

# Synchronization: Delivering tailored motion

Many applications require precise synchronization of one or more axes. Here we run through a few different approaches to coordinate motion.

## Chuck Lewin

Founder and V.P. of Engineering  
 Performance Motion Devices Inc.  
 Lincoln, Mass.

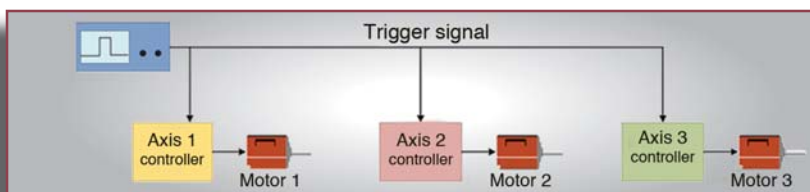
One simple form of motion synchronization, still widely used, is an external signal trigger that causes a synchronized change in the profile for one or more axes. Modern motion controllers allow a range of automatic trigger conditions (sometimes called *breakpoints*) so that axes can be started, stopped, or otherwise altered by

an external signal change, and so that the triggering condition can be nearly any combination of signal conditions.

In a typical application, the host controller preloads a new set of target profile parameters and sets a breakpoint event to occur when the external signal condition goal is reached. When the signal arrives at the target state, the motion controller automatically initiates the new motion profile.

For most applications, because moves are short, or because it is okay for axes to arrive at slightly different times, this is acceptable. However, a number of applications exist for which this approach won't work — including situations requiring continual synchronization between axes, and not just at the start of motion.

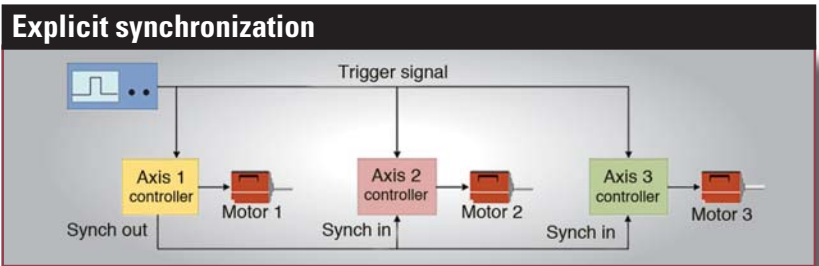
Why would two or more axes get out of synch during a motion? Motion controllers run their clocks



**One simple form of motion synchronization, still in widespread use, is an external signal trigger that causes a synchronized change in the profile for one or more axes.**

For most applications, moves are short, or axes can arrive at slightly different times without issue. Elsewhere, explicitly synchronizing controller servo loop rates is indispensable.

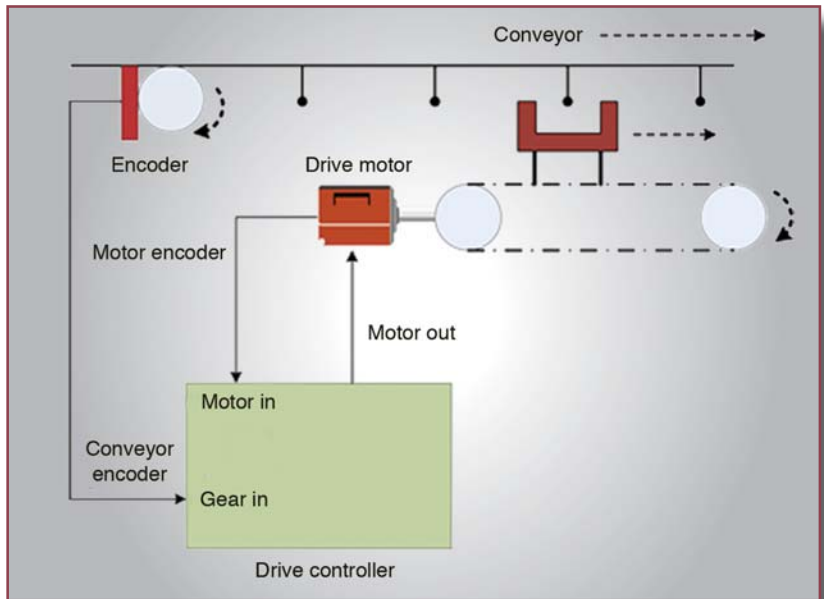
As shown here → one of the motion controllers serves as the master by generating a synch or strobe signal, which is then accepted by one or more slave controllers.



from an internal oscillator. Each oscillator provides a slightly different frequency, so drift can slowly accumulate between multiple axes.

Where it is problematic, one solution is to explicitly synchronize the servo loop rates of each controller. Explicit synchronization is done by designating one motion controller to serve as the “master” and generate a *synch* or *strobe* signal — in turn, sent to one or more slave controllers.

Alternatively, the same benefit of explicit axis synchronization occurs automatically using a multi-axis controller rather than a chain of single-axis controllers. This is because multi-axis controllers use one master internal clock, so their servo loops are explicitly synchronized.



Electronic gearing utilizes an external position data stream (nearly always quadrature encoder signals) to provide the master drumbeat by which all motions occur — as shown here. To make electronic gearing work, the host specifies the ratio of driven-motor output counts for each input encoder count.

### Get your gear on

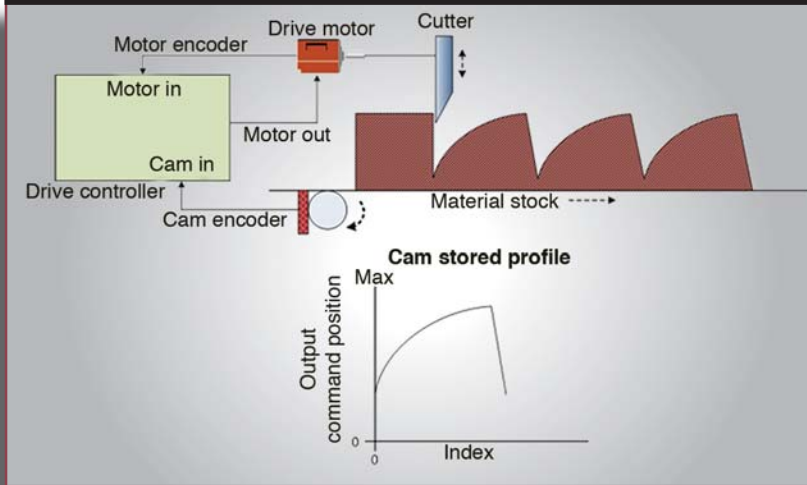
What if synchronization is more than a matter of keeping an axis coordinated with itself? Consider the application of a robot arm that must track the speed of an external conveyor. In this application, the conveyor speed may not be exactly known; therefore, the robot arm

Here and in the photo opposite, ION drives from Performance Motion Devices Inc. coordinate forces and eliminate the effects of gravity in Exerbotics equipment — for more flexible eccentric-resistance strength training. A precise load sensor measures the user’s force output at all phases of the lift.



Photos courtesy Exerbotics LLC.

### Sophisticated: Electronic cams



A more sophisticated version of an electronic gear is an electronic cam. This also takes an input position data stream (from an external encoder, for example) but uses this input position as an index to a lookup table — the contents of which determine the driven motor's output command position.

As with electronic gearing, cams can track an internal axis as well as an external encoder data stream. When tracking an internal signal, one axis acts as the master and the other axes act as followers using the specified cam profiles.

Cam lookup table schemes come in various flavors to control how the

output position is generated. A myriad of vendor-specific variations are available, but the basic idea is that cams can generate cyclical paths that always return to the same location, or paths that generate a net positive or negative position path.

must somehow be synchronized by a signal that changes constantly.

Enter electronic gearing, and its more sophisticated cousin, electronic camming: Electronic gearing utilizes an external position data stream, nearly always quadrature encoder signals, to provide the master drumbeat by which all motions occur.

To make electronic gearing work, the host specifies the ratio of driven-motor output counts for each input encoder count. Multiple axes can be driven at different ratios, allowing, for example, a continuous web to be guided by rollers of different diameters.

Additional features of electronic gears include continuously changeable gear ratios (resulting in smoother starts and stops) and the ability to track an internal axis position, rather than an external encoder data stream. This allows, for example, multi-axis straight-line moves to be drawn, even while the profile moves through a complex velocity profile such as an S curve.

### Cam we talk?

A more sophisticated version of an electronic gear is an electronic cam. These also take an input position data stream, such as from an external encoder, but use this input position as an index to a lookup table — the contents of which determine the output command position of the driven motor.

If cams control multiple axes, creating multi-dimensional shapes is easy. For example, to draw circles or ovals from a linear input motion, sinusoids are loaded into the lookup table, comprising the X and Y components of the circular shape. More commonly, cams are used to implement kinematically complex relationships between axes, such as those of *selective compliance assembly robot arm* or SCARA-style robots.

### World on a lookup table

To work in real applications, the cam profile must be at the correct phase as it moves through its motion. In other words, the cam profile must be at the correct output lookup index for a given input position. Otherwise, to borrow from our circle example, the drawn object might not be a circle but a tilted ellipse.



Off-the-shelf ION drives provide numerous options for synchronizing axes and generating highly precise motion.

To this end, motion controllers with cam profiles allow an offset to be inserted into the index look-up. While moving, introducing or changing this offset should be done smoothly so that the change in offset does not cause an abrupt change in cam output.

Finally, changing the cam shape (the loaded cam profile) while it is in use is not recommended. For systems constantly on the go, however, a common switchover scheme is to use a “foreground” cam profile for active motion and a “background” profile for loading new profiles. By carefully synchronizing changeover of the foreground with the background, it is possible to switch on the fly, with no abrupt changes in motion.

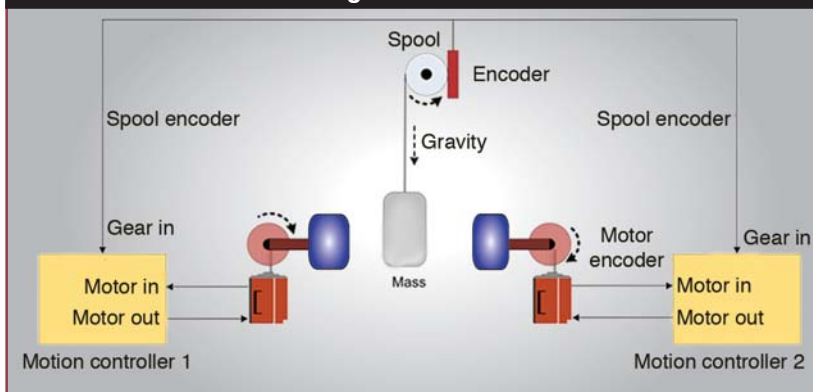
### Network is the motion computer

As is often the case, no motion controller discussion is complete without a mention of distributed digital control networks. In fact, a primary use of digital control networks for motion control such as CANopen, EtherCAT, and SERCOS is to provide tight multi-axis synchronization between a network of single axis modules.

Some digital networks, particularly those that employ the synchronizing protocol IEEE 1588 PTP (Precision Time Protocol), can synchronize axes to very high accuracies, just as if a physical strobe signal were connecting each module.

However, building a motion controller using one of these synchronized protocols may not be the best

### Coordinated even under high acceleration



**In this setup, a position encoder data stream is provided by a falling weight connected by a thin wire to a rotating spool/encoder. This encoder data stream is fed into two drives, which in turn drive two brushless dc servo motors.**

idea. Why? Though fast, the control languages provided by these systems need a lot of attention from the host. The most common approach requires the host to send a continuous stream of PVT (position, velocity, time) points for each controlled axis, generally about once every 10 msec or faster.

For CNC applications, plotters, and a few other applications, specific PC-based software programs can do the heavy lifting, but for general-purpose motion applications such as medical, scientific, and most machine automation, software would likely have to be written by the user.

### To the laboratory

Demonstration time: Refer to the above image — and assume that a position encoder data stream is provided by a falling weight connected by a thin wire to a rotating spool

encoder. This data stream is fed into two separate drives that in turn drive two brushless dc servo motors.

The position of each motor is commanded via the drives’ electronic gear profile mode. Everything is arranged so that the falling weight and the two driven motors arrive at the same point in time and space. Though the weight continually accelerates at  $9.8 \text{ m/sec}^2$ , high-speed cameras verify that the drives can easily track the changing speed of the falling 500-g weight.

To view videos from PMD’s lab that show motion synchronization in action, please click on the following links:

- [Synchronized motion move](#)
- [High-speed shot @ 420 fps of the same move above](#)



**For more information on PMD’s motion controllers & accompanying software, please contact:**

**info@pmdcorp.com**  
**781-674-9860**  
**www.pmdcorp.com**

**We would like to assist you in improving your motion system.**