Quick Start Guide to Using the PID Block in Logix 500

It is assumed that the reader has three important skills: a good working knowledge of PID, a good working knowledge of the Logix 500 platform, and a good working knowledge of the process and system to which the PID block is to be applied.

Note carefully that the methods used here are not universally applicable nor are they the only means of arriving at a PID solution to your control application. If you are not already familiar with PID control, we strongly recommend you read one of our Getting Started Guides to PID Control, and if you are not familiar with PLC programming, we strongly recommend you read our Getting Started Guide to Programmable Logic Controllers.

Step number one in the solution of any control problem is to first assure that the process can be controlled well. If, for example, you have an industrial oven which is heated by a gas burner, can you disconnect the mechanical linkage between the servo motor and the throttle and then adequately control the temperature by physically moving the throttle? If you can't, then maybe you have a problem that the most finely tuned PID loop can never control! Assuming that you can control your process manually, continue on…

1. When setting up a PID block, remember that 23 consecutive words beginning with the Control Block address are required. To avoid addressing conflicts, we recommend setting up a unique file such as N9 or N10 to use exclusively for PID blocks.

2. In this platform, the PID block operates with 14 bit (0-16383) integer numbers. If, for example, you have a PV (process variable) which is in floating point format, convert it first to integer format. If necessary, scale to 0-16383 bits. An important exception to this is that all of the newer SLC500 processors allow the range of –32768 to 32767 for Words 7 and 8, SMAX and SMIN (see Step 5).

Quick Start Guide to Using the PID Block in Logix 500

3. Write the SP (setpoint) in *engineering units* directly to the third word (which is Word number 2 since we start with Word number 0) of the 23 word PID file, as an integer. For example, If you assigned the Control Block to N9:0, then write the SP to N9:2.

4. Convert the PV (process variable) from engineering units (or perhaps you have a bit range) to a 14 bit (0-16383) value. This is commonly done using the SCP block. Enter the output address of the SCP in "Process Variable" in the PID block. NEVER write this value to Word 14 in the PID file. Word 14 should only be used for monitoring.

5. Click on "Setup Screen" at the bottom of the PID block. Note the Setpoint MAX and Setpoint MIN fields. These two fields not only establish scaling parameters for the data conversion to engineering units, but they effectively mark off the range for control (similar to setting input range on conventional temperature controllers). For example, if you have an oven which is expected to operate at say, 300C, then best control might be to set your limits here to 0 and 400. This range value is the basis upon which the gain reciprocal (or proportioning percent) specifies the actual proportioning bandwidth. For further details on SMAX and SMIN, see Page 7.

6. "Gain" or Kc is the inverse of proportioning percent (percent meaning the percentage of the range in which proportional control takes place). For example, if you desire a proportioning bandwidth of 30 degrees and the range is set at 0 to 300 degrees, the required gain would be 10.

7. "Reset" or Ti is the Integral term given as a time constant in minutes.

8. "Rate" or Td is the Derivative term given as a time constant in minutes.

Quick Start Guide to Using the PID Block in Logix 500

9. Loop Update will not be critical in most applications. Try a setting of 0.01 to 0.1. If you are working with a very fast moving process, you might consider using STI mode, which interrupts the processor scan to do a loop update every scan.

10. "Control Mode" means reverse-acting versus forward-acting. E=SP-PV is reverse-acting and E=PV-SP is forward-acting.

11. "PID Control" is simply auto versus manual. Leave it in Auto unless you want to manually control your process for setup.

12. Leave Limit Output CV on NO unless you have a good reason to limit your output.

13. Leave Deadband at zero unless you have a good reason to use one, such as avoidance of short-cycling a compressor or oil burner, although our recommendation even in those cases is to use a zero deadband, but instead limit short-cycling by manipulating the discrete output of a time-proportioning routine such as the one we suggest in our sample downloads. That way you keep the actual PID tuning nice and clean.

14. Finally we arrive at the PID block's output, the CV (control variable). Remember that the value written to the CV is limited to 0-16383, so use an SCP block to scale it to the appropriate value you need to drive your output. If your final control device is controlled by an analog output that is 0-32767 bits such as the 1746-NO4V module, all you will have to do is MUL the CV value by 2, otherwise use the SCP to scale the value. SimpleSolvers SignalScaler is very handy to use when programming scaling solutions such as this.

Quick Start Guide to Using the PID Block in Logix 500

15. In the event you wish to control a device that is either on or off (as opposed to modulated or throttled from low to high), you will need some additional ladder logic to convert your PID block's analog output value to a time-proportioning discrete signal. Although Allen Bradley provides a perfectly good solution of sample logic in their Publication 1747-RM011E-EN-P on page 9-31, we have developed an alternative routine that gives the user a greater degree of control over minimum on and off times, as well as avoiding periodic setting of minor error flags. See PID Example 2 in our Free Downloads section.

16. A quick word or two on the RG bit: The RG bit affects only two parameters, Gain (Kc) and Reset (Ti) and is normally off, or "0". This has importance only when you want to be able to change Gain and Reset values through the data table instead of directly in the PID block using Logix 500 (such as when you might want to be able to access these parameters through perhaps a touchscreen). Here's how they work:

RG bit = 0	RG bit = 1
Kc = value of Word 3 in PID block/10	Kc = value of Word 3 in PID block/100
Ti = value of Word 4 in PID block/10	Ti = value of Word 4 in PID block/100
Td = value of Word 5 in PID block/100	Td = value of Word 5 in PID block/100

17. The value of Rate (Td) always equals the value in Word 5 divided by 100.

18. For help with tuning your loop, we suggest reading our Simplified Guide to PID Temperature Control or our Simplified Guide to PID Process Control. Additionally, read our Quick Start Guide to Tuning the PID Block in Logix 500, a free download.

Quick Start Guide to Using the PID Block in Logix 500

PID BLOCK MAP

Note that all of the discrete commands and signals are contained in the first word (Word 0) of the 23 word PID block. The remaining 22 words all contain analog data.

WORD 0 (DISCRETE PID SIGNALS)

Bit	PID Symbol	Description	Choices Available
00	TM	Time Mode	Timed, STI
01	AM	Auto/Manual	Auto, Manual
02	CM	Control Mode	E=SP-PV, E=PV-SP
03	OL	Output Limiting	NO, YES
04	RG	Range Enhance	0, 1
05	SC	SP Scaling	(read only)
06	TF	Update Too Fast	(read only)
07	DA	Rate Action	0, 1
08	DB	PV in Deadband	(read only)
09	UL	Upper CV Alarm	(read only)
10	LL	Lower CV Alarm	(read only)
11	SP	SP Out of Range	(read only)
12	PV	PV Out of Range	(read only)
13	DN	PID Done Bit	(read only)
14	(not used)		
15	EN	PID Enabled Bit	(read only)

Quick Start Guide to Using the PID Block in Logix 500

PID BLOCK MAP (continued)

THE 23 WORDS OF THE PID BLOCK

Word	PID Symbol	Description
0	Discrete PID Signals	See the preceding table
1		Sub Error Code
2	**SP**	Setpoint
3	**Kc (X 10 if RG = 0)**	Gain (inverse of Proportioning)
4	**Ti (X 10 if RG = 0)**	Reset (Integral)
5	**Td (X 100)**	Rate (Derivative)
6		Feed Forward Bias
7	**Smax**	Setpoint Max
8	**Smin**	Setpoint Min
9		Deadband
10	(internal use)	Do not change
11		Output Max (% of CV)
12		Output Min (% of CV)
13		Loop Update Time
14	**PV** (scaled)	Scaled Process Variable
15	**SE**	Scaled Error
16	**CV** (%)	Output CV as percent
17	(internal use)	Do not change
18	(internal use)	Do not change
19	(internal use)	Do not change
20	(internal use)	Do not change
21	(internal use)	Do not change
22	(internal use)	Do not change

Quick Start Guide to Using the PID Block in Logix 500

MORE DETAIL ON SMAX AND SMIN

Note that in the PID Setup Screen's Inputs section, there is a grayed-out display field named "Process Variable PV". This value is more precisely called the SPV, or Scaled Process Variable, and is computed for you according to the values of SP (Word 2) and SMAX and SMIN (Words 7 and 8) as follows:

SPV = (PV/16383) * (SMAX – SMIN)

But when both **SMAX and SMIN = 0, SPV equals PV (Word 2).**

If SMAX is lower than SMIN, a processor fault will occur.

In all processors 5/03 and higher, values of approximately –32768 to 32767 can be entered for SMAX and SMIN.