Technical Note

Document Number	DTN0003		
Title	Introduction to PID Control		
Approved By			
Revision	**Date**	**Prepared By**	**Change History**
Draft	01/09/2004	Mark McCoy	

Introduction

The following document gives information on closed loop control using a PID algorithm. It also gives some details on the tuning of DC Servo Motor Control as an example to illustrate the principles involved. This example is based on a MoTeC M400/600/800/880 ECU.

What is PID?

PID stands for Proportional, Integral, and Derivative and is a type of feedback control system. It compares a measured value (from a sensor, say) against a desired value (the *setpoint* or *aim*) and adjusts outputs to reduce the difference (*error*) between the two.

The controller (or ECU in our case) uses a constantly updating calculation to control a physical system. It looks at the current value of the error, the integral of the error over a recent time interval, and the current derivative of the error signal to determine not only how much of a correction to apply, but for how long.

In the case of a Drive by Wire system the drivers' pedal position gives the Aim value while the angle of the throttle bodies Throttle Position sensor is the feedback channel. When there is a difference between the two signals, the duty cycle of the ECU outputs are changed to so that the feedback tracks the aim value.

The response time of any system is greatly dependant on the physical components (lag time). Eg: cam position changes quickly, while fuel cell gas pressure changes much more slowly. A Drive by Wire system uses a servo motor to drive the throttle blade to the Aim position where the response time of the system will be different again.

We can also consider the varying rate at which the driver requests different positions. In practice this can be accounted for when tuning the PID parameters by testing the systems' response to sudden (or *step*) changes in the aim value.

Note that in the real world a PID system will never track *exactly*[*] to the setpoint. Instead the tuning parameters include a '*deadband*' – an area around the setpoint where the controller doesn't try and drive the system any closer to the setpoint.

The output signal is the sum of the Proportional, Integral and Differential components. We will look at each of these in turn.

Proportional Component

The Proportional component is equal to the Proportional Gain ('P') multiplied by the Error.

The proportional parameter has an immediate response when there is an Error and it controls the bulk of the response in the system. The proportional component alone will never reduce the error to zero, since as the error decreases, the proportional response also decreases.

If the Proportional Gain is too small, the error will only be reduced by a small amount and response will appear to be slow. Too much proportional gain and the system will be unstable and hunting will occur. Hunting is the rapid over and undershooting of the system around the setpoint.

---

[*] On a physical system the feedback channel will always be changing, even if only slightly.

Integral Component

Integral is a more slowly changing factor there to reduce error over the longer term.

The size of the Integral component will keep increasing as long as there Error is not zero. Eg: boost and idle control valves require some duty cycle to hold a position, with an error value of close to zero, P & D components have little effect, but the I component will hold the valves position. If the control overshoots the setpoint, the error value changes polarity and a ‘negative’ term is added to the Integral component, reducing it.

Large amounts of Integral Gain can sometimes cause problems as well. “Integral Windup” is a situation that occurs when the error is constant, causing the Integral component to keep increasing the output to try to reduce this. If the conditions maintaining the error are removed the system can overshoot the aim value and then the control needs to reverse. Use the lowest Integral Gain that will ensure long term errors are within the deadband.

Some MoTeC PID functions include an ‘Integral Clamp’ parameter that sets the maximum allowable value of the Integral component so that you do not get Integral Windup.

Derivative Component

Derivative Gain has a damping effect on control of the system. It is there to improve the response time of the system. It is based on the rate of change of the Error value, so the derivative component will be larger for sudden changes in error than for gradual changes.

Again, the response time of any system is greatly dependant on the physical components (ie: lag time), so the Derivative Gain must be set accordingly. For example, if the Derivative Gain is too large, this will lead to ‘overshoot’ as the response goes past the aim value, and the system must reverse.

If there is high frequency noise on a measured position, a large Derivative gain will cause wild fluctuations. Therefore the size of the Derivative gain is dependant on the noise of the physical system. For a filter to be used on the feedback channel, the response time of the system must be of a lower frequency than any noise.

The derivative term tends to come into operation during the initial transient change in the system. It should have the effect of ‘flattening’ the response curve, reducing overshoot from the proportional response.

Feed Forward

Also known as “Linearization”. Some systems will always require a certain output duty cycle for operation, eg: where there is a known resistance in the system such as from the spring in a DBW throttle body. Feed-Forward is a predicted output value based on knowledge of how much output is needed to control a system to a certain Aim. This value can be used as a starting point for the systems control.

Eg: feed forward is used as the Average Position for the PD and PID Boost Control function. The Average Position sets a starting value duty cycle the valve works from. If you need roughly a known duty cycle to achieve your aim boost you would set this as your Average Position so the control loop has a close starting point.

A Real World Analogy

Think of a driver with no brakes wishes to stop a car at a set of lights. The driver is using the accelerator pedal to give the car forward movement to get to the lights. The closer the car gets the less the driver pushes on the accelerator pedal. The amount of throttle is the Proportional Gain.

The Driver is relying on the car to slow down because of rolling friction between the tires and the road. If the driver is trying to get to the lights quickly, more throttle will be used.

The problem is that if the driver relies solely on the rolling friction to stop the car, they may roll straight past the lights and then need to put the car into reverse and head back. This could happen several times before the car comes to rest at the lights and the faster the driver tries to get there, (better system response) the worse the over/undershooting problem becomes.

Now consider if the driver also has a braking system. When approaching the lights they can reduce the amount of throttle to slow the car and also apply the brakes to reduce the speed. The brakes act as the Derivative component of the system. It is logical to suggest that with the throttle and brakes the driver can now get to and stop at the lights with greater ease and generally more quickly, with less over/undershoot.

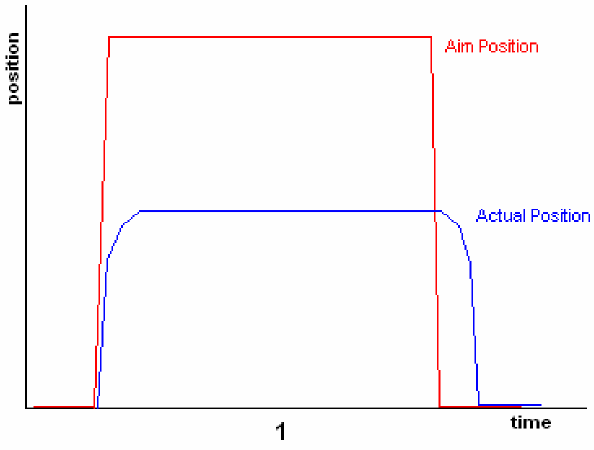
Now consider the driver has to do this when the lights are on a slight upward sloping hill. The driver can perform the stopping exercise using the throttle and brakes but the car will start rolling backwards when it is stopped. The driver now needs to apply a little bit of throttle (assume the brakes are ONLY for reducing speed and not to stop movement) to hold the car at the stopping point so it does not roll backwards, this is the Integral component of the system.

It can be seen that if the same driver has a very powerful car, the amount of throttle and brake needed to get to the set of lights is different to the amount of throttle and brake needed for a less powerful car. Obviously the high powered car will get the job done quicker but with more energy needed and therefore more stress on the equipment.

PID tuning

Example: a DC Servo motor.

The parameters for P, I and D are all set to zero to start with. A Small amount of P (figure 1) is added and the response of the motor to the input is started. It can be seen that there is a stepped response to the stepped Aim value but the aim position is not reached. The P is increased (figure 2) and the error (Aim minus Actual) is reduced.



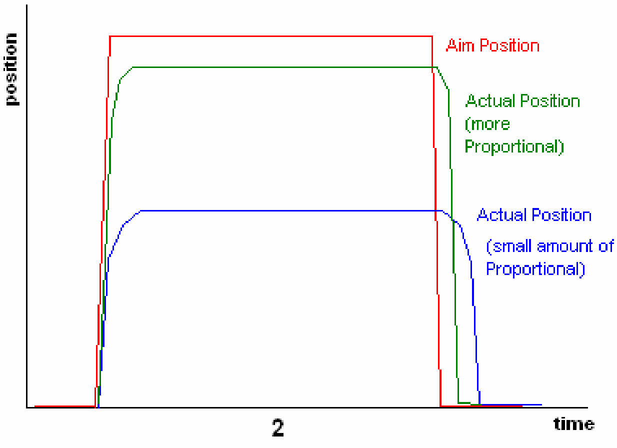
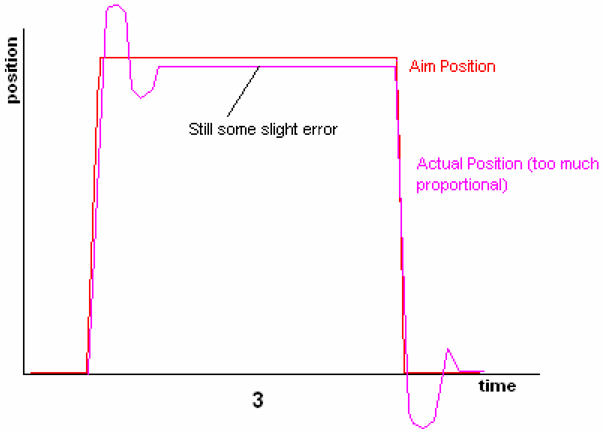
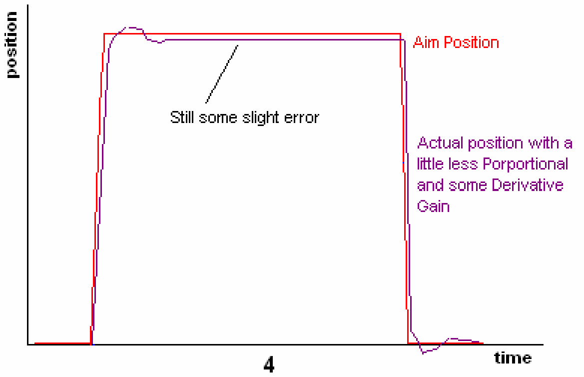


Figure 2 also shows that with added P there is a slightly quicker response to the stepped input, this shows as reduced ‘rounding’ of the actual position step.

P is increased even further so that response is quicker and the error between Aim and Actual position is a lot smaller but there is now some overshoot and undershoot (figure 3). Dropping the P amount slightly will reduce the hunting and adding some D will reduce it even further while keeping the response time acceptable (figure 4). The D parameter acts like a damper on the systems response. There is still some small error remaining though.

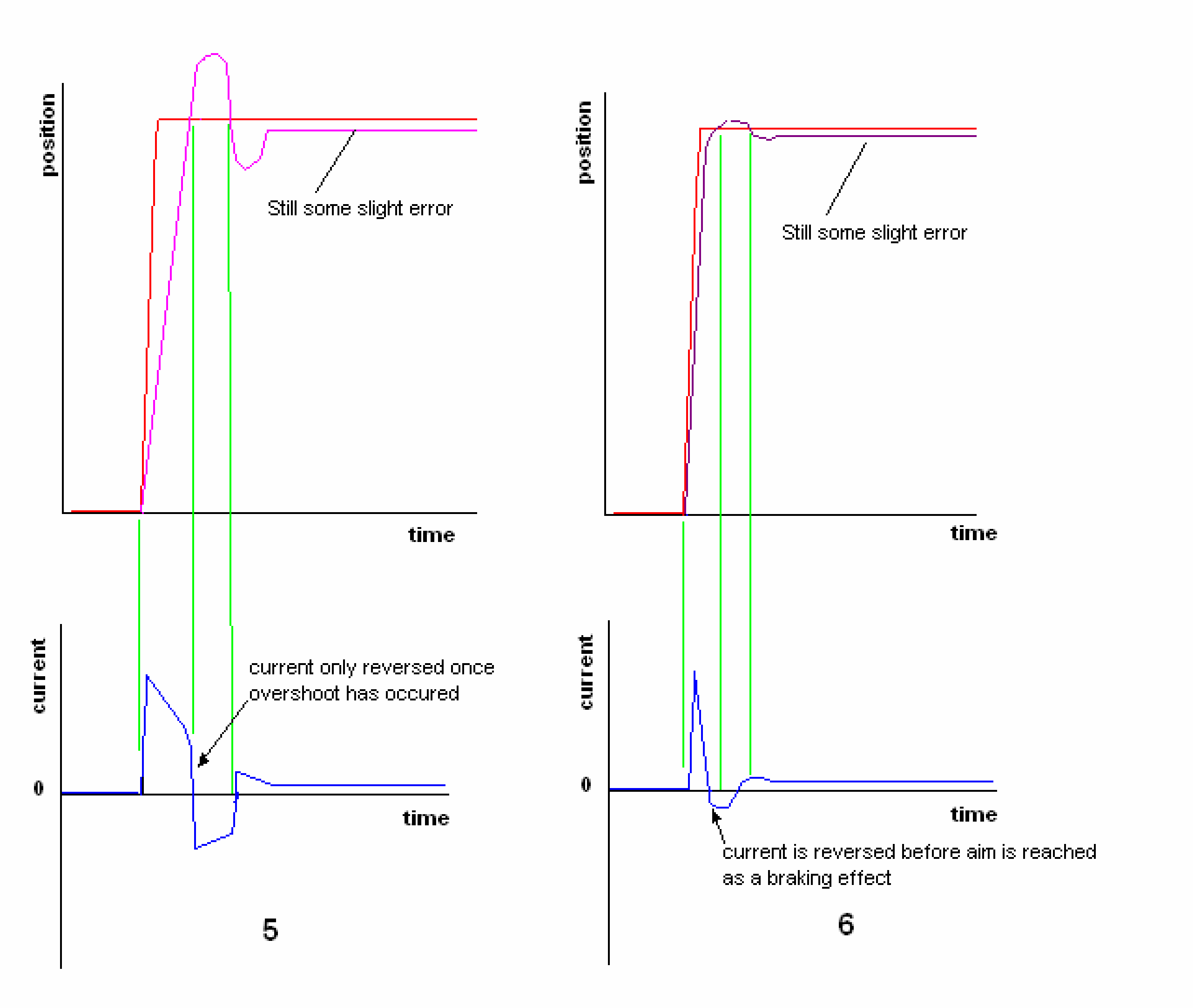




Some I can be introduced to the system to reduce the remaining error to close to zero, ie: to within the deadband range. The I will keep adding duty over the time that the error is not zero.

The reaction from the P and the additional D can also be seen by looking at the output current. Figure 5 shows the system with P component only; the current applied to the motor shows that it has no damping (braking) effect and the current is only reversed when the Actual position overshoots the Aim. In figure 6 there is a noticeable reversing of the current (braking) as the error gets close to zero. Fine tuning the P and D will make the system better. I is (generally) set last to eliminate any remaining error.

A system such as a DC Servo motor will almost always need full PID control because of its requirement to have very rapid and accurate response to a rapidly changing Aim request. Systems that have a slower response requirement may only need P and some I with no D. Some systems may require P and D only as any left over errors do not affect the system, boost control for example.



If the DC Servo motor is not able to achieve its aim position for some reason, say if something were to physically prevent movement, the integral gain would keep increasing the duty cycle to eliminate this error. If you were to suddenly release the motor there would be so much built up Integral component that it might overshoot wildly. Having a sensible Integral Clamp stops this from happening.

General PID Tuning

On-line trial tuning or, the "by-guess-and-by-golly" method

1. Enter an initial set of tuning constants from experience. A conservative setting would be a Proportional gain of 1 or less and an Integral gain of less than 0.1.

2. Put loop in automatic with process "lined out".

3. Make step changes (about 5%) in setpoint.

4. Compare response with diagrams and adjust.

This method may be feasible for some (but not all) vehicle systems. More robust starting parameters can be calculated mathematically using the Ziegler-Nichols tuning method. A discussion of this method is beyond the scope of this document. Many references can be found on the Internet by interested parties.