Experiment #6: Proportional-Integral-Derivative Control

PID is an acronym for Proportional-Integral-Derivative Control. In this section, we will explore each of these methods and how they work together to efficiently control a system.

Proportional Control

In the previous exercise, we used various means of cycling the heater of our incubator to maintain a desired temperature. Using differential gap, we created an allowable band in which the heater would cycle, causing the temperature to cycle above and below the setpoint. In Experiment #4, we saw how we could use pulse-width modulation (PWM) to add energy to our system in duty cycles between 0% (fully off) and 100% (fully on). In proportional control, we will combine a defined band of temperature and closed-loop feedback to adjust the PWM to our heater, maintaining a desired setpoint.

The first important step to understanding proportional control is to acknowledge that every system has both energy gains and energy losses. In our incubator, we add energy to the system by heating our resistor. The energy lost from our system is due to ambient losses. Heat changes in our container due to the temperature of the surrounding air. Consider another system, such as a fluid-flow system. Suppose we want a specific flow of oil at 10 gallons per minute. Energy is added to the system by a pump used to push the oil. A major loss in the system is from the restriction of the pipes to the oil flow. Friction between the oil and the pipe walls removes flow energy. In our automobiles, the engine adds energy to propel our car. The friction of the tires on the pavement, gravity and wind flow around the car all affect energy levels.

When the energy added to the system equals the energy lost, the system is balanced. In the incubator, if a 50% PWM added sufficient energy to make up for all the losses in the system, it would be balanced and the temperature would remain constant. If our losses exceeded the energy added to the system, the temperature would decrease. How far would it decrease? Until it found a new equilibrium of loss equaling gain. As our system cools less, there is less of a temperature difference between the inside and outside of the canister. The losses will decrease until they meet and equal energy added. Conversely, if gains are greater than losses, the temperature will increase until a new equilibrium is found at a higher temperature.

Think about when you are driving. You press the accelerator to add sufficient energy to propel the car to 65 MPH. On a flat, straight road, there may not be any need to change how far the accelerator is pressed to maintain speed. What happens when you start up a hill? Since your car is inclined, there is greater energy lost due to gravity pulling it. Without a change in how far the accelerator is pressed, the car begins to slow. Does it eventually come to a full stop? No, because the amount of energy from the engine reaches a point where it equals the new amount of losses, allowing a slower but stable speed. The system is back in equilibrium.

Let's think about our incubator operating in equilibrium. If using a PWM duty cycle (drive) of 50% adds exactly the amount of heat being lost by the system, the temperature will remain constant. What temperature will that be? That is dependent on the losses. Let's say the room temperature is 90 F and a PWM of 50% drive stabilizes the temperature at 101 F. If room temperature drops to 80 F, with the drive still at 50%, temperature might stabilize at 95 F.

But we want the temperature to stay at 101 F. How do we get the temperature to return to this value? We have two options: reduce the losses of the system, or increase the energy added to the system. Trying to adjust losses isn't a satisfactory solution to our problem. It could mean altering conditions that might be difficult or impossible to change. In our driving example, removing the hill from the road so we wouldn't slow down might sound like a nice idea, but isn't feasible. What do we do instead? We add energy (fuel) to the system via the accelerator.

In proportional control, feedback is used to monitor the system. For average losses, the amount of energy needed to maintain the setpoint is known as the "bias." The bias energy will equal anticipated losses, providing a steady-state system at the setpoint. As our losses change, however, the system is driven above or below the biased setpoint to compensate. The difference between the setpoint and the actual system drive is called the "amount of error." The amount of energy added to or removed from the system is proportional to the amount of error.

In our automobiles, we can use a cruise control to automatically adjust the fuel to our engine to maintain a constant speed, such as 65 MPH. At equilibrium on a flat, straight road as we rest our foot on the cruise-controlled accelerator, its position does not change while maintaining this setpoint. This would be its biased position. As our car proceeds up a hill, however, the speed decreases for the same accelerator position. The cruise control senses through feedback an error between the setpoint speed and the actual speed. It then drives our accelerator down, increasing the amount of energy and increasing speed. The further we fall from our setpoint, the greater the error and the further down the accelerator goes. If the hill is long enough, speed will increase until we arrive back very near our setpoint speed. Once it crests the hill, the car will increase speed because of the position of the accelerator, driving us beyond the setpoint. This causes another error condition and the cruise control will 'let up' on the accelerator, allowing the car to slow until the accelerator is up back at its biased position to maintain 65 MPH.

We will control the incubator in the same fashion. For anticipated losses, we will drive the heater at a biased PWM to maintain a constant temperature. As losses change, such as a disturbance in the system, we will drive the heater at a higher or lower PWM, based on the distance from the setpoint, to bring the temperature back to the setpoint.

Let's say our setpoint is 101.5 degrees, and a bias of 50% PWM is sufficient to maintain this temperature. If the temperature drops to 101.0, giving a -0.5 degree error, do we want to drive at 100% PWM, or something lower, to bring the temperature back to the setpoint? Let's look at a concept known as Proportional Band.

Exercises

Exercise #1: Proportional Band

For our incubator example, we want to maintain a temperature of 101.5 degrees. The steadier we maintain this temperature, the healthier our eggs will be. It is allowable, though, to go 0.5 degrees above or below this setpoint. So our setpoint will be 101.5 with a band of +/- 0.5 degrees, or 101.0 to 102.0 degrees.

If any error correction reaches the upper or lower limits, we want to take full action to return temperature back to the setpoint. Any error between the setpoint and the limits will provide a proportional amount of drive action. Figure 6.1 is a graphical representation of the incubator temperature verses the amount of drive needed for the given band.

Figure 6.1: Temperature vs. Drive
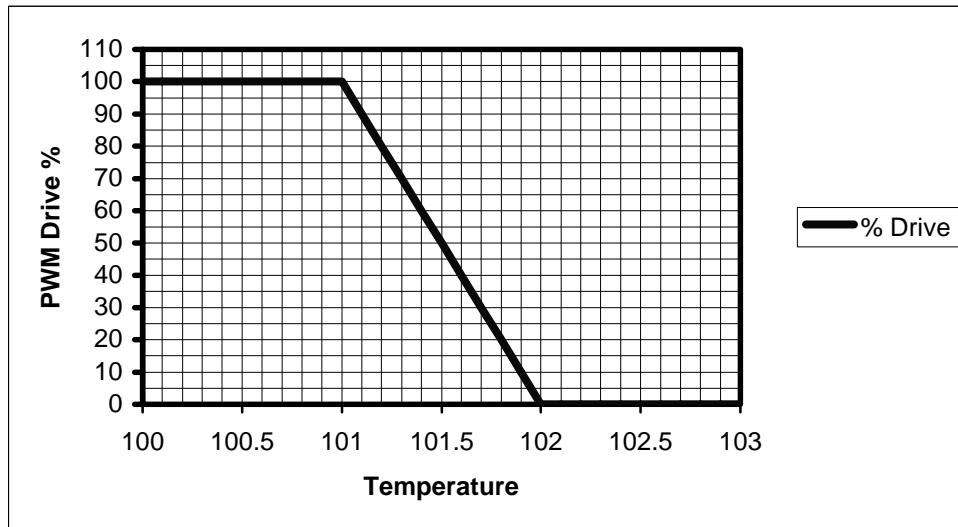
From Figure 6.1, if the temperature is at 101.5 F, 50% drive will be used to add heat energy to our system. If the temperature drops to 101 F, 100% drive will be used to increase the temperature, and any temperature in between will result in a proportional amount of drive.

If the temperature rises above 101.5 F, the drive will proportionally decrease to allow a lower temperature until at 102 F there is no drive (0%).

What would the drive be at 101.2 degrees? _____; at 101.7 degrees? _____.

This is termed a 100% Proportional Band because the drive covers 100% of our allowable band. Does this mean the temperature will not exceed our high and low limits? No. This simply means that at those limits our system will be taking full action to get the temperature back to the setpoint. The amount of drive that is added to or subtracted from the bias of 50% is the error drive. At 0.5 degrees away from the setpoint, we want our system to add or subtract 50% of drive to be fully on or fully off. To perform the math for this, we will need to consider the system gain. Over a change of one degree, we want to be able to change the drive 100%; our output change is 100% for an input change of one degree.

Proportional Gain = 100% / 1 degree

Since we have been working with tenths of degrees in the BASIC Stamp, for every tenth of a degree away from the setpoint, we will add or subtract 10% Drive for our error.

The Total Drive is the Bias Drive plus the Error Drive:

Total Drive = Bias Drive + Error Drive

The Error Drive is equal to the amount of error (Setpoint – Actual) multiplied by the Proportional Gain.

Error Drive = (Setpoint – Actual) x Proportional Gain

Let's try some numbers here. Assume that 50% bias is sufficient to maintain the incubator at 101.5 degrees under normal conditions. With normal conditions, the temperature would be 101.5 from the bias drive alone, thus no error.

Total Drive = Bias Drive + Error Drive

Total Drive = 50% + 0 % = 50%.

Looking at Figure 6.1, we see this is the correct drive for this situation.

Now, imagine we turn on a ceiling fan that blows air over our incubator, causing greater losses than expected. Energy removed is now greater than energy added, so the temperature drops. At 101.1 degrees, what amount of drive would our controller use to return the temperature to the setpoint?

Drive Error = (Setpoint – Actual) x Gain

Drive Error = (101.5 – 101.1) * 100 % = 40%.

Total Drive = 50% + 40% = 90 %.

Again, looking at Figure 6.1, we see this would be the correct amount of drive for the setpoint.

6

Now, let's turn off the ceiling fan and open a window so sunlight falls on our incubator. Losses would be less than anticipated, and the temperature would rise above the setpoint.

At 101.6 degrees, what would be the total drive of the system? _____.

Ok, enough talk! Let's look at this in our actual system. We have a few things to consider first. A real egg incubator is made with insulated materials and has high-energy heaters and stirring fans to equalize temperatures throughout the incubator. In our incubator:

- A bias of 50% PWM drive is insufficient to provide a temperature of 101.5.
- Every incubator will have a different stable temperature for 50% bias drive due to many factors.
- Our incubator is a fragile, steady state system. Factors such as room temperature and vents or fans blowing on the canister will shift the temperature.
- Moving or bumping the incubator will cause air mixing and affect the measured temperature.

Keeping all of this in mind, we will deviate somewhat from our incubator temperatures in testing out this section. You will use the program first to find the temperature at 50% bias drive for your incubator, and then use a band of that temperature +/- 1.0 degree for your experiments.

Note that this biased temperature band may shift over time because of varying conditions such as room temperature. You may occasionally need to redefine your bias drive.

Since our band is two degrees wide, the gain of our system will be:

Gain = 100%/2 degrees = 50%, or 5% drive change per tenth of a degree error

<u>Circuit Construction</u>

We will use the same circuit from Exercise #5, but we will connect the fan to Pin 19, the 5V supply, instead of Pin 20. In our control program, while individual drive values may exceed 0 and 100%, the total drive will not. Following is the full program for this section. We will change values in the program in testing the different areas of control.

```
'Program 6.1: PID Control with the StampPlot Interface

'Configure Plot
Pause 500                          'Allow buffer to clear
DEBUG "!RSET",CR                   'Reset plot to clear data
DEBUG "!TITL PID Control",CR       'Caption form
DEBUG "!PNTS 400",CR               '400 sample data points
DEBUG "!TMAX 600",CR               'Max 600 seconds
DEBUG "!SPAN 80,100",CR            '70-120 degrees
DEBUG "!AMUL .1",CR                'Multiply data by .1
DEBUG "!DELD",CR                   'Delete Data File
DEBUG "!CLMM",CR                   'Clear Min/Max
DEBUG "!CLRM",CR                   'Clear Messages
DEBUG "!USRS  ",CR                 'Clear User status bar
DEBUG "!SAVD ON",CR                'Save Data
DEBUG "!TSMP ON",CR                'Time Stamp On
DEBUG "!SHFT ON",CR                'Enable plot shift
DEBUG "!PLOT ON",CR                'Start Plotting
DEBUG "!MAXS",CR                   'Stop at Max
DEBUG "!RSET",CR                   'Reset plot to time 0


' Define constants & variables

CS          con   3               ' 0831 chip select active low from BS2 (P3)
CLK         con   4               ' Clock pulse from BS2 (P4) to 0831
Dout        con   5               ' Serial data output from 0831 to BS2 (P5)
Datain      var   byte            ' Variable to hold incoming number (0 to 255)
Temp        var   word            ' Hold the converted value representing temp
TempSpan    con   5000            ' Full Scale input span in tenths of degrees.
                                  ' Declare span.  Set Vref to .50V and
                                  ' 0 to 255 res. will be spread over 50
'(1/100ths).

Offset      con   700             ' Minimum temp. Offset, ADC = 0
```

```
' Declare zero Temp. Set Vin(-) to .7 and
' Offset will be 700 tenths degrees.
' At these 'settings, ADC output will be 0 - 255 for temps
' of 700 to '1200 tenths of degrees.

Error         Var    Word   ' Amount of drive due to error
Drive         var    Word   ' Amount of total drive
Integral      var    word   ' Amount of Integral Drive
Integral = 0

PWMCount      var    byte   ' Counter for amount time to apply PWM

LastTemp      var    Word   ' Holds last temperature for derivative drive
LastTemp = 0

Deriv  var    Word          ' Holds amount of Derivative Drive
Deriv = 0

IntCount      var    byte   ' Holds variable for counting cycles for integral drive
                            ' One cycle one PWM Time
PWMTime       Con    20      ' Number of repetitions for PWM, 20 = ~ 5 seconds

'********* ESTABLISH CONTROL SETTING ************

Bias          con    50      'Bias drive setting

Setpoint      con    924
'Initialize setpoint to YOUR bias Temp (mine was 92.4 F)

PropGain            con    5
'Set gain. 5 = 100% Proportional Gain, 1= 500%, 10 = 50%
IntTime       con    0
'How often to update integral amount in 5 second increments (6 = ~30 sec)
IntStep       con    0
'The amount to step integral drive each update
DerivGain     con    0
'Gain for each tenth degree change from previous


'************************************************
' Update Status text box
DEBUG "!USRS Setpoint:", DEC Setpoint, "  P-Gain:", DEC PropGain
DEBUG "  Int-Time:", DEC IntTime * 5,"  Int-Step:",DEC IntStep,"  Deriv-Gain:", DEC
DerivGain,CR
```

6

```
Main:
  GOSUB Getdata
  GOSUB Calc_Temp
  GOSUB Control
  GOTO Main

Getdata:                          'Acquire conversion from 0831
  LOW CS                          'Select the chip
  LOW CLK                         'Ready the clock line.
  PULSOUT CLK,10                  'Send a 10 uS clock pulse
                                  'to the 0831
  SHIFTIN Dout, CLK, msbpost,[Datain\8]         'Shift in data
  High CS                         'Stop conversion
RETURN

Calc_Temp:                        'Convert digital value to
  Temp = TempSpan/255 * Datain/10 + Offset      'temp based on Span &
RETURN                            'Offset variables.

Control:
  GOSUB PropCalc                  'Perform proportional error calcs
  GOSUB IntCalc                   'Perform Integral Calcs
  GOSUB DerivCalc                 'Perform Derivative calcs
  Drive = (Bias + Error + Integral + Deriv)     'calculate total drive
  IF Drive < 10000 then HighAdjust     'adjust for signed number exceeding 0-100
  Drive = 0                       'Min of 0

HighAdjust:
  If (Drive <= 100) Then DriveOK
  Drive = 100                     'Max of 100

DriveOK:
  Drive = 50        '***** TEMPORARY TO FIND INCUBATOR BIAS TEMPERATURE
  DEBUG DEC Temp,CR               'Plot and display data
  DEBUG "Temp:", DEC Temp, "  Bias:", DEC BIAS, "  Error:", SDEC Error
  Debug "  Int:", SDEC Integral, "  Deriv:", SDEC Deriv, "  Drive:",DEC Drive, CR
  FOR PWMCount = 1 TO PWMTime     'Pwm at 218 mSec for each PWMTime repetition
  PWM 8,drive * 255/100,218
NEXT
RETURN


'***********Proportional Drive Error
PropCalc:
  Error = Setpoint – Temp * PropGain  'Calculate error
Return
```

```
'********** Integral Drive
IntCalc:
  IntCount = IntCount + 1              'Add one to IntCount
  If IntCount = IntTime then IntAdjust
'Every IntTime counts (~5 seconds each) adjust Integral
Return

IntAdjust
  If Temp = Setpoint THEN IntDone:
'If temp > setpoint, subtract one step
  If Temp < Setpoint Then IntAdd
  Integral = Integral - IntStep
Goto IntDone

IntAdd:                               'Otherwise add it
  Integral = Integral + IntStep

IntDone:
  IntCount = 0                        'Reset integral counter
Return

'********** DERIVATIVE DRIVE
DerivCalc:

If LastTemp = 0 then DerivDone        'If first reading, skip
  Deriv = (LastTemp - Temp) * DerivGain
                                      'Calculate amount of derivative drive
                                      'based on the difference two temps
                                      'mutliplied by the deriv. gain
DerivDone
  LastTemp = Temp    'Store temp for next deriv calc
Return
```

6

Once the code is entered:

Enter and run the above program. Ensure the fan is pointed well away from the incubator.

1. Start StampPlot Lite and monitor the temperature.

2. Once the system is fairly stable, record the temperature at 50% Bias Drive.

3. In the control setting, make the Setpoint temperature the same as your bias-driven temperature. Remember, it is in tenths of degrees (924 = 92.4 degrees).

4. Remark out the line 'Drive = 50' in the Drive Heater section by placing an apostrophe in front of it.

5. Disconnect the StampPlot Lite program, download the new program, close Debug and reconnect on StampPlot Lite.

6. Reset the BASIC Stamp by pressing the reset button on the Board of Education.

After 60 seconds of running a graph (1st grid line), create a system disturbance by blowing on the incubator at very close range for 10 seconds and monitor the results.

Figure 6.2 shows the results of our test with a temperature biased at 92.4 degrees with a disturbance of 60 seconds. The analog values were adjusted to best display the plot.

Figure 6.2: Test Results with a Temperature Biased at 92.4 Degrees and a Disturbance of 60 Seconds
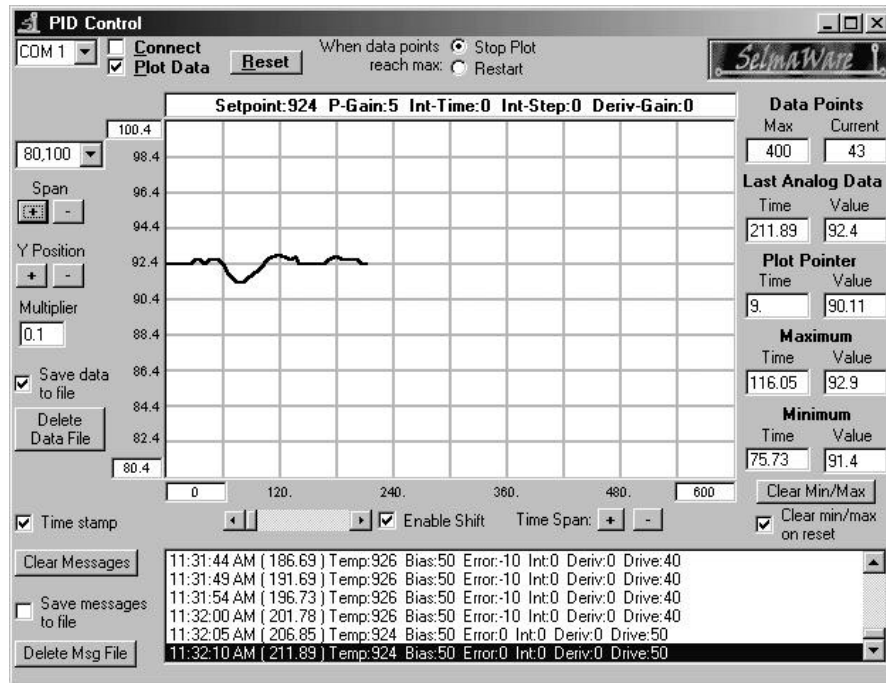
In the message section of StampPlot Lite, we can see the amount of Bias, Error, and (Total) Drive. In an example at 92.4 degrees, we see:

BIAS: 50   Error: 0   Drive: 50

The drive is only from the Bias, since we are at the setpoint.

At 92.6 degrees: BIAS, 50; Error, -10; Drive, 40. Since we are two tenths above, we have an error.

Drive Error = (Setpoint – Actual) x Gain
Drive Error = (92.4 – 94.6) * 50 % = -10%
Total Drive = 50% + (-10%) = 40 %

When the temperature drops below the setpoint, the error becomes positive, and the total drive rises above 50% in an effort to bring the temperature up. Figure 6.3 shows the amount of drive versus the temperature for our plot.

Figure 6.3: 100% Proportional Band with 92.4 F Setpoint, +/- 1 Degree Band
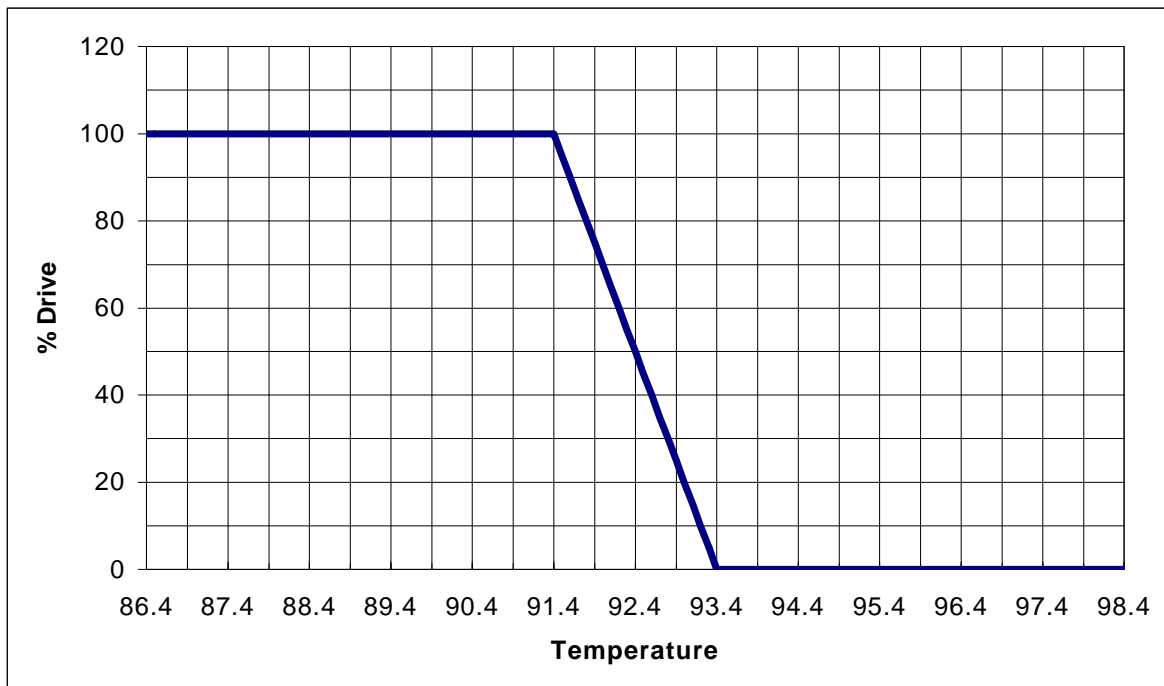
Notice that as the proportional control drives the heater to raise temperature, there is hunting. The temperature varies above and below the setpoint, hunting (adding and subtracting drive) until it finally stabilizes after several cycles.

The amount of drive we want to return to our setpoint is based on our proportional band and, therefore, our gain. For example, what if it was more important to stabilize our temperature than it was to return quickly and overshoot and undershoot getting there. In our cars, it would be an interesting ride if they sped up and slowed down repeatedly while re-adjusting for a new road condition!

In using a different proportional band setting, neither our setpoint nor limits change, we just set the gain so that we have the same amount of error drive over a greater range. For instance, in this example, our limits are still 91.4-93.4 degrees (92.4 +/- 1), but we will take full action over +/- 5.0 degrees. Full control would cover 500% of our allowable band.

Figure 6.4: 500% Proportional Band with 92.4 F Setpoint, +/- 5 Degree Band
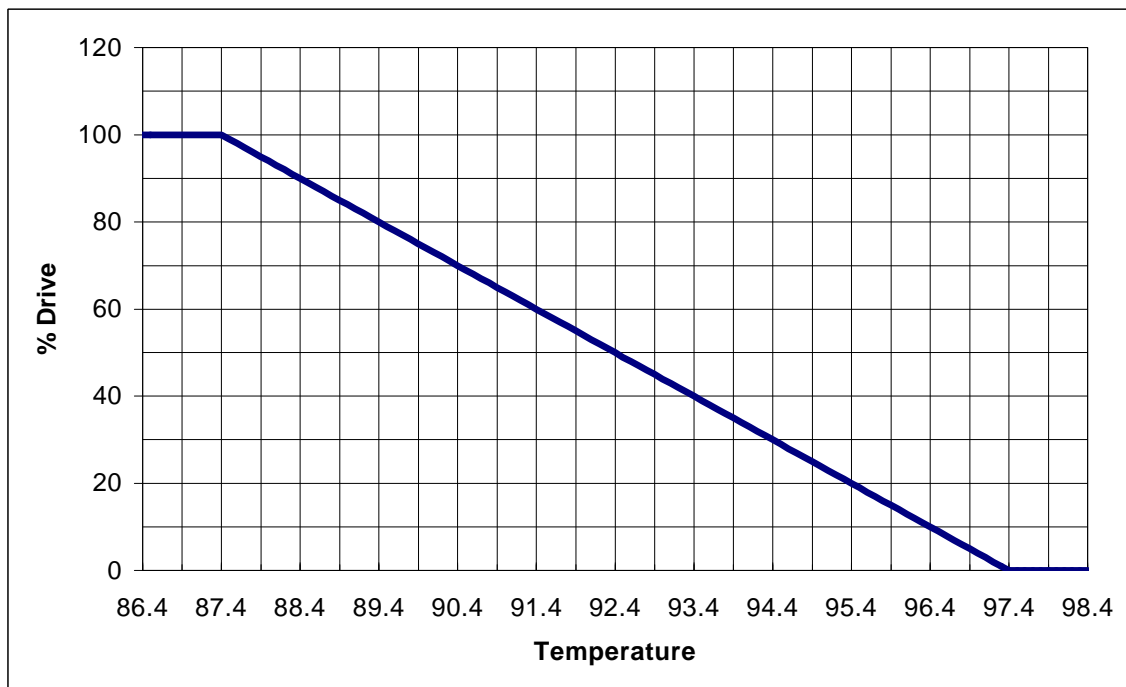
1. In your program locate `PropGain = 5` in the Control Settings section and change it to 1. (100% / 10 degrees = 10% or 1% per tenth of degree).
2. Re-run the plot with the same disturbance.

Figure 6.5 shows the plot of our control action. Notice that there is little overshoot and hunting, though it takes longer to reach our temperature since the drive for the same error is five times less.

Figure 6.5: System Response with 500% Proportional Band
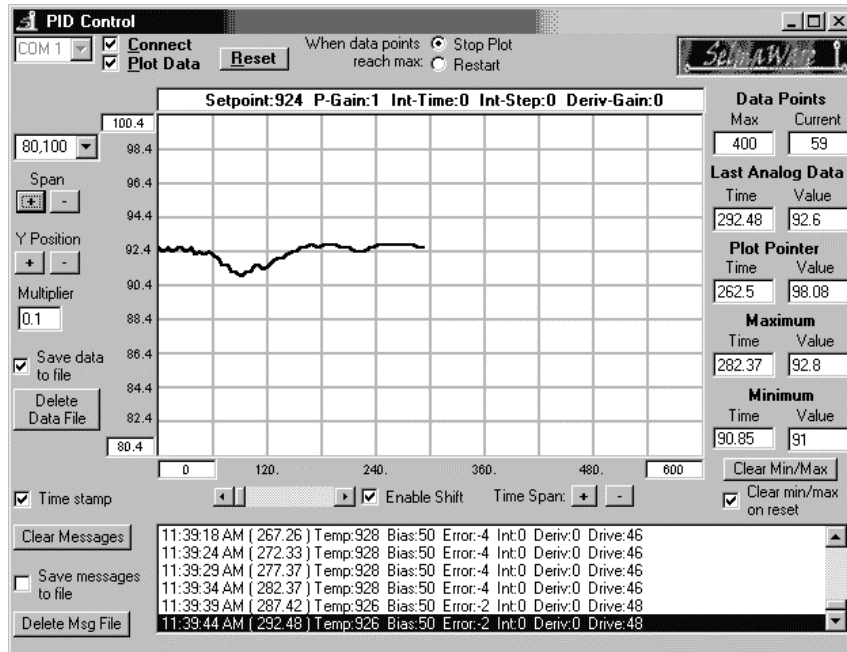
Conversely, let's say we needed faster response and overshoot wasn't a concern. We could take full drive over 50% of our allowable band, so that at +0.5 we would have 0% drive and at -0.5 we would have 100% drive. Figure 6.6 illustrates the Drive for 50% Proportional Band.

Figure 6.6: 50% Proportional Band with 92.4 F Setpoint, +/- .5 Degree Band
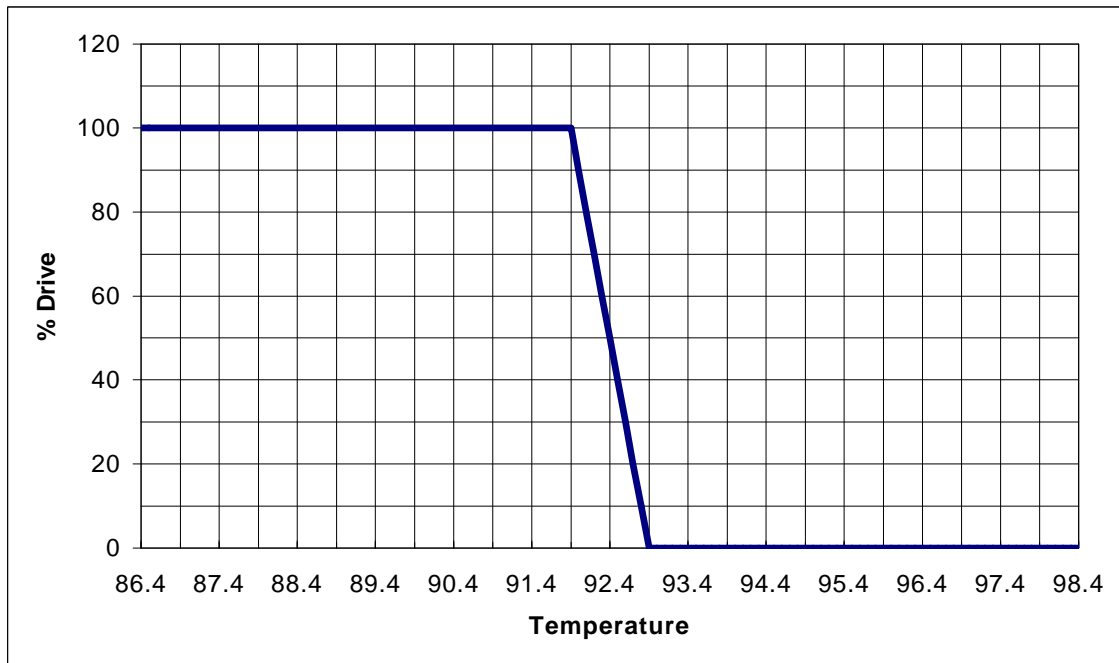
6

What would our new gain be? 100%/1 degree or 10%/tenth of a degree.

Once the plot is stable, change the gain to 10 and re-plot. Figure 6.7 shows our results. Notice that while response is faster, there is a lot more hunting before the temperature finally stabilizes.

Figure 6.7: System Response with 50% Proportional Band
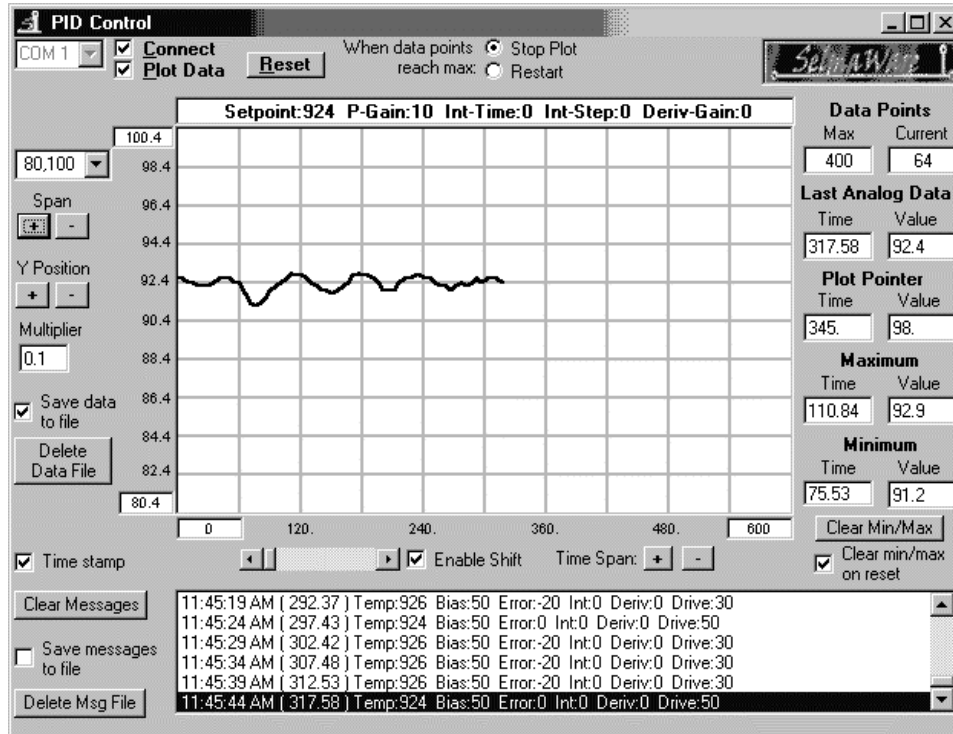
In system control, there can be too much of a good thing. When we try to drive a system faster than it can respond, the resulting action could destabilize the system and lead to undesirable behavior. Figure 6.8 shows a plot of our incubator using a 25% proportional band for a gain of 20. This means that for every one-tenth of a degree change, we adjust the drive by 20%!

Figure 6.8: System Response Using a 25% Proportional Band
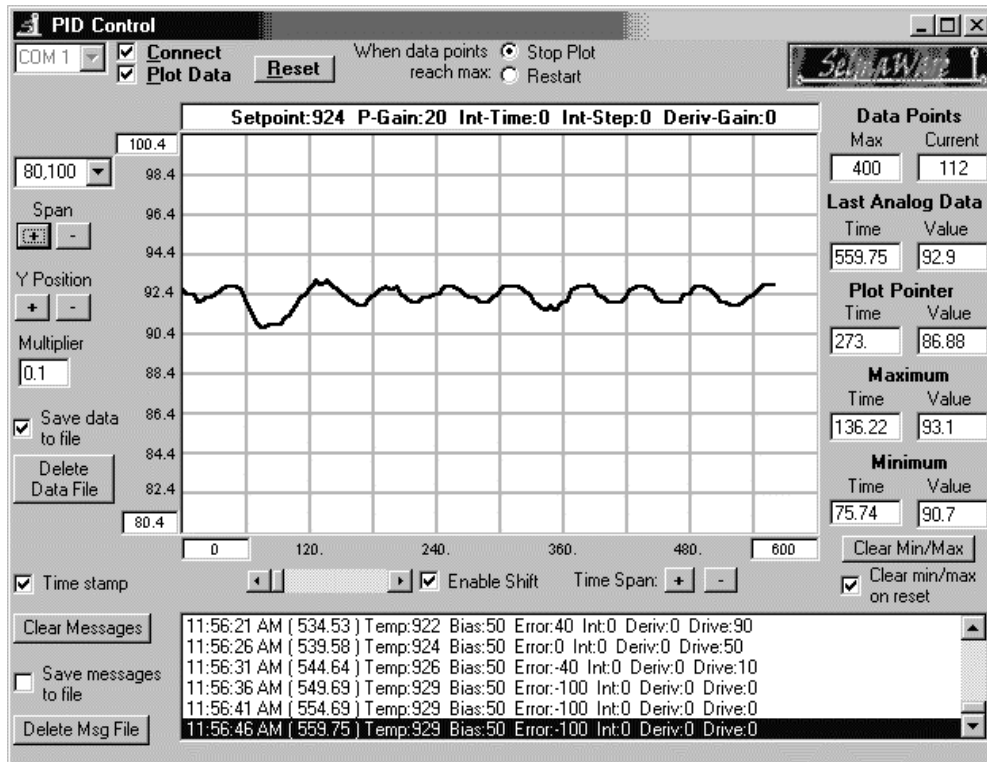
By driving the system faster than it can properly respond, the temperature not only hunts dramatically, but the oscillations begin getting larger, leading to system instability.

Challenge!

1. Adjust the system gain for a:
   - 100% Band
   - 200% Band
   - 50 % Band

2. Complete the following values for the Error Drive and Total Drive following a disturbance: (If the temperatures do not reach all values, fill in calculated values and circle)

Temp. from Setpoint	50% Band			100% Band			200 % Band		
	Bias Drive	Error Drive	Total Drive	Bias Drive	Error Drive	Total Drive	Bias Drive	Error Drive	Total Drive
+1.0									
+.8									
+.6									
+.4									
+.2									
0									
-.2									
-.4									
-.6									
-.8									
-1.0									

Exercise #2: Proportional-Integral Control

So far we've looked at what occurs when quick disturbances occur to our system in equilibrium. Proportional control may be used to drive the temperature back to the desired setpoint. But what happens when the disturbance is long enough to affect the equilibrium of our system over a long period of time?

Try this:

1) Setup Program 6.1 for a 500% Proportional Band (Gain of 1).
2) Allow the system to stabilize.
3) Point the fan at the incubator from a distance of about two feet (if you see no response after 30 seconds, move it closer in six-inch increments and try again).
4) Allow the system to stabilize with this new system loss.

Figure 6.9 shows our results of this test with an equilibrium setpoint of 92.4 F.

6

Figure 6.9: Long-Term Disturbance Effects
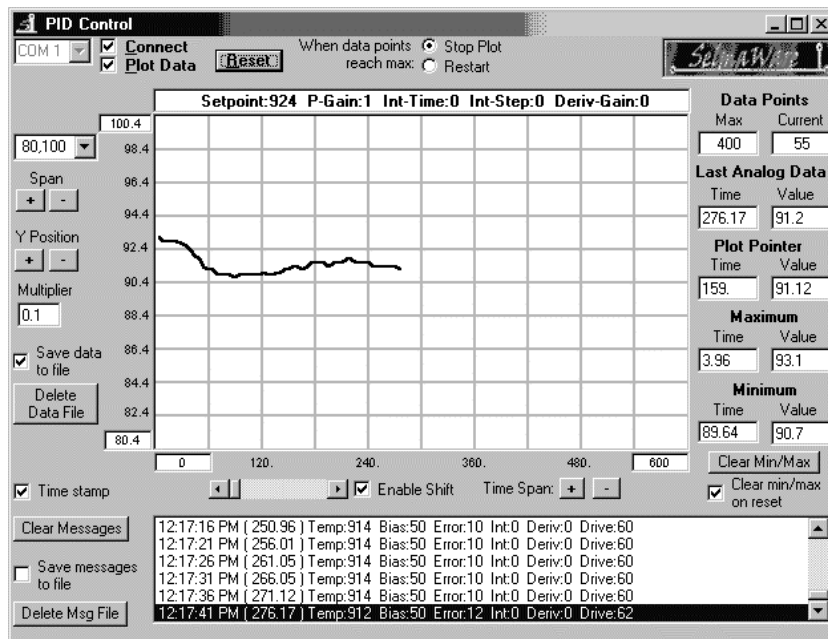
Notice that even with a drive of 62%, the temperature is approximately 1.2 degrees away from the setpoint with an error of 12%. With a short disturbance, this error would have been sufficient to drive the temperature back up. Now we have a long-term loss from our system and the error drive is insufficient to make up for it. As long as the system is out of equilibrium, where heat loss is greater than heat gain at 50% drive, the temperature will not be stable again at our setpoint.

If we were to increase the gain, to say 10, it would cause a greater drive and push the temperature closer to the setpoint, but there will always exist some error. Let's look at why. Let's say that we could reach the setpoint of 92.4 F. How much drive would there be? Since the Error drive would be 0, there would be 50%, and all from Bias. However, we know with our disturbance that 50% is insufficient to maintain that temperature; therefore, it would have to be less the 92.4 F, providing an error. If the temperature returns to the setpoint, there will be no error and thus no error drive, but we need the error to have sufficient energy to increase the temperature, so some error MUST remain. It can be confusing, but it has to work this way if you think about it.

With the long-term disturbance, what we reach is a new equilibrium where the energy added to meet the loss is equal to the bias + the Error drive at a lower temperature. In this case, the disturbance was a fan, but it may have been the temperature of the room changing over the course of the day, or even the season of the year. (Though it is easier to discuss greater losses, we could also have started at a lower room temperature and have it heated over the day, reducing our losses and causing our system to stabilize at a higher temperature with an error!)

So how do we maintain the desired setpoint in the face of long-term losses to our system? We use integral control. Integral control is used to slowly add to the total drive over a long period of time, driving the Actual value back to the setpoint and reducing the error.

Set the control settings to the following:

```
Bias        con   50          'Bias drive setting

Setpoint    con   YOURS       'Initialize setpoint to YOUR bias Temp in
                              'tenths of degrees

PropGain    con   1           'Set gain. 5 = 100% Proportional Gain,
                              '1= 500%, 10 = 50%
IntTime     con   6           'How often to update integral amount in
                              '5 second increments (6 = ~30 sec)
IntStep     con   2           'The amount to step integral each update
DerivGain con     0           'Gain for each tenth degree change from previous
```

We just added integral control to our process. In the `IntCalc` routine, every six repetitions (about 30 seconds), we adjust Integral drive by a step of two. If the temperature is below the setpoint, two will be added to the integral control. If above the setpoint, two will be subtracted. In this manner, the program will add or subtract drive over a long period of time, adjusting until it eventually drives away any error.

Run this program testing the long-term disturbance of the fan. Figure 6.10 shows our results.

Figure 6.10: Long-Term Disturbance Using Integral Control
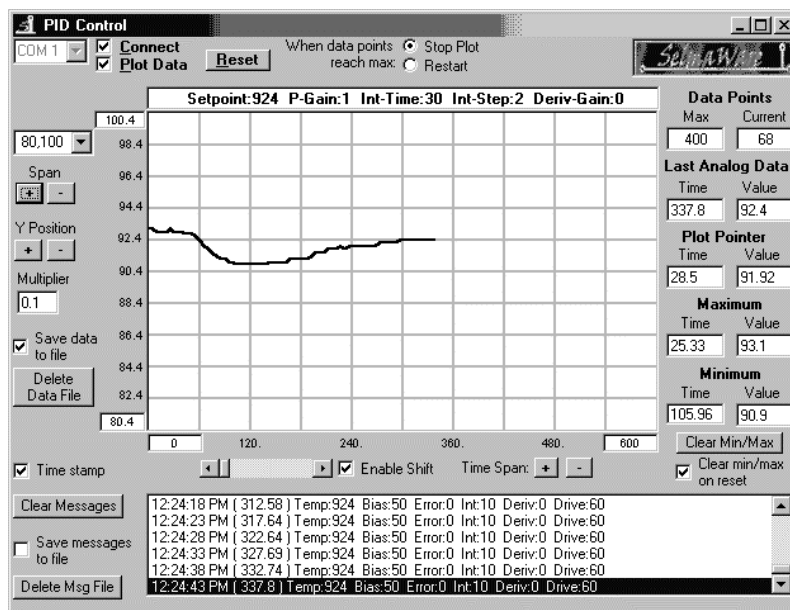
Notice that every 30 seconds or so (every half-division), the temperature bumps up slightly. At the end of the run, there is a 60% drive: 50% from bias, 0% from Error and 10% from integral.

Total Drive = Bias Drive + Error Drive + Integral Drive

Over long periods of time, as environmental temperatures change or perhaps our oil piping erodes and causes greater friction, integral control may be used to drive out these changes in equilibrium, returning the system to its desired operating point.

Of course, one drawback is what occurs when the long-term disturbance is suddenly removed. Figure 6.11 shows what occurs when the fan is turned off.

Figure 6.11: Control After the Disturbance Removal

When the disturbance leaves our system, all of that added integral drive still is present, causing the temperature to go way above the setpoint of 92.4 F. The error drive then has to compensate until, over time, the integral drive slowly is subtracted away.

Challenges!

1) If possible, heat the room to higher-than-normal temperature. Establish equilibrium with 50% bias at the higher room temperature. Return the room temperature to normal, and observe the effects on the system.

2) Change the amount of integral drive added (IntStep) from two to five. Test the system using the fan and notice the changes in the system's response time.

3) Increase the frequency adjustment rate for the integral drive (change IntCount from six to two). Test the system response.

Exercise #3: Derivative Control

In derivative control, we take action based on a change in readings over a very short time, typically from one reading to the next. This allows for an immediate amount of drive change to counter a rapid disturbance, quickly stabilizing the system.

Derivative Drive = (Change in temp) / (Change in time) x Derivative Gain

In our program's `DerivCalc` routine, we take the difference between the last temperature reading and the current temperature reading and multiply it by the derivative gain. Let's look at an example for a 500% proportional band (Gain = 1).

If a disturbance causes the temperature to drop from 92.4 degrees to 92.0 degrees, the amount of drive from the proportional error would increase the drive by only 4%. The rapid change in temperature, though, with a derivative gain of five, would add 20% drive instantly in an effort to dampen the sudden drop.

Last reading: 92.4. This reading: 92.0

Derivative Drive = (92.4-92.0)/1 unit * 5% = 20%

Total Drive = Bias + Error + Integral + Derivative

Total Drive = 50% + 4% + 0% + 20% = 74%

Remember, the amount of derivative drive is only based on the long-term change in temperature. If, at the next reading, the added total of 24% drive was sufficient to stop the sudden drop, and the reading was again 92.0 degrees, the only additional drive would be from Proportional at 4%.

Last reading: 92.0 This reading: 92.0

Derivative Drive = (92.0-92.0)/1 unit * 5% = 0%

Total Drive = 50% + 4% + 0% + 0%

If there were a sudden increase from 92.0 to 92.6, what would the total drive be?

Last reading: 92.0. This reading: 92.6

Derivative Drive = (92.0-92.6)/1 unit * 5% = -30%

Total Drive = Bias + Error + Integral + Derivative

**Total Drive = 50% + 2% + 0% + (-30%) = 22%**

As we can see, if the temperature suddenly jumped from below the setpoint to above it, the amount of derivative drive would be negative for the instant, setting the PWM drive very low in an effort to slow the rapid change.

Let's look at some examples using StampPlot Lite and our incubator. Configure your program for the following 100% Proportional band (PropGain = 5), the integral drive, and a gain of five for Derivative drive, and run your plot with a 10-second blow disturbance.

Figure 6.12 shows the result of our test. Compare it to Figure 6.2.

6

Figure 6.12: PID Control: 100% Proportional Band
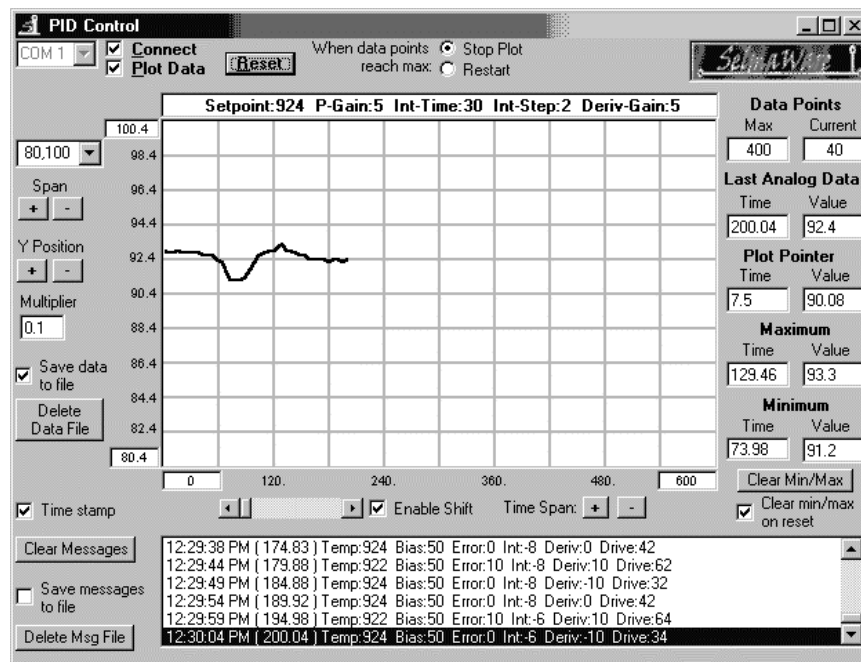
Compared to Figure 6.2, the sudden drop turned more quickly and the plot stabilized faster. Note in the message window the small fluctuations in the final temperature, creating derivative drive to try and stabilize the system.

Look back at Figure 6.8, where we plotted only proportional control with a gain of 20. The temperature oscillated endlessly, trying to stabilize. In your program, change the Proportional Gain to 20 and the derivative gain to 10 and try it again.

Figure 6.13 shows the results of our test. Revise your program as follows:

```
Bias            con   50     'Bias drive setting

Setpoint        con   YOURS  'Initialize setpoint to YOUR bias Temp in tenths

PropGain        con   5      'Set gain. 5 = 100% Proportioanl Gain,
                             '1= 500%, 10 = 50%
IntTime         con   6      'How often to update integral amount in
                             '5 second increments (6 = ~30 sec)
IntStep         con   2      'The amount to setp integral each update
DerivGain       con   5      'Gain for each tenth degree change from previous
```

Figure 6.13: PID Control for 25% Band
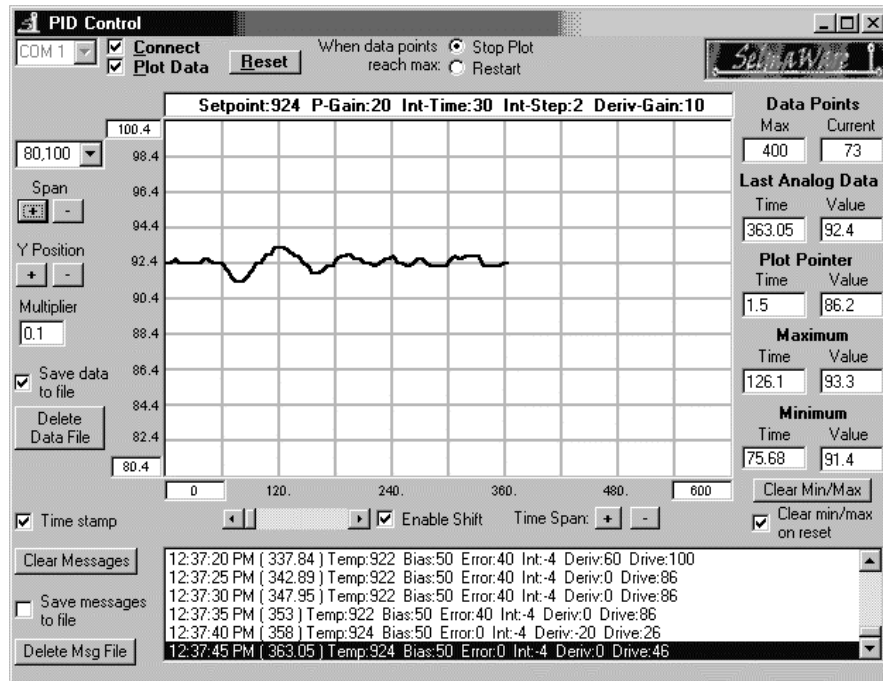
In this example with derivative control, the temperature more quickly stabilized as the derivative gain helped dampen overshoot and undershoot.

Once again, too much of a good thing can be bad. Improper derivative gains can totally destabilize a system where slight disturbances can dramatically affect the control. We didn't have any luck stabilizing our slowly responding system with derivative control. Can you?

Challenge!

For a 100% Proportional band, test various derivative gains with a disturbance. What setting allows the fastest return to the setpoint with minimal overshoot?

Proportional, Integral, Derivative Summary

Using PWM control, we can drive our system in small increments over a wide range. Through the use of proportional control, we can add to or subtract from that drive based on the error between the actual reading and the setpoint. By defining our band and adjusting the error gain, we can establish how quickly the system will take action based on the error. Small amounts of gain allow a slow, controlled return following a disturbance. High amounts of gain allow a faster return, though it may lead to instabilities.

Long-lasting disturbances prevent the actual value from reaching the setpoint because of the imbalance and the error drive needed to make up for the disturbance. Using integral control over long periods of time, we can add or subtract drive to compensate for the error and return the system to the setpoint.

Rapid disturbances can be effectively dampened through the use of derivative control. Derivative control adds to or subtracts drive based on changes in readings over a very short time. Effective use of derivative control can rapidly stabilize systems.

As a final example, consider that some inertial navigation systems use a tiny ball spinning at thousands of revolutions per second while suspended in a magnetic field to measure the speed and direction of a craft. Imagine the PID control employed to keep the ball suspended and rotating. Improper control would cause the ball to slam against the enclosure, destroying it. Not all systems are as slow as our incubator. Some must react to changes in the system within hundredths or millionths of a second. Proficiency at tuning some systems for proper PID control can take years of education and experience. Entire volumes have been written on the subject. Improper control of a system can lead to expensive, and sometimes disastrous, consequences.

Questions and Challenge

1. Would on/off control of the system be suitable for PID control? Explain.

2. Which type of control, proportional, integral, or derivative, would be best suited for the following?

   a. To return a system to the setpoint based on the difference between Actual temperature and the setpoint due to a short-lived disturbance: _____.
   b. To minimize the effect a quick disturbance has on the system: _____.
   c. To reduce the effect a long-term disturbance has on the system: _____.

3. A system has a setpoint of 101.5 degrees, and an allowable band of +/- 0.5 degrees. For a 50% proportional band, what would be the proportional gain? _____.

4. A system has a setpoint of 101.5 with a gain of 10%/0.1 degrees. If the Actual temperature were 101.2, what would be the drive due to proportional error? _____

5. A system increments the Integral drive by 1% every two hours. If a long-term disturbance creates an error of 8%, how long before the error is driven away? _____.

6. A system has a derivative gain of 10%/ 0.1 F. If the temperature dropped from 101.8 to 101.3 between readings, with a setpoint of 101.5, what would be the error due to derivative drive? _____.
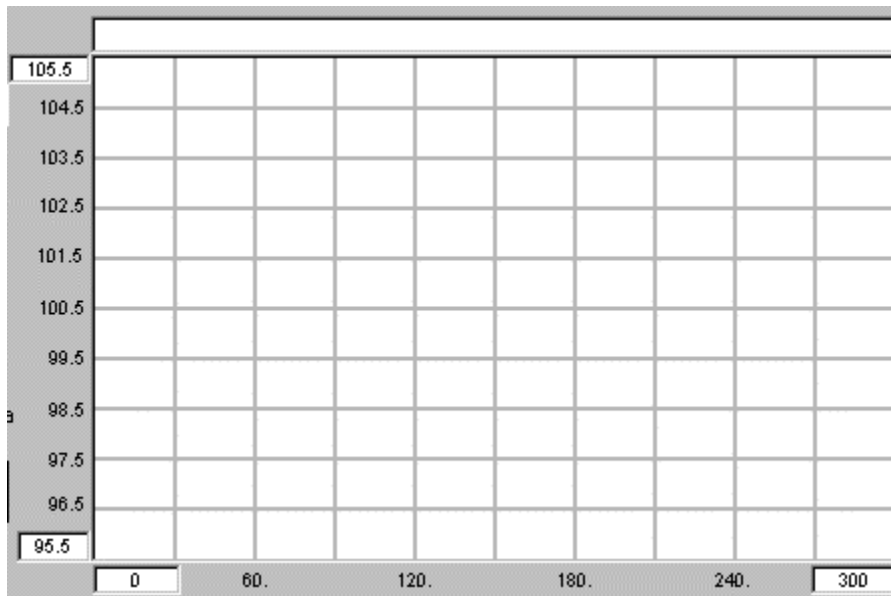
Final Control Challenge

From a cold condition (incubator at room temperature), find the values of PID control which will bring the incubator to an operating temperature of 95 degrees the quickest with minimal overshoot and hunting. Graph and record your results (note the graph scales):

PropGain: _____   IntTime: _____   IntStep: _____   DerivGain: _____

Graph your results:

Time first reached 101.5: _____   Maximum value reached: _____

Next minimum reached: (Time)_____   (Value) _____

Find a system:
Find an example of a system that either does or could employ PID control. Discuss how PID control may be implemented to control it.