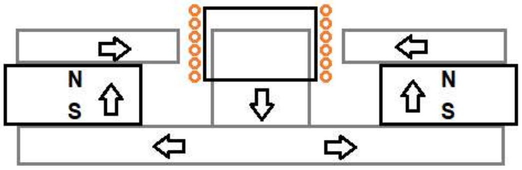
Stepping and Servo Motor Commutation

By Kevin McCarthy, Chief Technology Officer

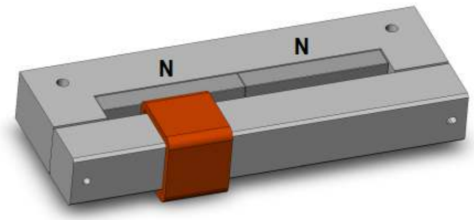
No Commutation

Commutation simply refers to the switching between phases of a multi-phase motor. There is one class of permanent magnet motors that requires no commutation whatsoever, and serves as a simple and effective solution, albeit to a small subset of applications. These are single phase motors, several physical linear motor variations of which are shown in Fig 1. The single phase, permanent magnet servomotor is a very simple affair, which can provide either torque (for angular rotation), or force (for linear motion). Single phase servomotors are two-lead devices, consisting of a single electrical coil placed within the DC magnetic field produced by a permanent magnet. “Back iron” is typically added to help channel the magnetic flux. Torque or force is directly proportional to current, and a single “H-bridge” (four FETs, with the coil in the horizontal leg of the “H”; Fig. 2) is all that is required to drive the motor. No commutation is required, and servo movement can be commanded as soon as the drive is powered up. Single phase motors do have one drawback, which is that practical implementations are restricted to fairly short angular and linear travels. When you try to scale single phase motors to longer travels, their back iron gets ever larger and more massive to avoid magnetic saturation, and depending on the design, the inactive coil leg length also grows. Beyond a reasonable cut-off in travel, it is worth taking on the additional complexity of commutation, and switch to a multi-phase motor.

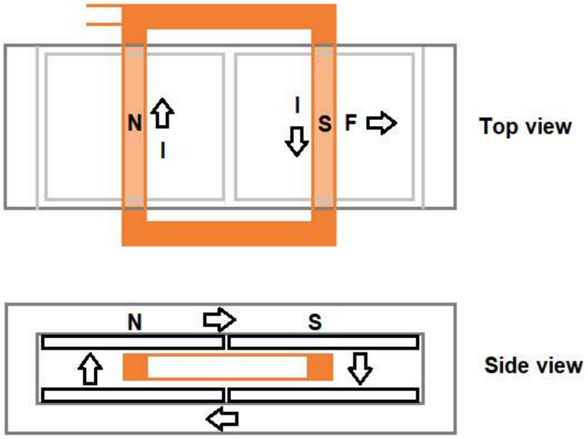
At Dover Motion, we have drawn the line at 25 mm of travel; our MMX-25 is single phase, while all longer travel models employ three phase motors. For single phase motors used for angular motion, a good maximum cutoff is about twenty degrees.



1a: Voice coil motor, side view

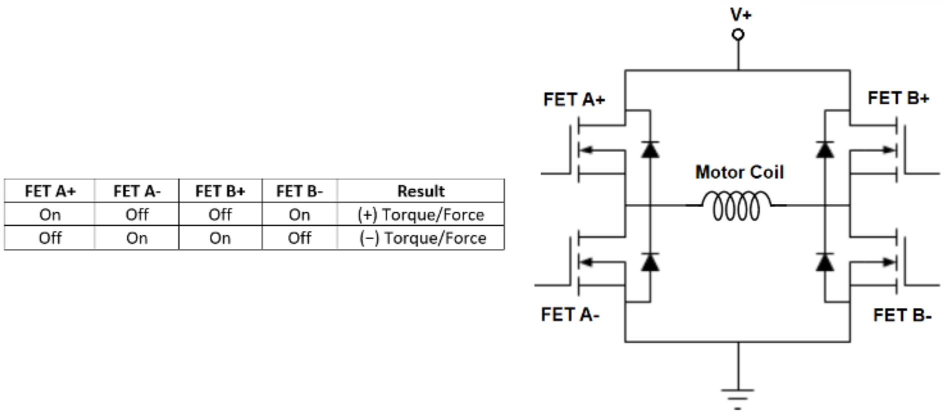


1b: One active leg hybrid single phase motor



1c: Planar single phase motor

Fig. 1 a-c: Examples of single phase linear motors



FET A+	FET A-	FET B+	FET B-	Result
On	Off	Off	On	(+) Torque/Force
Off	On	On	Off	(–) Torque/Force

Fig. 2: Single Phase (non-commutated) Motor Truth Table and Output stage

Mechanical (Brush) Commutation

There is another class of servomotors that resembles single phase motors in that they present as two-lead devices, requiring only a single H-bridge for drive, and no electronic commutation. These are brush motors, and while they are capable of continuous rotation, they are a multi-phase motor. In brush motors, commutation (switching among the motor's phases) is performed mechanically, via a pair of spring-loaded brushes in the motor stator, and a set of corresponding electrical contacts for each coil phase in the motor rotor. As one coil phase is leaving the region of strong magnetic flux, the brushes energize the next coil phase that is entering that flux region, with the permanent magnet field lagging the coil field by 90 electrical degrees. Rotary brush motors are both cost effective, and easy to drive and control.

It is also possible to make a linear equivalent to the rotary brush motor. These have a long series of coil phases in the linear motor stator track, with brushes in the permanent magnet moving carriage that only energize the coils directly beneath the moving carriage as it passes over them. This has an advantage in that no moving cables are required, but this is countered by the complexity, particulation, and radiated EMI of the long linear moving brush assemblies, and the cost of the long coil assembly. Moreover, brush commutation is inherently digital, not sinusoidal, so constant velocity performance isn't great, and audible clicking can be heard as coils are abruptly energized and deenergized. There are a few suppliers producing brush type linear motors, but for the most part, they have been superseded by brushless linear servomotors.

Two Phase Motors

For applications with travel beyond about 25 mm, or rotation more than ~ 20 degrees, and no brushes, the most likely choice is a brushless, multi-phase motor. Induction motors are another option, but that is a complex subject, not addressed here. A key question then is: How many phases are needed? Well, arbitrarily long linear travel and continuous rotation can be achieved with as few as two phases in a servomotor, but the vast majority of brushless servomotors use three phases. Why?

The reasons for this are more historical than technical. All early brushless servomotor drives employed digital commutation (covered in the following section), and in a two phase rotary motor, this resulted in significant torque ripple. By increasing the phase count from two to three, this torque ripple was reduced, and the increased drive complexity was to some degree offset by requiring only three half bridges (six FETs) instead of the eight FETs in a dual H-bridge, two phase motor drive. So both the motor and drive industry settled in on three phases as the standard. In the 80's and 90's, companies such as Pittman produced two phase brushless servomotors, but lacking widespread drive availability, they were eventually discontinued. Years later, sinusoidal commutation (see below) was introduced, and this eliminated the torque ripple problem of two phase motors, but by then, the market had moved on, and the three phase servomotor standard had become nearly completely dominant. But when designing a custom linear motor, two phases are entirely adequate, and most modern controls can operate them with both sinusoidal commutation and field oriented control.

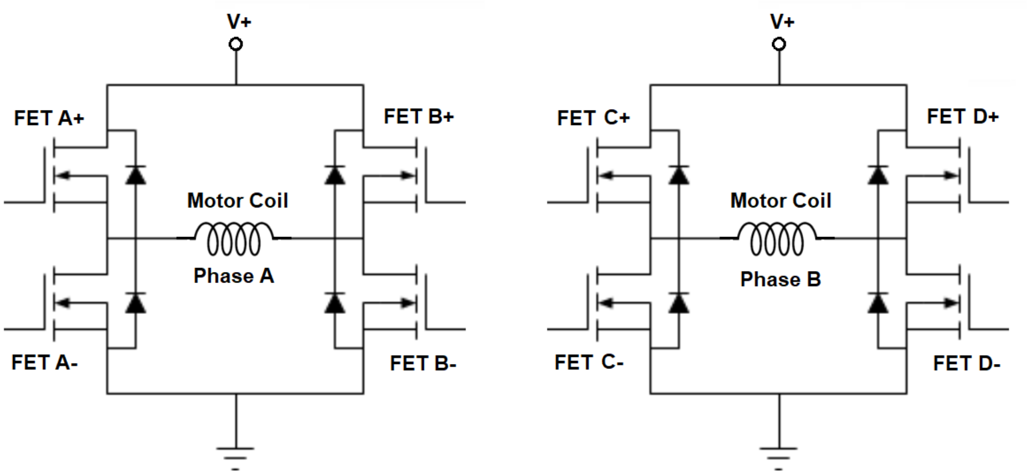
Viewed from another vantage point, though, two phase brushless motors vastly outnumber their three phase counterparts. The difference is that these aren't strictly "servomotors", because they are typically operated in open loop mode; we call them instead "stepping motors".

Stepping Motor Commutation

A stepping motor is essentially a high pole count (typically 50 poles, every 7.2 degrees for a 200 step per revolution motor) rotary brushless motor that is commutated as a function of time (one could also say hope), instead of by encoder-based position feedback. Just as brushless servomotors can be digitally or sinusoidally commutated, stepping motors face the same choice, only their terminology is different: digital commutation is referred to as "full step" or "half step" operation, while sinusoidal commutation is called "microstepping". Stepping motors require two full H bridges; an output stage diagram and full step truth table are shown below in Fig 3.

Since stepping motors have a high pole count, their inductance results in a significant drop-off of torque above a certain speed. As a result, their acceleration and deceleration should be reduced as speed increases to avoid having the motor fall out of its time-based synchronization and "stall". The maximum torque is extracted when the motion profile and load result when the coil field lags the magnet field by 90 electrical degrees (1.8 mechanical degrees in a 200 step per revolution motor).

Stepping motors can be turned into servomotors by adding an encoder, and a real-time closed loop control system, but as mentioned above, their high pole count (typically 50) limits their high speed torque. In contrast, three phase brushless servomotors typically have four or at most eight poles per revolution, and are accordingly much better suited to high speed operation.



	Step #	FET A+	FET A-	FET B+	FET B-	FET C+	FET C-	FET D+	FET D-
1.8°	1	Off	On	On	Off	Off	On	On	Off
1.8°	2	On	Off	Off	On	Off	On	On	Off
1.8°	3	On	Off	Off	On	On	Off	Off	On
1.8°	4	Off	On	On	Off	On	Off	Off	On

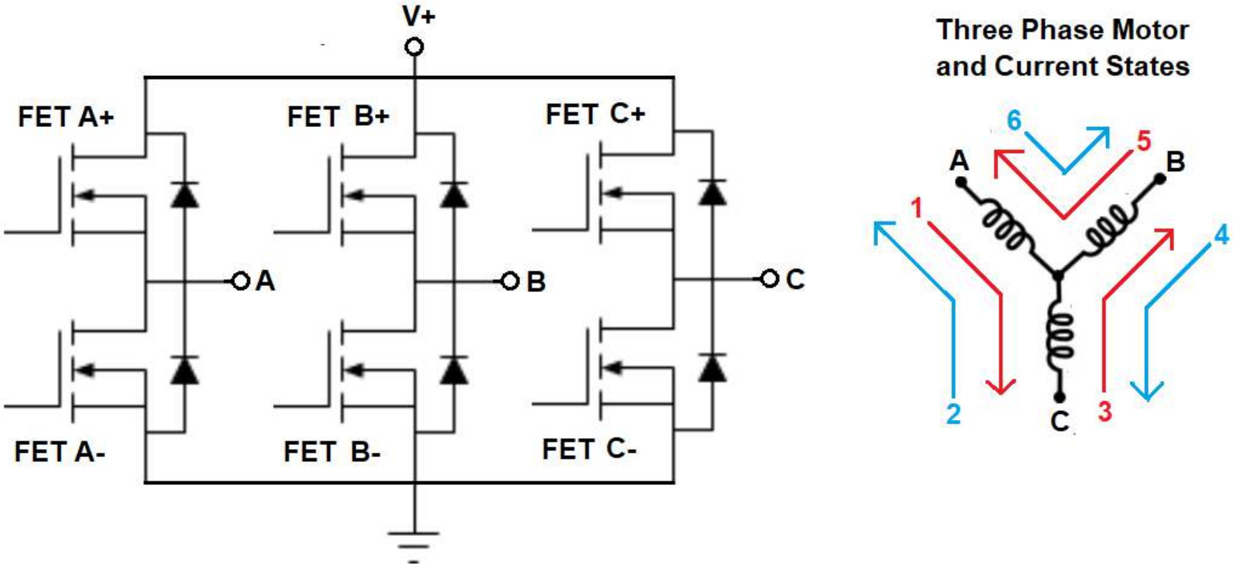
⇧ One pole (7.2°) ⇩

Reverse sequence or any one pair of coil leads to reverse direction

Fig. 3: Stepping motor output stage and full step commutation truth table

Digital Commutation

The remainder of this white paper will focus on brushless three phase servomotors. When these were first introduced into the market, more sophisticated sinusoidal commutation wasn't yet a thing, so the motor phases were switched in and out all at once, in a digital fashion. The position-based signals that determined where to perform this digital commutation were typically a triad of digital magnetic Hall effect sensors, and a "truth table" depicting the Hall effect sensor state and motor coil switching with FET states and current directions is shown below (Fig. 4). One benefit of the triad of Hall sensors is that they form a simple, three-bit absolute encoder for commutation, which means that no initial phase initialization step is required. The drive can begin moving as soon as it is powered up.



State	Hall A	Hall B	Hall C	FET A+	FET A-	FET B+	FET B-	FET C+	FET C-
#1	1	0	1	On	Off	Off	Off	Off	On
#2	0	0	1	Off	On	Off	Off	On	Off
#3	0	1	1	Off	Off	Off	On	On	Off
#4	0	1	0	Off	Off	On	Off	Off	On
#5	1	1	0	Off	On	On	Off	Off	Off
#6	1	0	0	On	Off	Off	On	Off	Off

Invert FET states to reverse direction

Fig. 4: Three Phase Motor Digital Commutation

Digital commutation remains a very viable technique for rotary brushless motors. The same principal was initially used for linear brushless servomotors, but since they lack the mechanical advantage that a ballscrew or leadscrew provides to a rotary motor, their coarse linear pole pitch produces noticeable velocity ripple, as well as audible clicking as coil phases are abruptly energized and deenergized. In a rotary motor, the steel housing serves as a Faraday cage to prevent EMI from the discrete switching from corrupting sensitive encoder signals. Linear servomotors are much more difficult to shield in this manner, which is another issue. As a result, linear brushless servomotors almost invariably employ sinusoidal commutation (below).

Sinusoidal commutation

Sinusoidal is to digital commutation as microstepping is to full-step operation; instead of abrupt digital switching of the motor coils, each phase is gradually ramped in and out of operation in a sinusoidal manner. Since there are three phases, these are each separated by 120 degrees around a full 360 degree electrical cycle.

The corresponding plot of phase current versus position is shown below (Fig 5). For stepping motors, the open loop microstepping current waveforms look similar, although there are only two coil phases, separated by 90 electrical degrees. High resolution information on rotor or carriage location is typically provided by an optical encoder, although some rotary motors instead use a resolver (a different type of position feedback device). Sinusoidal commutation provides exceptional constant velocity performance, and is entirely quiet in operation. In applications which are required to produce a force, the force can be extremely constant over travel. Some linear servomotors retain a triad of magnetic Hall effect sensors, which eliminates any need for a phase initialization step upon power-up, but such a sensor array increases the overall dimensions of the stage, and the additional signal lines and sensors add complexity and cost. In the absence of a three channel Hall effect or optical commutation sensor array, or an absolute position encoder, the stage will have to perform a phase initialization step upon power-up (see that section below). In recent years, simple sinusoidal commutation as a function of just the encoder position has given way to the more sophisticated and performant Field Oriented Control (next section).

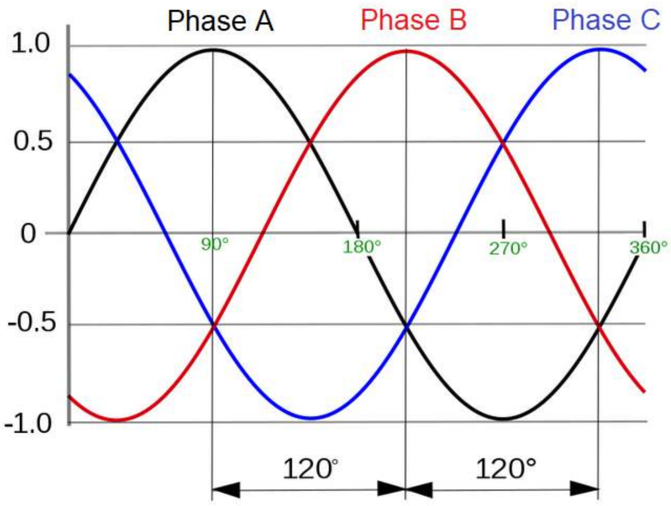


Fig. 5: Sinusoidal three phase currents vs electrical angle

Field Oriented Control (FOC)

A stepping motor at rest with a DC current applied to its windings develops no torque; if it did, it would begin accelerated rotation. The power dissipated in its windings serves only to create radially directed forces. It is only when its rotor angle is rotated by 90 electrical degrees (1.8 mechanical degrees in a 200 step per revolution motor) that its full “holding torque” is experienced. The same applies to a three phase brushless servomotor. The most efficient use of motor winding currents to create torque (or force, in a linear motor) is when the coil field lags the magnet field by 90 degrees. Ensuring that this condition is met during a variety of motion profiles and load conditions involves a fairly substantial bit of mathematics.

In Field Oriented Control, or “FOC”, the motor current is resolved into two orthogonal components. One, called the quadrature current I_q, is the torque or force that we are after. The other, called the direct current I_d, serves only to create useless radial force. FOC continuously adjusts the phase of the coil PWM voltages so as to drive the direct current I_d to zero, thereby maximizing the desired quadrature current I_q and the system efficiency. FOC’s algorithm to accomplish this involves a pair of mathematical transforms (Clarke and Park) that use the encoder based position and measured current to measure I_q and I_d in their moving reference frame, and use a PI loop to drive I_d to zero. Inverse Clarke and Park transforms are then used to determine the required winding currents. The computational power needed to implement the Clark and Park transforms and their inverses in real time is substantial, and simply wasn’t available when FOC was invented in the late 1960s. But today, we have the luxury of ample computing power, even in stage-embedded servo drives such as the Dover Motion SmartStage™ products, and FOC is our default and strongly recommended mode of operation.

Incremental vs. Absolute Encoders

The most common and cost-effective encoder feedback devices are incremental in nature, which is to say that they do not provide information on the stage location upon power-up. They do faithfully keep track of position from the moment of power-up, by sending two channels of position data to a counter in the motion controller. This data takes the form of two signals in 90-degree quadrature, typically called Channel A and Channel B. The phase relationship between the two channels allows the direction of travel to be determined. While some incremental encoders are digital in nature, higher performance encoders put out analog sine and cosine signals, which are then “interpolated” to achieve very high resolution. In the case of the Dover Motion “Ultra” encoder, the resolution of a single count can be programmed to be as small as 305 picometers, which is the diameter of a single aluminum atom!

One potential issue to be aware of with incremental encoders is that if their maximum velocity is exceeded, they will lose counts, and both a homing routine (to reestablish the correct position) and a new commutation phase finding routine (to restore the proper commutation phasor) will be required. Exceeding the encoder’s maximum velocity can occur due to either programming a move with too high a velocity, or by moving the stage quickly by hand while the servo is disabled.

Absolute encoders are fundamentally different, in that they know their absolute position immediately upon power-up. As a result, the relative orientation of the three phase motor coil and the permanent magnet array need only be recorded once, together with the distance between successive magnetic poles (a North-South pair). Once the controller has this information, the controller is immediately ready to move the stage correctly upon power-up, without the need for any subsequent phase initialization step. While that is useful, absolute encoders also have a few drawbacks. They transmit their position information via a digital serial word, whose protocol varies among manufacturers, and can be proprietary, locking you into buying the servo drive from that encoder vendor. Absolute encoders are considerably more expensive, typically have lower resolution, and are usually physically larger. Their lack of the real-time stream of position data that incremental encoders provide means that many applications that depend on that real-time data (such as TDI imaging) cannot be supported. For these reasons, absolute encoders remain a minority in the marketplace.

Phase initialization

When a stage with an incremental encoder and no Hall commutation sensors wakes up, it has no idea where it is in travel. As a result, servo movement cannot yet be commanded, as the phase relationship between the three phase motor coil and its permanent magnet array has not yet been established. The controller will have previously been told the distance between magnetic poles (25.4 mm for most Dover Motion linear motor stages), but the missing bit of information is that positional offset between the coil and the permanent magnets. This relationship repeats every magnetic pole (North and South pair), so what is important is merely the phase relationship between the two components within one pole. This is referred to as the phase angle, and can be expressed in units of either encoder counts (possibly with a prescalar), or degrees within the electrical cycle. In the above-mentioned case, 25.4 mm would correspond to 360 degrees of electrical phase. So upon power-up, discovery of this phase relationship is essential, and there are two different ways to accomplish this. One, which successively energizes two motor phases and measures the resulting stage movement to determine the phase angle, is referred to in our SmartStage™ and DMCM controller documentation as “Algorithmic Phase Initialization”, henceforth: “API”. The other is referred to as “Pulse Phase Initialization”, or “PPI”.

The Half Travel Vane

Before we begin discussion of each of the two phase initialization methods, it’s worth pointing out a very useful optical signal that we have long employed, and which has been designed into our current line of SmartStage™ products. This is referred to as a “Half Travel Vane”, henceforth: “HTV”. This term is a bit archaic, as it referred to a time years ago when a physical vane passed through and blocked light in a slotted opto-interrupter for half of the travel; the signal is purely optical now, so there’s no vane, but the nomenclature has persisted. The HTV can be thought of as a one-bit absolute optical encoder, which is high for one half of travel and low for the other. As a result, upon power-up, examination of the HTV provides an immediate confirmation of which half of travel the stage is in; more to the point, it also shows which way the center of travel lies. Put another way, it shows you the direction ***away*** from the limit switch and hard stop. The HTV also serves as a physical reference for homing routines, and for stages with high friction, it provides a specific known location where an optimal commutation phasor can be jam loaded. Since the polarity of HTV signal can be inverted in the SmartStage firmware, with that setting saved in NVRAM, it is up to the user to choose the state (logic high or logic low) for either half of travel.

Finding the position of the transition of the HTV from high to low (or vice versa) takes a bit of explanation. In SmartStage products and our DMCM (Dover Motion Control Module), a breakpoint can be set to take an action on a transition of the HTV (the signal is called “Axis In”). Such a breakpoint can be chosen as either “abrupt” (immediate, at that servo cycle), or “slow” (decelerating to a stop once the transition occurs). The optical sensor itself has a bit of hysteresis (the difference in position when approaching the transition from two different directions); this typically ranges from 25 to 75 µm. But this hysteresis can appear larger, depending on how you set up the breakpoint. The stopping distance using the “slow” breakpoint will simply equal the velocity squared divided by twice the deceleration. At a decel of 1 m/s², and an approach velocity of 20 mm/s, a “slow” breakpoint will coast to a stop 200 µm to either side, for a total hysteresis (including that intrinsic to the sensor) of ~ 450 µm. An alternative would be to choose an “abrupt” breakpoint; this has no hysteresis, but essentially infinite deceleration, and so runs the risk of servo instability, a following error trip, or an overcurrent fault for anything other than very slow velocities. Our recommendation is to choose between one of the three following options when finding the position of the HTV:

- Use an abrupt breakpoint, but at a slow velocity (<= 5 mm/s). This takes a little longer, but is very accurate.
- Use a slow breakpoint, but at modest velocities (10 - 30 mm/s), and a reasonably high deceleration (~ 3 - 5 m/s²).
- Perform a two-step HTV transition finding routine: Approach it at high speed using a slow breakpoint, then pull back a few mm, and approach it again from the same direction using slow speed and an abrupt breakpoint.

In all cases, choosing a single direction of approach (e.g., always approach the HTV in the positive direction) will always result in a more accurate HTV position.

Algorithmic Phase Initialization

In Algorithmic Phase Initialization, or API, a voltage is ramped for a programmable time to a programmable level across a specific pair of coils in the three phase motor, corresponding to 90 electrical degrees. Since the phase relationship between the motor coil and the permanent magnet array is unknown when the motor is poled in this fashion, the resulting motion can be as much as half a pole (12.7 mm in Dover Motion SmartStages), and as little as zero. The half pole motion would correspond to the initial phase having been at 270 degrees, with zero motion occurring had the motor already been at 90 degrees. With the motor and magnets now at 90 degrees of phase, a second voltage is ramped across coil legs, this time at 180 degrees. Since the controller already knows how many encoder counts correspond to a full electrical cycle, motion of one-quarter this value is expected during this second move. If the resulting motion is within 22.5 degrees of the expected value, no error is thrown, and the commutation phasor is then set at the value of 270 degrees. Why 270 degrees? Well, as described above under the Field Oriented Control section, the most efficient use of motor winding currents to create torque (or force, in a linear motor) occurs when the coil field lags the magnet field by 90 degrees: 180° + 90° = 270°. The motor can now be enabled, and closed loop servo motion begun. If, for a variety of reasons (see “Troublesome Mechanical Realities”, below), the motion during this second move comes up short, or has the wrong sign, the Commutation Error bit is set, and the resulting 270 degree phasor may not permit motion, or may do so inefficiently. The most likely cause of such a commutation error is that the stage was up against one of the hard stops at the start of commutation phase initialization.

For high resolution stages, choosing too high a value for the applied voltage during the open loop moves, or too short a ramp time, may result in fast stage motion that exceeds the encoder maximum velocity. The resulting encoder count loss will produce an incorrect phase initialization. Conversely, if the applied voltage is too low, or too short a ramp time is chosen, insufficient motion will occur; this is most likely when the stage has significant friction. Both the applied voltage and the ramp time should be chosen to best suit the stage type being commutated.

Depending on the position of either hard stop, the second phase initialization motion from 90 to 180 degrees may move away from the hard stop, and so allow for a good commutation phase finding. This isn’t something you would want to count upon. But not to worry; we have a very neat solution for that, as detailed in the section “Robust Phase Initialization Routine“.

Pulse Phase Initialization

In Pulse Phase Initialization, or PPI, a series of small, ramped values are applied to different coil phases, and the servo controller uses the resulting small encoder movements to calculate the relationship between the permanent magnet and coil fields. The benefit is that only very small motions are required; a potential downside is that the accuracy can be affected by issues such as mechanical friction. As a result, PPI is best used on stages with smooth bearing guideways, such as crossed rollers or air bearings. More advanced versions of PPI perform synchronous demodulation of a sinusoidal motor excitation, with the amplitude and phase of discrete Fourier components of the resulting motion used to achieve a high-performance phase initialization with minimal motion.

Jam Loading a Phase angle, aka “Direct Set”

Dover Motion SmartStage™ products and our DMCM controller all support the ability to directly set an initial phase angle into the motion controller. This can be a very helpful commutation tool, especially when used in conjunction with our one-bit absolute linear encoder, aka the Half Travel Vane. The process is as follows:

- As a one-time event (which can be performed during final test by Dover Motion), strong algorithmic commutation is performed, and a slow move with abrupt stop breakpoint to the HTV is initiated.
- Once servoed at the HTV transition, the phase angle is read and stored for future use.
- Later, at run time, with the potential for highly loaded cable and tubing chains and other customer paraphernalia, an algorithmic phase commutation initialization is performed at power-up. It may not be very accurate, but as per the next section, it doesn’t have to be.
- The stage then “limps” to the HTV using this initial commutation phasor.
- Once there, the previously determined value is jam loaded (aka, “direct set”), and voila, essentially perfect commutation!

- The only downside to this approach is in the event that the stage is up against a hard stop, such that upon algorithmic phase initialization, it can't even limp to the HTV. For that situation, we have another approach, detailed in the last section of this white paper.

Precision Commutation Phasor Determination

In most cases, the Algorithmic commutation described above will accurately determine a suitable commutation phasor. In the interests of completeness, a higher precision method is also available. In this case, the stage is initially commutated using the Algorithmic method. The servo loop is closed, and an external force is applied to the moving carriage. This force can be generated by a spring, or a pair of repelling magnets, but another approach is to simply place the stage on an incline. In this case, the carriage will experience an axial force in Newtons equal to m*g, where m is the total moving mass in kg, and g is the acceleration of gravity: 9.8 m/s^2. The magnitude of the applied force should be in the range of 20-30% of the motor's continuous force rating. The motor coil current required to oppose this applied force is then measured and plotted as the phasor angle is varied (via Direct Set, as discussed in the above section) over a range of +/- 20 degrees, as shown in Figure 6. The resulting set of data points (blue) is then fit with a parabola (red). The minimum of the best fit parabola is the optimum phasor angle.

Since precision stages typically employ tight tolerances on the dimensional offsets between the linear motor magnets, the coil, and reference signals such as limit switches or a Half Travel Vane, such a precision commutation phase finding will often need to be done only once for any given model (not distinct serial number) of stage. Once the precise commutation phase angle is found using the above technique, the stage can then be moved to a limit switch or Half Travel Vane position, and the phase angle at that reference position can be recorded as a "gold standard" for a future Direct Set operation.

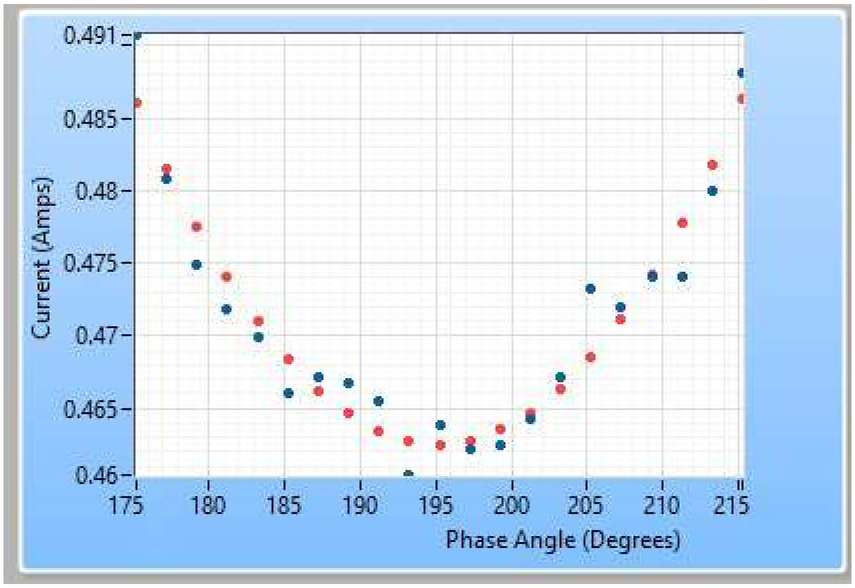


Fig. 6: Motor coil current vs phase angle

Commutation phasor precision: How good is "good enough"?

The math here is extremely simple; as the initial commutation phase value departs from perfect, motor efficiency suffers, but not abruptly. The relation is simply the cosine of the commutation phasor's angular error.

If the commutation phasor error is zero, then the full efficiency of the motor is realized. If the phase error is 45 degrees, then the resulting efficiency is cos (45 deg), or 71%. Not great, but not too shabby either. On the other hand, if a stage was already being pushed to near its thermal limits, that 29% inefficiency at a 45 degree commutation phasor error could easily overload it. Users should always set the parameters of their i2t motor protection to avoid any potential for thermal damage. If the commutation phase error is +/- 22.5 degrees, the resulting efficiency will be 92%. Unless the application has significant challenges (see the next section), it isn't difficult to keep the commutation phasor error to no more than +/- 10 degrees; at this level, the efficiency is over 98%. Moreover, as detailed below, there are ways to get a great commutation phasor, even when the stage is pushed up against the hard stop.

Commutation Phasor Checking

To provide full confidence that the commutation phasor is close to its optimal value, users can code a simple check after every power-on and its new commutation phase find: move to the half travel vane position, and compare the commutation phasor value there with the previously established value. If the match is close, all is good. If for any reason the commutation phasor measured at the half travel vane departs from its expected value by more than a reasonable amount (say, more than +/- 10 degrees), then either repeat the commutation phase find, Direct Set a previously determined value, or use the robust algorithm described at the end of this white paper.

Troublesome Mechanical Realities

While algorithmic phase initialization is almost always very effective, there are a few troublesome mechanical realities that can degrade its accuracy. These include:

- High friction guideways: Stiffly preloaded recirculating bearings can fall into this camp.
- Guideway misalignment: If the two guideways have an angular misalignment, this can increase force at points along travel, making commutation phase discovery less precise.
- Mounting to non-flat surfaces: We normally test our stages while they are bolted to either a granite surface plate, or a precision ground hard-coated aluminum plate. If the stage is subsequently bolted to an instrument surface that isn't flat, the force required to move can vary over travel, and corrupt commutation phase finding.

- Waking up against a hard stop: It turns out that with our hard-coded Algorithmic Phase Initialization routine, being up against the hard stop isn't always an issue, depending on the sign of our initial movement, which end of travel the stage is at, and the location of the hard stop with respect to the motor/coil phases. But in other cases, being up against a hard stop can degrade accuracy or cause the API algorithm to fail.
- External forces applied to a stage: The most common culprit here is one or more energy chains that are chock full of disorganized tubes and cables. These present an external force that varies over travel, and can push stages into a hard stop. The best solution is a well designed cable plant, with energy chain stuffing below the recommended maximum fill factor, and linear arrays of cables and tubes, instead of disorganized ones.
- Vertical orientation: Residual force bias from an incomplete gravitational counterbalance can challenge initial phase finding, depending on the magnitude of the imbalance. Such forces ideally bias the stage to a point in mid-travel, but in troublesome cases, the stage can find itself pegged at the bottom or top of travel.
- Encoder velocity limit exceeded: If this occurs, due to either a programmed move that exceeds the encoder's maximum velocity, or by moving the stage quickly by hand while the servo loop is disabled, then both the position and the commutation phasor will be incorrect, and a homing and new commutation phase find will need to be performed.

Robust Phase Initialization Routine

For stage systems that exhibit some of the qualities in the rogue's gallery above, we have a definitive answer that ensures perfect commutation every time. Full details are in separately available scripts and flowcharts, but a simplified description follows. Key to the approach is that just as a stepping motor can be open-loop commutated with respect to time, the same technique can be applied to three phase brushless servomotors. The servomotor is for all intents being operated as if it was a microstepping step motor.

- Upon power-up, the Half Travel Vane is read, revealing the direction to move to get away from any hard stop, and approach the HTV.
- Open loop motion in that direction is then initiated, with an abrupt stop and clear position error breakpoint set for the HTV.
- The stage begins a contoured velocity move towards the HTV. Initial motion can be irregular, but within less than one commutation cycle, synchronous motion begins. Note that this is open loop motion, and due to the coarse pole pitch, the moving carriage will not be stiffly following its command as it does during closed loop motion.
- When the HTV is encountered, motion stops, and the position error is cleared. Since open-loop motion is fairly mushy, external forces may result in the current position being a small distance off the HTV.

- After disabling the motor and transitioning to sinusoidal commutation, the position error is read, which reveals exactly how far from the HTV the stage is. This distance is scaled against the number of prescaled encoder counts per commutation cycle, and the previously determined phase angle of the HTV transition.
- A calculated phase value corresponding to the current location relative to the HTV is then jam loaded into the motion controller.
- Closed loop operation is now commanded, with very accurate commutation. To double check, the stage can now perform an abrupt stop breakpoint move to the HTV, and compare the phase value read at that point to the previously measured and recorded “gold standard” value. If the currently observed HTV phase angle is for any reason off, simply Direct Set that gold standard value.

That’s our robust phase initialization routine in a nutshell. It works exceptionally well!

And with that, patient reader, our stepping and servomotor commutation story draws to a close.

ABOUT THE AUTHOR...
Kevin McCarthy is the Chief Technology Officer of Dover Motion, and recently celebrated his 40th year at the company. He holds a B.S. degree in Physics from The Massachusetts Institute of Technology. Kevin can be reached at:
kevin.mccarthy@dovermotion.com