Wood Technology 2004
March 17-19, 2004
Oregon Convention Center
Portland, Oregon

Author: Peter Nachtwey

Session: Thursday March 18@ 1:00PM

Title:

System identification, simulation and auto tuning of Hydraulic Motion Control saves time, optimizes control and delivers smoother, more precise positioning when built upon a properly designed hydraulic system foundation.

Abstract:

Today's closed loop motion controls take advantage of advances in electronics, sensor technology and control systems techniques.

Motion Control. At the center of this automation approach is the electronic motion controller. Field proven motion control algorithms deliver precision in servo-hydraulic applications, whether for retrofits or new machines.

Simulation. Simulation helps identify design limitations before installation, making changes more economical. System identification determines how a system will respond. Auto tuning uses system identification data to reduce setup time and optimize performance.

Optimized Results. Together, system identification, simulation and auto tuning verify machine design and reduce startup time. Coupled with good hydraulic system design, these techniques deliver better engineering, testing, installation and production results.

Introduction

Delivering a well tuned hydraulic machine to Production quickly is the primary objective of the project engineer whether for a retrofit or new machine. By using a combination of system response identification, simulation oriented modeling, and auto tuning methods to optimize design, an engineer can deliver a production ready machine to operations quicker and better tuned than before.

Tuning a system for optimal performance using traditional methods is a time-intensive trial and error process with the objective of closing in on the correct PID gains until a satisfactory result is achieved.

Motion controllers have evolved over the years to augment trial and error testing with advanced tuning commands that use velocity and acceleration feed forward functions to vastly improve these methods for tuning PID gains. *(i.e. some motion controllers will provide a specific command to calculate velocity feed forwards in tuning procedures for hydraulic cylinder extend and retract movements.*

Evaluation methods such as the integrated square error (ISE) used along with plotting or graphing capabilities has taken much of the guesswork out of the motion tuning process. The interactive feedback provide by graphs is still the most robust method of tuning because it works with difficult system applications where auto tuning methods may not. Today, faster and more robust automatic tuning methods have begun to be used in production level motion controllers. Experienced motion control manufacturers are adding proven, pre-programmed and easier to use auto-tuning techniques to allow for quicker optimization of motion control.

Tuning Basics for Hydraulic Motion Control

Before tuning a system, proper selection and sizing of fluid power components is necessary. See **Figure 1** for a representation of a typical hydraulic system. This paper assumes this is accomplished. For suggestions and articles on this subject please contact the author or go to his company's website. This paper also addresses closed loop control of position and velocity with some open loop control. See **Figure 2** for a Block Diagram of a typical closed loop motion control system.

PID and Feed Forward Gains

In an auto tuning system, gains are determined by the transfer function or response of the actuator. In general, simpler systems have simple transfer functions and may not require all the gains of a PID to control an actuator. Likewise, a complex system may actually require more gains than those provided by a PID to tune the actuator. In these cases, an actuator tuned by a PID will be a compromise. The same can be said for feed forwards. The point is that one should have some idea of how the actuator responds to an input so the correct form of PID can be selected. Factors that should be considered include whether the system is going to be used in a velocity or position mode, how the system responds to a set change in the set point and how the set point will be changed in normal operation. For instance, a first order position system has two poles, one for the actuator and one for the integrator that converts velocity to position. Ideally, this system should have two gains to properly place the poles. The options are a PD controller or a PID controller. Note: Adding the integrator does not add a gain because the integrator has its own pole. If the system must position accurately, then the PID is the proper choice. Otherwise, the PD may provide better stability and response. Since a first order velocity system has one pole and it should use a PI controller. Again, the integrator has its own pole so the proportional gain is the only other gain required. Finally, how the set point is changed may influence whether the proportional and differential gains are in the forward path or in the feedback path. Normally a servo motion controller puts the gains in the forward path where the gains are multiplied by the error between the set point and the feedback position. The gains in the forward path add zeros to the system transfer function. This increases the bandwidth which allows the actuator to follow the motion profile easier. However, if the gains are in the forward path and the set point makes a step change, the output will instantly saturate. This can have many negative affects such as hydraulic shock causing leaks, servo motors tripping the over current breaker, damaging equipment and the spilling of coffee. When the gains are in the forward path a smooth target or trajectory generator must be used. If the set point is going to be changed by step changes, the proportional and differential gains should be in the feedback path. This keeps the PID from adding zeros and reduces the system bandwidth. Now the actuator will not respond as quickly to the step change but it also will not follow a motion profile as closely either. Therefore, choosing the correct form of PID is important!

When an advanced motion controller initiates a move to a new position, it does not change the set point in one step. Typically, the motion controller moves the target position toward the requested position in small increments every one or two milliseconds. The target generator (a function of the motion controller) calculates the target position, target speed and acceleration every time period, implementing motion profiles according to the requirements of the application. Motion profiles can include “S” curve or trapezoid ramping, linear or cubic interpolation, or “gearing” of the motion of one axis to that of another. In a typical implementation, there may be a different target generator routine for each motion operation. Once the target generator specifies the next motion, the motion controller initiates closed loop control using a combination of factors. The target generator is also a part of the control algorithm as the velocity and accelerations are multiplied by the velocity feed forward and the acceleration feed forward gains.

Proportional (P) control

This most basic type of closed loop control is just one basic building block of a motion control algorithm. Proportional only control is not used in high performance applications because the gain can only be increased so much before the system starts to oscillate. After that point is reached, increasing the proportional gain will increase the oscillations but not reduce the settling time. See **Figure 4a.**

Proportional-integral (PI) control

Most hydraulic control systems use PI control, which solves the error problem inherent in proportional-only control by guaranteeing that the error will eventually be zero. Even if the error is small, the integral drive term in the control equation will “wind up” (increase) until the positioner reduces the error to zero. The integral gain controls how fast the error is reduced to zero.

Proportional-integral-differential (PID) control

Full PID control adds differentiator gain, which affects how the drive signal responds to the rate of change in the error, and provides damping that reduces oscillations. PID control allows the placement of all the poles on a first order lag position system. This allows one to choose the desired response and calculated the gains. However, few hydraulic systems use the differentiator either because it is not understood or its effectiveness is limited by a noisy and relatively coarse position feedback signal. Ignoring the differentiator will not allow one to place all the poles properly so it may not be possible to tune out oscillations or overshoot with greatly limiting the response. All forms of PID control require an error before it can generate an output. If a significant amount of output is required to move the hydraulic cylinder, there will have to be a corresponding amount of error before the PID control will generate the required drive output.

Consider a system that requires 5 volts to go 10 inches per second, looking first only at the proportional gain. Assume the control has proportional gain of 10 volts per inch of error. From this data we can see that an error of one half inch will be required to move the positioner at 10 inches per second. The error would have to be one inch to move the positioner at 20 inches per second assuming the system is linear. These errors may be acceptable in some applications but for many they are not. Adding the integral drive term eventually reduces the following error to zero. The problem here is the word eventually. Normal hydraulic systems are point-to-point movement systems which occur quickly and don't give the integral term time to "wind up." To minimize the following error during these quick point-to-point moves requires augmenting the PID control with feed forward drive. Since PID control requires this error before there is a control output and goal is to reduce the error, one needs to enhance the PID and anticipate the control output as a function of velocity and acceleration in a predictive manner.

Closed loop control with feed forwards

Feed forwards – predictive functions supported by a hydraulic motion controller – provide a means of estimating how much drive output is required to move or accelerate. For instance, if one volt is required to move the cylinder at one inch per second, it will probably take two volts to move the cylinder at two inches per second and three volts to move at three inches per second. However, this only occurs in a perfect world. PID control together with feed forwards provides much better control than either by itself because feed forwards from a control design perspective are the inverse of the actuator transfer function resulting in a lead and lag action on the control output. Ideally, the feedback position should always be equally to the set point. To do this the system must have a gain of one. The feed forward attempts to achieve this goal by using the feed forward.

Two types of feed forward parameters are often used: velocity feed forward and acceleration feed forward. Each of the feed forwards has an important part to play in a different part of a motion profile. The velocity feed forward plays a part whenever the target speed is not zero, and the acceleration feed forward terms are only active while accelerating or decelerating.

One key point is that for proper hydraulic control, two sets of gains are normally required to support cylinder movement because the surface area of the piston being pushed is different while extending than when it is retracting. It is important to make sure that the motion controller is capable of using different gains for extending and retracting. **Figure 3** shows how a Motion Controller's gains and feed forward parameters are specified using one of the screens provided by my company's RMCWin tuning and programming software.

What values should be used for the gains?

In selecting and testing the parameter values, one of the first questions asked is "What values should I use for the gains?"

It is possible to base the parameter values on a mathematical system model constructed and simulated ahead of time, but this requires serious calculations and even then some of the specifications may have tolerances of ten percent or more. Another approach is to excite the actuator with a step jump and monitor the response. Values such as peak time, percent overshoot and final value can be used to calculate the PID gains.

Typically, motion designers employ a trial and error process of adjusting parameters and comparing the actual motion profile to the target motion profile until a satisfactory result is achieved. The PID and feed forward gains are adjusted so that the target and actual motion profile look similar. In ideal cases they should look the same, but most systems cannot be perfectly tuned and some differences between the target and actual motion profile remain. It may also be difficult to visually determine whether the error between the target and actual motion profile was reduced because of a gain change.

To solve this problem, the sum of errors squared, or the integrated square error (ISE) as it is more commonly known, can be calculated at several points along the motion profile. The goal is to reduce the ISE to as close to zero as possible. This data is used to tune the PID manually: Increase a gain and make a move – if the ISE is lower, the increased gain change was good. If the ISE increases, reverse the gain change and try a smaller increase, or consider a decrease in one of the gain parameters. Eventually the ISE is minimized and the system is tuned. **Figure 4a,4b & 4c** demonstrate graphical tuning tools that aid the process.

Beyond the Basics using Automatic Tuning Techniques

Improving tuning effectiveness beyond trial and error and beyond calculating velocity feed forwards is a valuable incremental advance in tuning. In order to obtain this improvement a method of incorporating the Plant transfer function needs to be accomplished. Auto tuning approaches have been around for some time but primarily are included by suppliers as first order response systems requiring a step input to estimate gain requirements. For real world applications, such as fluid power systems, going beyond the first order response systems is often necessary. Machines often simply don't respond well to step function inputs. (i.e., machine damage may occur in several areas of the machine).

Auto tuning can greatly reduce this risk and by using the motion controller to determine a first order or higher order system deciding what response is best for your application. Using standard control theory algorithms and extensive real world experience, motion control suppliers have moved beyond first order response auto tuning.

What is Applicable?

Proper selection of the appropriate control law is essential to matching the application to the control algorithm. Control laws such as P, PI, PI-D, I-PD method were not chosen for these motion control applications. We make use of Pole placement and integrated square error (ISE) calculations along with more specific and extensive graphic or plotting data acquired from the machine being controlled.

Today, Pole placement and ISE tuning techniques along with propriety enhancements can simplify tuning leading to a method where there is only one parameter to adjust. Using ISE in addition to Pole placement (or lambda) tuning methodology, a PID loop can easily be tuned in minutes instead of hours.

System Modeling, Response and Results

One of the benefits of using the “auto tuning” approach is allowing the designer to focus more on the response – the end result – rather than on the gains – the means to achieve the response. The auto tuner should be able to limit the choices to responses to virtually eliminate problems such as oscillations and system nonlinearities. Testing the resulting system at many frequencies for adequate gain or phase margins can do this.

Another way to obtain optimal tuning parameters is to build a mathematical model of the motion system, using the least squares identification method. This method allows us to identify the coefficients for an equation, whose response best matches the physical system's response to a control signal. This approach requires more programming and math, entirely feasible with today's processing power, and statistically will result in a more accurate model because it uses more data points (usually about 1000) to form the model. To collect the data points, actual position and the control output must be collected at regular intervals and stored in an array for processing by the algorithm.

An additional benefit of advanced auto tuning calculations is in determining velocity and acceleration feed forward values. When set correctly, the feed forward parameters should be able to do the majority of the work leaving the PID gains to take care of changes in the load and unmodeled non-linearities.

The S Plane and Pole placement

A pole is a concept from control theory that relates to an object's response to stimulus. One can observe different responses for an object on an s plane plot where poles should reside in the LHP or left hand plane.

Each real pole has a cutoff or corner frequency where signal attenuation and phase delays occur, and hence are frequencies that should be avoided for proper operation of a machine. At a pole's cutoff frequency, the effectiveness of the drive signal on the motion is attenuated by 3db and experiences 45 degrees of phase lag. The attenuation increases as the frequency of operation increases. As a general rule, it is best if the system is operated at 1/10 the pole frequency. At that rate, little attenuation and phase delay occur.

This philosophy works very well when tuning a motor with a rigid load on the shaft, but not all applications allow the axes to make step jumps in control output because the machine may be damaged. In addition, basing the gains around one or two sample points may be error prone.

Assuming the system gain and the cutoff frequency can be calculated, it is easy to calculate the PID parameters for a positioning system:

$Kp = 3*\lambda^2/ (G*\alpha)$

$Ki = \lambda^3/ (G*\alpha)$

$Kd = (3*\lambda- \alpha)/ (G*\alpha)$

Where:

Kp, Ki and Kd are the PID gains.

G is the system gain in velocity units/volt

α is the system pole frequency in radians per second

λ is the desired pole frequency or lambda in radians per second

In the above equation, the farther λ is from zero (i.e., the higher the corner frequency) the faster the system will respond. The advantage of using this mathematical approach is that it can calculate the gains close enough so that all is needed are final tweaks to adjust for those machine application requirements that auto tuning can't or doesn't take into account.

There are many books on control theory that can provide additional insight into mathematical modeling and control algorithms. My favorite is Digital Control System Analysis and Design by Charles L. Phillips and H. Troy Nagle. A caveat: This is not light reading. Suffice it to say that automatic optimization of control systems via mathematical modeling is quite possible and can be very useful – it is not "black magic".

An Implementation Approach

One can simplify tuning by looking at several readings, (as noted, a rule of thumb may be 1000 data points), and making the realistic assumptions that one can calculate the system gain and pole frequency leading to calculating PID parameters. Velocity and acceleration feed forwards are established once and are constant over the range of different pole placement when determining the desired gains for P, I and D. Changing the PID gains changes the location of the poles of a system and it is not clear how the poles are changing when the PID gains adjust. Pre-programmed calculations can do this for an application by using a simple method such as a slider bar that allows one to move the poles, thereby selecting the response, without worrying about the PID gains..

Use of a Slider Bar and understanding the desired response…….

A slider bar or similar method can be used to move the poles from locations that result in slow system response to those that result in an aggressive response. (Limits are established such that the response isn't too slow or too fast, avoiding step response problems alluded to earlier.) This allows the user to focus on the response rather than the gains as a means to achieve the response. Furthermore, validation of range for pole placement can be done by doing Bode plot calculations to ensure the calculations are proper. See **Figure 5a & 5b** for examples of gain and frequency response.

Caution: Auto tuning is not the answer to every tuning need.

Look out for the potential obstacles. Machines seem to have certain personalities and don't always do what you want them to do. Auto tuning requires some consistency in response to machine issues like non-linearity, natural resonances, dead bands and feedback delays or noise. All may affect the overall effectiveness of the auto tuning process. Thus, auto tuning gets us closer to a perfectly optimized loop and helps us to do faster tuning, yet it may still leave us wanting for more final tweaks in order to optimize control. Better system identification data and modeling can help this as technology improves over time.

Conclusions:

Selection decisions for your hydraulic system and motion controller are basic to optimizing your machines operation and using auto tuning techniques can be a great time saver when initially tuning a system. Designing out obstacles, that could negate intended results, during the initial stages of a project can only improve auto-tuning results, leading to faster and better motion control. Motion control manufacturers with experience in understanding fluid power applications can further assist end users and systems integrators to improve productivity, increase machine life and reduce system startup times by using these tools.

Figures 1 through 5.

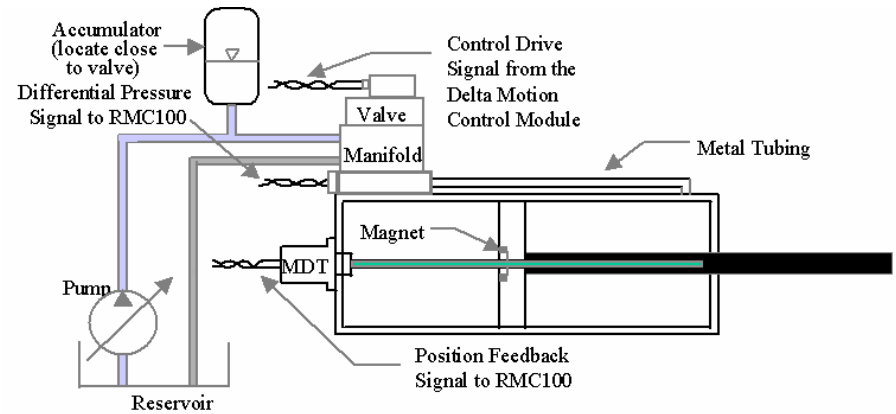


Figure 1. Typical Hydraulic system design for linear motion of a cylinder and using position and pressure feedback devices.

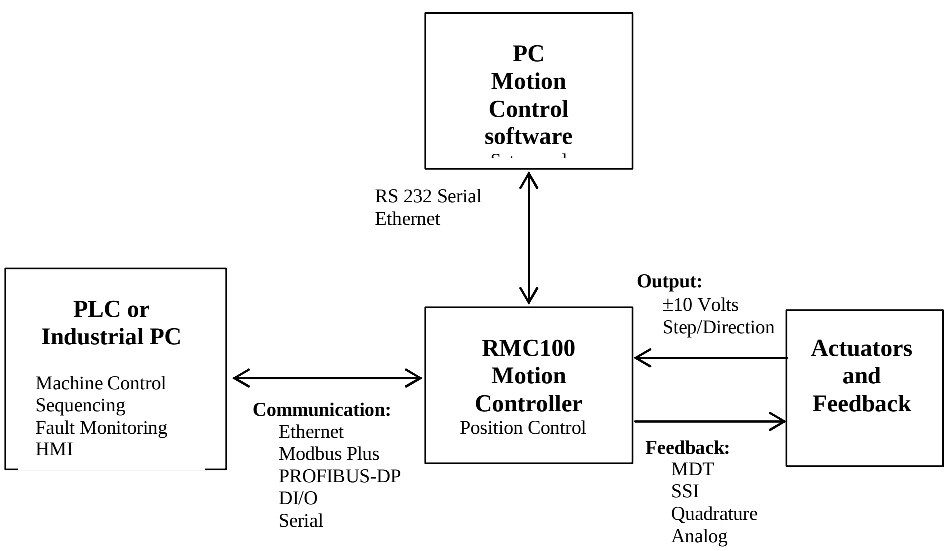


Figure 2. Block diagram representation of a basic closed loop motion control system, including communications to PC for the Motion control software and to a PLC for non motion control functions.

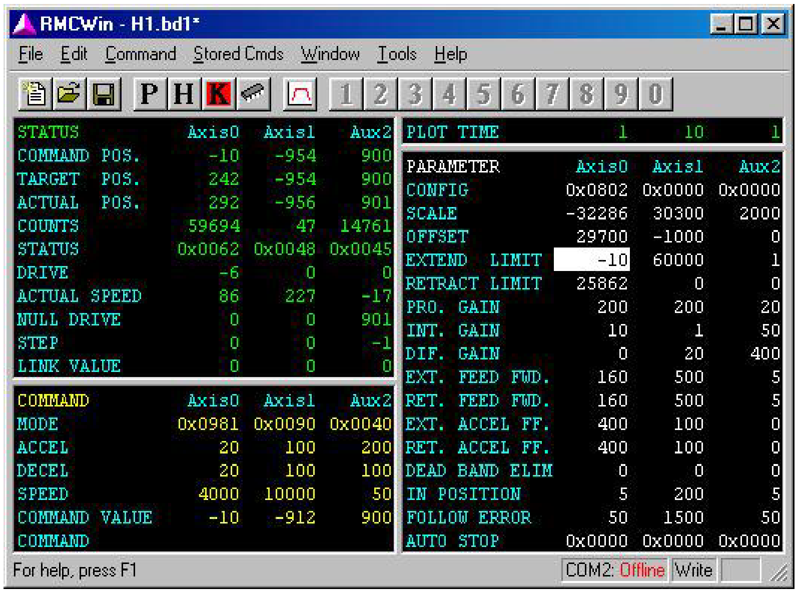


Figure 3. Motion Controller Parameters, such as gains, feed forwards in the Delta RMCWin tuning and programming software.

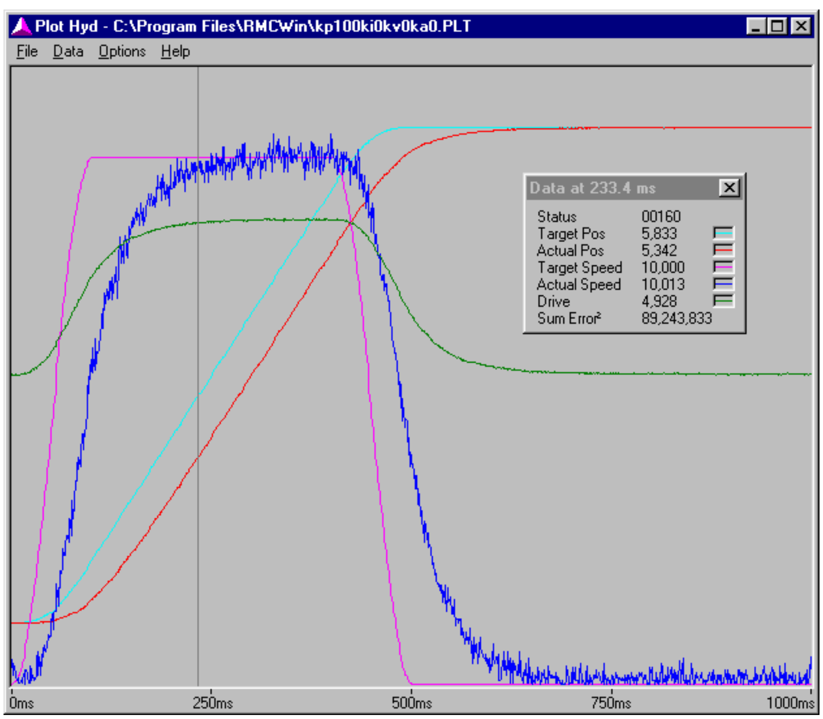


Figure 4a. Proportional control. Using the graphing capability of Delta Computer Systems' RMCWin development software, you can see how the actual position lags the target position (red vs. cyan curves). This system is not optimized.

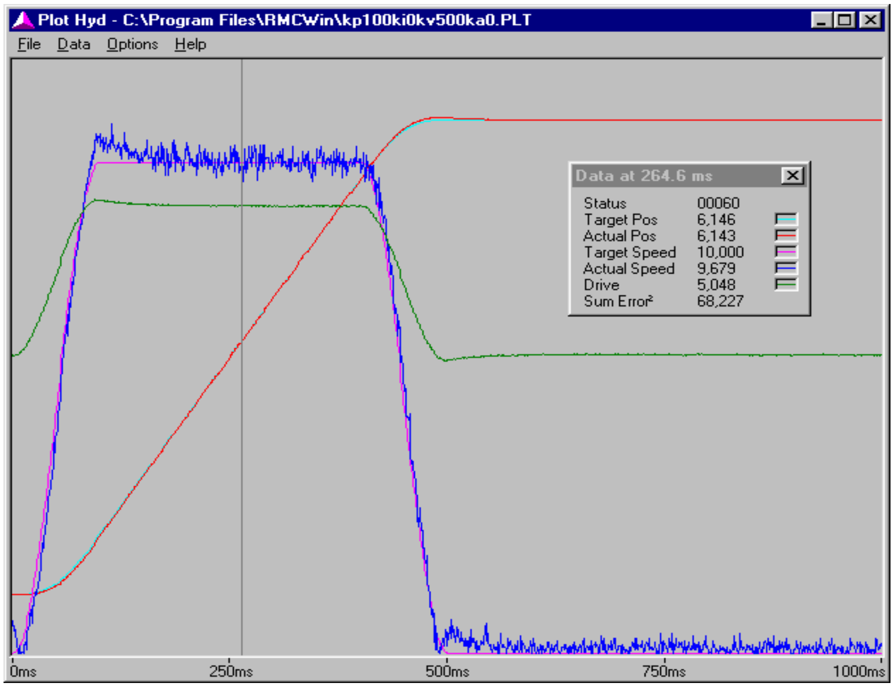


Figure 4b. PID control. Adding integral and differential factors improves the system control and reduces the error (compare cyan and red curves – now overlapping – to those in fig. 4a), but you can see some overshoot in the actual speed (blue) curve compared to the target (purple).

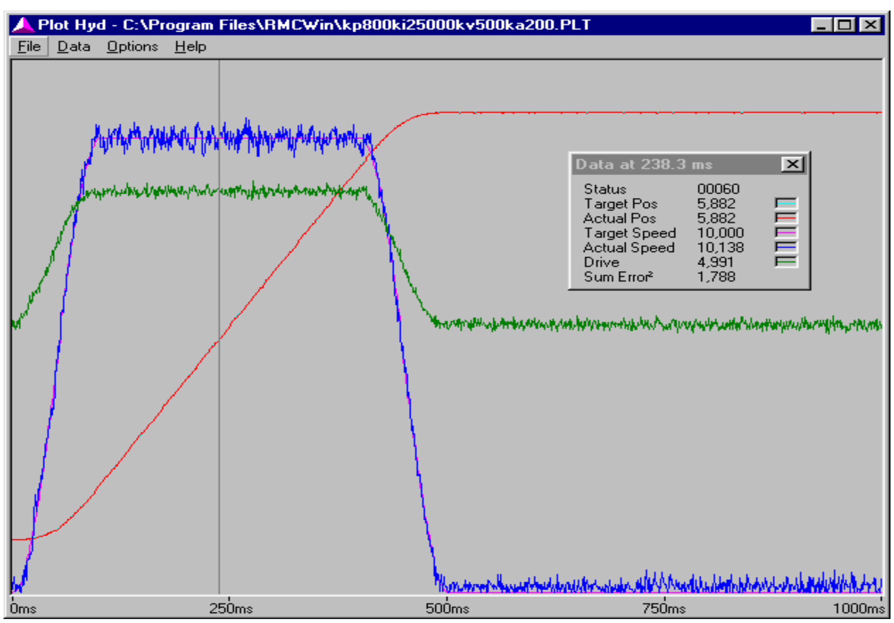


Figure 4c. PID control with feed forward. Adding feed forward greatly reduces the sum of the errors, resulting in actual speed and actual position tightly matching the target profiles.

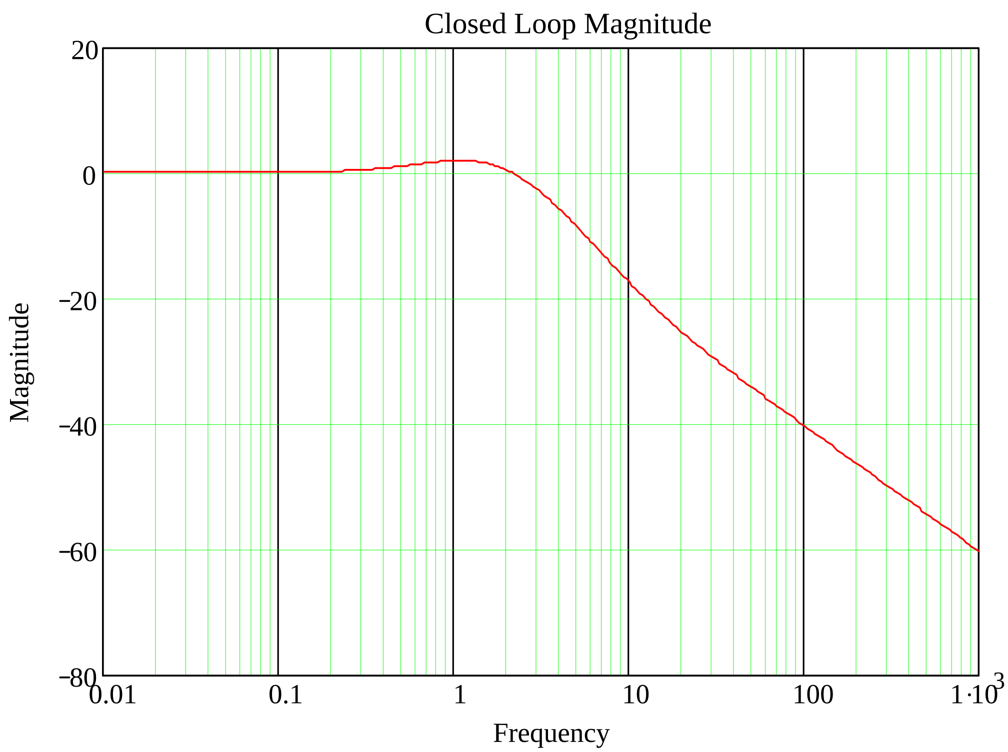
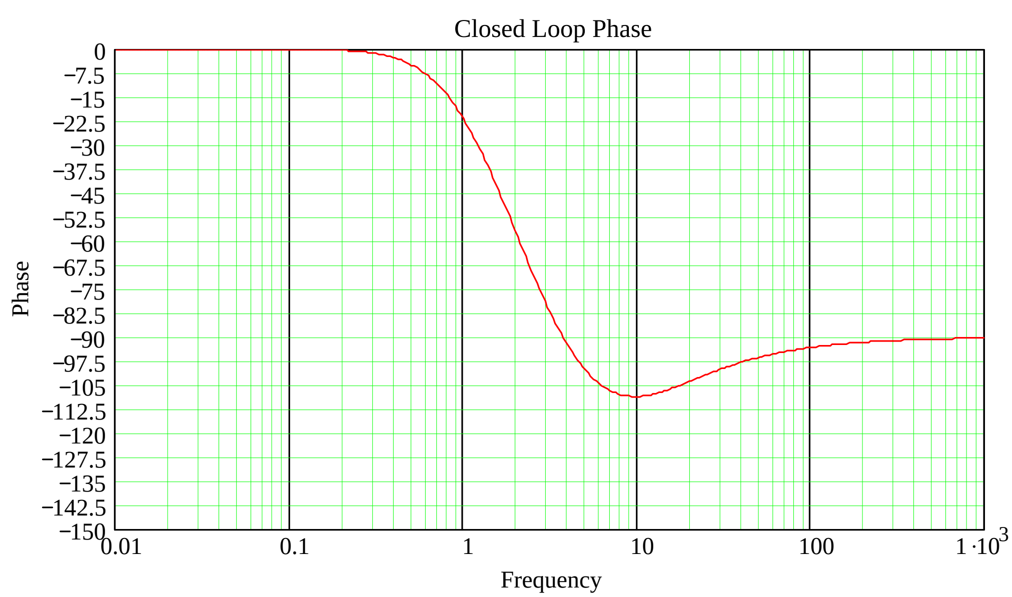


Figure 5a. Bode Plot example above for a closed loop system

Figure 5b. Phase Angle example below for a closed loop system



Peter Nachtwey, President of Delta Computer Systems, Inc. will be presenting this technical paper at the Oregon Convention Center during the Wood Technology Clinic and Show on Thursday March 18 from 1PM- 2:15PM.

Delta's motion controllers will be demonstrated at **Booth # 617**

Additional details and information about Delta Computer System, Inc. are available on the website at www.deltamotion.com or by contacting Bill Savela, Delta Computer System, Inc. 11719 NE 95th Street, Suite D, Vancouver, WA 98682. Telephone 360-254-8688, Fax 360-254-5435, email bsavela@deltamotion.com

deltamotion.com