# Application Note #5523

Connecting to a Linear Motor with Sinusoidal Commutation and Sin/Cos Feedback

## Table of Contents

<table>
<thead>
<tr>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>1</td>
</tr>
<tr>
<td>Example System</td>
<td>2</td>
</tr>
<tr>
<td>Start with Specifications</td>
<td>2</td>
</tr>
<tr>
<td>Connecting Hardware</td>
<td>3</td>
</tr>
<tr>
<td>Configuring Sine/Cosine Feedback</td>
<td>4</td>
</tr>
<tr>
<td>Configuring Sinusoidal Commutation</td>
<td>5</td>
</tr>
<tr>
<td>Tuning</td>
<td>5</td>
</tr>
</tbody>
</table>

## Description

The popularity of using linear motors to replace linear actuators and pneumatics has drastically increased as the technology has improved. So too has the popularity of using sinusoidal commutation to drive these motors rather than using hall sensors to detect which phases should switch—and it’s no wonder since this commutation method provides lower torque ripple and smoothes out any audible “clicking” trapezoidal drives can have. Although the position
feedback types for these motors is as varied as ever, from traditional, standard quadrature, SSI, BiSS, to basic analog feedback from a force actuator—Sin/Cos feedback has seen its great deal of use with these motors. This application note is meant to take out the mystery of connecting, configuring, and tuning linear motors by practical example system using a Copley Servotube to a DMC-31012.

**Example System**
The DMC-31012 is a single-axis motion controller that can accept Sin/Cos feedback and also has a built-in 800 W internal amplifier that can drive either 3-phase brushless motors or DC-brushed motors. This makes it perfect for driving our example linear motor, a Copley STA1104-1116 Servotube.

**Start with Specifications**
Before setting up any electro-mechanical system it is important to know what resolution the system is and how much error is allowable for the system. Often times this error is given in mechanical terms (inches, mm, etc) and must be translated into electrical terms, for most system this is “counts” in terms of feedback.

The unit of counts comes from the dominate system used for position feedback, the optical-quadrature encoder, where each output “pulse” from the encoder is a known mechanical distance. These pulses are then counted by a controller to interpret them into a distance moved, a speed reached, etc. Although the Copley motor referenced in our example utilizes Sin/Cos feedback, the Galil interpolates this signal into counts. For instance, one-period of the Sin/Cos feedback is a known distance, in this case 25.6 mm/period (this value should be provided by the manufacturer of the stage). If the Galil interpolates this signal to 1024 counts/period, the resolution of the system would held by the relationship below:

\[
Resolution = \frac{25.6 \text{ mm}}{\text{period}} \left( \frac{\text{period}}{1024 \text{ counts}} \right) = \frac{.025 \text{ mm}}{\text{count}}
\]

*Equation 1: Calculating resolution of the example system*

For this particular application, ± .002 inches of error is the acceptable range. Luckily, .025 mm is approximately .0009 inches—well within the required resolution with a significant safety factor. Thus, we know that an interpolation of 1024 counts/period or greater is acceptable. Lower values could also be used, but again, a safety factor is always recommended. How much is dependent on the design and purpose of the system.

Later for tuning of the system, it is helpful to know how much error is acceptable in the units of counts. Using the same information already given above and that .002 inches is equal to .0508 mm, we can calculate the acceptable error in counts:

\[
Error = .0508 \text{ mm} \left( \frac{1024 \text{ counts}}{25.6 \text{ mm}} \right) = 2.032 \text{ counts} = \pm 2 \text{ counts}
\]

*Equation 2: Calculating error of the example system*

Note that the counts are rounded down to the nearest integer. Reason being is that a pulse is either high, or low, thus there is no such thing as fractional counts.
Once the specifications have been translated from mechanical to electrical, a better understanding of the system is reached, and it is time to hook-up the components and begin configuring the system.

**Connecting Hardware**

Wiring the Sin/Cos feedback from the Copley Servotube to the DMC-31012 is a fairly straightforward endeavor. Table 1 shows the wiring of the DMC-31012 to the Copley Servotube. Note: These pin-outs are simply for example and reference and are not meant to substitute either the DMC-31012 or Copley manual for pin-outs. Please refer to the appropriate documentation for each component as different hardware revisions or models may have different pin numbers.

<table>
<thead>
<tr>
<th>DMC-31012 15-pin HD D-sub</th>
<th>Copley STA1104-1116 15-pin HD D-sub</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin #</td>
<td>Label</td>
<td>Pin #</td>
</tr>
<tr>
<td>1</td>
<td>MI+</td>
<td>--</td>
</tr>
<tr>
<td>2</td>
<td>MB+</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>MA+</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>GND</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>MI-</td>
<td>--</td>
</tr>
<tr>
<td>7</td>
<td>MB-</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>MA-</td>
<td>2</td>
</tr>
<tr>
<td>15</td>
<td>+5V</td>
<td>5</td>
</tr>
</tbody>
</table>

Table 1: Pin-outs for wiring feedback to a DMC-31012 from a Copley STA 1104-1116

Connecting the motor phases may require some additional wiring to get the proper phase configuration that will allow sinusoidal commutation of the motor. Galil’s TechTalk blog provides an in-depth summary on how to wire a sinusoidal motor for the first time and how to test the system for proper commutation:

TechTalk: Wiring a Brushless Motor for Galil’s Sine Amplifier

Below in Table 2 are the results obtained by following the TechTalk article above for this example DMC-31012 and Copley motor system. Again, this application note is in no way a replacement for the manual or performing the individual steps required to get proper wiring in the TechTalk article linked above.

<table>
<thead>
<tr>
<th>DMC-31012 4-pin Molex</th>
<th>Copley STA1104-1116 15-pin HD D-sub</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pin #</td>
<td>Label</td>
<td>Pin #</td>
</tr>
<tr>
<td>1</td>
<td>Phase A</td>
<td>13</td>
</tr>
<tr>
<td>2</td>
<td>Phase B</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>Phase C</td>
<td>15</td>
</tr>
<tr>
<td>4</td>
<td>N/C</td>
<td>--</td>
</tr>
</tbody>
</table>

Table 2: Pin-outs for wiring motor phases from a DMC-31012 to a Copley STA 1104-1116
Important Note about Noise
When using analog feedback types, either in the form of a potentiometer, force transducers, or in the case of our example, Sin/Cos feedback, a designer should already be aware that noise will inevitably have an effect on the feedback signal. A designer should be even more diligent in abating this effect with the Copley motor in our example because this Servotube has the motor phases on the same 15-pin D-sub connector as the feedback lines—thus, it is important that noise be taken into serious consideration and will have a serious effect on what configuration settings should be chosen. This logic is explained throughout the follow sections of this application note. Hardware steps can also be taken in order to reduce the effect of noise, these steps are explained in detail in the application note linked below:

Application Note 5438: Electrical Noise in Motion Control Circuits

Configuring Sine/Cosine Feedback
In order to be able to input Sin/Cos feedback into a DMC controller one must be sure that they have the correct hardware to do so. In the case of the DMC-30000 one simply needs to order the DMC-31XXX part number, where the “X” represents other ordering options. For other DMC controllers, be sure to check the manual, specification sheets, or part number generator for the correct ordering configuration.

Once the correct hardware is ordered, one simply needs to configure the DMC controller to interpolate the Sin/Cos period into position data. This is simply done by issuing the \texttt{AF} command.

The DMC-31XXX controller can be configured to interpolate \(2^n\) counts/period, where \(n\) is the argument fed into the \texttt{AF} command. Using our example, where we want 1024 counts/period, \(1024 = 2^{10}\), thus we'll issue \texttt{AF 10} to configure the A-axis for 1024 counts/period interpolation.

But why use \texttt{AF 10} instead of the maximum \texttt{AF 12}? Isn’t the highest resolution always the best? The answer is: not necessarily. The reason is noise.

Many products will often compromise amplifier performance and use lower current-loop bandwidth amplifiers to “filter out” the controller’s response to noise on the analog feedback lines. Galil does not make this compromise and our amplifiers remain high performance and high bandwidth. This may mean that Galil’s amplifiers can respond to the changes on the error output due to noise if using an analog feedback type. To prevent the effect of noise, interpolating using \texttt{AF 10} rather than \texttt{AF 12} allows us to filter out as much noise as possible without having to sacrifice the specifications of our application or amplifier performance.

During tuning one can then play with the PID gains and servo loop to meet the strict \(\pm 2\) count error requirement without introducing unnecessary response in the system. Tuning is also a good time to adjust the \texttt{AF} settings to see if higher values are possible—This does end up being the case, as discussed further in the Tuning section below. But before tuning, one must first ensure that the system is properly commutated and configured and a starting point of \texttt{AF 10} is perfect for this.
Configuring Sinusoidal Commutation

Although briefly discussed in the TechTalk article for Wiring a Brushless Motor in the “Connecting Hardware” section above—it did not go into detail on how to chose whether to use **BX** versus **BZ**, or how **BM** values are calculated. This section will go into further depth and explain why certain values and commutation methods were chosen over others in the system.

It was decided, partially from experience, that **BZ** would be a better commutation method for this particular system. Again, it comes down to noise. **BX**, although appropriate for many models of linear motors and stages, does not tend to fair well in noisy environments. The reason is that **BX** excites the motor phases and uses an algorithm to calculate where in the magnetic cycle the system currently is. The benefit of **BX** is that it will only move the motor a few counts, but it’s algorithm is susceptible to be effected by noise. In comparison, the **BZ** command is more “brute-force” and forces the motor to a zero magnetic phase and sets this point of the commutation phase to zero; this method makes **BZ** better for noisy environments or when a single-sided frictional offset (such as gravity) exists.

Another commutation parameter to be specified is **BM**. This is the “brushless modulus” of the system, or simply put, the length (in counts) for which one magnetic cycle completes. This data is usually provided by the motor manufacturer. Luckily, the Copley motor’s Sin/Cos period is the same length as the motors magnetic cycle. Thus, for an **AF 10**, which represents 1024 counts/period also represents 1024 counts/magnetic cycle. Thus our command would be **BM 1024**.

Lastly, to bring all these commands together to complete our commutation is shown below:

```
AF 10; ' Need 2^10 or 1024 counts/period
BA A; ' Set A-axis for sinusoidal commutation
BM 1024; ' 1024 counts/magnetic cycle
BZ A= 4; ' Use 4V to force motor to the 0 magnetic cycle
```

For a full explanation of each of the commands used above, one should refer to their specific command reference for their controller.

For further explanation of sinusoidal commutation, refer to the application note below:

Application Note #1501: Brushless Sine Drives


Tuning

As always with tuning, it important to tune with a move that would be expected of the system with realistic, but aggressive accelerations, decelerations, and speeds to ensure that any other move could easily be performed with the system given the PID’s chosen for the more aggressive move.

For our example, an acceleration of 100000 counts/s^2, deceleration of 20000 counts/s^2, a slew speed of 100000 counts/s, and a point-to-point move of about six inches, or 6144 counts was
used to create or profile for tuning. We repeated this point-to-point move “back-and-forth” to easily see the systems response to changing tuning parameters. The program used for tuning is shown as below:

```
#a
AC 100000;' Acceleration
DC 20000;' Deceleration
SP 100000;' Speed
BA A;' Specify A-axis for sinusoidal commutation
BM 1024;' Specify there are 1024 counts/magnetic cycle
BZ -4;' Commutate motor using 4V
step= 6144;' Define move using variable step
wait= 250;' Define wait period between moves
WT 2000;' Wait 2 seconds
DP 0;' Define position as 0

#b
PA step;' Move distance specified by variable step
BG A;' Begin motion
AM A;' After motion is complete
WT wait;' Wait designated amount
PA 0;' Move back to original position
BG A;' Begin motion
AM A;' After motion complete
WT wait;' Wait designated amount
JP #b;' Repeat move indefinitely
EN
```

Using the program above and graphing the actual encoder position (_TPA), reference or ideal position (_RPA), torque (_TTA), and position error (_TEA) in real-time with GalilSuite—we are able to see the immediate response of our system to a change PID parameters in real time using the a variation of the tuning methods described in the following application notes:

Application Note 5491: Manual Tuning using the Velocity Zone method

Application Note 3413: Manual Tuning Methods

Different engineers will prefer different methods, and some work better for some systems over another. Choose which method gives you the better results for your system.

In the end, with an AF 10 we found that the following tuning parameters gave us an error within ±2 counts and a settling time that is near-instantaneous as shown in Figure 1.

```
TM 2000;' Sero-update rate
AG 2;' Amplifier gain (Amps/Volts)
AU 2;' Amplifier current-loop gain
FV 24;' Feed-forward velocity
```
Tuning Parameters 1: Appropriate parameters for AF 10, BM 1024.

FA 4;' Feed-forward acceleration
KP 150;' Proportional gain
KD 3000;' Derivative gain
KI 10;' Integral gain
IL 2;' Integrator limit
PL 0.5;' Pole filter

Figure 1: AF 10 system response to the tuning program and gains given. Key to graph below:
- **Red:** Ideal, instantaneous velocity \(\frac{dx}{dt} - RPA\) in 2000 counts/division
- **Green:** Actual position \(_TPA\) in 5000 counts/division
- **Blue:** Position error \(_TEA\) in 4 counts/division
- **Pink:** Torque \(_TTA\) in 5V/division
- **Black:** Shows when in motion (high) and motion complete (low), \(_BGA\)

Nothing is particularly special about these tuning parameters and they were achieved by simple trial-and-error, but one may scratch their head at TM 2000. The TM parameter controls the rate at which the servo-loop samples—for many engineers this is the first parameter they lower to get better response time and performance. But for systems with sluggish mechanics there may not be enough “information” or position error for parameters like KD (damping) to compensate for higher-frequency errors (such as noise) at such servo-update rates. By lowering the servo-update rate one allows for additional position error to build up and allows KD to have a greater effect, essentially dampening out the errors caused by noise. In this case, increasing TM improved the stability of the system so much so that increasing the system to interpolate at greater feedback resolutions of AF 11 and AF 12 did not derogate the performance of the system in any significant way. Below are the tuning parameters used at both AF 11 and AF 12 as well as the system response shown in Figure 2 and Figure 3 respectively. Note, that due
to the increase of AF, AC, DC, SP, BM, and PA will all be required to change proportionally with the change of resolution. With change in the resolution will also require tuning of the system.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>TM 2000;</td>
<td>Sero-update rate</td>
</tr>
<tr>
<td>AG 2;</td>
<td>Amplifier gain (Amps/Volts)</td>
</tr>
<tr>
<td>AU 2;</td>
<td>Amplifier current-loop gain</td>
</tr>
<tr>
<td>FV 12;</td>
<td>Feed-forward velocity</td>
</tr>
<tr>
<td>FA 4;</td>
<td>Feed-forward acceleration</td>
</tr>
<tr>
<td>KP 75;</td>
<td>Proportional gain</td>
</tr>
<tr>
<td>KD 1500;</td>
<td>Derivative gain</td>
</tr>
<tr>
<td>KI 5;</td>
<td>Integral gain</td>
</tr>
<tr>
<td>IL 2;</td>
<td>Integrator limit</td>
</tr>
<tr>
<td>PL 0.5;</td>
<td>Pole filter</td>
</tr>
</tbody>
</table>

*Tuning Parameters 2: Appropriate parameters for AF 11, BM 2048.*

*Figure 2: AF 11 system response to a proportional tuning program and gains given. Key to graph below:*

- **Red:** Ideal, instantaneous velocity \( \frac{d}{dt} RPA \) in 20000 counts/division
- **Green:** Actual position \( TPA \) in 5000 counts/division
- **Blue:** Position error \( TEA \) in 8 counts/division
- **Pink:** Torque \( TTA \) in 5 V/division
- **Black:** Shows when in motion (high) and motion complete (low), \( BGA \)
TM 2000;' Sero-update rate
AG 2;' Amplifier gain (Amps/Volts)
AU 2;' Amplifier current-loop gain
FV 6;' Feed-forward velocity
FA 1;' Feed-forward acceleration
KP 40;' Proportional gain
KD 700;' Derivative gain
KI 2.5;' Integral gain
IL 2;' Integrator limit
PL 0.5;' Pole filter

Tuning Parameters 3: Appropriate parameters for AF 12, BM 4096.

Figure 3: AF 12 system response to a porportional tuning program and gains given. Key to graph below:

Red: Ideal, instantaneous velocity ($\frac{\text{counts}}{\text{sec}}$, RPA) in 10000 counts/division
Green: Actual position ($\_\_\_\_\text{TPA}$) in 5000 counts/division
Blue: Position error ($\_\_\_\_\text{TEA}$) in 16 counts/division
Pink: Torque ($\_\_\_\_\text{TTA}$) in 5V/division
Black: Shows when in motion (high) and motion complete (low), ($\_\_\_\_\text{BGA}$)