Introduction to the Proportional Integral Derivative Controller

Course No: E03-045
Credit: 3 PDH

Jeffrey Cwalinski, P.E.

CED engineering.com
PDH FOR THE PROFESSIONAL

Continuing Education and Development, Inc.

P: (877) 322-5800
info@cedengineering.com

Introduction to the Proportional Integral Derivative Controller

Table of Contents

List of Figures

List of Tables

1.0 INTRODUCTION

In an industrial process a parameter such as temperature or flow may need to be regulated to a certain value to ensure that some end product meets specifications. This regulating is done by a device called a controller. The specific type of controller that will be considered here is the Proportional Integral Derivative 'PID'. The PID controller acts on an error signal, the difference between the desired value and the output of a process. The output of the PID controller generates a signal that is applied to a control device. The control device responds in such a way to cause the output of the process to match the desired value. Also, the PID controller compensates for unexpected changes in the system that otherwise would cause the output of the process to unacceptably deviate from the desired value.

Since control system analysis uses Laplace equations a knowledge of Laplace equations and transforms is needed, as well as, knowledge of block algebra.

2.0 OVERALL PID CONTROLLER DESCRIPTION AND COMPARISON TO ON/OFF CONTROLLER

The PID controller is part of a system that produces an output, called the Controller Output 'CO'. The Controller Output is sent to a control device, such as, a heater controller, valve position controller, or motor speed controller. The control device manipulates a physical parameter called the manipulated variable 'MV' such that it causes a change in that quantity being measured and controlled in response to a change in desired value called the Setpoint 'SP' or Disturbance. That quantity being measured and controlled is called the Process Variable 'PV'. Examples of a process variable could be temperature, pressure, or flow. A Disturbance is an undesired change to the system being controlled which causes the process variable to deviate from the setpoint. For example, loss of thermal insulation on a heat exchanger which is being kept at a constant temperature could cause temperatures to decrease. Eventually, the controller will adjust the manipulated variable such that the process variable matches the setpoint. The setpoint maybe operator entered or calculated by another device and input. If the setpoint is changed, then the controller will adjust the manipulated variable such that the process variable matches the setpoint.

Three constants called the Proportional Gain 'K_P', Integral Time 'T_I', and Derivative Time 'T_D', are adjusted to provide a specified rise time, overshoot, settling time, and steady state value of the process variable. The PID is a closed loop control system since the process variable, which is measured by a sensor, is fed back and compared to the setpoint to produce an Error signal '$e(t)$', which is the difference between the setpoint and process variable. This is different from an open loop control system where the value of the process variable depends only on the setpoint. That is, the value of the process variable does not influence the control actions. The PID controller applies a calculation using the error signal and the three constants. The equation adjusts the controller output which in turn adjusts the control device, which manipulates a physical parameter, and drives the process variable such that the error signal is zero.

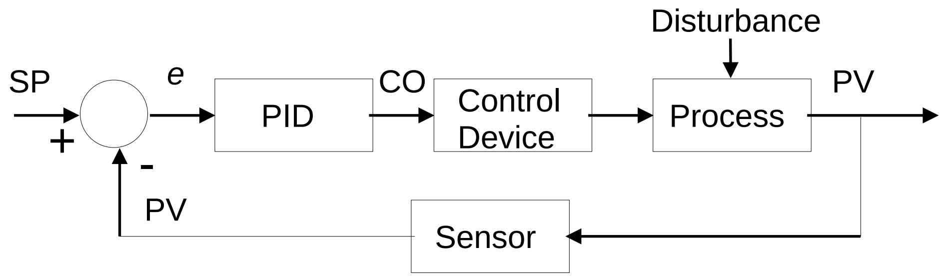


Figure 1 Simplified diagram of the PID controller system

Terms, such as, overshoot, rise time, etc. will have their engineering text book definitions. Also, for this discussion the value of the transfer function for the Sensor will be 1. In this case the Sensor block can be removed and replaced with a line so that the output, PV, is connected to the minus terminal of the summing node. It will always be assumed that the setpoint is a unit step function.

The PID controller can be pneumatic, such as the AMETEK-PMT Model 40 Pneumatic Pressure Controller; or analog electrical using operational amplifiers and discrete electrical components (resistors and capacitors), or panel mounted digital stand alone, such as the Eurotherm EPC 3000 programmable temperature controller, or implemented in a programmable logic controller (PLC).

There needs to be some discussion regarding the manipulated variable and control device since there is a possibility of confusion. Basic first principles need to be used to determine the manipulated variable. For example, if temperature is being controlled, then temperature is the process variable. Although the amount of heat is being changed, if electric heaters are used, it is the electrical current thru the heaters that needs to be manipulated to change the temperature. A panel mounted, stand-alone device, such as the Watlow EZ-Zone PM Express controller, could monitor the temperature, contain the tuning constants, and perform the PID calculation. The result of the PID calculation is the controller output which could be a 4 to 20 mA signal and sent to a heater controller which is the control device. The heater controller could be a Watlow DIN-A-MITE A, which would control the current thru the heaters. For the 4 – 20 mA signal the value of 4 mA would correspond to the heaters being off, 0% power and current, and the value of 20 mA would correspond to the heaters being fully on, 100% power and current. As such, it is the current through the heaters that is being manipulated by a control device that receives an input signal from the PID controller. Another example would be a PID controller sending the controller output to a valve position controller, the control device, where the manipulated variable would be the position of the valve, fully shut, 0% open to fully open, 100% open. Another example would be a PID controller connected to a variable speed drive for a motor where the speed of the motor would be the manipulated variable: off, 0 RPM to full speed, 3600 RPM.

The PID controller is an alternative to the On/Off controller. An On/Off controller produces an output that is in one of two control states, On or Off, there is no intermediate state. An example of an On/Off controller would be a thermostat in an electric oven which could use a thermistor to sense the temperature and a contactor to turn the heaters on and off. The oven temperature is the process variable, the desired cooking temperature is the setpoint, the contactor is the control device, and the electrical current is the manipulated variable. When the oven temperature is below the setpoint the thermostat output actuates the contactor/heater combination, and electrical current causes the heaters to increase the temperature of the oven. When the oven temperature reaches the setpoint the contactor/heater combination is no longer actuated and electrical current no longer flows. This process repeats for as long as it is desired to cook whatever is in the oven.

While the On/Off controller is simple in nature it is oscillatory around the setpoint. For this example let us assume an oven temperature setpoint of 350°F. The temperature change from the electric heaters to the air in the oven and the temperature change from the air to the thermostat will not be immediate. When the oven is first turned on the temperature starts to rise from ambient room temperature. It will take time for the air temperature to affect the temperature of the thermostat. During this time the heaters will be on and the air temperature will rise above the setpoint. Eventually, the thermostat reaches the setpoint and turns the heaters off. Since the temperature of the air is above the setpoint heat will be transferred to the thermostat increasing its temperature above the setpoint. When the air starts to cool to the setpoint the thermostat turns the heaters on and it takes time for the electric heaters to heat up to temperature. During this time the air temperature continues to fall below the setpoint before eventually rising. For an oven set to 350°F oscillations between 325°F and off at 375°F were measured.

If the oscillations occurred at a high frequency the contactor would rapidly operate on and off, it would “chatter”. This could cause premature failure of the contactor. This can be mitigated by introducing a dead band, also known as a hysteresis loop, around which there would be no control action.

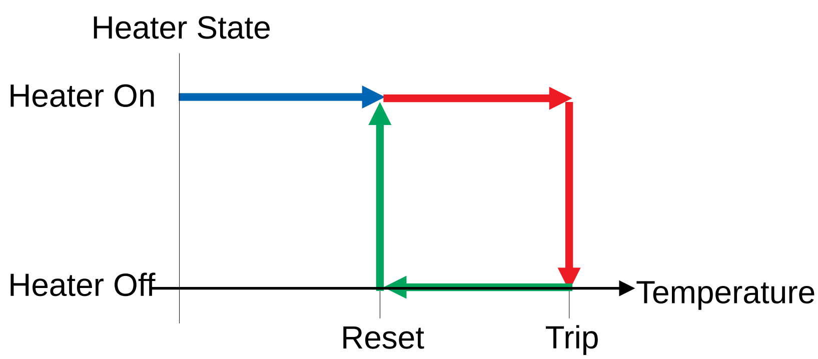


Figure 2 A hysteresis loop

The Trip temperature could be the temperature the oven is set at. The reset temperature could be set by the manufacturer as some number of degrees, say 25°F, below the trip temperature. For temperatures below the trip temperature the heaters would be on and the temperature would increase following the horizontal blue line to the horizontal red line. When the trip temperature is reached the heaters would turn off as indicated by the vertical red line. The temperature would decrease following the horizontal green line. When the reset temperature is reached the heaters would turn on as indicated by the vertical green line and temperature would increase. The cycle would repeat until the oven is turned off.

Less chatter is achieved by a greater difference between the trip and reset points but, the tradeoff is increased temperature variability.

In the above case, which is for a residential application, On/Off control is adequate. The deviations from the setpoint can be compensated for or ignored if they are small. There are processes where bringing a process variable to a setpoint quickly, with minimal or no overshoot, and maintaining the process variable at the setpoint is critical. One can make a reasonable argument that the manufacture of pharmaceuticals and very large scale integrated circuits are two of them.

While the PID does have its advantages compared to On/Off control it does have a disadvantage, it must be tuned. The proportional gain, integral time, and derivative time tuning constants must be determined. There are a number of methods for doing this, the Ziegler and Nichols method is one popular method. However, for the majority of the time an experienced operator that understands the system can determine tuning constants that produce minimum rise time, overshoot, settling time, and steady state error of the process variable.

3.0 THE PID EQUATION

The PID equation can be written in several forms. Three of the more common equations are below.

The Parallel Equation $CO(t) = K_p * e(t) + \frac{1}{T_i}\int e\,(t)dt + T_D\frac{de(t)}{dt}$

The ISA or Ideal Equation $CO(t) = K_p\left(e(t) + \frac{1}{T_i}\int e\,(t)dt + T_D\frac{de(t)}{dt}\right)$

The Series or Interacting Equation $CO(t) = K_P\left[\left(\frac{T_d}{T_I} + 1\right)e(t) + \frac{1}{T_I}\int e\,(t)dt + T_D\frac{de(t)}{dt}\right]$

As said, there are several variations of the equation and which variation used is dependent on the manufacturer of the controller. For example, some controllers manufactured by the company Eurotherm Ltd. use the ISA Equation. The ISA equation is used with the Ziegler and Nichols tuning method. There are many reasons given for the Series or Interacting Equation form. They range from backward compatibility when replacing pneumatic controllers with electronic controllers, to allowing a modular hardware design for each term in the equation. As such, the control engineer could determine which actions, proportional, integral, or derivative are required to control the process and buy the required hardware. For this discussion the Parallel Equation will be used.

The error term in the above equations is the difference between the setpoint and process variable$e(t) = SP(t) - PV(t)$. It can be positive, or negative, or zero which is the desired value.

A block diagram of the Parallel equation is below.

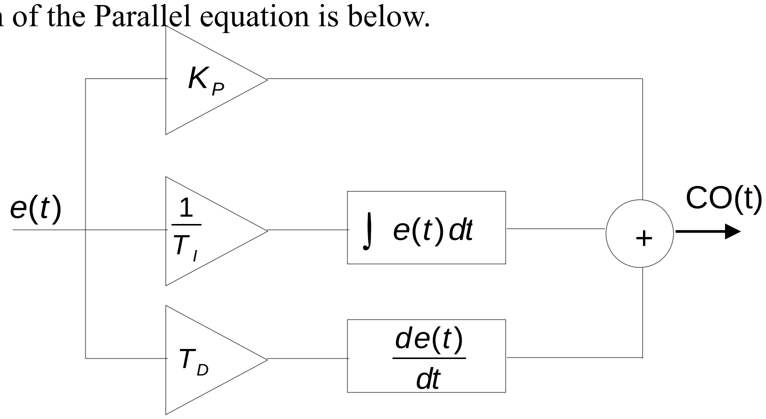


Figure 3 A block diagram of the Parallel equation

3.1. The Proportional Term

The first term of the equation $K_p * e(t)$, is called the proportional contribution and it relates to the magnitude of the error at the present time. The term $K_p$is called the Proportional Gain constant, it has no units. When it is multiplied by the error the product takes on the engineering units of the error. As an example, if *e(t)* is in meters then the product K_P**e(t),* will be in meters.

The PID controller can be operated as a proportional only controller, or P controller. The remaining two terms are removed from the diagram and equation. Considering only the gain and error the controller output equation is now $CO(t) = K_p * e(t)$. As the error gets greater the controller output, the controller output, gets larger and drives the control device to minimize the error. As the process variable approaches the setpoint the error decreases and the control device output decreases

At this point a question that maybe entering the readers mind is "Why can't proportional action alone bring the process variable to the setpoint? Why can't a large gain constant be used to quickly bring the process variable to the setpoint?" This is explained by the mass/spring/damper system shown below.

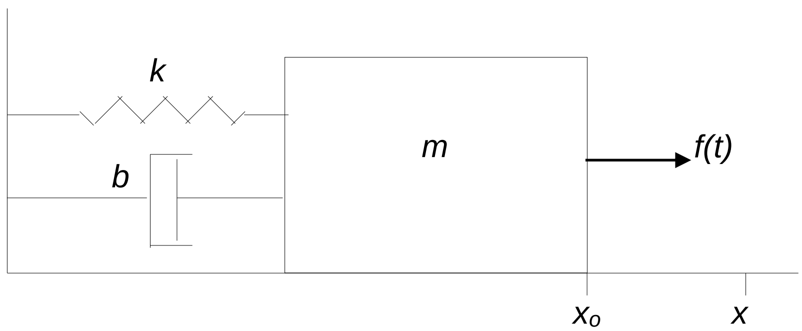


Figure 4 Mass/spring/damper system

The time domain equation for force is $f(t) = m\frac{d^2}{dt}x(t) + b\frac{d}{dt}x(t) + kx(t)$, where *m* is mass in kilograms (kg), *b* is viscous friction constant in Newton-seconds/meter (Ns/m), *k* is the spring constant, in Newtons per meter (N/m), *x* is distance in meters (m), *t*, time in seconds (s), and there is some type of control device that applies a force, *f(t)*, to the mass.

Applying the Laplacian to the time domain equation yields $F(s) = ms^2X(s) + bsX(s) + kX(s)$ and the transfer function, distance as a function of applied force is $\frac{X(s)}{F(s)} = \frac{1}{ms^2+bs+k}$.

The distance *x* is our process variable and x_o is the initial position. We will arbitrarily let a force of 6 Newtons be required to oppose the spring force and keep the mass stationary at any distance. It is our desire to have the control device apply a force to move the mass 2 meters to the right of x_o (that will be considered the positive *x* direction) and have the mass remain stationary at that position. Our setpoint is 2 meters. Since the force is being changed to control the distance, the force, *f(t)*, is the manipulated variable. The equation for the manipulated variable for this system is $f(t) = K_P * e(t) = K_P(2 - x(t))$.

We will let K_P be some arbitrary value greater than three. When the mass is in its initial position, x_o, the value of the process variable is 0 meters and the error is $e(t) = SP(t) - PV(t) = 2 - 0 = 2$ meters, which when multiplied by K_P will generate a force causing the mass to move. As the mass moves the error and subsequently the force are reduced. When the mass reaches 2 meters the error is zero, the product with K_P is zero, the force is zero, and the spring causes the mass to move back toward the initial position. As the mass moves to the left the error increases, the force increases, and the mass moves to the right. Eventually the mass will reach some distance, but what distance will that be?

If we solve for the process variable with a gain of, $K_P = 5$, the previously stated 6 Newtons, and setpoint of 2 meters, $6 = 5(2 - x(t))$, we get a distance of 0.8 meter. So, the mass will reach 0.8 meter, applied force will be 6 N, and the mass will remain at 0.8 meter when $K_P = 5$.

The spread sheet and graph below shows how the distance changes as the proportional gain increases.

Proportional Gain K_P	Distance m
3	0.00
4	0.50
5	0.80
6	1.00
7	1.14
8	1.25
9	1.33
10	1.40
11	1.45
12	1.50
13	1.54
14	1.57
15	1.60
16	1.63
17	1.65
18	1.67
19	1.68
20	1.70

Table 1 Spreadsheet showing the increase in distance with respect to proportional gain

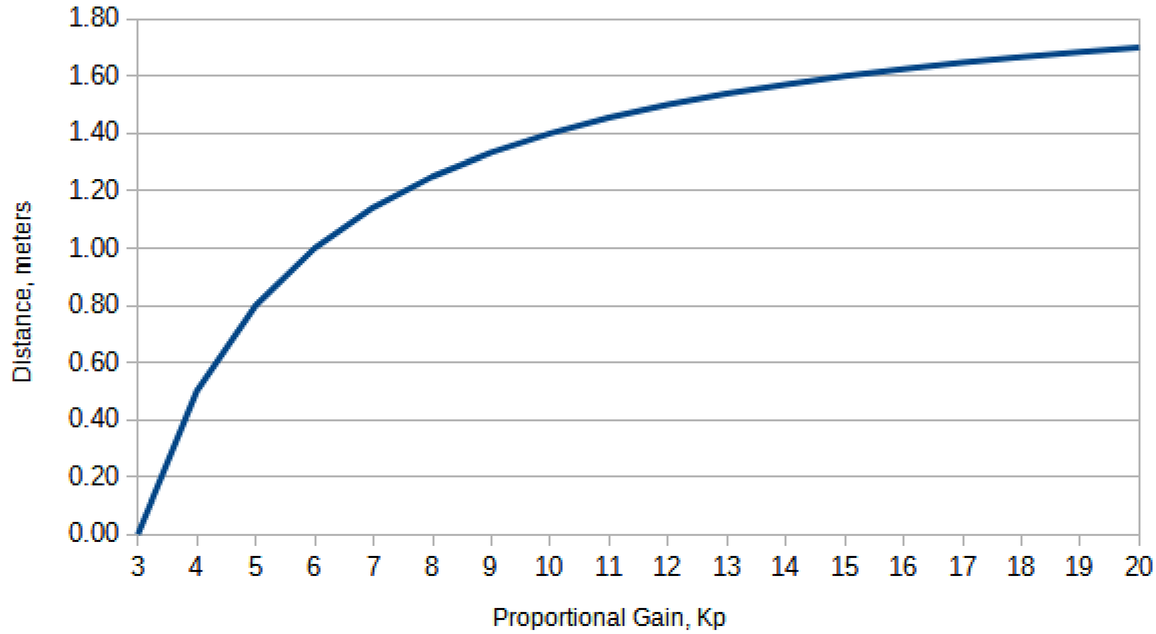


Figure 5 Graph showing the increase in distance with respect to proportional gain

As you can see the mass approaches 2 meters but never reaches it. There is a steady state error or offset.

The theory also predicts that there will always be a steady state error if only the proportional gain is used and we will find it using the system below. Steady State error is the difference between the setpoint and the actual value of the process variable in the limit as time goes to infinity.

For convenience let us write the Parallel PID equation again

$$CO(t) = K_p * e(t) + \frac{1}{T_i}\int e(t)dt + T_D\frac{de(t)}{dt}$$

We will apply the Laplacian operator to the entire Parallel PID equation to obtain

$$CO(s) = K_P E(s) + \left(\frac{1}{T_I}\right)\frac{1}{s}E(s) + T_D sE(s).$$

For information a block diagram of this Laplace equation is shown below

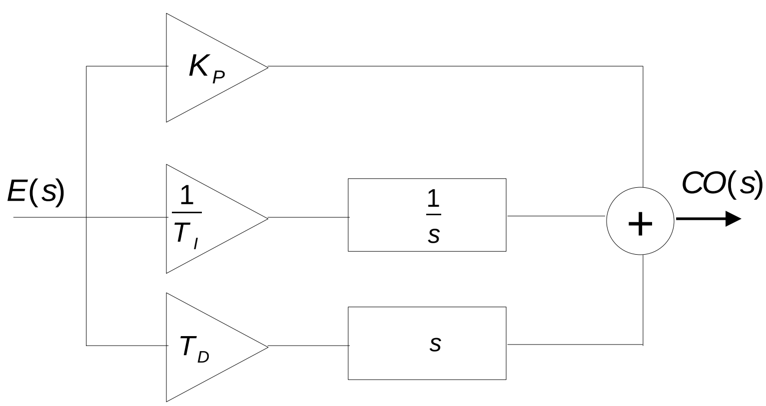


Figure 6 Block diagram of Laplace equation

Considering the proportional contribution only we have $CO(s) = K_P E(s)$. The integration and derivative contributions are removed from the diagram above. We now have a proportional only controller. It is not necessary to use all three contributions. Contributions that are not used are simply removed from the equation and diagram.

For this example, the transfer function of the mass/damper/spring will be used with m = 0.5 kg, b = 5.0 Ns/m, and k = 3.0 N/m. A block diagram of the proportional only controller which is controlling a process described by the equation, which was derived above, $\frac{X(s)}{F(s)} = \frac{1}{0.5s^2+5s+3}$ is shown below.

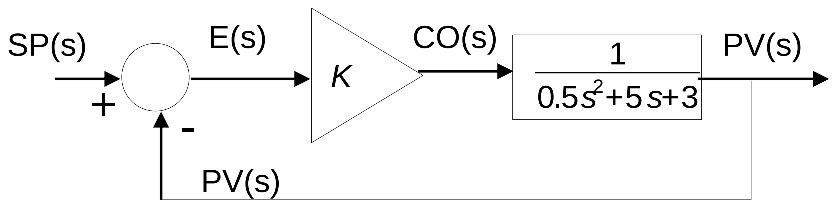


Figure 7 Block diagram of the proportional controller

Let the setpoint change be a step function with a magnitude of 1, $SP(s) = \frac{1}{s}$.

Using block algebra the proportional constant and the equation for the process can be combined and simplified to give us the forward path gain, $G_{FP}(s) = \frac{2K_P}{s^2+10s+6}$. The goal is to solve for the error *E(s),* in terms of the input, *SP(s)* and the forward path gain $G_{FP}(s)$.

First solving for error *E(s)*, in terms of *SP(s)* and *PV(s)* results in $E(s) = SP(s) - PV(s)$.
Second solving for the process variable, *PV(s)*, in terms of error, *E(s)*, and the forward path gain, $G_{FP}(s)$ results in $PV(s) = E(s)G_{FP}(s)$.

Substituting for $PV(s)$yields $E(s) = SP(s) - E(s)G_{FP}(s)$. After some manipulation $E(s) = \frac{SP(s)}{(1+G_{FP}(s))}$.

Replacing $SP(s)$ and $G_{FP}(s)$with their functions above and simplifying results in $E(s) = \left(\frac{1}{s}\right)\frac{1}{1+\frac{2K_P}{s^2+10s+6}} = \left(\frac{1}{s}\right)\frac{s^2+10s+6}{s^2+10s+2(3+K_P)}$.

The Final Value Theorem allows us to find the value of a function of time at time equals infinity using the function's Laplace Transform. It states $\lim_{t\to\infty} f(t) = \lim_{s\to 0} sF(s)$ where, in this definition, *f(t)* and *F(s)* are general polynomial functions, not force. From this it follows that $\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s)$.
For our system with a unit step input $\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left(\left(\frac{1}{s}\right)\frac{s^2+10s+6}{s^2+10s+2(3+K_P)}\right) = \frac{3}{3+K_P}$ is the steady state error. From the above it can be seen that proportional control alone can only minimize the steady state error, not remove it.

At this point a review of system block algebra is in order.

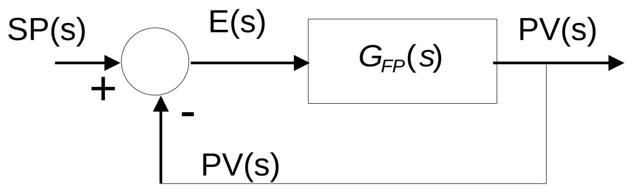


Figure 8 Block algebra diagram of the proportional controller

The system can be simplified to:

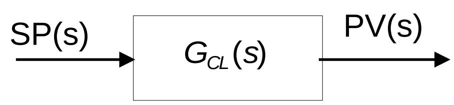


Figure 9 Simplified block algebra diagram of the proportional controller

Where the closed loop gain, $G_{CL}(s)$, is $G_{CL} = \frac{G_{FP}}{1+G_{FP}}$ and the transfer function for the system is $\frac{PV(s)}{SP(s)} = G_{CL}$.

Now find the final value of the transfer function when the input is a step function. The closed loop transfer function for this system with proportional gain only is $\frac{PV(s)}{SP(s)} = \frac{2K_P}{s^2+10s+2(3+K_P)}$.

Again, the Final Value Theorem can be used to find the final value as a result of a step input, $\lim_{t\to\infty} PV(t) = \lim_{s\to 0} s\left(\left(\frac{1}{s}\right)\frac{2K_P}{s^2+10s+2(3+K_P)}\right) = \frac{K_P}{3+K_P}$. As a final check the value of the final value can be subtracted from the input, 1, and the difference will be the final value of the error.

Even though a proportional controller will never reach the desired setpoint it still can be used in some applications. For example, if the tolerance on the process variable is greater than the steady state error, then a proportional controller may be adequate.

Changing K_P affects the response time of the system as shown on the graph below. Multiply the example closed loop transfer function by the unit step input, $PV(s) = \left(\frac{1}{s}\right)\frac{2K_P}{s^2+10s+2(3+K_P)} = \left(\frac{1}{s}\right)\frac{2K_P}{\left(s+5+\sqrt{(19-2K_P)}\right)\left(s+5-\sqrt{(19-2K_P)}\right)}$. Then take the inverse Laplace transform which results in the time domain equation of

$$f(t) = \left(\frac{K_P}{3+K_P}\right)\left[1 - \frac{\left(5-\sqrt{(19-2K_P)}\right)e^{-\left(5+\sqrt{(19-2K_P)}\right)t} + \left(5+\sqrt{(19-2K_P)}\right)e^{-\left(5-\sqrt{(19-2K_P)}\right)t}}{-2\sqrt{(19-2K_P)}}\right].$$

Note in the case where K_P equals 9.5 there is a double pole at -5 and the equation for the process variable is now $PV(s) = \left(\frac{1}{s}\right)\frac{2K_P}{(s+5)^2}$. The inverse Laplace transform of this is $f(t) = \frac{2K_P}{25}[1 - e^{-5t} - 5te^{-5t}]$. Below, these functions of time are graphed below for various values of K_P.

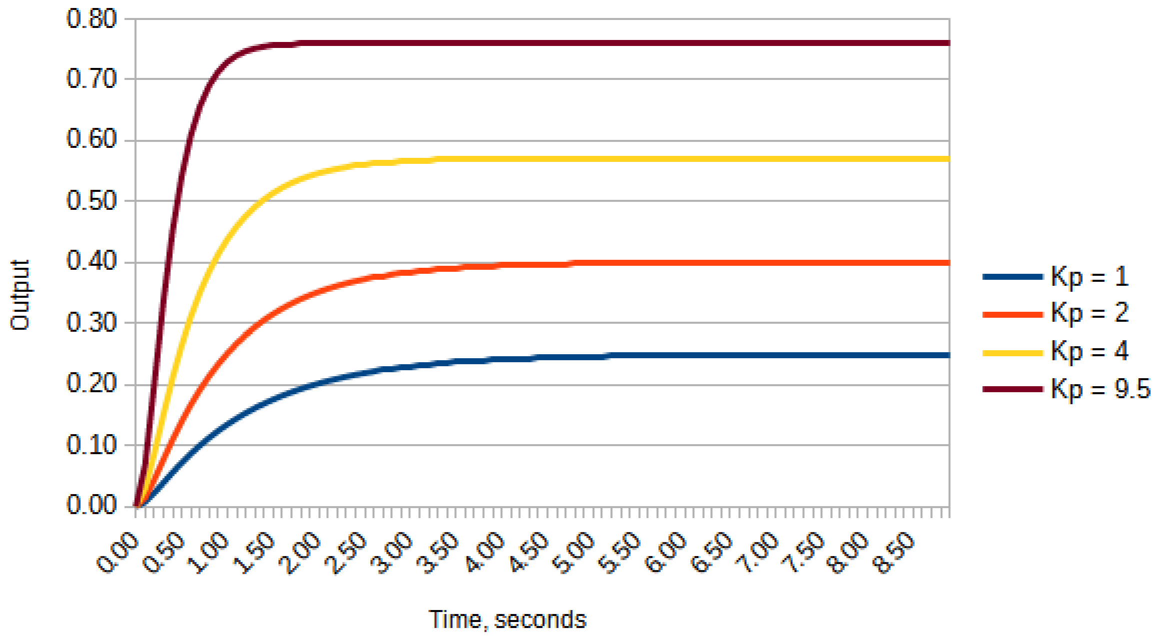


Figure 10 Effect of KP= 1, 2, 4, 9.5 on the response time of the system

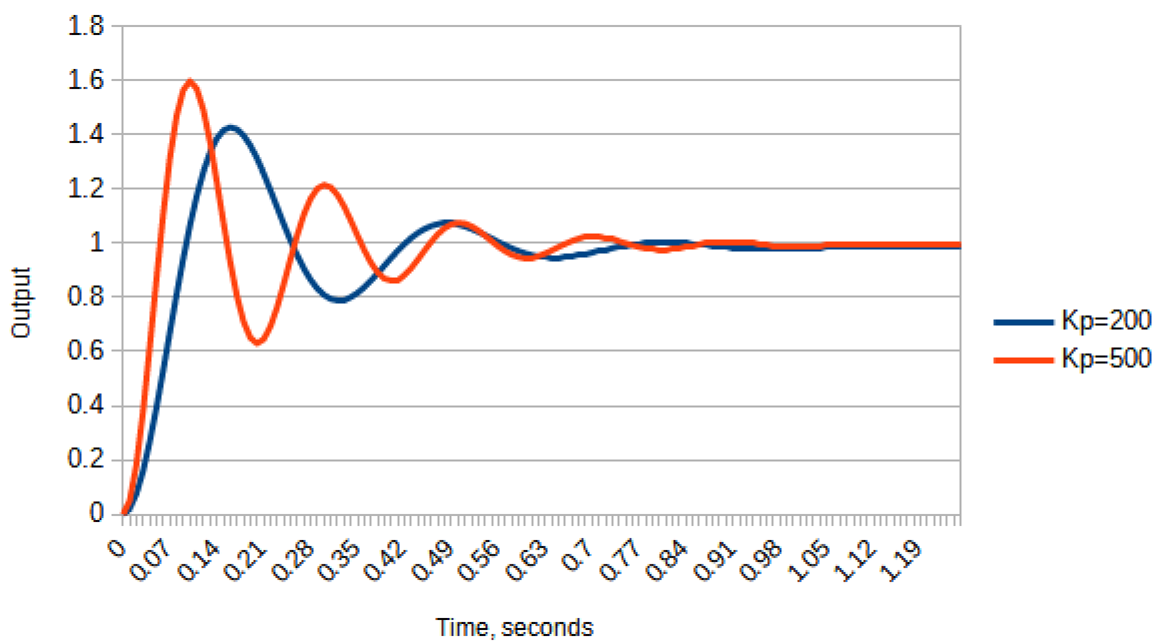


Figure 11 Effect of KP= 200 and 500 on the response time of the system

Block diagrams for a mass/spring/damper system with coefficients m, b, and k for mass, viscous damping and the spring constant respectively are shown below. From these diagrams equations for the natural frequency, damping ratio, damped frequency, and overshoot can be derived.

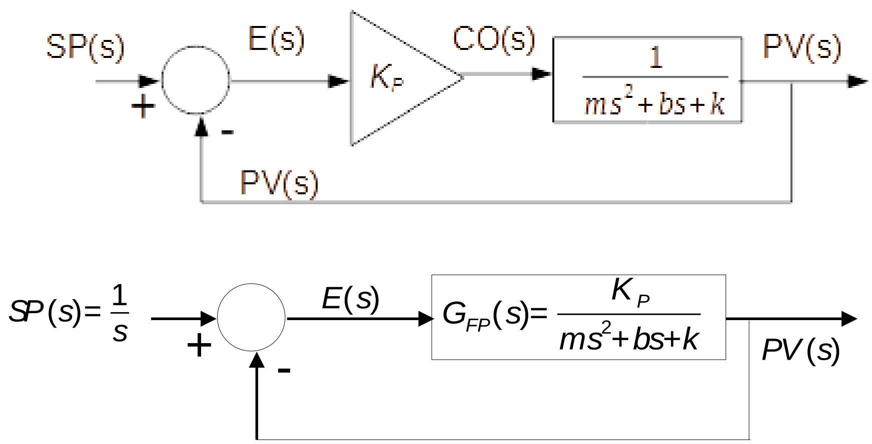


Figure 12 Block diagrams for a mass/spring/damper system

The transfer function is $\frac{PV(s)}{SP(s)} = \left(\frac{K_P}{m}\right)\frac{1}{s^2+\left(\frac{b}{m}\right)s+\left(\frac{k+K_P}{m}\right)}$. The denominator is in the form of a second order characteristic equation, $s^2 + 2\zeta\omega_n s + \omega_n^2$. From this the natural frequency, in radians/sec, is $\omega_n = \sqrt{\frac{k+K_P}{m}}$, damping ratio is $\zeta = \frac{b}{2\sqrt{m(k+K_P)}}$, overshoot is defined as $Overshoot,\% = 100 * e^{\frac{-\zeta\pi}{\sqrt{(1-\zeta^2)}}}$, and the damped frequency, that is the frequency of oscillation of the graphs above, is $\omega_d = \omega_n\sqrt{(1-\zeta^2)}$.

KP	Transfer Function Poles	Final Value for Unit Step	Natural Frequency, ωn, Radians/sec	Damping Ratio, ζ	Damped Frequency, ωd, Radians/sec	Overshoot	Rise Time, seconds, from data	Settling Time, seconds, from data
1	-0.88, -9.12	0.25	2.83	1.77	N/A	N/A	2.5	N/A
2	-1.13, -8.87	0.40	3.16	1.58	N/A	N/A	2	N/A
4	-1.68, -8.32	0.57	3.74	1.34	N/A	N/A	1.35	N/A
9.5	-5, -5	0.76	5.00	1.00	N/A	N/A	0.7	N/A
200	-5 ± 19.52i	0.99	20.15	0.25	19.52	0.4472	0.06	0.71
500	-5 ± 31.32i	0.99	31.72	0.16	31.32	0.6056	0.03	0.74

Table 2 Tthe effect of changing KP on different parameters including the final value

Increasing K_P decreases the rise time and the system reacts faster to a change in input or the response is more aggressive, and eventually oscillations develop. For this system with K_P less than or equal to 9.5 the poles in the s-plane are always on the negative x-axis and the system is over damped. When K_p is equal to 9.5 there is a double pole on the negative x-axis and the system is critically damped. In each case the system is stable, there is no overshoot, and the output takes the form of an exponential curve.
If K_P is increased beyond 9.5 the quantity under the square root sign is negative, and the system is under damped. It will oscillate and exponentially decay. The graph above shows the mass/damper/spring system output for values of K_P equal to 200 and 500. As the value of K_p increases the response time decreases and the system reacts faster to a change in input, but it overshoots more, and has a higher frequency of oscillation.

The data that was used to generate the graphs was analyzed to determine rise time and settling time. Rise time is defined as the time it takes for a signal to go from 10% of its final value to 90% of its final value.

The settling time is the time it takes the output to settle withing 2% of the final value for the under damped conditions. For $K_P = 200$ the settling time is 0.71 second, and for $K_P = 500$ the settling time is 0.74 second. There is a small increase in settling time as K_P is increased.

Now a discussion on some other terminology that is in use. Sometimes the term Proportional Band *PB,* is used instead of the Proportional Constant*,* K_P. The relationship is $PB\% = \frac{100}{K_P}$. It can also be defined as $ProportionalBand\% = 100\left(\frac{ProcessVariable\%Change}{Controller\%Change}\right)$. To demonstrate this let us have a variable frequency drive (VFD) connected to a motor/pump combination. The discharge side of the pump fills a tank. A level sensor sends a signal to the VFD representative of tank level. The tank has a minimum level of 0" and a maximum level of 50". The system is programmed to keep the tank level at 25". Also, when the tank level is 0" the VFD will command the motor to run at 3600 RPM, and when the tank level is 50" the VFD will stop the motor, 0 RPM.

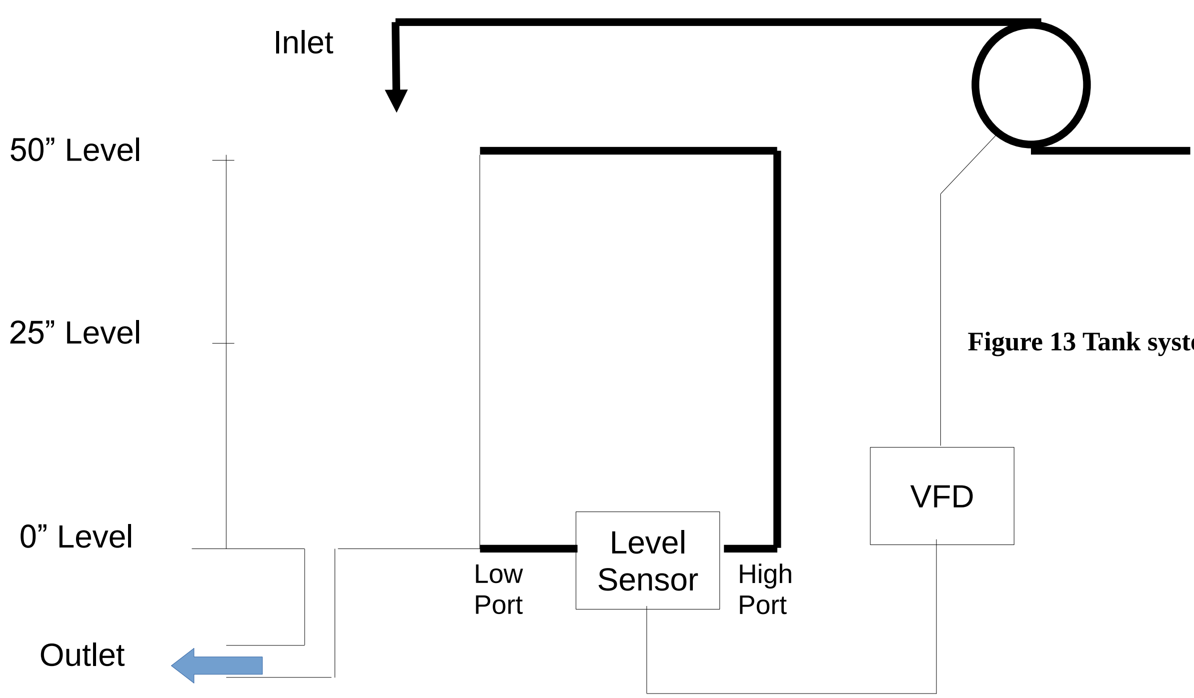


Figure 13 Tank system

Since a 100% change in process variable, 0" to 50", will result in a 100% change in motor speed, 3600 RPM to 0 RPM the proportional band is 100%. It is shown on the graph below.

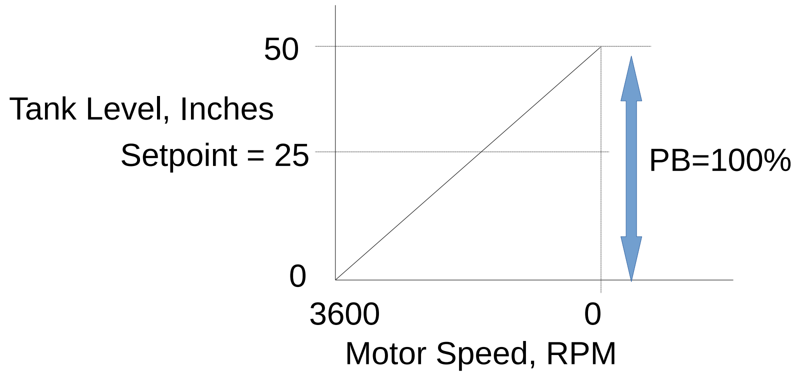


Figure 14 Graph showing the change in motor speed with respect to the tank level

Now, consider the case where the taps for the level sensor's low port and high port are relocated to the 15" and 35" levels respectively as shown in the diagram below.

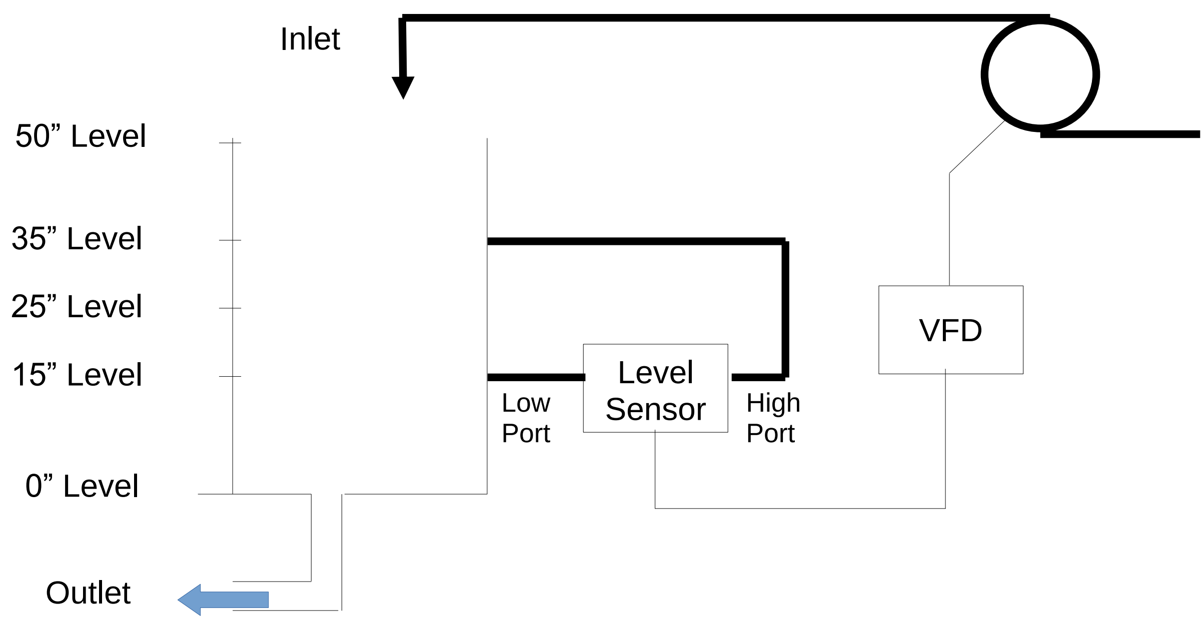


Figure 15 Tank sysem where the low and high port are relocated

The VFD will command the motor to run at 3600 RPM when the level is at 15" and 0 RPM when the level is at 35". Since a $\frac{35-15}{50} * 100\% = 40\%$ change in process variable will result in a 100% change in motor speed, 3600 RPM to 0 RPM the proportional band is 40%. It is shown on the graph below. The proportional gain would be 2.5.

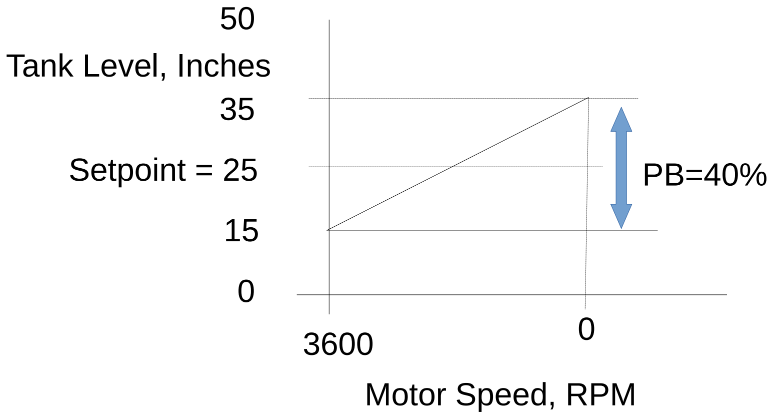


Figure 16 Graph showing the change in motor speed with respect to the tank level

It is interesting to note that as the proportional band gets smaller and graph becomes more horizontal the proportional controller acts more like an On/Off controller where a small change in level, the process variable, causes the control device VFD to run the motor at full speed or 0 RPM.

3.2. The Integral Term

The second term of the equation, the integral contribution is next, $\frac{1}{T_I}\int e\,(t)dt$. This contribution relates to the accumulation of error over time.

As expected, the engineering units of the integral contribution must be the same as the engineering units of the proportional contribution in order to be added together. The integral is an area, as such, it is a product of the Y and X axis. The X axis is time and, to be consistent with the example for the proportional contribution, the Y axis will be meters. The integral now the product of meters and seconds. To be added to the proportional error the integral must be multiplied by $\frac{1}{T_I}$ to obtain meters.

The integral contribution causes the offset error from the proportional contribution to go to zero. To see how this is done let us go back to the mass/spring/damper example where the setpoint is 2 meters and the force required to hold the mass stationary is 6 Newtons. If K_p is set to 15 which, from the previous chart, will result in the mass moving a distance of 1.60 meters and then stopping. In this case a proportional error of 0.4 meters generates an applied force to the mass of 6 Newtons, the force necessary to hold the mass in place at 1.60 meters. For this discussion, we need to consider a graph of error (on the Y axis) as a function of time (on the X axis). The integral term is the area under this error curve. The error needs to be sampled at a constant time interval, Δt. The integral error can be calculated by multiplying the sampled error by Δt, and then summing this error with previous integral errors to form the integral contribution. The integral contribution is added to the proportional contribution and this sent to the control device which applies a force to the mass and the mass moves. This process repeats on a periodic basis, Δt. The force increases and the mass eventually reaches 2 meters. At this point the error is zero, thus the area under the error curve is zero and additional integral contributions are zero. Although additional integral contributions are now zero, there is a past history of integral contributions. As such, the integral contribution is now constant. Since the error is zero, the proportional contribution goes to zero. Thus the steady state error has been eliminated.

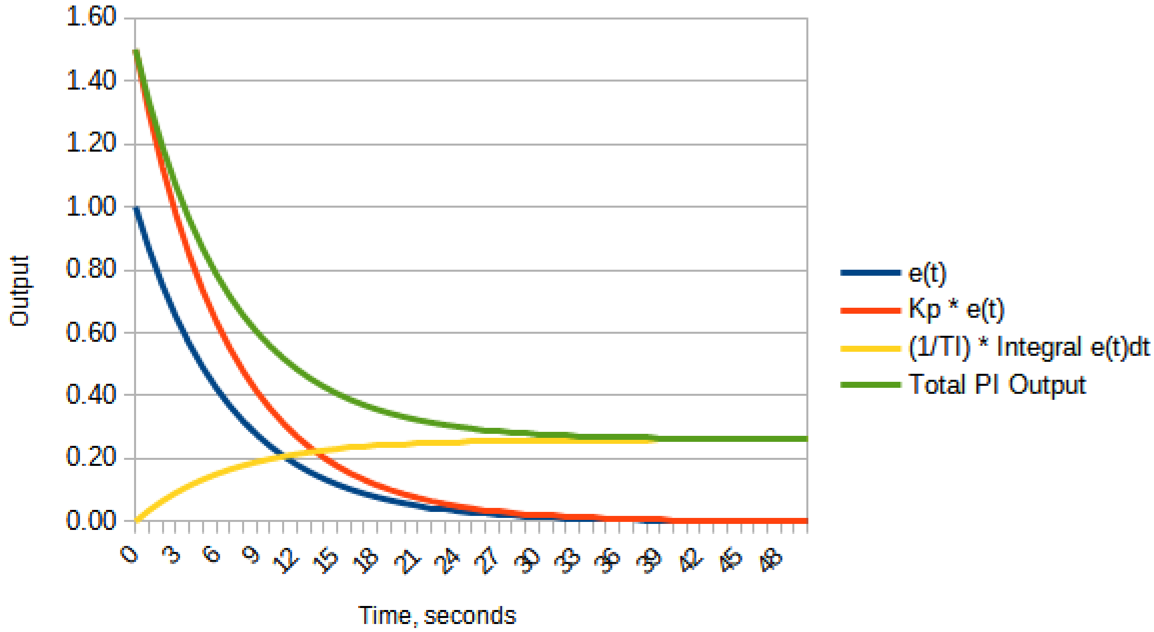
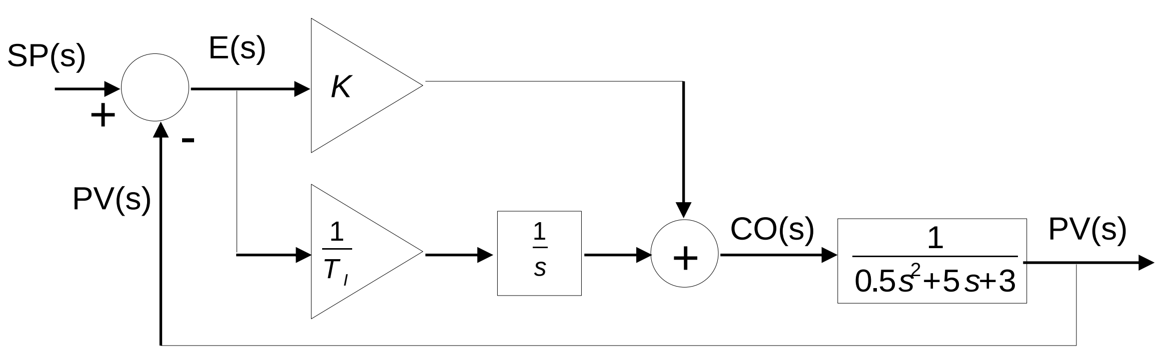


Figure 17 Graph showing the integral contribution output with respect to time

Note that the time Δt is not the same as the time T_I. The time Δt is the sampling time and is dependent on the equipment. For example, if the PID controller was is being implemented in a PLC, then the time Δt could be the program scan time. That could be on the order of milliseconds. The time T_I is a constant that is input by the operator. This constant determines how fast the steady state error is removed. The smaller the value of T_I, the large the integral contribution, and faster the steady state error is removed.

The diagram below shows the block diagram of the proportional integral controller (abbreviated PI controller) and mass/spring/damper transfer function that we previously used.



This diagram can be simplified by combining terms

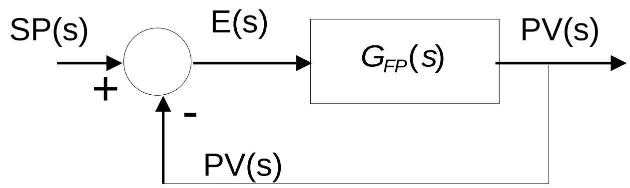


Figure 18 Block diagram of the proportional integral controller and mass/spring/damper transfer function

Where the forward path gain is $G_{FP}(s) = \frac{2K_P\left(s+\frac{1}{K_P T_I}\right)}{s(s^2+10s+6)}$. Using the same method as was used in the analysis of the proportional gain to find the closed loop transfer function $\frac{PV(s)}{SP(s)}$ results in $\frac{PV(s)}{SP(s)} = \frac{2K_P\left(s+\frac{1}{K_P T_I}\right)}{s^3+10s^2+2(3+K_P)s+\frac{2}{T_I}}$ and error with a step function input is $E(s) = \frac{1}{s}\frac{s(s^2+10s+6)}{s(s^2+10s+6)+2K_P\left(s+\frac{1}{K_P T_I}\right)}$. From this $\lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left(\frac{1}{s}\frac{s(s^2+10s+6)}{s(s^2+10s+6)+2K_P\left(s+\frac{1}{K_P T_I}\right)}\right) = 0$, the error is zero. For *SP(s)* equal to the step function $\lim_{t\to\infty} PV(t) = \lim_{s\to 0} s\left(\left(\frac{1}{s}\right)\frac{2K_P\left(s+\frac{1}{K_P T_I}\right)}{s^3+10s^2+2(3+K_P)s+\frac{2}{T_I}}\right) = 1$, the output is equal to the input. As such, there is no steady state error.

The integral contribution has a draw backs. First, it can introduce overshoot and oscillations as shown on the graphs below for a step input. For $T_I = 0.01$ second and $K_P = 4$ combination, and $T_I = 0.01$ second and $K_P = 1$ combination the systems are unstable and are not plotted.

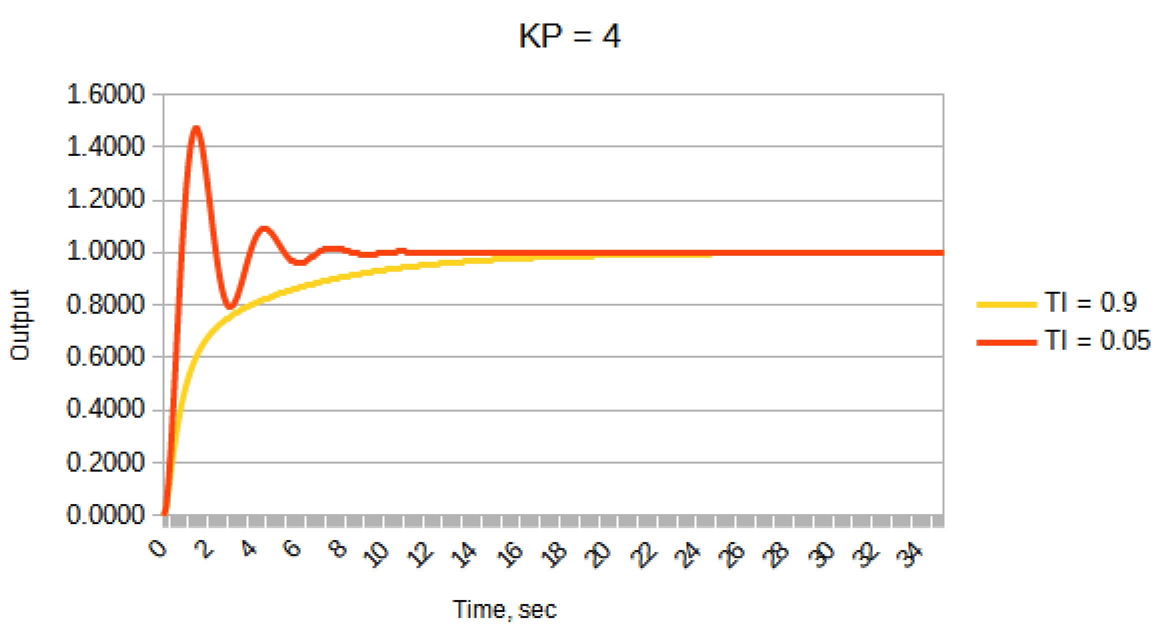


Figure 19 The effect of KP =0.4and TI=0.9 and 0.05 on the output of the integral system

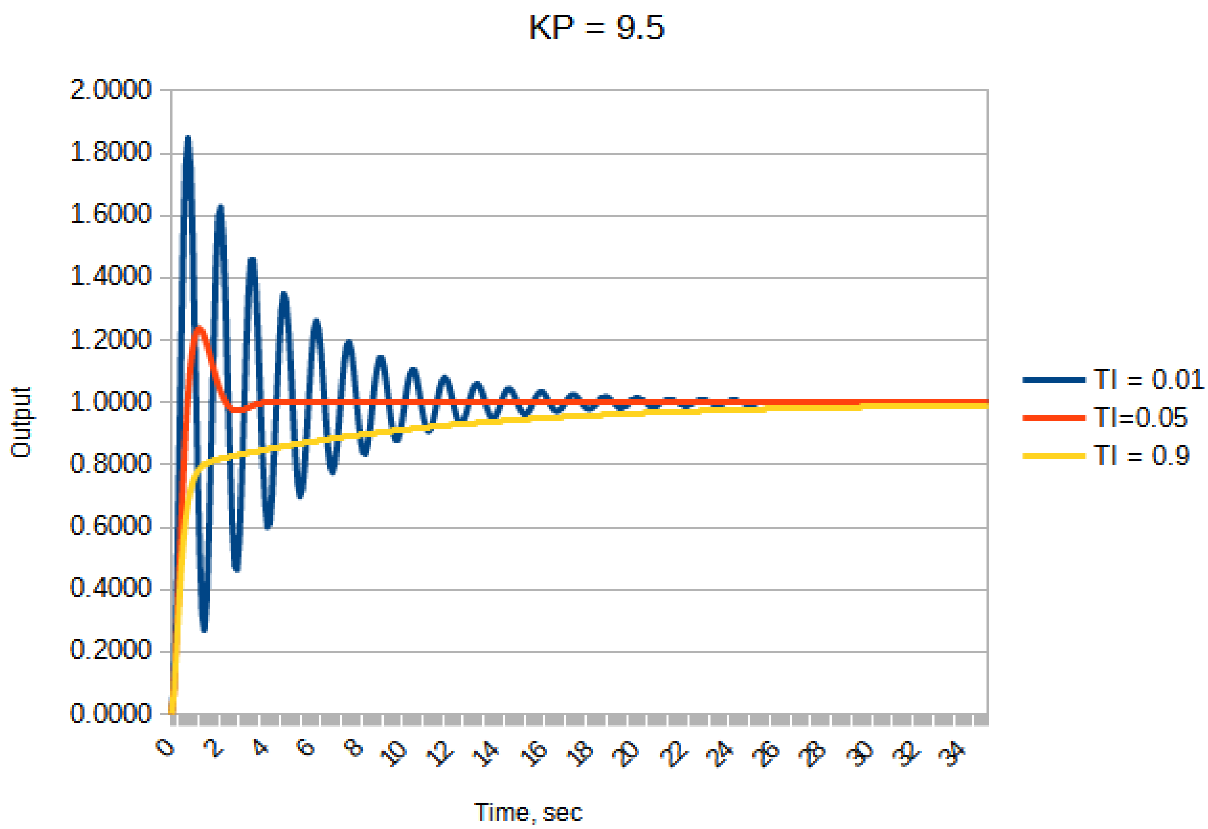


Figure 20 The effect of KP =9.5 and TI=0.01, 0.05, 0.09 on the output of the integral system

The second draw-back is that the integral term does have an issue which is called windup. Windup can occur when the actuator is at its extreme end of range and the process variable is not at the setpoint. This causes the integral contribution continues to increase. This is illustrated by the example below of an integral only controller where the temperature sensor compares the temperature, the process variable, to a setpoint, and changes the position of a valve, the manipulated variable. Upstream of these components is an isolation vale that is either open or shut. The system provides hot water for heating from an ambient temperature of 70°F to 100°F, which is the setpoint.

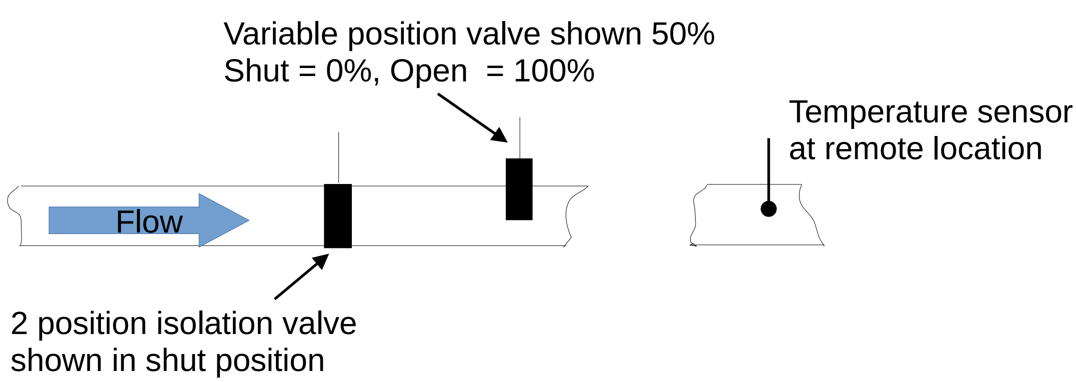


Figure 21 Example of an integral only controller

Before the system is started the isolation valve is inadvertently shut. Therefore, when the system is started there is an error of 30°F shown in yellow. This error is integrated and the integral contribution, the green line, increases. It causes the valve position to change to 100%, full open, also the green line. The valve position remains at 100%, the green line, while the integral error continues to increase, the red line. The valve cannot open past 100%, it has reached its limit. The area bounded by the red curve and the green line at 100% is called the integral wind up. The problem with integral wind up is that it takes time for it to be removed. During this time the actuator will remain at its limit, in this case 100%, causing the process variable to overshoot.

At 2 seconds the isolation valve is opened, hot water starts to flow, the error decreases, and the integral increases, but at a slower rate toward 200%. At 2.4 seconds the error reaches zero, the temperature is 100°F, but integral output is well above 100% keeping the valve fully open. After 2.4 seconds the process variable exceeds the setpoint of 100°F. This causes a negative error and negative area which causes the integral to decrease. At 3.9 seconds the integral error and valve position are both at 100% and the temperature has overshot the setpoint. At this point the valve position starts to decrease. Eventually, the valve position settles and the error is zero.

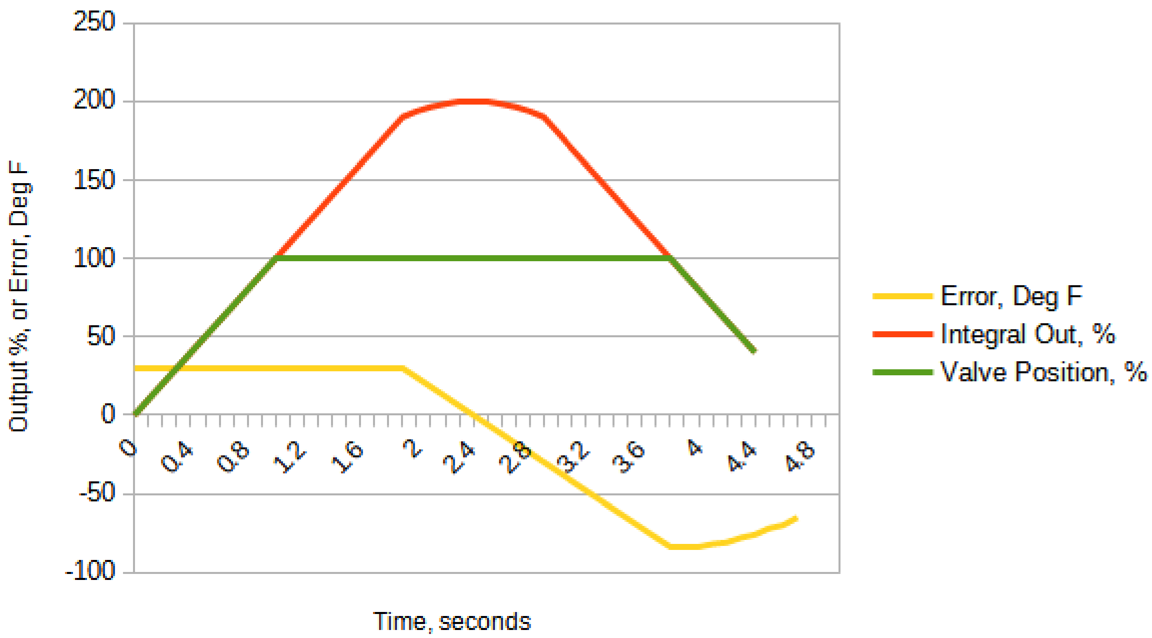


Figure 22 Graph showing the % errors in an integral only controller system

One possible solution to integral wind up is to clamp the integral output to 100%.

Another issue arises if the control element has hysteresis. For example, if the valve stem in the control valve above is slightly bent and there is a position or two where it sticks. In this case the integral term can cause oscillations. The integral would build up until there is enough force to free the valve and the valve could jump towards its new position and overshoot. If it got stuck again during it travel the integral would build up until the valve breaks free. This process could continue until the process variable equals the setpoint, if it ever does equal the setpoint. The valve could oscillate or hunt for the proper position as it gets stuck and frees itself. The best solution is to fix the sticking problem that is to fix the valve. Otherwise an integral dead band that ignores small errors could be programmed into the controller and might solve the issue.

At this point the question "If the integral term only can produce zero error why use the proportional term?" may arise. An integral only controller can be used, but without the quick response provided by the proportional term the response could be slow.

3.3. The Derivative Contribution

The third term of the equation, the derivative contribution is next $T_D \frac{de(t)}{dt}$. This term is an anticipatory function based on the rate of change of the error. It should be noted that the engineering units of the derivative term must be the same as the engineering units of the proportional and integral contributions. The derivative is a slope, as such, it is a quotient of the Y and X axis. The Y axis is distance in meters and to be consistent with the examples for the proportional and integral contributions the X axis will be time in seconds. As such, the derivative has units of $\frac{meters}{seconds}$ and, the derivative must be multiplied by T_D, in seconds, to obtain meters.

The derivative contribution adds a contribution to minimize sudden changes in the process variable based on the time rate of change of the error. This change could be due to a disturbance, or overshoot, or change in setpoint. To see how this is done let us go back to the mass/spring/damper example. We will operate a proportional-derivative controller with a setpoint of 5 meters and sufficient time has passed so the system is in steady state. The value of the proportional constant will be such that in steady state the mass will be at 4 meters. This gives us an error that is a constant 1 meter. Since the derivative of a constant is zero there will be no derivative contribution to the controller output. If for some reason, a disturbance is introduced to the system that makes the mass move to 7 meters, then the error, $e(t) = SP(t) - PV(t) = 5 - 7 = -2$ feet. If this happens over an interval of 4 seconds, for example, then the derivative contribution is $\frac{-2}{4} T_D = -0.5 T_D$. The value of the output will decrease and force will decrease allowing the mass to move back to the setpoint. As the mass returns to the setpoint the time rate change in error decreases and the derivative contribution decreases. Eventually, the error is constant and the derivative contribution is zero.

Since derivative only controllers are not used there will be no analysis using the Limit Value Theorem as was done for the proportional controller and the integral controller. An analysis will be done for a proportional-derivative controller.

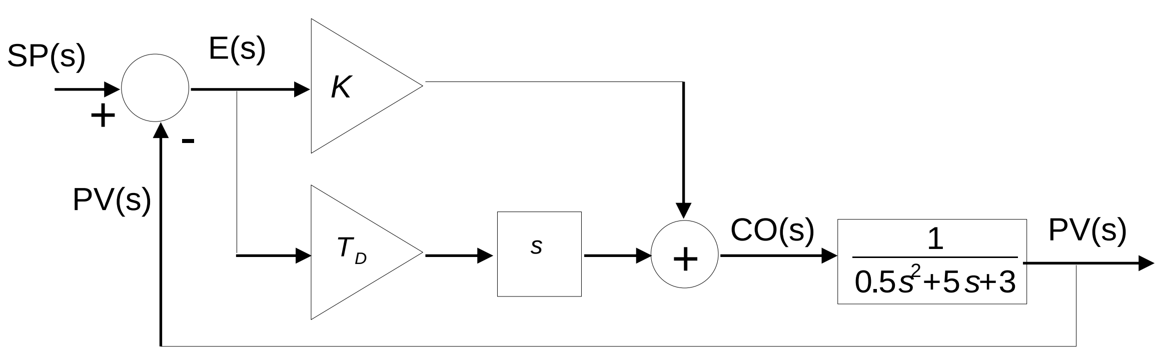


Figure 23 Block diagram of the deriviative contribution

This can be simplified to

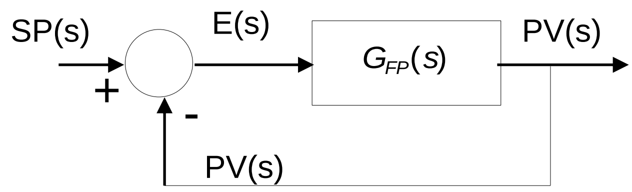


Figure 24 Simplified block diagram of the derivative contribution

Where $G_{FP}(s) = \frac{2T_D\left(s+\frac{K_P}{T_D}\right)}{s^2+10s+6}$ and $\frac{PV(s)}{SP(s)} = \frac{2T_D\left(s+\frac{K_P}{T_D}\right)}{s^2+(10+2T_D)s+2(K_P+3)}$. Using the Limit Value Theorem and a step input results in a final value for the process variable of $\frac{K_P}{K_P+3}$. There is no contribution from the derivative term since the input is a constant. Also, note that this is the same result that was obtained for the proportional only controller.

The equation for the error is $\frac{E(s)}{SP(s)} = \frac{1}{1+\frac{2T_d\left(s+\frac{K_P}{T_d}\right)}{s^2+(10+2T_d)s+2(K_P+3)}}$ using the Limit Value Theorem and a step input results in an error of $\frac{3}{K_P+3}$. Also note that this is the same result that was obtained for the proportional only controller.

Below is a graph of the response of the PD controller to a unit step input with K_P = 200 and T_D = 1, 10, and 25. It shows that as T_D increases, and K_P is held constant, overshoot decreases and rise time

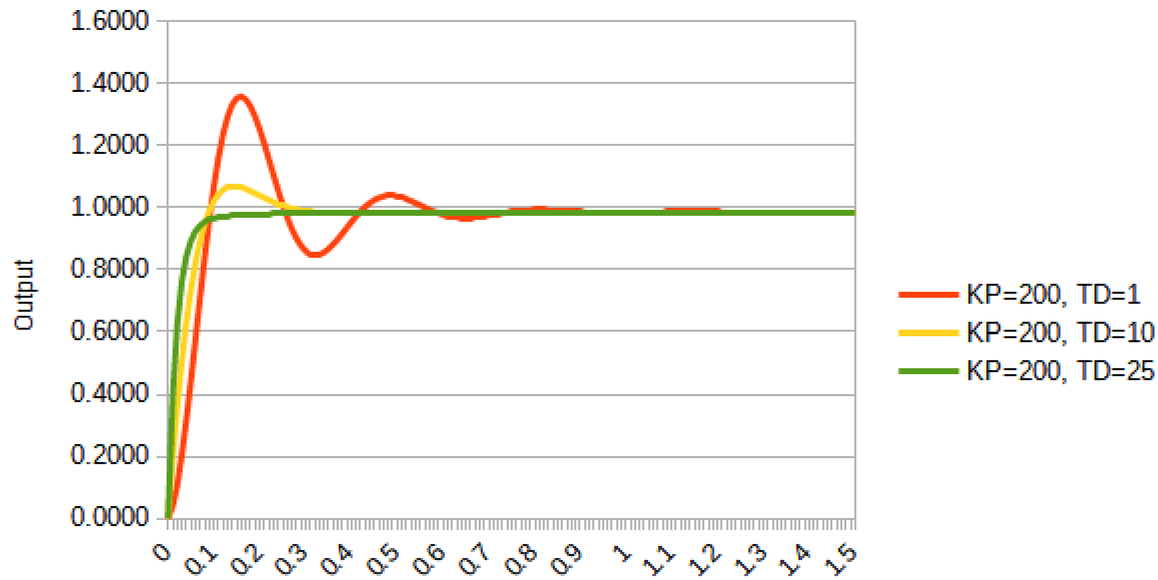


Figure 25 Graph showing the response of the PD controller to a unit step input with KP = 200 and TD = 1, 10, and 25

decreases. An analysis of the data shows that for the under damped conditions the time to settle to 2% of the final value is 0.73 second for $T_D = 1$, and 0.23 second for $T_D = 10$.

The derivative contribution is not without issues. Two of them are discussed below.

The graph below shows the response of a system to a step input. The error is Δe and the sampling time is Δt. The derivative $\frac{de}{dt}$ can be calculated as $\frac{\Delta e}{\Delta t}$.

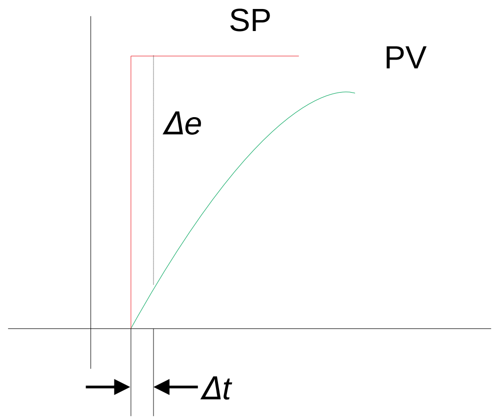


Figure 26 Graph showing the response of a system to a step controller

Since Δt is small and Δe could be large immediately following a change in setpoint the result can be a large spike in the derivative contribution. This may or may not be an issue since it exists for a very short time. However, it could cycle equipment and over time cause premature wear. This could be mitigated by performing the derivative calculation on the process variable which could change at a slower rate. This is shown below.

From the definition of error $\frac{de(t)}{dt} = \frac{d\left(SP(t)-PV(t)\right)}{dt} = \frac{dSP(t)}{dt} - \frac{dPV(t)}{dt}$. Since the setpoint is constant $\frac{dSP(t)}{dt} = 0$ and $\frac{de(t)}{dt} = \frac{-dPV(t)}{dt}$. Programmable logic controller software may contain functions for PID control which allows the programmer the choice of performing the derivative calculation on the error or process variable.

The second issue has to do with noise on the feedback signal used to calculate the error. This noise could be variations in the process variable or caused by the sensor that monitors the process variable and generates the signal used in the error calculation. The small sampling time could cause a large derivative contribution, as discussed above. Also, rapid changes in slope will cause the derivative contribution to quickly oscillate between a positive value and a negative value. A low pass filter or moving average can reduce these negative effects, however this will introduce a delay in the system.

3.4. Putting All the Terms Together

The transfer function for the mass/spring/damper system PID controller is
$\frac{PV(s)}{SP(s)} = \frac{2\left(T_D s^2 + K_P s + \frac{1}{T_I}\right)}{s^3 + (10+2T_D)s^2 + (6+2K_P)s + \frac{2}{T_I}}$.

The graph below compares the controller output for a step input for:

1. Proportional only controller with K_P = 9.5 which provided the most aggressive response with out oscillation.
2. Proportional and integral controller with K_P = 15 and T_I = 0.1 second which provides an aggressive response with some overshoot.
3. Proportional, integral, and derivative controller with K_P = 15, T_I = 0.1 second, and T_D = 1 second. The derivative term does minimize the overshoot at the expense of reaching the final value later in time. For this transfer function the poles are located at p_1 = -0.71643, p_2 = -3.66349, and p_3 = -7.62008; and zeros are located at z_1 = -14.30074 and z_2 = -0.69926. For a step input the time domain equation for the process variable is

$$f(t) = 1 + 0.0320034e^{-0.71643t} - 1.47627e^{-3.66349t} + 0.444271e^{-7.62008t}.$$

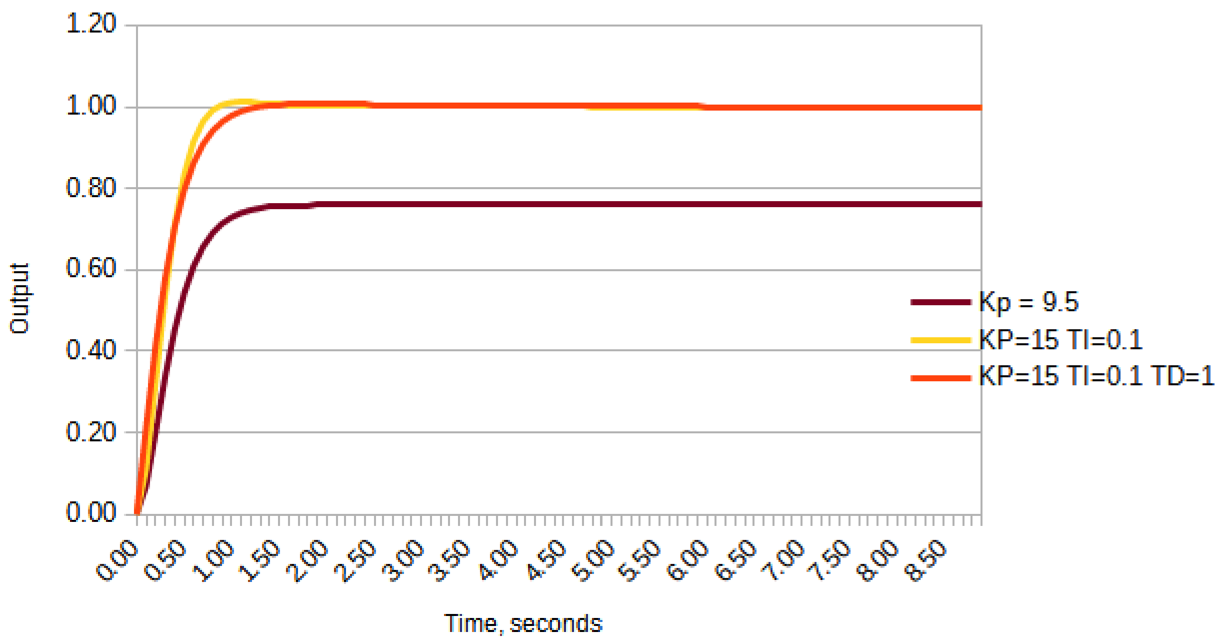


Figure 27 A graph comparing the controller output for different controller types

4.0 SUMMARY

The PID controller is widely used in industry to control a process. It's not necessary to use all three constants. The proportional only controller can be used depending upon the value of the proportional constant the process variable can quickly rise to a value near the setpoint. The proportional alone will never bring the process variable to the setpoint. It will always have a steady state error. This steady state error may be acceptable depending upon system requirements. Introduction of the integral term eliminates the steady state error. However, it can cause overshoot and is subject to an undesirable effect called windup. An integral only controller can be used, but it will lack the quick response provided by the proportional term. The Proportional-Integral controller is used most. The introduction of the derivative term can reduce overshoot and the effects caused by disturbances. However, the small sampling time amplifies noise and methods to mitigate this may be required. Selecting the correct constants is called tuning. While procedures exist a well-trained operator that understands the system can tune the system very well.

	Overshoot	Settling Time	Steady State Error
Increasing K_P	Increases	Small Increase	Decreases
Increasing T_I	Decreases	Decreases	Steady State Error always equals zero
Increasing T_D	Decreases	Decreases	No effect

Table 3 Tuning the system