CHAPTER 6

Four Types of Controllers

Chapter Outline

The proportional-integral-differential (PID) controller[32,47] is perhaps the most common controller in general use. Most programmable logic controllers (PLCs) support a variety of processes with this structure; for example, many temperature, pressure, and force loops are implemented with PID control. PID is a structure that can be simplified by setting one or two of the three gains to zero. For example, a PID controller with the differential ("D") gain set to zero reduces to a PI controller. This chapter will explore the use of four variations of P, I, and D gains.

When choosing the controller for an application, the designer must weigh complexity against performance. PID, the most complex of the four controllers in this chapter, can accomplish anything the simpler controllers can, but more complex controllers require more capability to process, in the form of either faster processors for digital controllers or more components for analog controllers. Beyond that, more complex controllers are more difficult to tune. The designer must decide how much performance is worth.

Control System Design Guide. DOI: 10.1016/B978-0-12-385920-4.00006-0

The focus in this chapter will be on digital controls, although issues specific to analog controls are covered throughout. As discussed in Chapter 4, the basic issues in control systems vary little between digital and analog controllers. For all control systems, gain and phase margins must be maintained, and phase loss around the loop should be minimized. The significant differences between controller types relate to which schemes are easiest to implement in analog or digital components.

The controllers here are all aimed at controlling a single-integrating plant. Note especially that the PID controller discussed in this chapter is for a single-integrating plant, unlike a PID position loop, which is for a double-integrating plant. As will be shown in Chapter 17, a PID position loop is fundamentally different to the classic PID loops discussed here.

6.1 Tuning in this Chapter

Throughout this chapter, a single tuning procedure will be applied to multiple controllers. The main goal is to provide a side-by-side comparison of these methods. A consistent set of stability requirements is placed on all of the controllers. Of course, in industry, requirements for controllers vary from one application to another. The requirements used here are representative of industrial controllers, but designers will need to modify these requirements for different applications. The specific criteria for tuning will be as follows: In response to a square wave command, the high-frequency zone (P and D) can overshoot very little (less than 2%), and the low-frequency zone can overshoot up to 15%. Recognizing that few people have laboratory instrumentation that can produce Bode plots, these tuning methods will be based on time-domain measures of stability, chiefly overshoot in response to a square wave. This selection was made even though it is understood that few control systems need to respond to such a waveform. However, a square wave is often the signal of choice for exposing marginal stability; testing with gentler signals may allow marginal stability to pass undetected.

This chapter will apply the zone-based tuning method of Chapter 4. Each of the four controllers has either one or two zones. The proportional and differential gains combine to determine behavior in the higher zone and thus will be set first. So the P and D gains must be tuned simultaneously. The integral gain determines behavior in the lower zone.

The higher zone is limited by the control loop outside the control law: the plant, the power converter, and the feedback filter. Note that sampling delays can be represented as parts of these components; calculation delay and sample-and-hold delay (see Section 4.2) can be thought of as part of the plant and feedback delay as part of the feedback filter. The lower zone is limited primarily by the higher zone.

The tuning in this chapter will set the loop gains by optimizing the response to the command. Higher loop gains will improve command response and they will also improve the disturbance response. Depending on the application, either command or disturbance response may be the more important. However, command response is usually preferred for determining stability, for a practical reason: Commands are easier to generate in most control systems. Disturbance response is also an important measure, as will be discussed in detail in Chapter 7.

When tuning, the command should be as large as possible to maximize the signal-to-noise ratio. This supports accurate measurements. However, the power converter must remain out of saturation during these tests. For this chapter, the example systems are exposed only to the relative quiet of numerical noise in the model; in real applications, noise can be far more damaging to accurate measurements.

6.2 Using the Proportional Gain

Each of the controllers in this chapter is based on a combination of proportional, integral, and differential gains. Whereas the latter two gains may be optionally zeroed, virtually all controllers have a proportional gain. Proportional gains set the boundaries of performance for the controller. Differential gains can provide incremental improvements at higher frequencies, and integral gains improve performance in the lower frequencies. However, the proportional gain is the primary actor across the entire range of operation.

6.2.1 P Control

The proportional, or "P," controller is the most basic controller. The control law is simple: Control $\propto$ Error. It is simple to implement and easy to tune. A P-control system is provided in Experiment 6A and is shown in Figure 6.1. The command (R) is provided by a square wave feeding through a DSA to a sample and hold block. The error (E) is formed as the difference between command and feedback. That error is scaled by the gain K_P to create the command to the power converter. The command is clamped (here, to ± 20) and then fed to a power converter modeled by a 500-Hz, two-pole low-pass filter with a damping ratio of 0.7 (equivalently, the filter Q is also about 0.7). The plant (G) is a single integrator with a gain of 500. The feedback (F) passes through a sample-and-hold. The sample time for the digital controller, set by the *Live Constant* "*TSample*," is 0.0005 seconds. The response vs. command (F/R) is shown on the *Live Scope*.

The chief shortcoming of the P-control law is that it allows DC error; it droops in the presence of fixed disturbances. Such disturbances are ubiquitous in controls: Ambient temperature drains heat, power supply loads draw DC current, and friction slows motion. DC error cannot be tolerated in many systems, but where it can, the modest P controller can suffice.

6.2.1.1 How to Tune a Proportional Controller

Tuning a proportional controller is straightforward: Raise the gain until instability appears. The flowchart in Figure 6.2 shows just that. Raise the gain until the system begins to overshoot. The

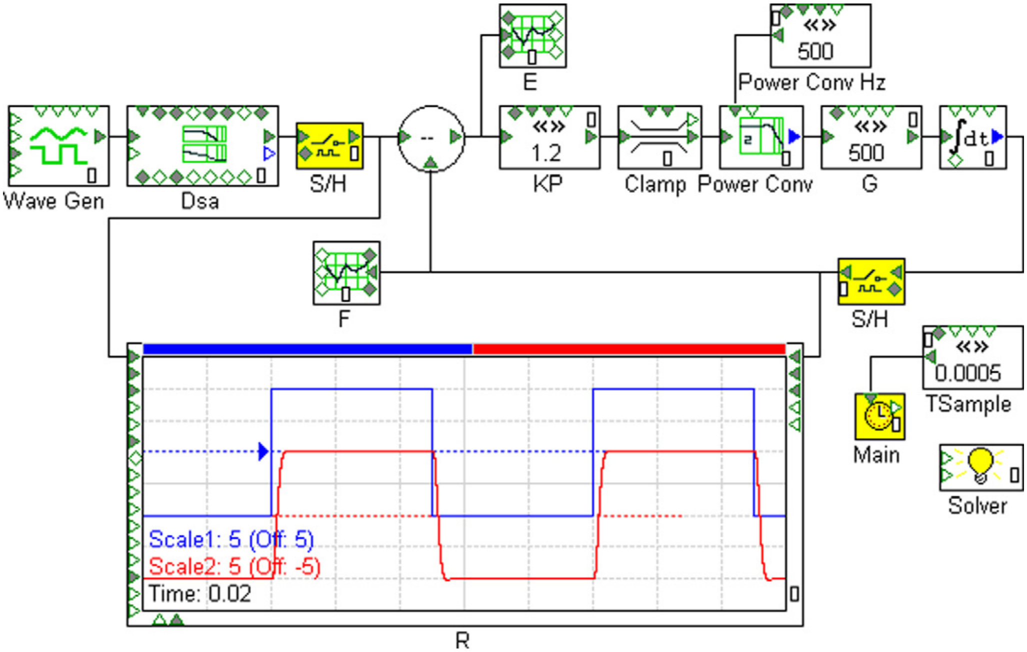

Figure 6.1:
Experiment 6A: A P controller.

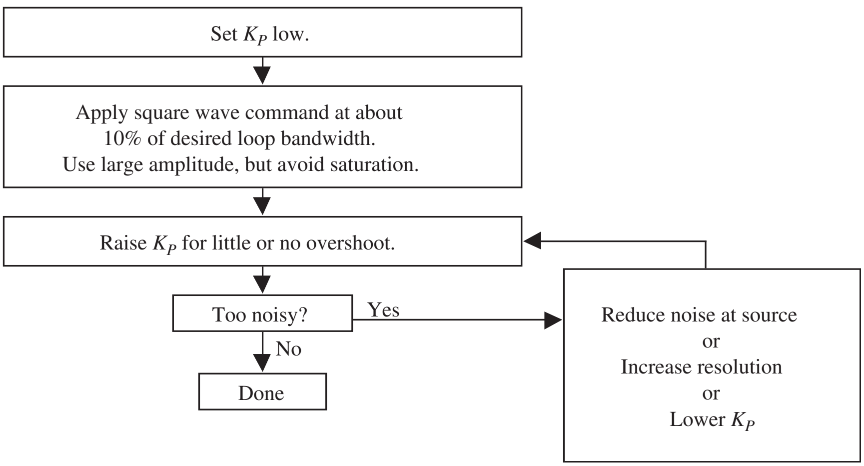

Figure 6.2:
Tuning a P controller.

loss of stability is a consequence of phase lag in the loop, and the proportional gain will rise to press that limit. Be aware, however, that other factors, primarily noise, often ultimately limit the proportional gain below what the stability criterion demands.

Noise in a control system may come from many sources. In analog controllers, it is often from electromagnetic interference (EMI), such as radio frequency interference (RFI) and ground loops, which affects signals being connected from one device to another. Noise is common in digital systems in the form of limited resolution, which acts like random noise with an amplitude of the resolution of the sensor. Independent of its source, noise will be amplified by the gains that set the high-frequency zone, such as the proportional gain.

Noise is a nonlinear effect and one that is generally difficult to characterize mathematically. Usually, the person tuning the system must rely on experience to know how much noise can be tolerated. Higher gain amplifies noise, so setting the gain low will relieve the noise problem but can also degrade performance. In real-world applications, setting the proportional gain can require balancing the needs for high performance and low noise. Things are simpler for tuning the examples in this chapter; since these simulated systems have only the very small numerical noise of the model, stability will be the sole consideration.

Figure 6.1 shows the step response of the P controller tuned according to the procedure of Figure 6.2. The result was $K_P = 1.2$. The step response has almost no overshoot. Using Experiment 6A, the closed- and open-loop responses can be measured. As shown in Figure 6.3, the closed-loop response has a comparatively high bandwidth (186 Hz) without peaking. The open-loop plot shows 65° PM and 12 dB GM (Figure 6.4).

6.3 Using the Integral Gain

The primary shortcoming of the P controller, tolerance of DC error, is readily corrected by adding an integral gain to the control law. Because the integral will grow ever larger with even small DC error, any integral gain (other than zero) will eliminate DC droop. This single advantage is why PI is so often preferred over P control.

Integral gain provides DC and low-frequency stiffness. Stiffness is the ability of a system to resist disturbances. Integral gains operate in the low-frequency zone and so improve stiffness at relatively low frequencies (those well below the bandwidth). When a DC error occurs, the integral gain will move to correct it and the higher the integral gain, the faster the correction. Don’t confuse DC stiffness with dynamic stiffness, the stiffness across the entire range of operation. A system can be at once quite stiff at DC and not stiff at all at higher frequencies. These concepts are discussed in detail in Chapter 7. For the present, be aware that higher integral gains will provide better DC stiffness but will not substantially improve stiffness near or above the system bandwidth.

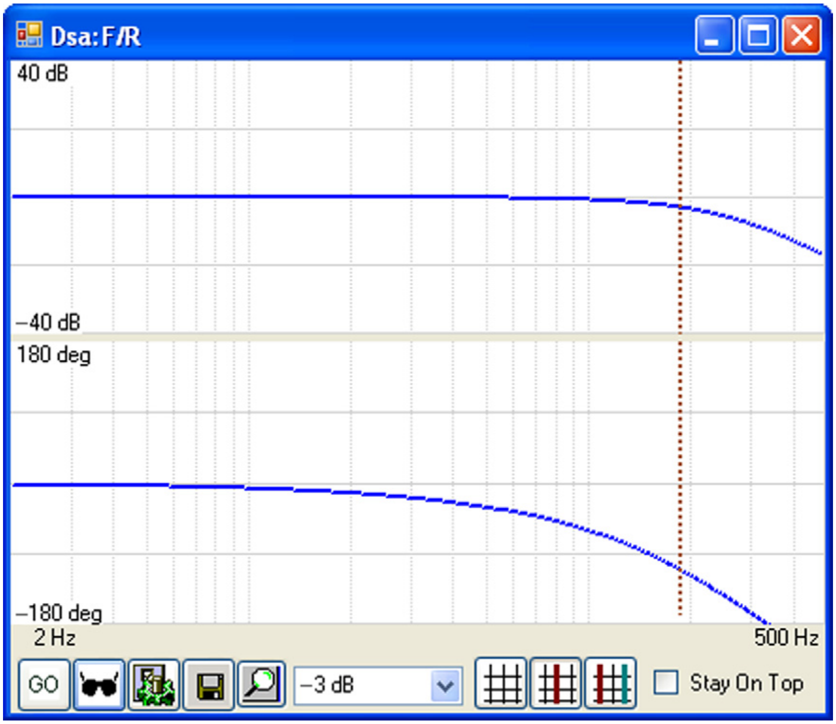

Figure 6.3:
Closed-loop Bode plot for proportional system (186 Hz bandwidth, 0 dB peaking).

Integral gain comes at a cost. PI controllers are more complicated to implement; the addition of a second gain is part of the reason. Also, dealing with power converter saturation becomes more complicated. In analog controllers, clamping diodes must be added; in digital controllers, special algorithms must be coded. The reason is that the integral must be clamped during saturation to avoid the problem of “windup,” as discussed in Section 3.8. Integral gain also causes instability. In the open loop, the integral, with its 90° phase lag, reduces phase margin. In the time domain, the common result of adding integral gain is increased overshoot and ringing.

6.3.1 PI Control

With PI control, the P gain provides similar operation to that in the P controller, and the I gain provides DC stiffness. Larger I gain provides more stiffness and, unfortunately, more overshoot. The controller is shown in Figure 6.5. Note that the K_I is in series with K_P; this is common, although it’s also common to place the two gains in parallel.

It should be noted that the implementation of Figure 6.5 is for illustrative purposes. The PI controller lacks a windup function to control the integral value during saturation. The standard

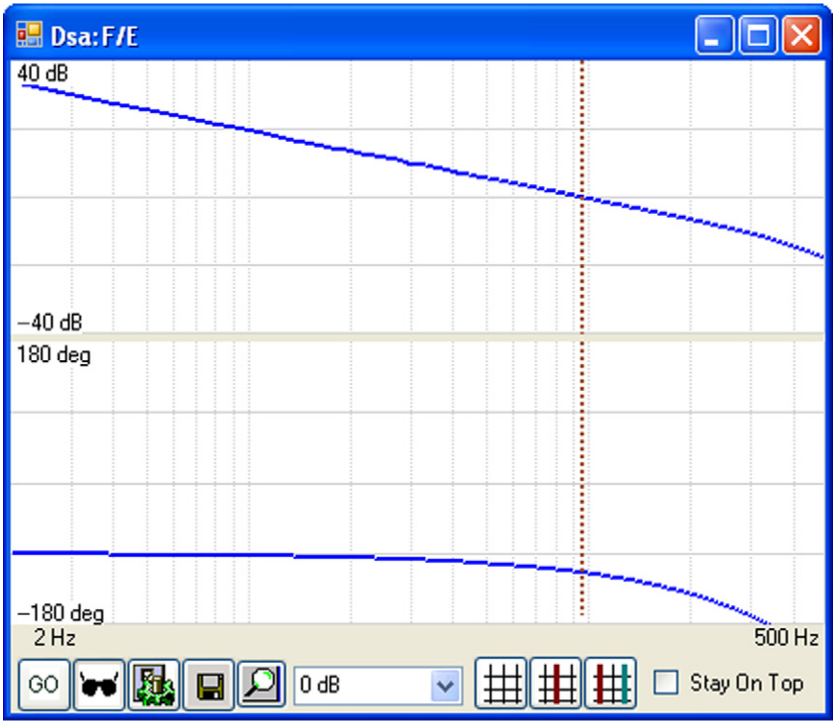

Figure 6.4:
Open-loop Bode plot of proportional system (65° PM, 12.1 dB GM).

control laws supported by *Visual ModelQ* provide windup control and so would normally be preferred. (In addition, models using them take less screen space.) However, Experiment 6B and other experiments in this chapter break out the control law gains to make clear their functions. Because the purpose of this section is to compare four varieties of PID, the clarity provided by explicitly constructed control laws outweighs the need for wind-up control or compact representation.

6.3.1.1 How to Tune a PI Controller

PI controllers have two frequency zones: high and low. The high zone is served by K_P and the low by K_I. As Figure 6.6 shows, the process for setting the proportional gain is the same as it was in the P controller. After the higher zone is complete, K_I can be tuned. Here it is raised for 15% overshoot to a square wave. Again, a square wave is an unreasonably harsh command to follow perfectly; a modest amount of overshoot to a square wave is tolerable in most applications.

As Figures 6.5, 6.7, and 6.8 show, the PI controller is similar to the P controller, but with slightly poorer stability measures. The integral gain is high enough to cause a 15% overshoot to

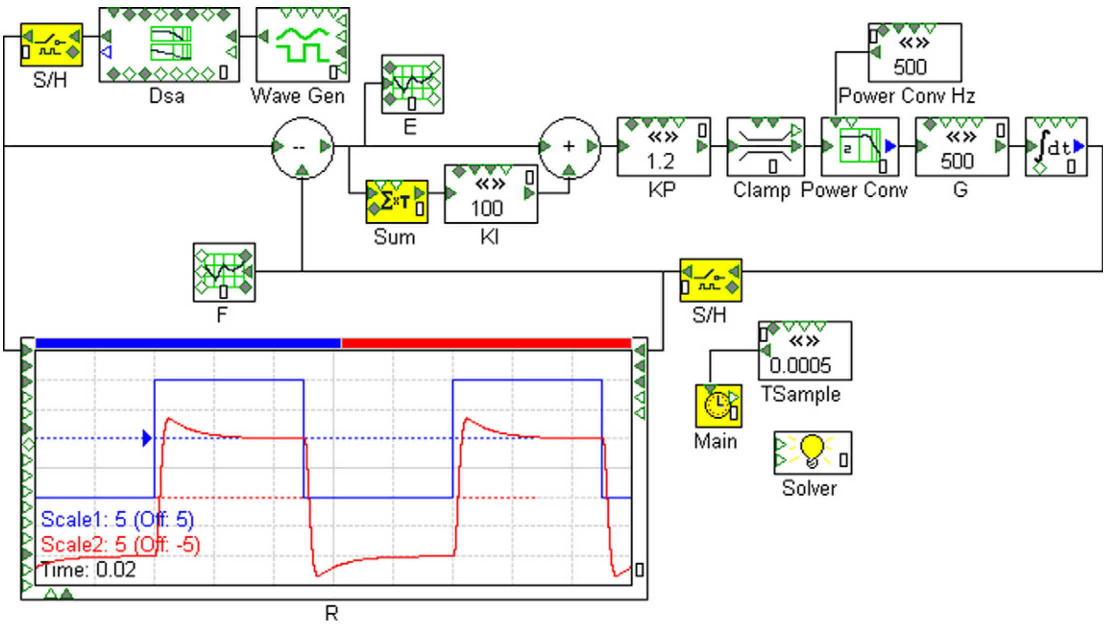

Figure 6.5:
Experiment 6B: A PI Controller.

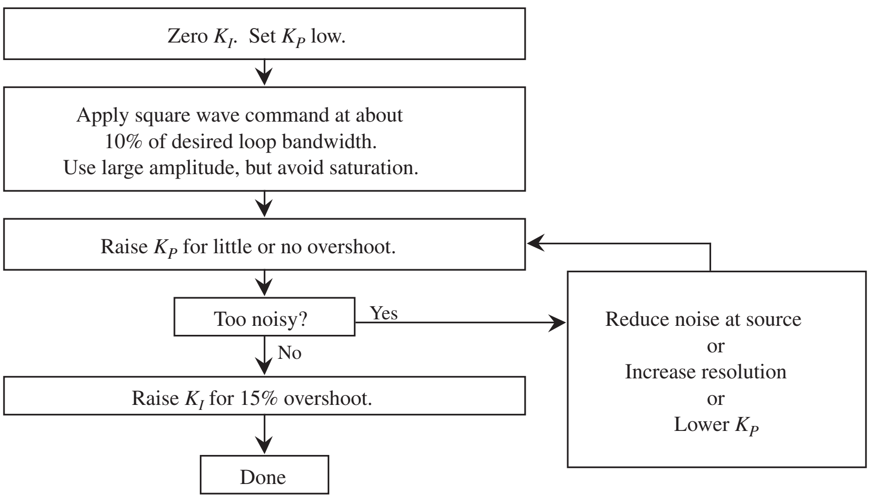

Figure 6.6:
Tuning a PI controller.

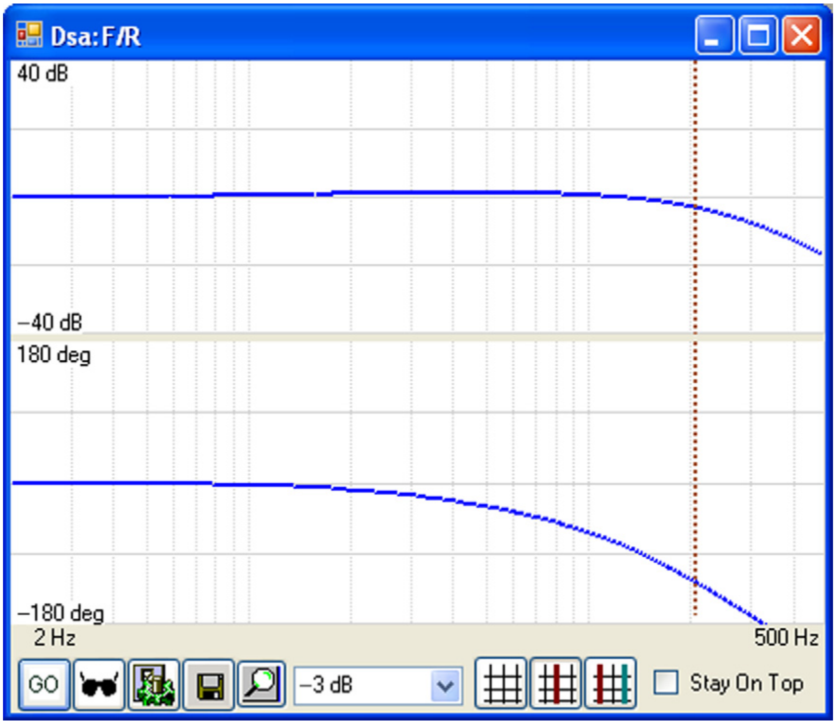

Figure 6.7:
Closed-loop Bode plot for a PI controller (206-Hz bandwidth, 1.3 dB of peaking).

a step. The bandwidth has gone up a bit (from 186 Hz to 206 Hz), but the peaking is about 1.3 dB. The PM has fallen 9° (from 65° to 56°), and the GM is nearly unchanged, down 0.4 dB to 11.7 dB.

6.3.1.2 Analog PI Control

A simple analog circuit can be used to implement PI control. As shown in the schematic of Figure 6.9, a series resistor and capacitor are connected across the feedback path of an op-amp to form the proportional (R_L) and integral (C_L) gains. Clamping diodes clamp the op-amp and prevent the capacitor from charging much beyond the saturation level. A small leakage path due to the diodes is shown as a resistor. The input-scaling resistors are assumed here to be equal ($R_C = R_F$).

The control block diagram for Figure 6.9 is shown in Figure 6.10. Note that the gains in this figure are constructed to parallel those of the general PI controller in Figure 6.5. Tuning the analog controller is similar to tuning the general controller. Short (remove) the capacitor to convert the system to a P controller, and determine the appropriate setting of R_L, as was done for K_P. Then adjust C_L for 15% overshoot. The analog controller will behave much like the digital controller.

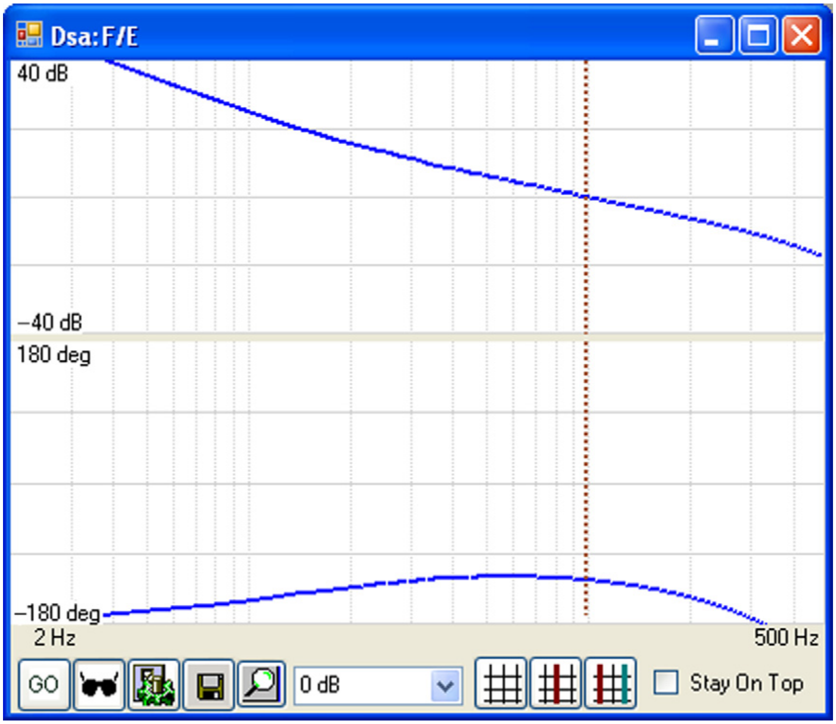

Figure 6.8:
Open-loop plot of PI controller (56°, PM 11.7 dB GM).

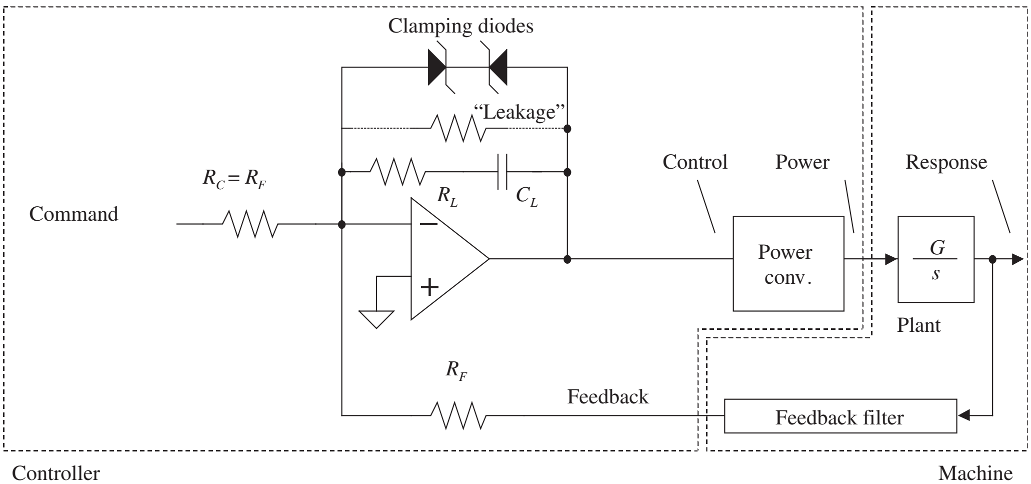

Figure 6.9:
Schematic for analog PI controller.

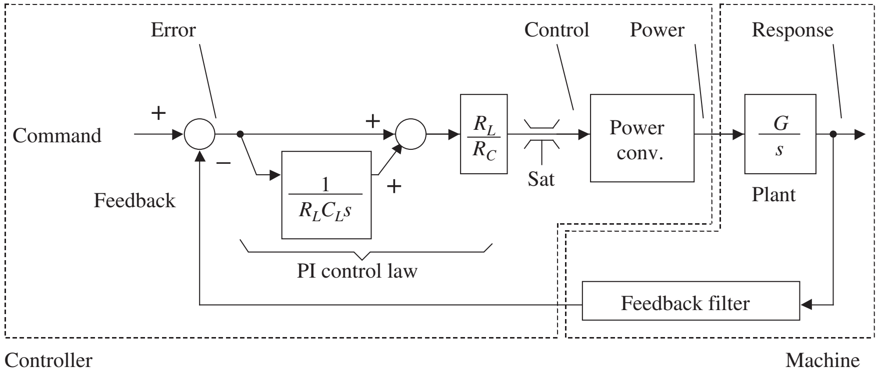

Figure 6.10:
Block diagram of analog PI controller.

One compromise that must be made for analog PI control is that op-amps cannot form true integrators. The diodes and capacitor will have some leakage, and, unlike a true integrator, the op-amp has limited gains at low frequency. Often, the PI controller is modeled as a lag network, with a large resistor across the op-amp feedback path, as shown in Figure 6.9. This "leaky" integrator is sometimes called a *lag circuit*. In some cases a discrete resistor is used to cause leakage intentionally. This is useful to keep the integral from charging when the control system is disabled. Although the presence of the resistor does have some effect on the control system, it is usually small enough and at low enough frequency not to be of much concern.

6.4 Using the Differential Gain

The third gain that can be used for controllers is the differential, or "D," gain. The D gain advances the phase of the loop by virtue of the 90° phase lead of a derivative. Using D gain will usually allow the system responsiveness to increase, for example, allowing the bandwidth to nearly double in some cases.

Differential gain has shortcomings. Derivatives have high gain at high frequencies. So while some D does help the phase margin, too much hurts the gain margin by adding gain at the phase crossover, typically a high frequency. This makes the D gain difficult to tune. It may at once reduce overshoot by virtue of increased PM and cause high-frequency oscillations owing to the reduced GM. The high-frequency problem is often hard to see in the time domain because high-frequency ringing can be hard to distinguish from normal system noise. So a control system may be accepted at installation but have marginal stability and thus lack the robust performance expected for factory equipment. This problem is much easier to see using Bode plots measured on the working system.

Another problem with derivative gain is that derivatives are sensitive to noise. Even small amounts of noise from wiring or resolution limitations may render the D gain useless. In most cases, the D gain needs to be followed by a low-pass filter to reduce the noise content. The experiments in this section assume a near-noiseless system, so the D filter is set high (2000 Hz). In many systems such a filter frequency would be unrealistic.

6.4.1 PID Control

The PID controller adds differential gain to the PI controller. The most common use of differential gain is adding it in parallel with the PI controller as shown in Figure 6.11. Here, a low-pass filter with a break frequency (2000 Hz by default) is added to the derivative path. As with the PI controller, the differential and integral gains will be in line with the proportional gain; note that many controllers place all three gains in parallel.

6.4.1.1 How to Tune a PID Controller

A PID controller is a two-zone controller. The P and D gains jointly form the higher-frequency zone. The I gain forms the low-frequency zone. The benefit of the D gain is that it allows the P gain to be set higher than it could be otherwise. The first step is to tune the controller as if it were a P controller, but to allow more overshoot than normal (perhaps 10%), understanding that the D gain will eventually cure the problem. Typically, the P gain can be raised 25–50% over the value from the P and PI controllers. The next step is to add a little D gain to cure the overshoot induced by the higher-than-normal P gain. The P and D gains together form the high-frequency zone. Next, the integral gain is tuned, much as it was in the PI controller. The expectation is that adding the D gain will allow the P and I gains to be about 20–40% higher.

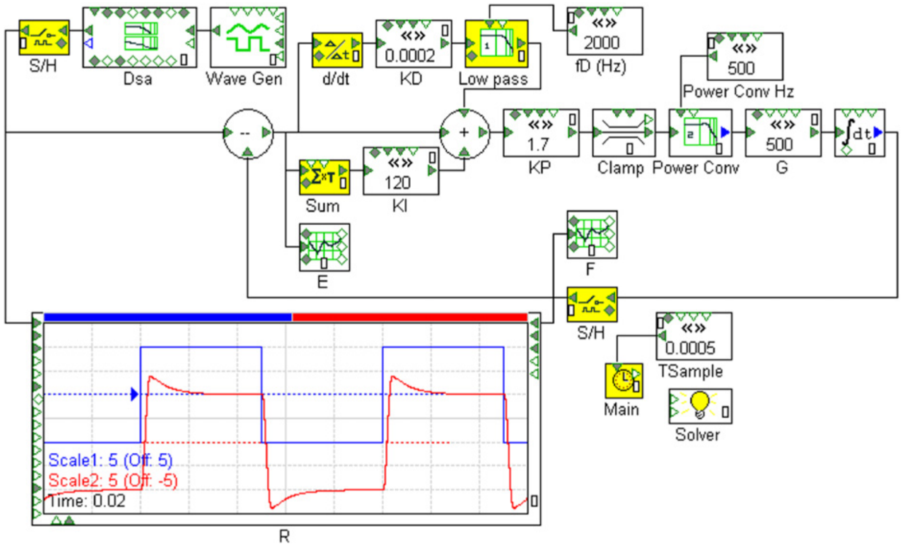

Figure 6.11:
Experiment 6C: A PID controller.

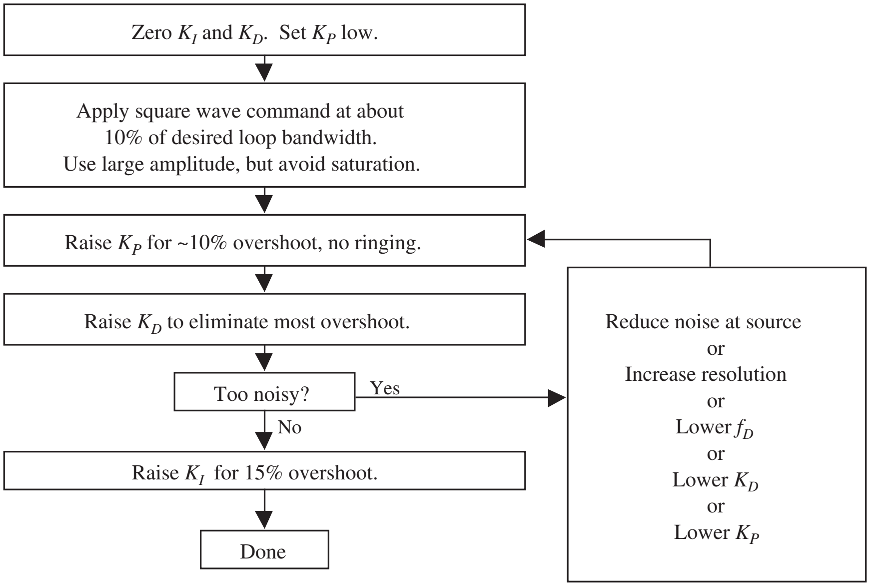

Figure 6.12:
Tuning a PID controller.

The results of the tuning procedure in Figure 6.12 are shown in Figures 6.11, 6.13, and 6.14. The PID controller allowed the proportional gain to increase to 1.7, about 40% more than in the PI controller (Figure 6.5), and the integral gain to increase to 120, about 20% more than the PI. However, the PID controller overshoots no more than the PI controller.

The closed-loop Bode plot of Figure 6.13 shows a dramatic increase in bandwidth; the PID controller provides 359 Hz, about 70% more than the 206 Hz provided by PI (Figure 6.7). Notice, though, that the phase lag of the closed-loop system is 170°, which is about 45° more than the PI. That makes this PID system more difficult to control as an inner loop than the PI controller would be. More phase lag at the bandwidth means an outside loop (such as a position loop surrounding this PID velocity controller) would have to deal with greater lag within its loop and thus have more stability problems.

The open-loop plot of the PID controller in Figure 6.14 shows a PM of 55°, about the same as the PI controller. However, the GM is about 8.5 dB, 3 dB less than the PI controller. Less GM is expected because the high-frequency zone of the PID controller is so much higher than that of the PI controller, as evidenced by the higher bandwidth. Reduced GM is a concern because the gains of plants often change during normal operation. This is of particular concern in systems where the gain can increase, such as saturation of an inductor (which lowers the inductance) in a current controller, declining inertia in a motion system, or declining thermal mass in a temperature controller; these effects all raise the gain of the

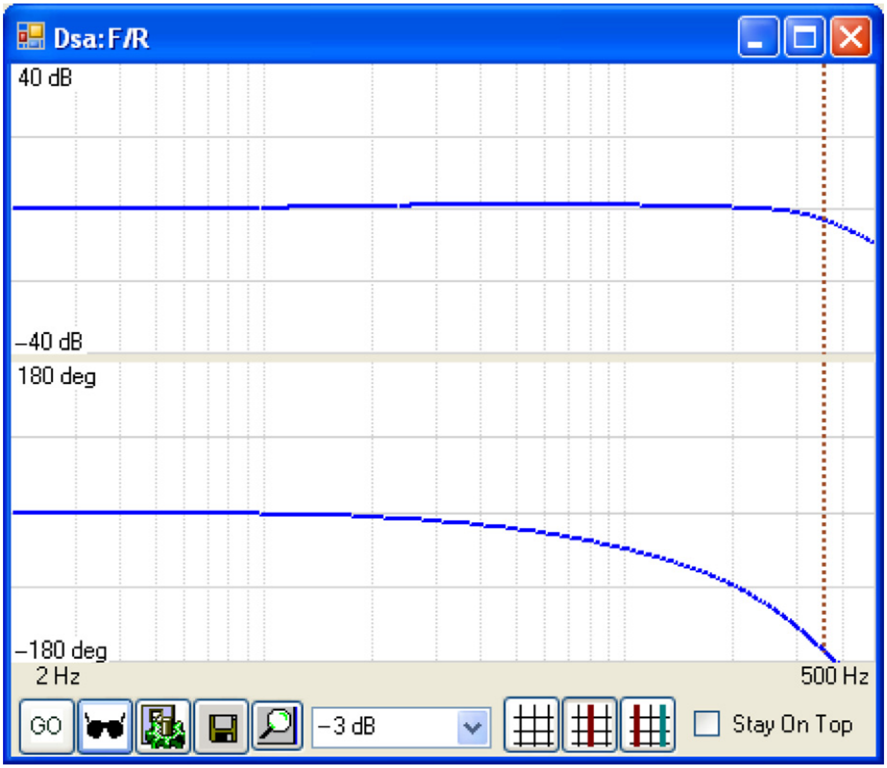

Figure 6.13:
Closed-loop Bode plot of PID controller (359-Hz bandwidth, 1.0 dB peaking).

plant and chip away at the GM. Given the same plant and power converter, a PID controller will provide faster response than a PI controller, but will often be harder to control and more sensitive to changes in the plant.

6.4.1.2 Noise and the Differential Gain

The problems with noise in the PI controller are exacerbated by the use of a differential gain. The gain of a true derivative increases without bound as the frequency increases. In most working systems, a low-pass filter is placed in series with the derivative to limit gain at the highest frequencies. If the noise content of the feedback or command signals is high, the best cure is to reduce the noise at its source. Beyond that, lowering the frequency of the derivative's low-pass filter will help, but it will also limit the effectiveness of the D gain. Noise can also be reduced by reducing the differential gain directly, but this is usually a poorer alternative than lowering the low-pass filter frequency. If the signal is too noisy, the D gain may need to be abandoned altogether.

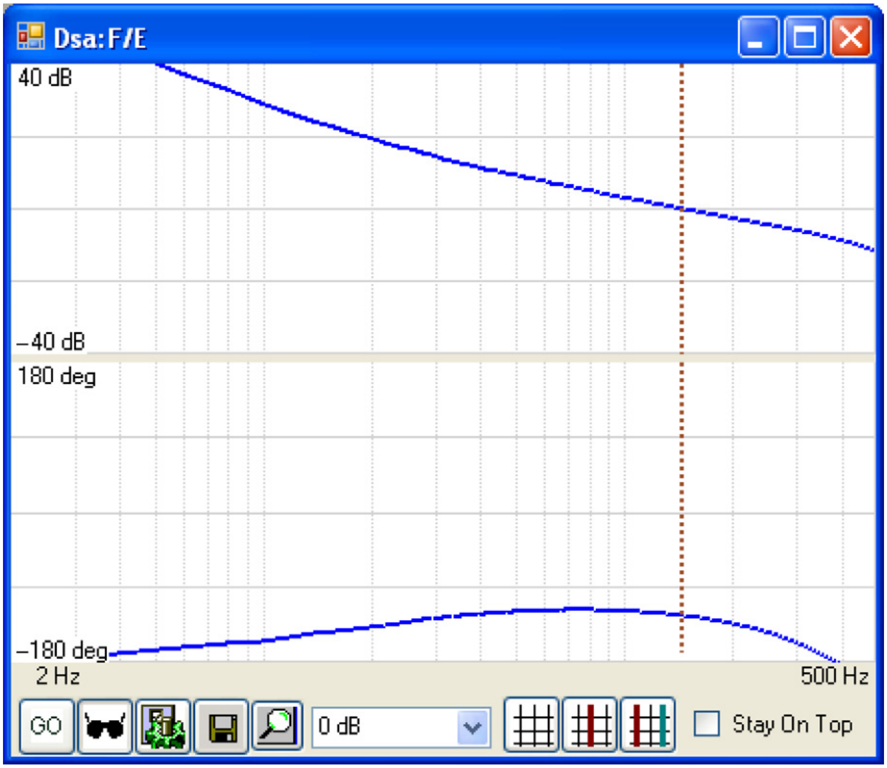

Figure 6.14:
PID controller open loop (55° PM, 8.5 dB GM).

6.4.1.3 The Ziegler–Nichols Method

A popular method for tuning P, PI, and PID controllers is the Ziegler–Nichols method. This method starts by zeroing the integral and differential gains and then raising the proportional gain until the system is unstable. The value of K_P at the point of instability is called K_{MAX}; the frequency of oscillation is f_0. The method then backs off the proportional gain a predetermined amount and sets the integral and differential gains as a function of f_0. The P, I, and D gains are set according to Table 6.1[32].

Table 6.1: Settings for P, I, and D Gains According to the Ziegler–Nichols Method

	K_P	K_I	K_D
P controller	0.5 K_{MAX}	0	0
PI controller	0.45 K_{MAX}	1.2 f_0	0
PID controller	0.6 K_{MAX}	2.0 f_0	0.125/f_0

If a dynamic signal analyzer is available to measure the GM and phase crossover frequency, there is no need to raise the gain all the way to instability. Instead, raise the gain until the system is near instability, measure the GM, and add the GM to the gain to find K_{MAX}. For example, if a gain of 2 had a GM of 12 dB (a factor of 4), K_{MAX} would be 2 plus 12 dB, or 2 times 4, or 8. Use the phase crossover frequency for f_0. A flowchart for the Ziegler–Nichols method is shown in Figure 6.15.

Note that the form shown here assumes K_P is in series with K_I and K_D. For cases where the three paths are in parallel, be sure to add a factor of K_P to the formulas for K_I and K_D in Table 6.1 and Figure 6.15. Note, also, that these formulas make no assumption about the units of K_P, but K_I and K_D must be in SI units (rad/sec and sec/rad, respectively). This is the case for the *Visual ModelQ* models of this chapter, but often is not the case for industrial controllers which may use units of Hz, seconds, or non-standard units. Finally, the Ziegler–Nichols method is frequently shown using T_0, period of oscillation, instead of the f_0 when $K_P = K_{MAX}$; of course, $T_0 = 1/f_0$.

The Ziegler–Nichols method is too aggressive for many industrial control systems. For example, for a proportional controller, the method specifies a GM of just 6 dB, compared with the 12 dB in the P controller tuned earlier in this chapter (Figure 6.5). In general, the gains from Ziegler–Nichols will be higher than from the methods presented here. Table 6.2 shows a comparison of tuning the P, PI, and PID controllers according to the method in this chapter and the Ziegler–Nichols method. (The value $K_{MAX} = 4.8$ and $f_0 = 311$ Hz were found using

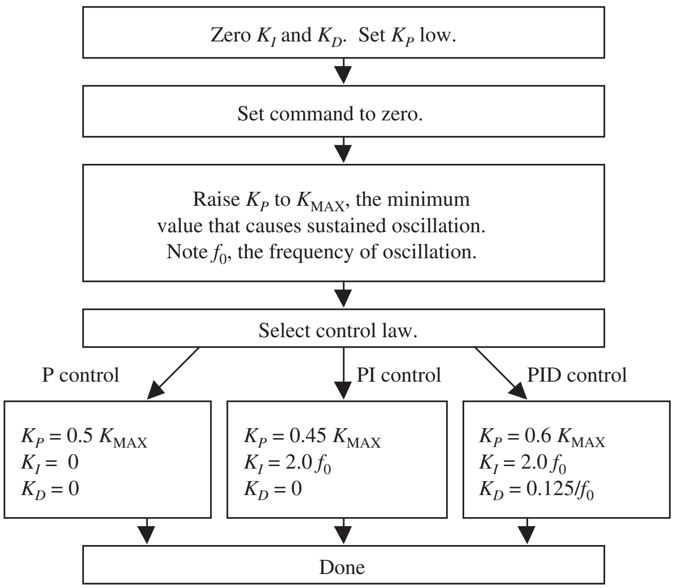

Figure 6.15:
Ziegler–Nichols method for tuning P, PI, and PID controllers.

Table 6.2: Comparison of Results from Tuning Method in this Chapter and the Ziegler–Nichols Method

	Method of Chapter 6	Ziegler–Nichols Method $K_{MAX} = 4.8$ and $f_0 = 311$ Hz
P controller	$K_P = 1.2$	$K_P = 2.4$
PI controller	$K_P = 1.2$	$K_P = 2.2$
	$K_I = 100$	$K_I = 373$
PID controller	$K_P = 1.7$	$K_P = 2.9$
	$K_I = 120$	$K_I = 622$
	$K_D = 0.0002$	$K_D = 0.0004$

Experiment 6A.) Both sets of gains are stable, but the Ziegler–Nichols method provides smaller stability margins.

6.4.1.4 Popular Terminology for PID Control

Often PID controllers involve terminology that is unique within controls. The three gains, proportional, integral, and differential, are called *modes* and PID is referred to as *three-mode control. Error* is sometimes called *offset. The* integral gain is called *reset* and the differential gain is often called *rate*. The condition where the error is large enough to saturate the loop and continue ramping up the integral is called *reset windup*. Synchronization, the process of controlling the integral during saturation, is called *antireset wind-up*. You can get more information from PID controller manufacturers, such as the Foxboro Company (www.foxboro.com).

6.4.1.5 Analog Alternative to PID: Lead-Lag

PID presents difficulties for analog circuits, especially since extra op-amps may be required for discrete differentiation. The lead-lag circuit of Figure 6.16 provides performance similar to that

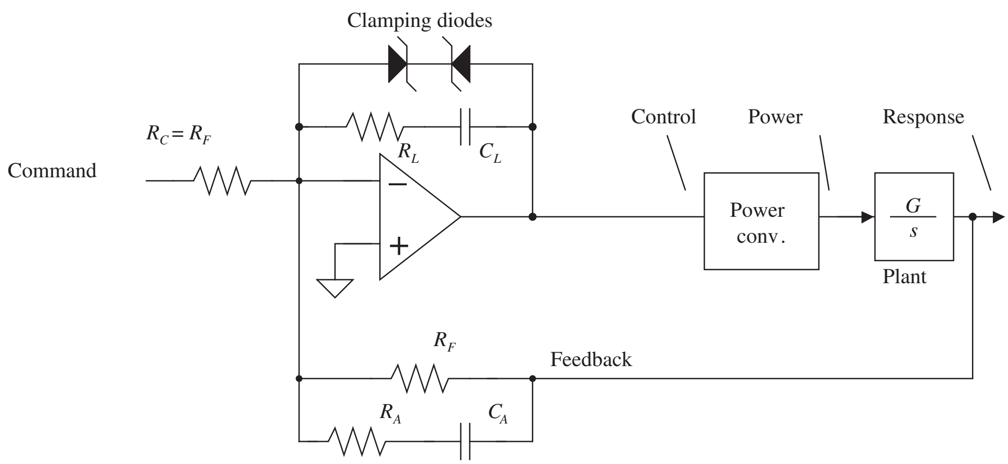

Figure 6.16:
Lead-lag schematic.

of a PID controller but does so with a single op-amp. The differentiation is performed only on the feedback with the capacitor C_A. The resistor, R_A, forms a low-pass filter on the derivative with break frequency of $R_A \times C_A/2\pi$ Hz. Because the differential gain is only in the feedback path, it does not operate on the command; this eliminates some of the overshoot generated by a fast changing command.

Tuning a lead-lag circuit is difficult because the tuning gains are coupled. For example, raising C_A increases the effective differential gain but also increases the proportional gain; the derivative from C_A is integrated through C_L to form a proportional term, although the main proportional term is the signal that flows through R_F to R_L. Lead-lag is often not used in digital controls because numerical noise caused by the lead circuit (here, R_A and C_A) is fed to the integral (here, C_L); such noise can induce DC drift in digital systems, which could be avoided with the standard PID controller. On the other hand, lead circuits are sometimes used by digital designers to a larger extent than is practical in analog lead circuits. For example, multiple digital lead circuits can be placed in series to advance the phase of the feedback to increase the phase margin; this is usually impractical in analog circuits because of noise considerations.

Tuning a lead-lag controller (Figure 6.17) is similar to tuning a PID controller. Set R_A as low as possible without generating excessive noise. Often, R_A will be limited to a minimum value based on experience with noise; a typical value might be $R_A \geq R_F/3$. When tuning, start with a proportional controller: Short C_L and open C_A, raise R_L until the system just overshoots, and then raise it, perhaps 30% (how much depends on R_A, because lower R_A will allow C_A to cancel more overshoot from a large R_L). Start with low C_A and raise it to cancel overshoot. Then set C_L to a high value and reduce it to provide a predetermined amount of overshoot.

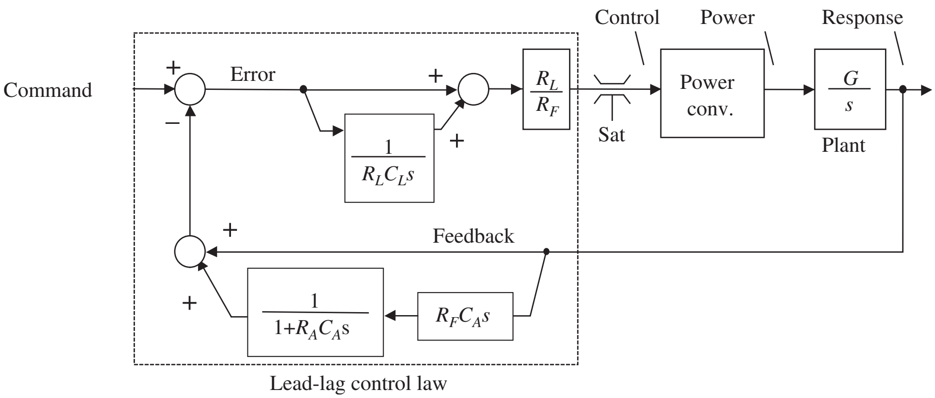

Figure 6.17:
Alternative controller 4, a lead-lag controller.

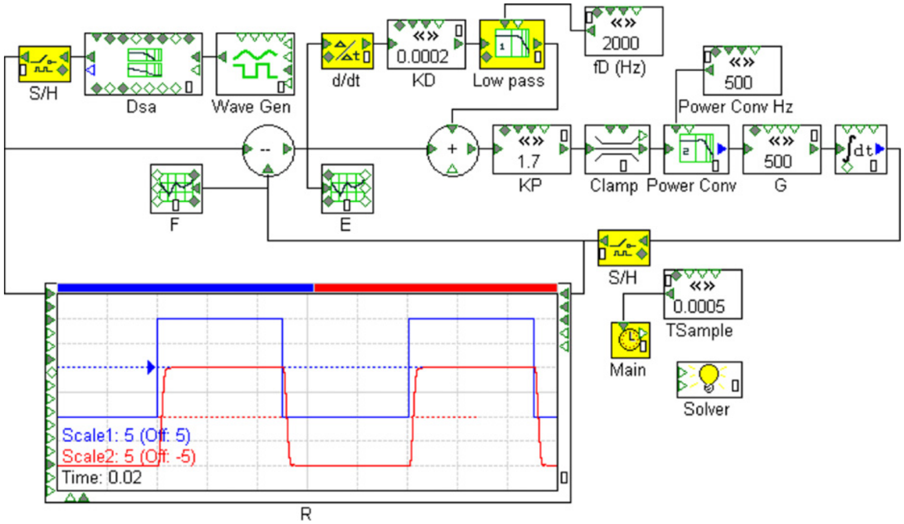

Figure 6.18:
Experiment 6D: A PD controller.

6.5 PD Control

The fourth controller covered in this chapter is a PD controller; the P controller is augmented with a D term to allow the higher proportional gain. The controller is shown in Figure 6.18. It is identical to the PID controller with a zero I gain.

6.5.1 How to Tune a PD Controller

Tuning a PD controller (Figure 6.19) is the same as tuning a PID controller, but assume K_I is zero. The effects of noise are the same as those experienced with the PID controller.

The results of tuning are shown in Figures 6.18, 6.20, and 6.21 using Experiment 6D. The step response is almost square. The introduction of the D gain allowed the P gain to be raised from 1.2 to 1.7. This allows much higher bandwidth (353 Hz for the PD controller compared with 186 Hz for the P controller), although the closed-loop phase lag at that bandwidth is much higher (162° for the PD controller compared with 110° for the P controller). As with the PID controller, the PD controller is faster, but more susceptible to stability problems. Also, the GM is smaller (8.8 dB, 3 dB lower than for the P controller). The PD controller is useful in the cases where the fastest response is required.

6.6 Choosing the Controller

The results of tuning each of the four controllers in this chapter are tabulated in Table 6.3. Each has its strengths and weaknesses. The simple P controller provides performance suitable for

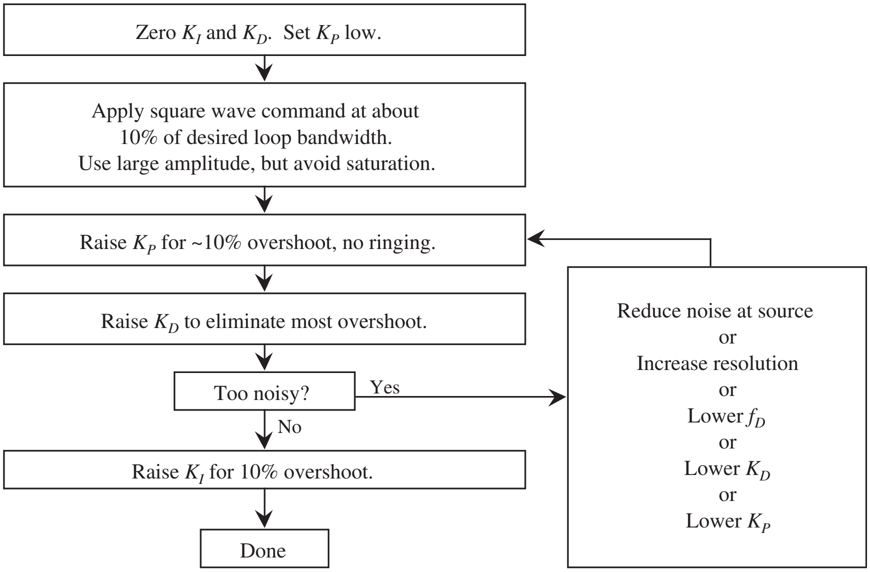

Figure 6.19:
Tuning a PD controller.

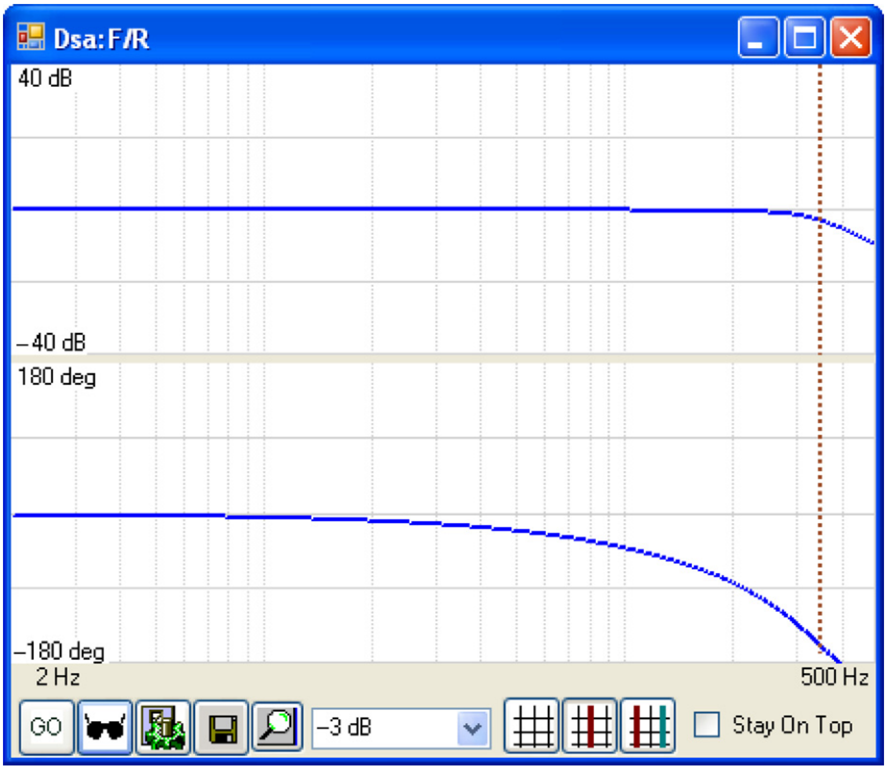

Figure 6.20:
Closed-loop Bode plot of a PD controller (353-Hz bandwidth, 0 dB peaking).

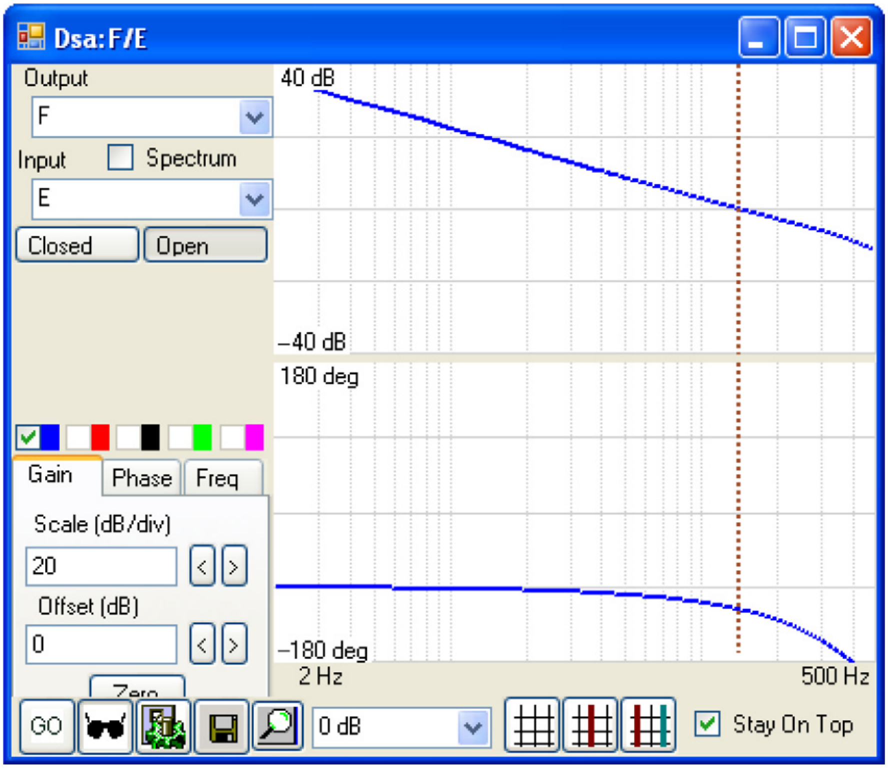

Figure 6.21:
Open-loop Bode plot of a PD controller (63° PM, 8.8dB GM).

Table 6.3: Comparisons of the Four Controllers

	P	PI	PID	PD
Overshoot	0%	15%	15%	0%
Bandwidth	186 Hz	206 Hz	359 Hz	353 Hz
Phase lag at BW	110°	126°	169°	162°
Peaking	0 dB	1.3 dB	1.0 dB	0 dB
PM	65°	56°	55°	63°
GM	12.1 dB	11.7 dB	8.5dB	8.8 dB
K_P	1.2	1.2	1.7	1.7
K_I	—	100	120	—
K_D	—	—	0.0002	0.0002
Visual ModelQ Experiment	6A	6B	6C	6D

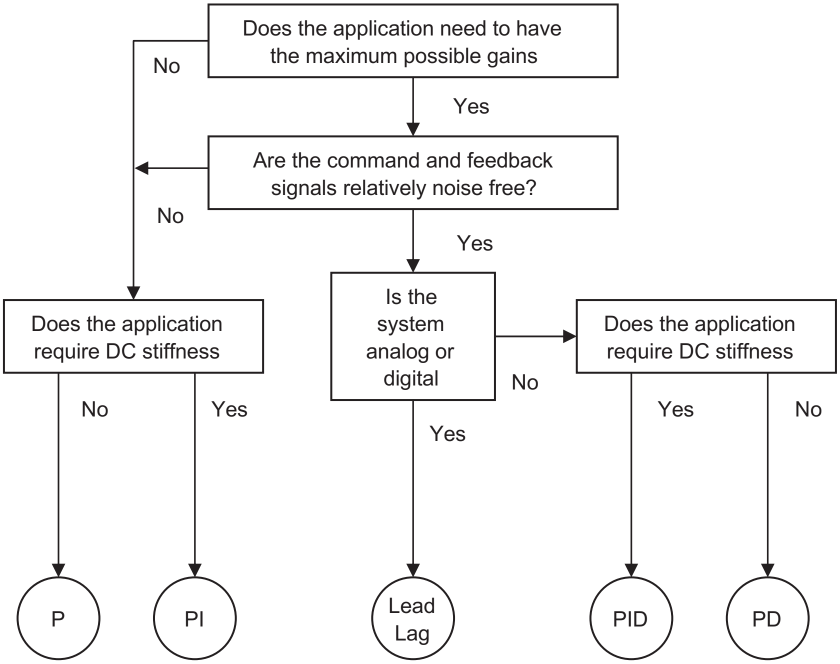

Figure 6.22:
Selecting the controller.

many applications. The introduction of the I term provides DC stiffness but reduces PM. The D term provides higher responsiveness but erodes gain margin and adds phase shift, which is a disadvantage if this loop is to be enclosed in an outer loop.

The chart in Figure 6.22 provides a procedure for selecting a controller. First determine whether the application needs a D gain; if not, avoid it, because it adds complexity, increases noise susceptibility, and steals gain margin. If so, make sure the application can support D gains; systems that are noisy may not work well with a differential gain. After that, examine the application for the needed DC stiffness. If none is required, avoid the integral gain. If it is needed, use PI or PID.

6.7 Experiments 6A–6D

All the examples in this chapter were run on *Visual ModelQ*. Each of the four experiments, 6A–6D, models one of the four methods, P, PI, PID, and PD, respectively. These are models of digital systems, with sample frequency defaulting to 2 kHz. If you prefer experimenting with an analog controller, set the sample time to 0.0001 second, which is so much faster than the power converter that the power converter dominates the system, causing it to behave like an analog controller.

The default gains reproduce the results shown in this chapter, but you can go further. Change the power converter bandwidth and investigate the effect on the different controllers. Assume noise is a problem, reduce the low-pass filter on the D gain (f_D), and observe how this reduces the benefit available from the derivative-based controllers (PID and PD). Adjust the sample time and observe the results.

6.8 Questions

1. Retune the proportional controller of Experiment 6A with the power converter bandwidth set to 100 Hz. Use the criteria of Chapter 6 (no overshoot for proportional gain, etc.). Measure the overshoot, peaking, and bandwidth of the closed-loop system; measure the PM and GM of the open-loop system.
2. Repeat Question 1 for the PI controller of Experiment 6B.
3. Repeat Question 1 for the PID controller of Experiment 6C.
4. Repeat Question 1 for the PD controller of Experiment 6D.
5. For the tuning procedures used in Questions 1−4, list two steps you could take to make the tuning more aggressive (that is, producing smaller margins of stability).