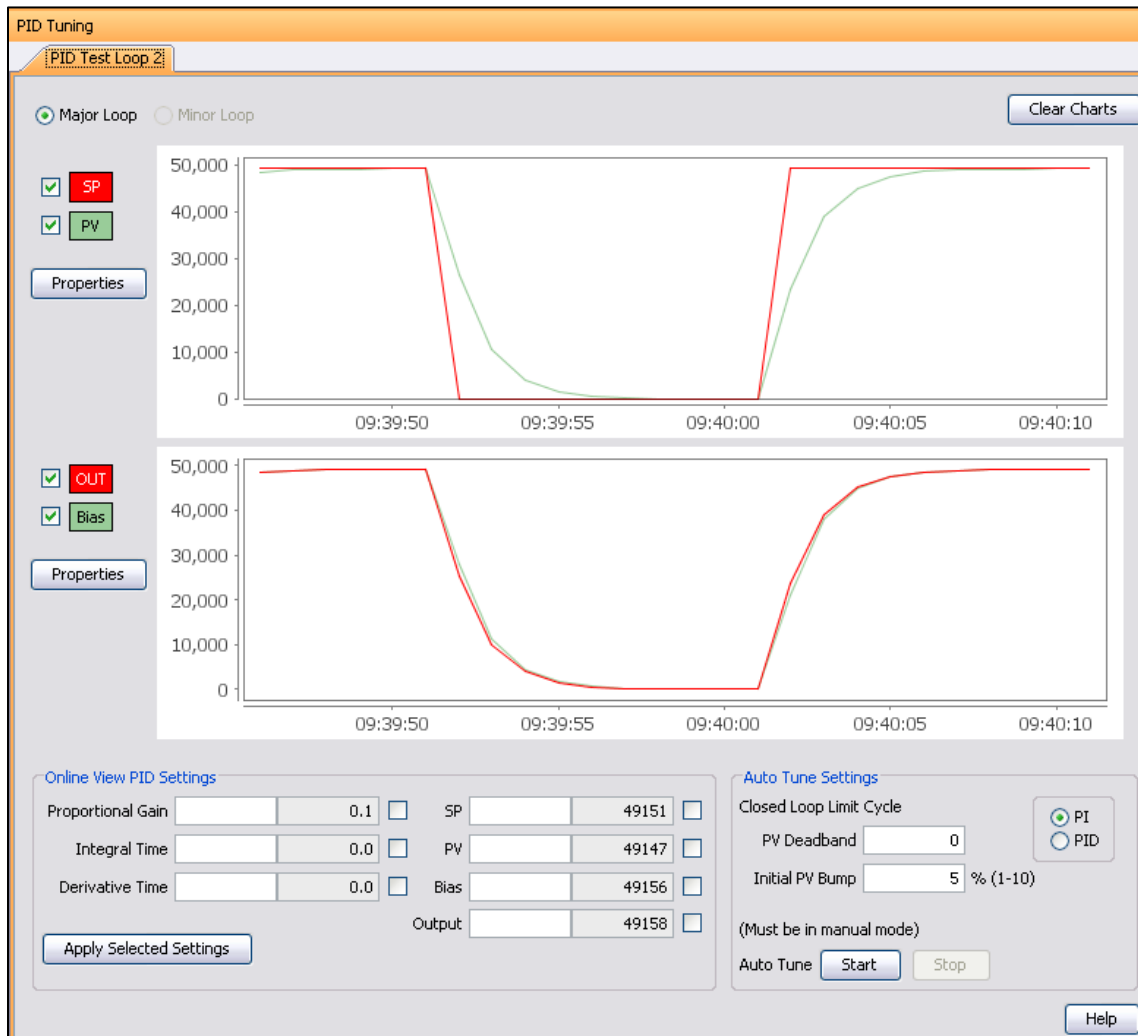
High High Limit Value	[Text Box] [Dropdown] [...]	High High Limit Bit	[Text Box] [Dropdown] [...]
High Limit Value	[Text Box] [Dropdown] [...]	High Limit Bit	[Text Box] [Dropdown] [...]
Low Limit Value	[Text Box] [Dropdown] [...]	Low Limit Bit	[Text Box] [Dropdown] [...]
Low Low Limit Value	[Text Box] [Dropdown] [...]	Low Low Limit Bit	[Text Box] [Dropdown] [...]
Deviation Limit Yellow	[Text Box] [Dropdown] [...]	Deviation Limit Yellow Bit	[Text Box] [Dropdown] [...]
Deviation Limit Orange	[Text Box] [Dropdown] [...]	Deviation Limit Orange Bit	[Text Box] [Dropdown] [...]
Rate of Change	[Text Box] [Dropdown] [...]	Rate of Change Bit	[Text Box] [Dropdown] [...]
Alarm Hysteresis	[Text Box] [Dropdown] [...]		

☐ Show Instruction Comment

OK Cancel Help

- 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

