SERVO DRIVES

By Bill Colton
Baldor Electric Co.
Commerce, CA

(**Note:** This article assumes a basic understanding of motors. Its purpose is to take some of the mystery out of servos.)

Servo motors come in so many types and flavors that it is difficult to define them in a way that is accurate and universally acceptable. It is possible, though, to describe some of the things commonly found in servo drives, as well as typical configurations.

Servo drives are designed to convert electrical power into precision-controlled motion–e.g., controlled torque (torque servo), controlled speed (velocity servo) or controlled position (positioning servo). This typically requires at least three elements: the motor, feedback of some sort, and an amplifier.

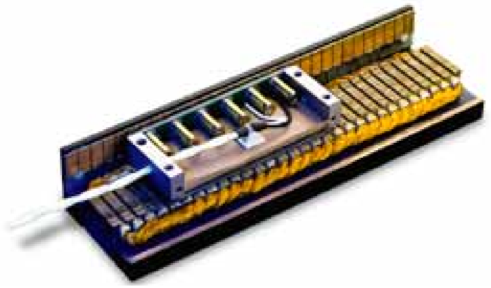
THE MOTOR

DC brush motors

DC motors can be either rotary or linear. Rotary DC motors typically are long and thin, which allows for quick acceleration due to the lower inertia, as well higher speed due to the lower centrifugal forces of a smaller diameter armature (see Figure 1). The armature is skewed to help reduce low-speed "velocity ripple."

Figure 1. Rotary brush DC servo motor.

Figure 2. Linear brush DC servo motor.



Linear DC motors have the commutator and windings along the path of travel, and power may be supplied to the brushes by a bus bar or an umbilical cord (see Figure 2). The "moving short" with the brushes has a permanent magnet, which is attracted to the energized stationary coil. A linear bearing is used to create an air gap and low friction.

DC brushless motors

Brushless DC motors may be either rotary or linear and come in many varieties. They are probably the most prevalent kinds of servo motors due to their quick response time, low inertia-, weight- and size-to-torque ratios, and reasonable cost.

Rotary brushless DC motors have either ceramic or rare earth magnets banded onto the rotor (see Figure 3). Ceramic magnets (typically ferrite) cost less but have higher inertia and size per torque than high-performance earth magnets (typically samarium cobalt or neodymium-iron-boron).

Figure 3. Rotary brushless servo motor.

The stators of these motors are wound with basically standard, but low-inductance windings. Some may be epoxy encapsulated for protection from elements (e.g., machine tool coolant) and to provide mechanical rigidity to reduce wire-to-wire abrasion from high current, shock and machine vibration. Rotary brushless DC motors exhibit low inductance and small electrical and mechanical time constants.

There are two main types of linear brushless DC motors–iron core and cog-free. Both types support the moving short with one or more linear bearings, which provide an air gap and reduce friction.

The iron core motor has one or more columns of magnets with alternating poles (north-to-south). Higher force motors may have several rows of magnets (see Figures 4, 5 and 6). The coils in the moving short are energized, which attracts them to the magnets and moves them along the column.

Figure 4. Brushless motor magnet track and moving short (coil with attached payload).

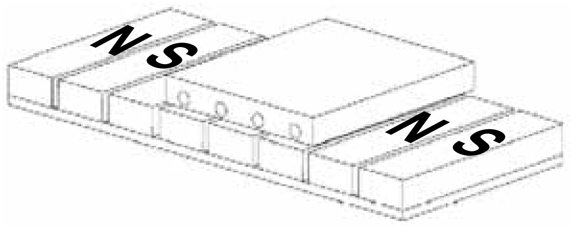


Figure 5. Single row of magnets in a long column.

Figure 6. Multiple row of magnets in a long column.

The payload usually is attached to this piece.

The laminated iron core magnifies the flux density. Since the core is also attracted to the magnets, a velocity ripple will occur during movement. If this is an issue for the application, a cog-free linear motor may be a better option.

For applications that only require point-to-point moves, cog-free linear motors also provide the most force in the smallest package and at the lowest cost.

A cog-free linear motor consists of a column of two magnets of like polarity attached to either side of a machined channel. As with the iron-core style, polarities along the magnet track assembly of cog-free linear motors alternate north-south-north-south. A coil of epoxied magnet wire supported by a linear bearing rides between the magnets. Energizing the coil creates the attraction to move it along the magnetic track. Since there is no iron core to fight the smooth attraction of the coil, the velocity ripple is minimized.

Figure 7. Cog-free brushless linear servo motor.

AC induction servo motors

AC induction servo motors can be linear or rotary. The rotary design typically is long and thin, making it suitable for higher speeds and quicker acceleration and deceleration profiles due to the lower inertia. A separate constant-velocity blower motor is often attached to the back of the servo motor for cooling during low-speed operation. The stator has a standard, low-inductance, three-phase AC motor winding, which sometimes has special volts/hertz ratings and/or wye-delta switching.

Figure 8. Rotary induction servo motor–standard inertia.

Linear induction servo motors operate from the same principle as rotary induction motors. They have a three-phase coil (typically on the moving short) powered by an umbilical cord or bus bar and ride on two linear bearings above an aluminum “reaction plate” that is mounted on steel (see Figure 9).

Figure 9. Linear induction servo motors.

The advantage of linear induction servo motors over brushless servo motors is their maximum speed. Since they do not have to fight the back EMF (electromotive force) of the motor through the magnets, they can reach velocities of more than 2000 in/sec (5080 cm/sec) or 110 miles/hour (177 km/hr) with enough travel.

FEEDBACK

Servo systems receive feedback from the motor and sometimes from the product and/or process. The focus of this article is servo drives, so elements of process feedback are not included.

Brush-style rotary DC servo motors may use tachometer feedback (typically 7 volts/1000 rpm), encoder feedback, and/or resolver feedback. Brush-style linear DC servo motors use encoder or laser feedback.

Brushless rotary DC servo motors may use hall feedback and/or encoder feedback, or resolver feedback. Brushless linear DC motors may use linear encoder feedback with halls or sinusoidal commutated linear encoder feedback.

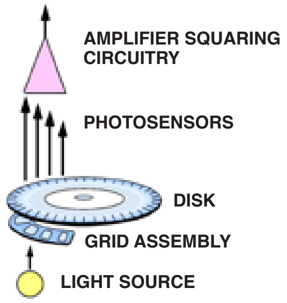
Induction rotary motors use a rotary encoder, whereas induction linear motors use a linear encoder.

Brief descriptions of the various feedback devices can be helpful in understanding how they work.

Figure 10. Tachometer.

Tachometers employ a permanent magnet field. The armature, which is coupled to the motor shaft, induces a DC voltage proportional to velocity as the shaft spins (see Figure 10). This voltage is brought out through a commutator and brushes to the amplifier for speed and direction information.

Figure 11. Encoder.

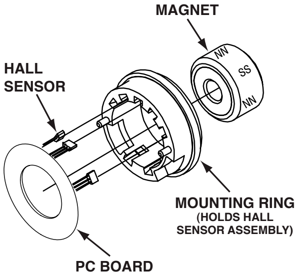


Encoders employ an LED to emit light that is directed through a slit called a gate and then through a scale (commonly glass) with very fine, uniformly etched marks. When these marks align, light passes through and hits a photo cell; otherwise, between marks, no light hits the photo cell. As light strikes the photo cell, a pulse (generally 5, 12 or 15 volts) passes back to the amplifier or motion controller (see Figure 11).

There are typically 2 or 3 signals. The first is called A; if the signal is differential, A/ will also be present. The second signal, B (and B/), has the same ppr–pulses per revolution (rotary)–or ppi–pulses per inch (linear)–as A, but it is offset 90 degrees. This makes it possible to obtain velocity, direction and position information. The third signal, called C, Z or I (with corresponding C/, Z/ or I/), occurs only once per revolution. This signal is only found on rotary encoders.

Hall feedback may be part of the encoder or a separate device. A hall device is a magnet with a sensor that indicates the polarity (and therefore the location) of the magnet (see Figure 12). It is used for commutation of the brushless motor. When part of the encoder, hall feedback is really just another encoder channel. This is often called “encoder with halls,” “hall tracks” or “comm tracks.”

Figure 12. Hall feedback device.



Hall feedback is used strictly for motor commutation, so the device must be aligned with the magnets according to a timing circuit that the amplifier can use.

Resolvers are essentially rotating AC transformers that employ a small AC input for the power circuit.

A rotor is attached to the main motor shaft. As it passes through the alternating stator polarities, it generates a voltage that cuts the “output windings” (two windings 90 degrees apart electrically).

The output is two sine waves 90 degrees out-of-phase. The number of sine waves per rotation depends on the number of poles–ranging from 1 to 80 in common, commercially available motors, but going much higher in very special ones. The output goes into the amplifier and is “decoded” to encoder output.

Resolvers are also used for commutation, so the output must be aligned with the magnets using a timing circuit that is compatible with the amplifier. The encoder signal can be brought out to the motion controller.

THE AMPLIFIER

The amplifier converts the output power from the distribution panel to controlled output that will cause the motor to move at the correct velocity. Most servo amplifiers are PWM (pulse width modulated) style.

The converter section of a single-axis amplifier uses a diode bridge to change AC input power to DC power, which is then “smoothed out” by the addition of a capacitor. This DC section is called the bus (see Figure 13).

The output is then “chopped up” by transistors (typically FETs or IGBTs). These devices rapidly turn the DC power to the motor on and off. If the transistor is closed for long periods (motor receiving power) with short open periods (motor not receiving power), the average terminal voltage goes up. If the transistor is open for long periods (motor not receiving power) and closed for shorter periods (motor receiving power), the average voltage goes down. The amplifier can close an “upper transistor” in one phase and a “lower transistor” in another phase, creating a path

Figure 13. Converter section of a single-axis amplifier.

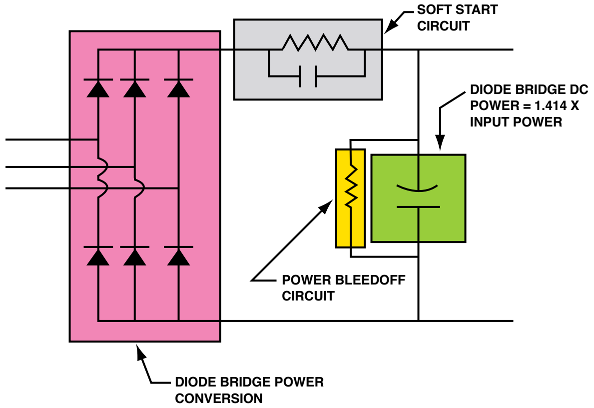


Figure 14. Typical single-axis, three-phase output. For DC, eliminate one set of transistors.

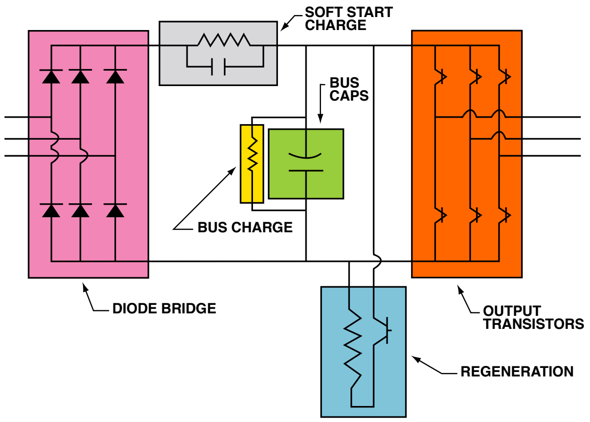
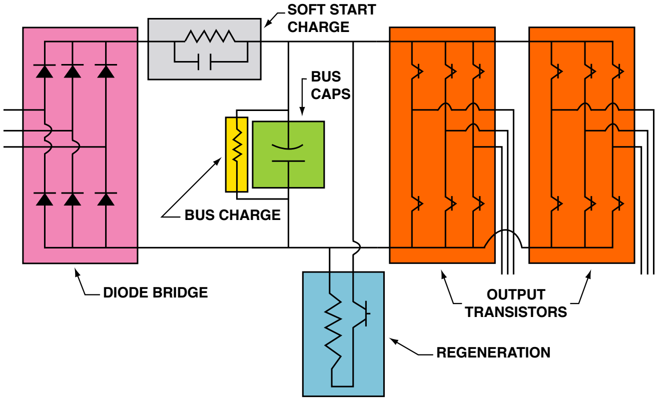


Figure 16. Single-axis amplifier.

Figure 17. Multi-axis brushless servo.

Figure 15 Single-axis AC power amplifiers and motors.

Figure 18. Multi-axis, three-phase output.



through the winding (see Figure 14). If the motor is AC, the polarities can be reversed by switching from an upper to a lower transistor and back on each of the phases (see Figures 15 and 16).

Another topology called multi-axis essentially splits the amplifier after the bus (see Figure 17). The bus is bought out to terminals and then connected to output (transistor) sections. The bus must be sized to handle all axes of continuous and peak current that the application requires.

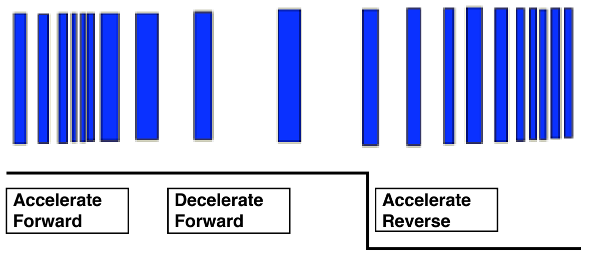
Notice that two output sections are driven from one power converter. Output sections can be added in series as long as there is enough power–e.g., 2 - 8 axis is very common (see Figures 18 and 19).

Both single-axis and multi-axis amplifiers have a regeneration section, which is required for stopping quickly. To do so, the motor becomes a generator (the rotating magnetic field bypasses the stationary magnetic field), and the energy is carried to the bus via the lead wires and through the transistors. When the bus is raised to a sufficient level, a regenerative transistor closes to transfer the excess power to a shunt resistor. Another method is to use a transistorized or SCR-based front-end converter to regenerate the power back onto the line.

Figure 19. Chassis-mount and NEMA 1 multi-axis brush type servo amplifier.

The devices described here are called amplifiers because they typically take a control signal from a motion controller or CNC and amplify this command to higher power motion. The signal is almost always a pulse and direction input (from a stepper program), a ±10VDC signal, or a digital command. The prevalent method today is ±10VDC, although digital will probably take over the favorite spot soon.

Figure 20. Pulse train.



There are three kinds of control signals: **pulse and direction**; **±10VDC**; and **serial command and feedback**.

- **Pulse and direction** is a pulse train (see Figure 20). Each pulse commands a velocity or position, depending on how the amplifier is set up–velocity following or position following. There will be a ratio between the input pulse train and the output velocity or position. The faster the incoming pulses, the faster the motor will move to keep up. There is a second input for direction. This input is either high or low, indicating forward or reverse direction.

  The pulses are generally 5 - 15VDC. This is typical for stepper-indexer output, and today many servo amplifiers can read and run from this type of input. Input of this kind usually gives instructions to the amplifier and lets the amplifier close the velocity and/or position loop. This does not allow for real time trajectory correction. In other words, corrections can be made for speed variations (due to speed feedback to the amplifier) but not for position (because the typical indexer does not take and correct for position feedback).
- **±10VDC** is a more traditional and popular amplifier input. This can be either a speed command or a torque command, depending on the system configuration.
  - If a speed command, then +10VDC = full speed forward; +5VDC = half speed forward; 0VDC = 0 speed; -5VDC = half speed reverse; and -10VDC = full speed reverse.
  - If a torque command, then +10VDC = peak torque (or force) forward; +5VDC = half torque (or force) forward; 0VDC = 0 torque (or force); -5VDC = half torque (or force) reverse; and -10VDC = full torque (or force) Reverse.

One of the amplifier specifications will be analog input resolution, which is typically rated in bits. This is how it works. If a ±10VDC velocity command with maximum speed of 6000 rpm (rotary servo) is used with an amplifier that has 9-bit analog input, the analog signal can be divided in 2^9 parts (or 512 pieces).

10VDC / 512 pieces = 0.01953125 volt pieces

6000 rpm / 512 = 11.71875 rpm increments

Therefore, for every 0.01953125 volt increment, the motor will accelerate another 11.71875 rpm.

When trying to position accurately at low speeds, 11 or 12 rpm may not allow for good, smooth positioning. Additionally, the controller giving the signals will have an analog-out bit resolution. It does no good to go from a 9-bit amplifier to a 16-bit amplifier if the controller is 9 bit.

Serial commands and feedback is increasingly popular. For this to become a mainstream topology, more work will be needed to increase the speed of information transfer. Most servo loops close every 250 microseconds to 2 milliseconds. That means a lot of information must be transmitted very quickly. Most research and development in this area seems to be heading toward a ring topology with fiber transmitters.

THE CONTROLLER

The controller, which ties all parts of the system together, comes in four main flavors.

- **Standalone card**–Requires power supplies, mounting and a place to reside.
- **Resident card**–Resides inside the computer in ISA, VME or PCI format, using the computer's mounting and power supply (Figure 21).
- **Standalone box**–Requires a single power supply with the program resident inside of the box (Figure 22).
- **Intelligent amplifier**–Resides inside an amplifier package, using the mounting and power supply from the amplifier (Figure 23).

At the heart of all these controls is a clock and an encoder reader that determine velocity and position, and software that calculates and relays trajectory information to the amplifier via a ±10VDC signal or digital (serial) command.

Depending on the controller, programming languages range from Basic-like commands to Visual Basic "boxes" to C or C++. The program tells the machine what to do and interfaces with the operator and safety devices (e.g., machine guards and E-stops). It also tells the controller how the servo should perform (tuning the gains for servo motor/amplifier performance,

Figure 21. PCI bus-style controller.

Figure 22. Standalone combination amplifier and controller.

Figure 23. Boxed 4-axis servo controller.

the desired motion, as well as machine compliance). It frequently will take several hours to several days or even weeks to tune a servo system. The software usually costs more than the hardware.

The motion controller mathematically generates a trajectory or path–where the part should be at each specific servo loop (time period). It also receives position feedback (typically encoder or laser) that tells it the actual position. The controller then compares the trajectory with the actual position and generates a change in analog output (±10VDC) to help correct for the difference (or error).

The process of calculating trajectory, comparing it to actual position, and correcting for error is repeated at each servo loop. The amount of change generated will depend on the tuning gains that are programed into the controller.

TOPOLOGY

Feedback can come from the motor to the motion controller; from the motor to the amplifier and the amplifier to the motion controller; or from the work piece to the motion controller. The advantages of having the feedback on the motor are lower cost and a less complicated package. The advantage of having feedback on the work piece is increased accuracy in product positioning.

Consider a motor coupled to a lead screw with the load riding on linear bearings. The screw might rotate a 1/16 of a revolution before the load starts to move (a form of linear backlash and coupling slop). With a 1024 PPR or 4096 quadrature count encoder, 1/16 x 4096 = 256 encoder quadrature counts. (Quadrature counts = 4 times the pulse per revolution.) Having two offset channels makes it possible to read the leading and trailing edges of both channels, which yields four times the resolution.

If positioning from the back of the motor, the controller thinks the load has moved a distance of 256 quadrature counts, which in fact has not happened. Since this affects system accuracy, many motion controllers correct for this somewhat in the software with a feature called backlash compensation.

Additionally, ball screws are not perfect, so distances for one revolution will change slightly along the length of the screw. This can also be adjusted for with lead screw compensation, which typically is based on a table in the software that converts the actual move counts to calculated move counts produced empirically.

Figure 24. Tachometer attached to motor; encoder attached to motor or load.

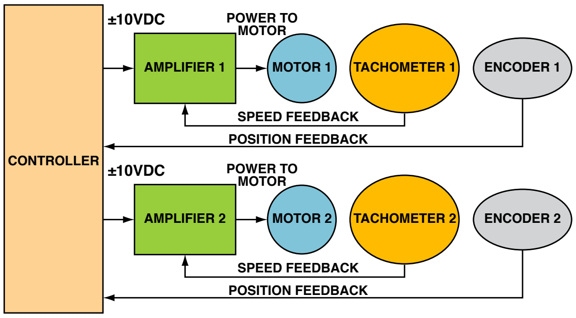


Figure 25. Encoder or resolver attached to motor; amplifier buffers feedback to controller.

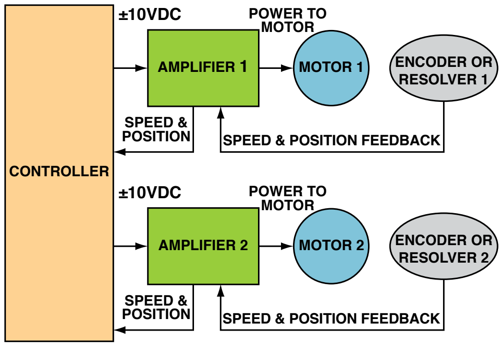
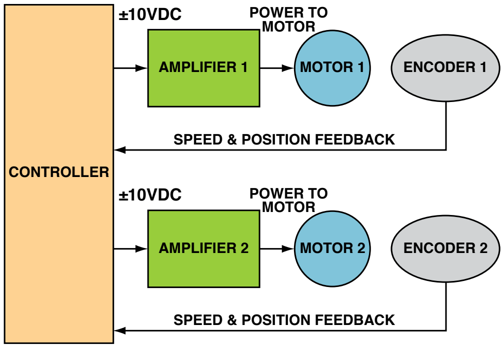


Figure 26. Torque servo with controller controlling velocity and position; encoder may be attached to motor or load.



It is much more difficult to account for thermal growth. Direct feedback makes it possible to locate and position the work piece without making calculations based on testing and tables.

For high-performance servos, the amplifier is set up in velocity control, and the controller demands velocity and closes the position loop. In lower performance servo systems, the amplifier closes the torque loop, and the controller closes the velocity and position loops.

As an example, in the topology shown in Figure 24, the tachometer is attached to the motor, and the encoder may be attached to either the motor or the load.

In Figure 25, an encoder or resolver is attached to the motor, and the amplifier buffers the encoder information back to the controller. In the case of a resolver, the amplifier converts the sine waves to a digital encoder-like signal and buffers this back to the controller.

With the torque servo in Figure 26, the controller governs velocity and position. The encoder may be attached to the motor or the load.

Loops

What are the torque, velocity and position loops, how are they “closed” in different areas, and why is this important?

Imagine a person driving a car. The driver knows the destination and needs to steer the car in the right direction to reach it. If the driver notices that the car is veering too close to the line, he corrects by steering back to the middle of the lane. This is “closing a position loop.”

The driver also knows where he is going (trajectory), sees where he is (position feedback), compares that with where he should be, and then corrects for the error. Since the driver knows the command, compares to feedback, and corrects for the error, he is closing the position loop.

As he proceeds, the driver monitors the speedometer. He knows how fast he is supposed to be traveling and compares that with the actual speed. If necessary, he changes the commanded torque to the engine to maintain the correct speed. In other words, he is closing the speed loop.

By pressing the accelerator, the driver changes the commanded torque. The car senses the change in accelerator position and delivers more fuel to the engine, causing it to run faster and produce more torque. The car is closing the torque loop.

In this example, the driver functions like a servo controller that closes the position and speed loops (steering and commanding a change in torque to maintain desired speed). The driver also resembles an amplifier that closes the torque loop–i.e., changing engine torque in response to a command from the accelerator.

If cruise control is used, the driver still controls position, but the automobile regulates velocity and torque automatically, like a velocity-controlled amplifier and a position controller. The position controller delivers speed commands to the amplifier, which automatically adjusts torque in relation to load variations in order to maintain demanded speed.

What happens if both the amplifier and the controller try to control speed? They will fight each other, and control will be terrible. Therefore, a critical step during setup is to determine if the amplifier is speed-controlled or torque-controlled. To do so, uncouple the motor from any load and put a small voltage into the command input (approx. 1 volt). If the motor runs up to full speed, the amplifier is torque-controlled; if it runs at about 1/10 of the maximum speed, the amplifier is velocity-controlled. When tuning the servo, it is important to know whether to put in a velocity feedback term or leave that up to the amplifier.

TUNING THE SERVO

Tuning a servo is more of an art using science than pure science. Typically, the amplifier is tuned to the motor for the torque loops (pretty straight forward), and then the motor/amplifier (also called a drive) is tuned to the application (the speed loops). The last step is to tune the motion controller to the customer’s desired control (the position loops).

Sometimes it works well to have a very stiff velocity control (whether done in the amplifier or motion controller) with a low position gain setting. At other times, it is better to have a sloppy velocity control and overpower it with higher position gain settings.

Many controllers now have electronic scope features for making a move, recording the resulting encoder counts over time, and graphing the position, velocity and following error. This makes it possible to change tuning parameters and view the effects again and again to obtain the desired performance.

Basic loops and their effects

Most loops (but not all) are based on error. For instance, if a motor is not exactly where it should be at a certain time, it is considered an error that must be corrected. Review of the trajectory vs. feedback is done in the summing junction. The output of the summing junction is error, which is amplified through a formula using some of the tuning parameters to determine the magnitude of the correction needed at the motor.

Torque or current gains

The following kinds of parameters are used to correct for errors.

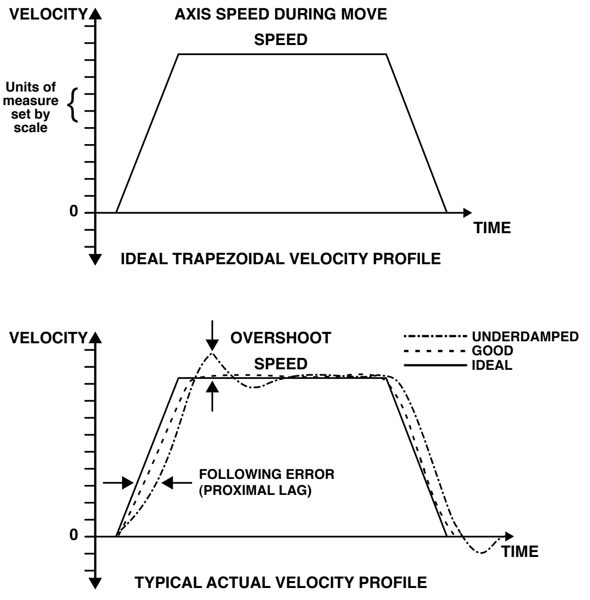
Proportional current gain–is multiplied by the error to determine an immediate step change in current to correct for the error. This gain must be used cautiously (especially on brush-style motors) to avoid flashing the commutator with an extremely abrupt change in current. Setting it too high can also cause torque ringing seen as heat, velocity ripple, and vibration. Proportional current gain is an immediate correction to error.

Integral current gain–this gain corrects for long-term errors. The current needed to correct for the error will increase or decrease relative to how long the error lasts.

Speed or Velocity Gains

Proportional speed gain–this is an immediate speed change to correct for error. If speed drops from demand due to load changes or voltage sags, this gain controls how much change will be made instantaneously to correct for that error. Immediate correction on a large inertia load will require a large change; making a large change on a small inertia load will create ringing. Figure 27 shows an ideal trapezoidal move and how under-damped, good and ideal

Figure 27. Trapezoidal moves.



systems would really look on a scope.

Speed integral gain–adjusts for long-term error. For example, consider the proportional gain that multiplies the value of the error times its gain. As the error gets smaller, so does the product of the gain times the error, which reduces the probability of achieving the target speed. This is where integral speed gain kicks in, because it looks at error over time. The longer the error lasts, the greater the correction that is applied.

Differential speed gain–looks at the slope of the error and multiplies the rate of change times the gain to correct for a radically increasing error. This gain should be treated delicately; otherwise, it will cause oscillation.

Position gains

Position gain–this multiplies the error times the gain for a velocity command to correct for the error. This gain must be used carefully. Too high of a position gain without damping (especially with a tight speed loop) can cause severe oscillations.

Velocity feedback–is a damping term that allows higher position gains for a stiff system, while decreasing the ringing.

Position integral gain–looks at long-term position error and multiplies by the gain to correct for the inadequacies of only a proportional gain. Again, this gain must be used very carefully. Extremely low numbers (if any) usually will be sufficient. Too high of a gain may create violent action.

Velocity feed forward–This gain is based not on error but on *anticipated error*. Assume, for example, that a motor should accelerate from 0 to 200 rpm. Using the gains previously described, the trajectory would be set but no analog output to the amplifier would occur until an error occurs.

Velocity feed forward (as opposed to feed back) outputs an analog signal (magnitude set by the gain) appropriate to the trajectory speed to create speed as it is supposed to occur–instead of after the speed exceeds 200 rpm and an error occurs.

Tuning procedure

The gains all wrap around each other, so it takes lots of experience to set up a well-tuned servo.

All of the parameters are set in the amplifier or the motion controller. The torque gains will be found in the amplifier, whereas the speed gains may be found in either the amplifier or the controller. The position gains will be found in the motion controller.

The typical procedure for tuning a servo is to set the torque gains first. Speed gains are then tuned to the mechanical system, and velocity feed back is increased until there is good resistance. At this point, position gain is increased very slowly until a tight loop is obtained. The need for position integral gain is then determined with a scope, and velocity feed forward gain is set to minimize following error and overshoot.

File types

Two main files typically are included in the software. The configuration file is the basic setup of the machine. It usually includes the servo gains, scale factors, max and/or positioning speeds, following error, acceleration and deceleration rates, S-curves, and so forth. These items are related to the type and size of machine and motors.

The program file is the operation of the application. This normally includes the operator interface, instructions for movement and process of handling interrupts, as well as fast position latch, error handling, homing routine, and so on. Usually, the program file can be used on different size machines and motor/amplifier combinations with a change of items in the configuration table.

Software vs. firmware

Firmware is the programming done at the factory for the basic operation of the logic. Firmware variants may include different keywords, support for different protocols (such as profibus, can open, RS232, RS485, etc.), and recognition of options like keypads or additional I/Os.

Usually, the appropriate firmware must be selected when choosing the controller. Some (but not all) firmware can be updated in the field (e.g., firmware that is stored in flash memory or on an EEPROM).

The firmware and hardware give the user the tools to do the job. Application software is then written by the user or a systems integrator based on the firmware logic.

CONCLUSIONS

There are many parts to a servo system that are individually critical yet interrelated. The motor is the prime mover; it converts electrical energy into either rotational or linear movement. The speed feedback monitors the velocity of the motor. The current feedback closes the torque loop. The position feedback tracks the location of the load. The amplifier converts the line power to a different voltage and/or frequency to control the current and velocity of the motor and load. The controller is the brain of the operation, calculating the trajectory, comparing with the feedback and correcting for error.

Successful installation and operation of servo systems requires people with good mechanical, electrical, electronic, application and programming experience and capabilities. It also is important to allow time for tuning the system, and to expect programming bugs. They happen; they always happen. These are complicated systems with a lot of interaction–PLAN FOR IT!

Place this Tech Note in Section 6
of your *EASA Technical Manual* for future reference.
Note its location in Section 15, "Future Tech Notes."

Version 0609