

## Re: Shameless Plug

**Source:** <http://sci.tech-archive.net/Archive/sci.electronics.design/2004-12/0032.html>

---

**From:** Guy Macon (<http://www.guymacon.com>)

**Date:** 11/30/04

Date: Tue, 30 Nov 2004 09:11:10 +0000

Tim Wescott wrote:

>

>*I will be presenting two topics at the 2005 Embedded Systems Conference  
>San Francisco next March -- see <http://www.esconline.com/sf/> for show  
>details.*

>

>*"Basic Control Theory for the Software Engineer" is as much information  
>on the z-transform as I can fit into 90 minutes. It gives a  
>high-altitude overview of designing software control loops in a  
>systematic manner.*

>

>*"PID Without a PhD" is a primer on developing PID controllers in  
>software, tuning them without using higher math, and avoiding some of  
>the common pitfalls for this popular controller form.*

Cool! From many years teaching "practical PID" to technicians and engineers, here are some questions that came up a lot:

"Why do so many of the setups around here have only P and I or have D set to zero? How do I decide when to use D?"

"Why do some controller boards have an option to reverse the phase of the D? What is that good for?"

"How do I describe a thermostat with hysteresis using the same language that I use to describe a PID controller? It seems like P is infinity."

"Why do half the engineers call it Proportional-Integral-Derivative" and others call it "Proportional-Integral-Differential?" When I did a Google search on "proportional integral differential" I got 18,600 hits while "proportional integral differential" only had 3,060 hits, but most of the "proportional integral differential" hits seem to be by scientists and equipment manufacturers. Which is correct?"

BTW. for what it's worth, I found that relating position servos

and velocity servos to a person controlling a car (speed and position within the lane) was helpful. I also found it helpful to show how to use a stopwatch and odometer to derive speed with no speedometer, a stopwatch and speedometer to derive distance without an odometer, and a speedometer and odometer to derive elapsed time with no stopwatch. Your audience is different, of course – this worked really well with mechanical engineers, but software engineers are quite different.

Another gotcha that sometimes trips up software engineers: non-monotonic ADCs causing a "bad spot" that has positive feedback.