Chapter 8

Motor Styles and System Considerations

Features

Tying the Servo Knot
Brush, Brushless, DC, AC, or Stepper?
Smart Drive Considerations
Tachometer or no Tachometer?
Mechanical Considerations for Control Selection
Differential or Single-ended?
Other Than Quadrature Feedback
System Friction and DAC Resolution
System Response and Gain Loops
Tuning Considerations and the Optimum Profile
Closing the Servo Loop

Tying the Servo Knot

With all of the available technology and all of the expertise at hand, the world of motion control should be very well defined. It is surprising to discover that problems plaguing us years ago are still haunting us. There are an ever increasing number of questions being asked relating to servo tuning, the servo control loop, and the how different gain structures work.

In previous chapters, various methods were provided for incorporating the constituent parts of a servo system. The intent of this chapter is to pull the pieces together (tie the knot) by integrating real world outside forces into the solution. The objective is to give you more ammunition when contemplating your system requirements and selecting your control. The bottom line is to make your system functional to the required specification **the first time around**.

Brush, Brushless, DC, AC, or Stepper?

It is the system load, inertia, profile, environment, and other such factors, which will govern the choice of your motor package. The electrical and physical differences of available motor types ensure enough performance variety for you to successfully choose the right one for your application. Only you, as the engineer of the project, can judge which motor/amplifier package will satisfy your system requirement and budget.

The following information presents some simple criteria that you can use to help you decide which type of motor might be best suited for your system:

DC Brushless

- High speed—short index moves
- Heavy loads (high torque control)
- Short duty cycle moves
- Resolver to Quadrature converter option (no physical encoder requirement)
- High power to size ratio
- Low to medium armature inertias
- High accel./decel. capability

PWM Motor

- Long high speed moves
- Low to Medium speed short moves
- Light to Medium loaded index moves
- Brush replacement required
- Medium power to size ratio
- Low to medium armatures inertias
- Medium accel./decel. capability

Stepper

- Light load moves
- Speed/torque dependency
- Positioning = +/- 3% of a full step
- No encoder requirement
- High power to size ratio
- Medium armature inertias
- Low speed operation
- Low to medium accel./decel. capability

General purpose DC

- Low to Medium speed
- Light to heavy loads
- High inertia armatures
- Velocity mode or low accuracy positioning
- Not designed for true servo application
- Brute force on/off control
- Low to medium accel./decel capability

AC (Brushless, Servo, Vector, etc.)

- Low to high speed
- Low to Medium accel./decel. capability
- Light to Heavy load capability
- Medium to high armature inertias
- Brute force on/off control
- Moderate servo positioning capability

This information is based on my experience in the field, and is also based on general information supplied by various motor manufacturers. Its purpose is to provide you with a starting point. As you gain experience, you should modify this source listing or develop your own list of guidelines. Knowing what you are comfortable with can save you hours of tedious testing on subsequent projects. You should not have to *reinvent the wheel* each time you need to choose a motor.

As an added suggestion, you should always maintain at least two motor vendors and two motor brands with which you are comfortable. Second sourcing will ensure the best pricing; and depending on the motor style, it may contribute to short order lead times.

Smart Drive Considerations

The motion control computer is the work horse for controlling move profiles that must be continuously altered, in either their velocity or position trajectories due to changing system conditions. However, there are many instances when a reasonably *smart* drive is perfectly capable of running a system without the help of a computer control (see Chapter 1).

With good timing charts (Chapter 4) and a good knowledge of the system's operation, a smart drive can save you hundreds, perhaps thousands, of dollars in equipment and labor.

In many cases, the motion control computer can actually inhibit the performance you are trying to achieve. Below is an outline of questions you need to ask in order to understand whether or not your system will function under simple *velocity control.* If your system is capable of operating with a simple smart drive, do not complicate your problem by incorporating a computer. It could become just that, a problem!

First, if your system is to run **only** in a *velocity mode* with no specific stopping position, it is a very good candidate for a smart drive.

The next series of questions to answer are:

What velocity stability does your system require? *Can it be achieved with a simple voltage control? If not, can it be achieved with tachometer loop?*

If a stopping position is required, what is its tolerance? *Can a **controlled ramp** be used to meet this tolerance when a stop signal (switch) is given?*

If your system requires multiple velocities in a given move, is the time required for the smart drive to make the real world velocity change within acceptable timing chart requirements? *Can alternate timing, or the acceleration or deceleration be flexed to meet the need?*

If the system requires acceleration and/or deceleration to change while the move is in progress, can the change(s) can be done in an allowable timing chart window? *By the drive itself?*

If your system is to be operated by some form of PLC (Programmable Logic Controller), would it be more cost effective for you to place the motion control portion of the system in the PLC structure you are already planning? *Can the PLC, as an ON/OFF type of control, fulfil the motion requirement while providing the drive to handle the actual move?*

Does your system experience relatively small changes in load weight when moving?

If you answered yes to most of the questions posed, it would be well worth your time to investigate smart drive alternatives.

Tachometer or No Tachometer

In any system design effort, accumulate as much information about the system as possible to provide for first-time on-line success. Ask questions directed toward determining whether a *tachometer velocity loop* should be used. Ask yourself questions concerning velocity stability and move distance, as well as acceleration and deceleration requirements. Although all systems need answers to these three questions, it is not immediately obvious how a tachometer loop can improve the positioning requirement or the DAC[1] resolution.

The effect of the tachometer loop on position and DAC resolution is demonstrated in the following example:

Example 1: Tachometer loop for an Inline System

DAC resolution, and Velocity considerations:

Assume a system must operate at a velocity of 1 RPM.

Load weight = 500 Lbs.
Friction Coeff. = 0.07
Screw Radius = 0.5 inch
Screw Pitch = 5

500-pulse quadrature encoder (10,000 counts per inch)

$$\textit{Static Friction Force} = \frac{(500)\ (0.7)}{5} = 3.5\ \textit{inch-pounds}$$

Motor Torque Constant (Kt) = 5 Inch-pounds / Amp.
Amplifier gain = 25 Amps / Volt

Breakaway Friction = Running Friction = 3.5 Inch-pounds

To break friction, a current of 0.7 Amp. (3.5 / 5) must flow through the motor. This equates to a motor amplifier input voltage of 0.028VDC (0.7/25), which is supplied by the DAC.

---

[1] DAC is Digital to Analog Converter

If the DAC is 12-bit (4096 steps), and the full ±10VDC analog signal voltage is used, then one DAC step is equivalent to 0.0048VDC. Therefore, it will take 6 DAC steps (5.7344 rounded up) to break friction and maintain the running operation.

In other words, if gain parameters are set to produce 6 DAC steps with one error count, the equivalent DAC resolution will be reduced to 682 steps (4096/6) which is only a 2.6 times resolution improvement over the 8-bit DAC versus the otherwise 16 times resolution improvement. The integral parameter in the PID loop is normally used to remove the friction error element. Once the system breaks friction, if the integral, proportional and derivative terms are set correctly, the DAC will be properly scaled. However, if an integral gain factor is not part of your gain structure, it is likely that the system could suffer a reduced DAC resolution. You will need to verify this to insure system performance.

If a 2000 RPM motor is to be used with the 12-bit DAC operating the motor package in a voltage mode, the capability of the DAC will be 0.976 RPM-per-DAC-step (2000 / 2048). However, the RPM resolution with a misapplied gain structure could be 5.865 RPM-per-DAC-step (2000 / 341), since the velocity will constantly interpolate between 6 DAC steps and 0 DAC steps to achieve the 1 RPM specified.

By including a *tachometer velocity loop* in the motor package and properly setting the motor amplifier gain, the motor amplifier will, with the lowest DAC voltage (0.0048VDC) applied to it, ensure that the motor will rotate (break friction). Note that this movement can start out at a velocity higher than 1 RPM due to the gain of the amplifier required to break friction. However the control with proper gain settings will interpolate through 1 DAC step, shifting to achieve the 1 RPM-per-second specified. Not only will there be a significant improvement in the smoothness of the acceleration profile, but as an added benefit, the controller gain structure can be maintained at a more optimum setting than with no tachometer loop incorporated. This will then increase the responsiveness and stability of the overall system.

A byproduct of the tachometer loop is to place the motor into a velocity mode, which ensures that the velocity of the motor is more predictable, and in the case of higher speeds, more stable. It can also be simulated with greater accuracy.

Position, and Move Distance Considerations Using a Current System

Use the same system parameters as in the above example requirement:

Move the system, which is at a standstill, at some velocity to a position that is three encoder counts away.

If the system needs 6 DAC steps to break friction, the trajectory generator will move only three counts to completion (see Chapter 11). Therefore, the error generator will only develop three error counts for the move. The two worst case gain situations will be low gain for long move stability and no motion for the short moves, or high gain for short move capability with possibly unstable long moves. Obviously, two different gain settings might be required. This represents a software cure.

What if we use a *tachometer velocity loop* again, giving the motor amplifier the ability to force the motor to break friction with a 0.0048VDC single DAC step input. The motor amplifier will force motion on the first DAC step, insuring a completed move. As previously demonstrated, the unit can now maintain the low gain DAC requirement for meeting the long move stability, and it will also have the short move dimension capability as well.

The tachometer loop will take the motor drive package out of the current mode providing the system higher velocity and position stability with changing load requirements without having to *model* the system to locate its *weak* points or constantly change its gain characteristics.

In the discussion on gain structures, you will see how the tachometer loop can be the hardware counterpart of software gain structures such as feed-forward and feedback, and it can also help maintain low following error.

Mechanical Considerations for Control Selection

The general machine operating requirements lead to the addition of sensors and other monitoring or controlling devices that will in some way function with the product motion. For instance, in a tension system there will be some form of tension monitor, which will feed tension information to the motion control unit. The motion controller will have to analyze the information and adjust the velocity of the system accordingly.

On the other hand, a system that makes wood products might have multiple product sensors, limit switches, and possibly product measuring equipment. All these sensor signals may have to be handled in *real-time* rather than by polling. A smart motion controller (possibly standalone) might be needed to help the host computer or PLC in real time operation.

By developing a machine or axis **timing chart** (see Chapter 12), the various sensor and interlock requirements can be determined. From this, the software can be outlined (not written) to separate the general operation from the *real-time* operation. Profiling the system would then allow selection of the motor amplifier package; and finally, with all of the *creeping feature-isms* accounted for, you can then choose the control. As stated in Chapter 2, *creeping feature-isms* are usually paid for out of project profit money. If you can figure out these features in the beginning, the project will be more of a dream than a nightmare.

Differential or Single-ended?

The problem with interfacing equipment, is that all of the electronic knowledge that people have forgotten about, usually comes into play. Impedance matching, frequency, resistance, voltage, current, power dissipation, noise, and other considerations must be planned for when interconnecting system components.

The selection of a differential or single-ended signaling device should be done *up front* while working out system interface requirements. Since it is possible to obtain a differential or single-ended interface for most any electronic device (such as general purpose signaling, computer I/O or encoding), the implementation now becomes environment, distance, and budget oriented.

The main advantage to using a differential driver over the single-ended unit is the high degree of noise immunity offered. This becomes a very significant factor in working environments where wire

lengths can exceed 100 ft. Differential signals also work well with simple twisted-pairs that save the cost of shielded cable. Decide with caution which approach you should implement. Many applications require shielded signal transmission.

The use of differential signals whenever possible will help maintain that extra degree of isolation electronic systems need. In cabinets where high power motor amplifiers and low frequency switching are done, the shielded differential approach can be a lifesaver.

Other than Quadrature Feedback

There is no rule that only encoders can present the feedback to the servo loop. As a matter of fact, there are many aircraft simulators around that input a variety of force information through strain gauge signals that are integrated into the various motion calculations. The resulting solutions are then applied to cylinders, motors, or other types of prime movers to adjust for the sensed pressure vectors. The key here is to insure that the feedback resolution is tighter than your tolerance requirement. Remember that all computers, counters, converters, and similar equipment have propagation delay, math round out, and tracking errors.

All of these factors can add up to an inability to achieve the position accuracy and move stability you may require. Since the end result is to be stable and repeatable, a feedback device with a tight resolution can usually eliminate errors caused by computer math and general servo hunting. I generally use a 5-to-10-times rule for this scenario. For example, if I am constrained to a 0.001 inch tolerance, my feedback sensor is calibrated to 0.0002 or 0.0001 inch. By doing this, the system will have the resolution to perform within the required tolerances.

If it's repeatable . . . it's predictable . . . and, therefore, it can be controlled.

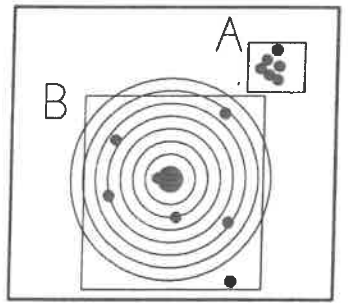

Figure 8.1: Repeatable versus predictable results.

System Friction and DAC Resolution

It is not widely known that the static friction in a given system can indirectly alter the resolution of the controller DAC. Because of this, many systems lose velocity stability and/or the low count positioning accuracy they were aspiring for. Just what part friction plays is the topic of this section.

In systems where a motor is used to move the load, the motor must produce enough torque to overcome the *system friction force*, prior to motion. This friction force is separate from the system acceleration torque. The friction force (created by the system friction times the system weight times mover Arm length) is a constant and needs only to be exceeded to produce motion. The motor used to generate the motion will develop the required friction breakaway torque when a proper current value is passed through it. The torque that a motor can produce with a given current is equal to:

Motor Torque = (Motor Current) (Motor Torque Constant)

When the system is sized and a motor is selected, the motor torque constant is chosen to ensure that the required system torque is met. The selected torque constant is to be considered part of the motor performance requirement, even if the motor torque constant is selected only to maintain low motor current. The torque constant will then govern the amount of current the drive amplifier needs to deliver to the motor in order to move the system load.

The motor drive amplifier will have a current gain rating in Amps-per-volt. It is this parameter that interfaces between the DAC and the motor. The motor amplifier gain is specified after motor selection. Knowing the system static friction force (breakaway and running), and DAC voltage resolution required, verify whether a tachometer loop is necessary, to keep the entire DAC resolution intact. For a detailed explanation of this effect refer to *Example 1* on Tachometer Loops.

System Response and Gain Loops

Gain structures, how they work and which would best fit a given application is perhaps the most discussed topic in the motion industry. Each time a new gain structure appears, it is generally offered as a remedy to whatever problem exists. User confusion is often created when insufficient information about the gain method is presented, and the result is that the user finds it difficult to figure out how to apply it.

Therefore, the topic of this section is . . .

Gain structures and what they do.

The gain structures discussed are:

- PID and Lead/Lag (PD)
- FeedForward, FeedBack, and Adaptive
- Notch
- Self optimizing

PID and Lead/Lag (PD)

As will be presented in Chapter 11, the classic PID structure consists of four independent elements.

The PID equation is:

Output = SetPoint + Proportional + Integral + Derivative

The PD loop uses the same calculations as the PID loop, but it usually incorporates some additional form of gain structure into the equation for added flexibility and control.

FeedForward, FeedBack and Adaptive

It should be obvious that when using a PID gain structure, an error between the desired and actual positions **must** exist in order to generate a DAC output. It is common knowledge that in order to simulate the response of a system, you need to know the total system inertia, motor parameters, the system resolution, gearing and general profile information.

The objective of the feedforward, and feedback gain structures is to estimate how the system will function in subsequent updates and to make corrections **now** for those estimates. The corrections are generally done by changing the value of the desired position that will increase or decrease the error value, causing the error generator to produce a larger or smaller DAC output. Note that the system will still work with the lower PID or PD gain structure needed originally to maintain the system at the proper level of stability. Therefore, the purpose behind this gain structure is to maintain the following error as close to zero as possible. There are a couple of inherent problems with this method . . .

Problem 1: Since the system is simulated in order to estimate the *future*, all information used for the simulation must be correct in order for it to work.

Problem 2: To optimize the estimates, the controller must have the ability to record the error along the path for future comparison and utilization. Most systems do not have the memory capacity to retain the amount of data it would take to do this, or have the CPU overhead available to continually do the required update simulations.

Notch Filter

The *notch filter* is a software constructed filter system similar in principle to those used in communications. The application of notch filtering with respect to servo systems is to automatically change the motor response, and thus the system, in areas of mechanical instability or resonance. The function of a notch filter is to eliminate these instabilities by dampening the system response.

There are certain things about the system that you must know and fix both in time and position in order for the notch filter structure to work. If system resonant frequencies change due to load or system mechanical changes, or if your system operation does not repeat its performance in the areas in which the filter parameters had been calculated, the notch gain structure might not handle the system as expected.

This is not to say it does not work: it does; but for it to work, the systems with which it is to be used must be capable of being modeled. Therefore, any changes in the system using the *notch filter* approach will require remodeling, and thus, re-tuning.

Self Optimizing Torque Profile

There are some gain structures that use the motor and system parameters to compute optimum motor performance while the system is operating *on-the-fly*. However, the function of a well-tuned system is to optimize the required system profile, not the capability of the motor. Therefore, this type of structure is useable only in very specific applications.

Remember:

Optimizing the profile has nothing directly to do with the motor, it has to do with the *machine timing requirements.*

Tuning Considerations and The Optimum Profile

Just what is the *optimum* profile, and when can the tuning operation be considered complete? Having been involved in the tuning of hundreds of systems, I feel very comfortable in stating . . .

A properly tuned system is one that performs as required.

The tuning operation begins at the specification. What is the profile requirement for your system in order to operate as planned? This is found by the machine timing diagrams (see Chapter 4). The timing diagrams not only give you machine timing, they also provide you with the interlock requirements regarding monitors, limits, multiple machine interlock, and motor duty cycle information among other things. Also, duty cycle information is necessary for properly sizing the motor amplifier package. It insures the package selected will not electrically or physically fail.

Once you have worked out the machine timing, **stick to it**! If you need to make the system run faster, do it before it is built. Several months ago, I was approached with a tuning problem. I asked the customer about the machine timing requirements for the axes in question. I found that 0.05 second timing windows were *optimized* to 0.2 seconds. Needless to say, the inertia torque-loading went up as a square function, and the motors could not be rendered under acceptable control due to the higher torque-loading during the acceleration ramps. However, when we tuned the axes to the timing chart requirements originally worked out by the *primary* designers, the 0.05 second profile worked!

The purpose for a *timing chart* is to aid in the design and understanding of the machine functionality. The timing of the mechanics cannot be changed without verifying the torque and acceleration capability of the motor with respect to the system inertia and timing requirements. Motor torque-loading is a direct result of the acceleration (or deceleration) rate times the inertia. If you are optimizing the profile when the machine is built, optimize it to what your profile was specified to be, which is not necessarily the best it can be.

The ability to control a piece of equipment
– is only limited by –
the ability of that piece of equipment!

Do not attempt to force the system to do something it was not designed to do or capable of doing.

Stop when the system profile requirement has been met,
not when the motor response capability has been peaked!

Closing the Servo Loop

As discussed in other sections, understanding how system friction can distort the DAC resolution is important to consider when selecting the motor and motor amplifier package. The motor amplifier gain must be chosen to allow one step DAC changes to move the load if the full 2048 DAC steps are to be realized.

Once you know the size of the motor and the gain of the motor amplifier, decide whether to include a tachometer loop on the motor. The positioning and velocity stability requirement(s) will provide you with that answer. The gain values will be directly related to move velocity and distance. By adding a tachometer loop to the system, while still operating the motor amplifier in the *current mode,* you can usually realize an optimum gain structure for both fast-and-slow, and long-and-short move situations.

The tachometer loop also simulates to a degree the feed-forward and feedback gain structures. The tachometer actually senses and forces a response to changes in the velocity of the motor prior to the motion controller even detecting changes that have occurred (update phenomenon using CPUs versus analog control within the motor amplifier).

The encoder resolution is then selected by multiplying the *real* resolution requirement by a factor of five to ten. Then, by using a trajectory simulation package (see Chapter 10), you can plot the response curve of the trajectory generator and the estimated system performance. As you test the required profile, changes in the encoder resolution and gain values will either improve or degrade the *actual* versus *desired* trajectory plots. When complete, the simulator will have helped verify the selected resolution and gain entries.

Using the proper tools to solve motion problems is no different than using the proper tools to change a light switch. The objective is to understand how to use the tools, and how to apply the numbers you are receiving. In simple terms, proper understanding will enhance profitability.

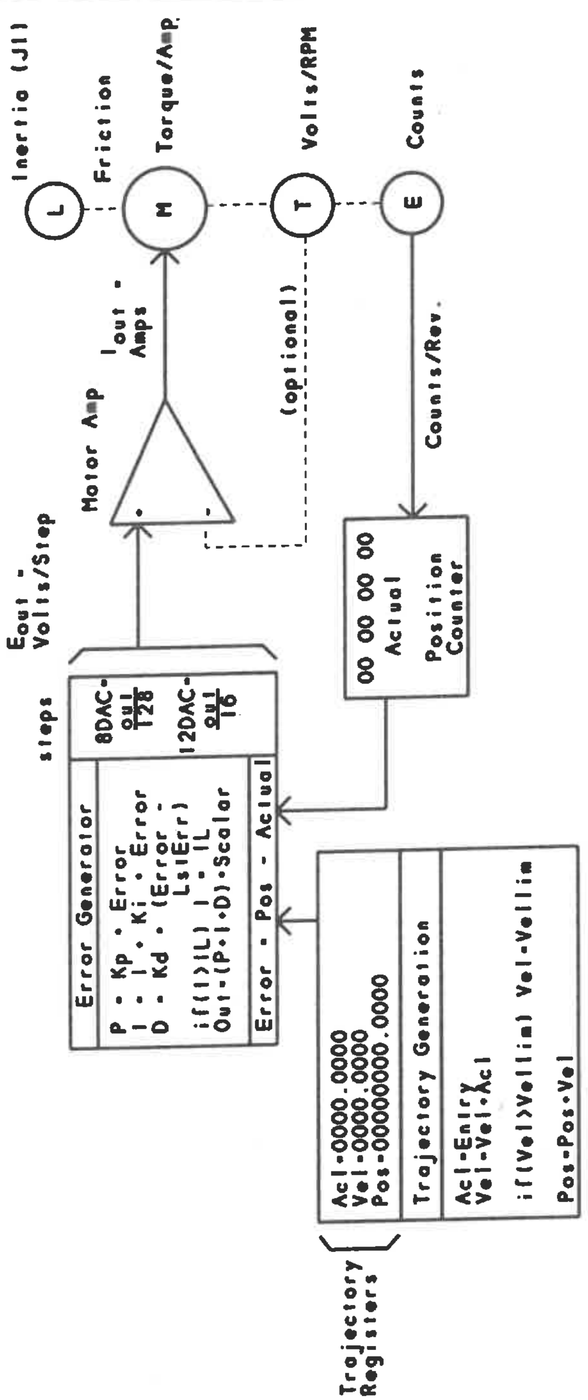

Figure 8.2 Typical trapezoidal generator and system.

Notes: