



# Tough PLC can handle 8 Auto-tuned PID Loops with Unique PID Monitor

## What is PID Loop?

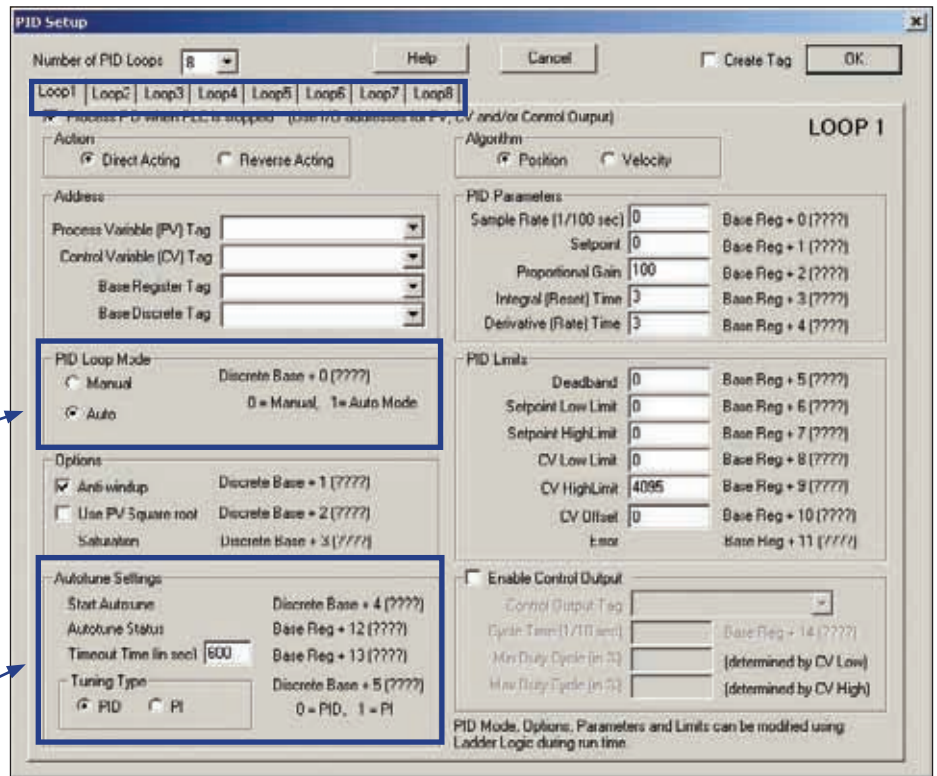
PID is one of the most popular control algorithms used in the industry, to control the variables involved in an industrial manufacturing process, for the proper operation of the process. PID stands for Proportional, Integral and Derivative control algorithm. With a proper choice of P, I, and D settings, a user can maintain a process value very close to the setpoint. In addition, if the setpoint changes, the PID algorithm can quickly bring the process back under control. Tough PLC supports up to 8 PID loops. For each loop you have to define several parameters, as shown below in the PID Setup window. You may change most of these parameters during run time, using Tough PLC editor in online mode.

## PID Loop Auto Tuning

To achieve a stable and responsive process control, it is very important to select the proper PID parameters. Experienced users can estimate good starting values for these parameters and later tweak them to optimize the PID loop performance. This is called as the manual tuning of the process. Whereas, those who want help in estimating the starting values of the parameters like P, I, and D coefficients, Tough PLC provides an Autotune feature.

## Autotune Control

Each PID Loop is controlled by the Start Autotune discrete variable (which is at Discrete Base+4). If the variable goes from false to true, and the loop is in manual mode, Tough PLC would start autotuning that loop.



## Autotune Setup

The Tough PLC can autotune PID loops, i.e. it can estimate the values for the Proportional Gain, Integral (Reset) time, and Derivative (Rate) time for PID loop. The dialog box allows you to setup the loop for autotune. Tough PLC uses Ziegler-Nichols method to estimate the PID parameters.

## Start Autotune

Shown on the dialog box for information only.

The Start Autotune discrete is at Discrete Base+4. Tough PLC initiates autotuning of a loop when this bit transitions from 0 to 1. Autotuning of the loop is started regardless of the selected "PID Loop Mode" of the loop. Once Autotune is started, you can stop it by setting this bit to 0.

## Timeout Time (in sec)

User programs Autotune timeout in seconds in this register. If Tough PLC can not finish autotuning within this time, the Autotune is aborted. User should program this field based on the dynamics of the process.

# Autotune Status

Shown on the dialog box for information only. During Autotune, Tough PLC reports the status of Autotune in the register.

Register Value	Description
0	Tuning in Progress
1	Tuning Done
2	User cancelled tuning
3	Control Value could not be incremented
4	The tuning algorithm failed to fit the curve
5	Division by zero error
6	Could not determine dead time
7	one or more of P, I or D was out of range

*Note: Autotune is performed by Tough PLC observing open loop response to a step change in the control value. Before starting autotune, the process should be in a steady state. During Autotune, watch the process variable closely for it to be within the safe limits.*

Tough'n'Smart™

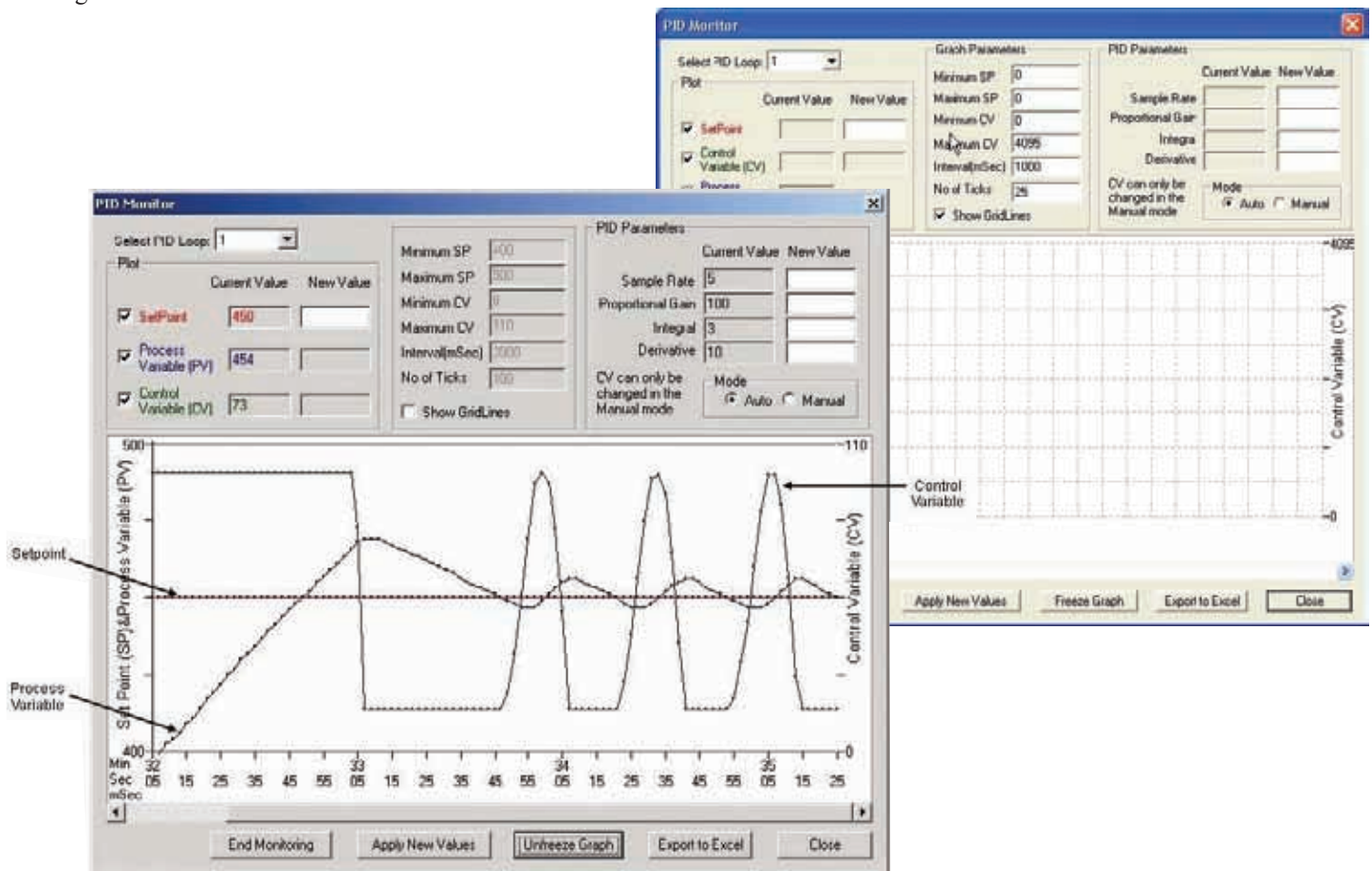
Tough  
PLC

## Tuning Type

User selects if PI or PID tuning is required.

## PID Monitor

You can use the PID Monitor function to monitor and make real-time changes to your PID Loop. In order to use it, you must be connected to the PLC and select Main Menu > Tough PLC > PID Monitor. A PID Monitor window will show up. Here you can change the current values of the parameters by entering a value in the New Value field. Once all of the parameters are defined, press the Apply button and then Start Monitoring button at the bottom, to begin monitoring your PID Loop. A graph will begin to appear as shown in the image below.



In the illustration above, the Setpoint and Process Variable were set to 450 and are represented in the graph by the line running through the middle of the graph. The Minimum SP of 400 is shown at the bottom left and the Maximum Limit of 500 is shown at the top left of the graph. The Control Variable was set to 230 and is represented on the right side of the graph.