PROGRAMMABLE PROTECTION FEATURES

HOW ADVANCED DRIVES & CONTROLLERS ARE OFFERING MORE MACHINE PROTECTION THROUGH PROGRAMMABLE FEATURES & QUICK-STOP CAPABILITIES

WHITE PAPER

MOOG ANIMATICS

Machine and work-in-process product protection is an important consideration for any motion control system. Regulations and governing bodies that create them have worked hard to outline and organize safety measures to ensure that operations meet at least the basic requirements for safe operation and the safety of surrounding personnel. Beyond the component specifications, what about the programming of the motion system itself?

Programmable motion controllers are now offering ***new capabilities and features that allow for additional protection against machine damage and aim to lower maintenance costs*** through automatic fault detection and programmable limits.

This white paper will identify and explain some of the newest innovations in programmable protection features for advanced controllers as well as provide examples for their use and implementation.

The following information is not meant to supplement or replace knowledge or implementation of any safety standards and regulations, nor is it meant to be a guide to those basic safety standards and regulations.

Warning: Any machinery in motion can be dangerous. It is the responsibility of the machine designer and user to design and implement all safety protection, error handling and risk assessment as part of any motion control system and machine design. Moog Animatics shall not be liable or responsible for any incident or consequential damages whether arising from breach of expressed or implied warranty, arising in tort, at law or in equity, or any law giving rise to a claim of strict liability, or for any other cause, whatsoever.

Overtravel Limits

Hardware Overtravel Limits vs. Programmable Software Overtravel Limits

Hardware overtravel limits are a common method of limiting movement in motion control systems. A hardware limit is typically an external sensor or switch in a hard fixed position that limits motion and causes a motion fault if exceeded. In some advanced controllers, such as the SmartMotor by Moog Animatics, hardware overtravel limit inputs are enabled by default to protect the motor upon startup and must be cleared before proceeding. Positive and negative hardware limits may be set individually or together. Often hardware limits and limit switches are used to control the movement of a machine, as safety interlocks or to count objects passing a point.

An example of a limit switch would be a sensor that detects a lever arm moving beyond a certain point. If the operator unknowingly moves the lever arm past the limit switch (where it could potentially damage product or other parts of the machine), the limit switch will cause a motion fault within the motor and motion will be stopped. The hardware limit is a normally closed input to the motion control system, meaning that if a cable happened to come lose or there was a malfunction in the hardware limit, a motion fault would occur and motion would stop.

Software overtravel limits often exist in motion control systems as an alternative to the hardware limits and offer their own distinct advantages. Software overtravel limits are 'virtual' limit switches that can interrupt motion with a fault in the event the actual position of the motor exceeds the desired region of operation. In like manner with hardware overtravel limits, software overtravel limits are also directionally sensitive, any motion commanded further in the direction of the exceeded limit will cause a fault.

Since software limits don't rely on hardware components, utilizing software limits in machine design will lower the total machine build cost

One advantage of software overtravel limits over hardware overtravel limits is the lack of components. Since software limits don't rely on hardware components, utilizing software limits in machine design will lower the total machine build cost and will do so exponentially with each additional limit established (as compared to hardware limits).

Another distinct advantage of software overtravel limits is their flexibility. Either or both positive and negative limits can be changed at the discretion of the user or machine builder without having to rearrange any hardware. If the machine designer wanted a significantly wider range of motion, there would be no hardware changes and no additional cabling needed utilizing software overtravel limits.

Some controllers offer other software limits such as position error limits and velocity limits. Some more advanced controllers will also offer derivative error limits (also known as rate-of- change of error limits).

Non-Programmable Protection Features

Peak Overcurrent

Peak overcurrent is a hardware circuit board based, non-programmable protection feature that most industrial drives are built with. It limits the absolute maximum current to protect the drive and the motor. Peak overcurrent limits are set higher than continuous overcurrent limits as an instantaneous limit where high current cannot be maintained over long periods of time.

Continuous Overcurrent

Continuous overcurrent is the maximum sustained current allowable for the protection of the motor and the drive. Its set point typically includes a thermal based algorithm for the heating effect caused by continuous current over time. In both continuous and peak overcurrent, levels are set by the drive manufacturer to protect against accidental current overloading of the system. Almost all industrial drives have this feature.

Bus Overvoltage and Undervoltage

Bus overvoltage sets the maximum upper limit of voltage to protect against damaging the insulation and components on the circuit board. Bus undervoltage sets the minimum lower limit of voltage in the case that power drops out. As voltage decreases, current increases, so a low voltage could damage equipment with a surge of current. Typically these limits are hardware circuit board based. Some products such as the Smartmotor have the settings stored in EEPROM so that custom values may be used for battery powered applications to prevent damage to battery packs.

Thermal limits

In most drives, thermal limits are established by the component manufacturer and unchangeable. Upper thermal limits protect against physically overheating components that could be damaged such as the insulation and motor windings.

In some drives such as the integrated SmartMotor, while upper thermal limits cannot be exceeded without creating a fault, they may be programmed to a lower value at the discretion of the end user or machine designer for more accurate, application-specific protection.

In some drives such as the integrated SmartMotor, while upper thermal limits cannot be exceeded without creating a fault, they may be programmed to a lower value at the discretion of the end user or machine designer

Software Programmable Protection Features

Position Error Limits (often referred to as Following Error Limits)

Most drives and controllers have some kind of position error limit. Position error is the difference between where the actual position is compared to where it should be. Position error limits can be set to guard against any actual motion that is outside the range of programmed trajectory based on the position counter inside the drive/controller. All servos will build up torque to compensate for falling behind the expected position. This torque build-up is proportional to the amplitude of the position error.

> ***Position error limits can be set to guard against any actual motion that is outside the range of programmed trajectory based on the position counter inside the drive/controller.***

While position error limits have their advantages, they are not the optimal choice for all applications. If a routine is being performed where product could get caught in the wrong part of the machine (such as build-up at the end of a conveyor line) and a product gets pinched in-between machine components, the following error would increase until the position error limit is reached. However, before that limit is reached the motor will build up torque to compensate for falling behind in position. As demanded torque increases, supplied torque increases until the position error limit is reached. This increased torque could damage the product caught in the machine process. In the case of accidental personnel interference (if someone's hand was to get caught in the process where position error limit is established), that torque build-up could injure the person's hand even before the position error limit is reached.

To give an added level of protection, some advanced controllers (including the integrated SmartMotor) offer a derivative error limit, explained in detail at the end of this section.

Velocity Limits

Another type of software limit is the maximum allowable velocity limit. Velocity limit, like position limit, is a simple programmable feature that's common to most drives/controllers. Velocity limits are used to restrict maximum speeds of equipment in motion to help protect the machine from possible damage and are especially useful in vertical load applications.

Take the example of a long ball screw actuator. If you exceed the critical speed rating of a ball screw actuator, it will go into a whipping or lashing vibration until it finally fails. This can occur in both closed loop and open loop systems and a simple velocity limit would have avoided the failure.

In vertical load applications, velocity limits can protect both the equipment in motion and the load in most applications. Assume a velocity limit of 2000RPM has been set within the controller/drive and the vertical load starts falling. If the speed of the falling load exceeds the 2000RPM velocity limit, a motion fault will occur and a predetermined response will take place. In the case of the Moog Animatics' SmartMotor, if the velocity limit is exceeded, the motor windings will short out dynamically to bring the load to a stop by default.

Software Programmable Protection Features

Derivative Error Limit

Some advanced motion controllers have derivative error limits; the rate of change of following error in units of velocity.

Unlike the previous example of position error limit torque build-up, the derivative error limit can be implemented for an almost immediate stop where torque build-up is reduced to near zero.

Take for example an actuator moving one inch per second from left to right. If its path was blocked, it would push against the blockage, possibly causing damage because the motor would begin to supply more torque as following error increases, in the attempt to catch up to the commanded position. Torque would increase until it reached the standard following error limit.

> ***Since a derivative error limit looks at how quickly that rate of following error build up is changing over time, it catches the fault much faster than a traditional position error limit, saving time and protecting products and the machine in the process.***

One of the most common and beneficial applications of derivative error limit is in pressing applications. In the example of the small parts press, derivative error limit can be used to capture the amount of applied torque before the breakage point is exceeded. Derivative error limit can also be set to a very low value so that once contact is made, the limit will be reached and the motor will stop.

For a detailed example of derivative error limit, see the ***Application Note: Derivative Error Limit*** at the end of this article.

Fault Stop Action

Some drives have the option of specifying the automatic action that will take place when a specific condition (such as a limit) is reached. Most drives will specify in the product manual what the default action is, such as for exceeding overtravel limits. The ability to specify the fault stop action with regards to an application offers the machine designer flexibility as well as added protection for the equipment and products in process that would not be available with simple drives. Below are the three most common actions that can be specified in the case of a fault.

Dynamic Braking or Mode Torque Brake – Dynamic braking is common to the motion control industry but is not always associated with servo motors. Dynamic braking shorts out the windings of the motor bringing any motion of the motor shaft to a sudden stop. In some cases dynamic braking is the default action for following error limits, derivative error limits, overtravel limits, thermal limits and overcurrent limits. Dynamic braking uses the motor's own back EMF to bring it to a sudden stop. However after it's stopped it may slowly drift because nothing is actively holding it in place. For an application example of dynamic braking (mode torque brake), see the ***Application Note: Dynamic Braking (Mode Torque Brake), Trajectory Overshoot Braking & Freewheel*** at the end of this article.

Decelerate to Stop – In some applications, stopping abruptly (as with dynamic braking) is not the ideal situation. A centrifuge application with a quick stop may cause a sudden shift in the load or liquid to spill. In this case, decelerating to a stop would be the ideal action from the result of a fault. However, while decelerating to a stop, the machine will keep moving until it goes through its specified deceleration move and then hold the end position. Vertical load applications are another example where decelerating to a stop may be the ideal action because holding a load after deceleration may prevent further damage to the machine or products in process. However, decelerating to a stop may not be the ideal situation where a product is being pressed or squeezed.

Freewheel – This action entails letting loose of the load all together. With this action, the moment of inertia of the load will keep it moving after the fault occurred and be depended upon only friction or gravity to naturally slow the load to a stop (similar to placing a moving car into neutral instead of using the brakes to stop). Freewheeling is not recommended for vertical applications. For an application example of freewheeling, see the ***Application Note: Dynamic Braking, Trajectory Overshoot Braking & Freewheel*** at the end of this article.

The ability to specify the fault stop action with regards to an application offers the machine designer flexibility as well as added protection for the equipment and products in process that would not be available with simple drives.

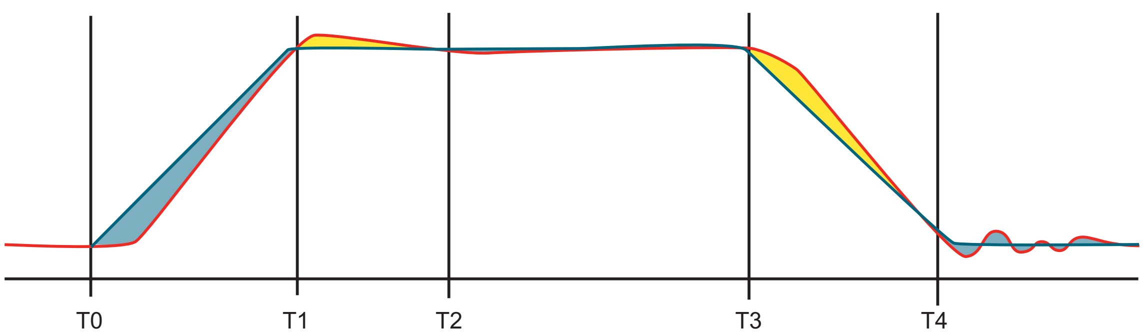
Trajectory Overshoot Braking

Trajectory overshoot braking (TOB) is a special method of commutation drive control that detects and corrects motion at the shaft only at points where the actual motion is beyond the commanded trajectory path of motion. In most motor drive systems, when the speed of the shaft exceeds the commanded trajectory, the system will drive current in the opposite direction in an attempt to correct the overshoot, or the speed will exceed the set velocity limit and the system will fault.

With TOB, when dynamic position exceeds the commanded trajectory, the drive switches from normal commutation mode to dynamic braking mode. In the SmartMotor, the update rate of TOB is equal to the PID scan rate and applies to the entire trajectory path in any closed loop mode of operation.

TOB gives the user 60% more stopping power than standard means of decelerating, thereby providing a smooth but powerful deceleration to a stop even in applications with a high moment of inertia mismatch. In comparison to velocity or following error limits, which fault and stop the system in motion when there's an excess overshoot, TOB decreases the chance that the motor reaches that fault in the first place, thereby increasing throughput and productivity. In addition, since the system is using the motor's own power to stop, versus power from the power supply, this method is also more energy efficient.

The graph below depicts a typical Trapezoidal velocity move profile:



The green line depicts the calculated trajectory while the red line depicts an exaggerated but more realistic servo move. At time T0, the command is given to start the move. From T0 to T1, the shaft begins to accelerate but lags behind the commanded trajectory. The motor speed increases to catch up with the commanded trajectory, eventually reaching a velocity higher than the commanded velocity (which could exceed the velocity limit and cause a fault).

TOB decreases the chance that the motor reaches that fault in the first place, thereby increasing throughput and productivity.

At time T1, the speed of the shaft exceeds the commanded speed. With trajectory overshoot braking, the drive cycles between normal mode and dynamic braking mode until the shaft speed is in line with the commanded speed. Directly before time T3, the motor shaft speed was matched to commanded velocity. However, at T3 there is a commanded deceleration and the shaft overshoots velocity once again. With TOB, through the same dynamic braking method, the result is a very fast correction to overshoot, allowing for exceptionally smooth and quiet deceleration to zero.

For an application example of trajectory overshoot braking, see the ***Application Note: Dynamic Braking (Mode Torque Brake), Trajectory Overshoot Braking & Freewheel*** at the end of this article.

Application Note: Derivative Error Limit

Industry: Sheet Metal Fabrication
Application: *Electric Resistance Welding*
Challenge: Improve quality of welding process by ensuring equal pressure of the resistive welding clamp across two metal pieces being bonded together.

Situation

Resistive welding clamps are used to spot weld two pieces of sheet metal together (such as on a car body). The quality of the weld must be held to strict tolerances to avoid fatigue failure once in service.

Problem

As the resistive welding clamps tighten down on the two pieces of sheet metal, pressure should be equally applied to both surfaces. Equalizing actuators try to center the clamping force between the two pieces of metal. As soon as one half of the clamp meets the metal, that half (the equalizing half) stops moving until the other half of the clamp can come in contact with the opposite side. Optimally, the clamp is trying not to bend the metal at all in either direction. If either side of the clamp applies more force than the other, the sheet metal will be stressed and the weld could later fail or the sheet metal could crack loose around the spot weld.

Solution

There are two possible solutions in this application, the first involving position limit. If the following error is set to 500 encoder counts and it's moving at 500 revolutions per second, it will take one second before the position limit is reached. Unfortunately this could be enough time to dent the metal sheet.

The second, preferred solution, would be to use derivative error limit (such as with the SmartMotor). Instead of looking at how far behind the encoder count has fallen from the expected position, the controller looks at how fast the following error is increasing. So instead of building 500 encoder counts of error and risk pressing into the sheet metal with too much force, derivative error limit will reach the fault and stop motion within one to two servo samples (a matter of microseconds). In this reduced time, the clamps will not dent or deform the metal. This improves quality of the weld and ultimately increases productivity by increasing throughput and lessening the chance of a fault which would cause a stop in production.

Application Note: Dynamic Braking, Trajectory Overshoot Braking & Freewheel

Industry:	Architecture
Application:	Automated Sliding Glass Doors
Challenge:	Protection of store patrons & employees as well as large expensive glass doors
	Smooth, safe and quick-stop motion capabilities
	High torque but quiet operation for retail operation

Situation

An architect had specified very large glass doors for the retail entrance of a well-known consumer product technology leader headquartered in Silicon Valley. Each door was 10 ft. by 8 ft. and weighed roughly 2000 pounds. The doors were programmed to be controlled by a manual switch or sensors for daily operation when the store was open for business.

Problem

Such large doors required a significant amount of torque to open and close. However to protect the doors themselves (as well as any store patrons and employees), the motion control systems operating the doors needed to respond properly and quickly when operated manually or in emergency exit situations. For aesthetic reasons, the company didn't want an external controller to operate the doors despite the complex requirements.

Solution

The architect and door manufacturer chose Moog Animatics' SmartMotor because of its programmable protection features and integrated nature (no external controller needed). The customers wanted the doors to be able to detect if something or someone was obstructing the path of motion, which was accomplished using following error and derivative error limit to stop in the quickest manner possible. In the case that the derivative error limit was reached because of an obstruction, the SmartMotor on each door was programmed to use dynamic braking (also known in the SmartMotor as Mode Torque Brake). Dynamic braking ensured that no more active force would be applied to the obstruction once it was detected.

However, dynamic braking normally applies a counter force once motion has stopped. In an emergency situation the doors needed to have the option of being manually pushed open. The SmartMotor was therefore programmed to turn off dynamic braking after coming to a stop and then to freewheel.

In the situation whereby the door was in motion but needed to stop and move in the opposite direction (such as if commanded by the wall switch), the doors used trajectory overshoot braking to quickly and efficiently decelerate to a stop and accelerate in the opposite direction without exceeding any following error or velocity limits.

The combination of these three programmable protection features allowed a SmartMotor with no external controller and minimal cabling to operate effectively and safely while maintaining the sleek aesthetic brand identity that the technology company required.

To learn more about how Moog Animatics can offer programmable protection in your application, please call 408.748.8721 or email us at sales@animatics.com***.***

Moog Animatics www.animatics.com
3200 Patrick Henry Drive Santa Clara, CA 95054
Tel: (408) 748 8721 Fax: (408) 748 8725 E-mail:sales@animatics.com