Save with the Right Distributed Motion Control

Written by: Peishan Juan, Advantech Corporation, Industrial Automation Group

In today's automation applications, it seems there's no room for dumb machines and there's plenty of space for intelligent choices. A case in point is distributed motion control, in which intelligence and control are pushed out from a central server to the edges of an automation system. There are some very good, bottom line reasons for doing this, as it can save significant time and money.

However, not all distributed motion control approaches are the same. A wrong choice about which approach to implement can lead to problems. Correcting that error can be expensive, time-consuming, or both. Hence, there's a need to pay attention to the degree of time-determinism, the amount of integration in control loops, the types of motors that can be managed, the openness of the software interface, and other specifications. Dealing with such issues upfront makes the eventual implementation smarter and more cost-effective.

It is important that the motion control solution be appropriate for the problem at hand. Over and under specifying should be avoided, two guiding principles that can be found in the design and implementation of Advantech's distributed motion control products.

This white paper will look at the reasons behind the trend toward distributed motion control, examine when it makes sense to take this approach, and detail some important specifications. Finally, it will look at how Advantech answers these needs and show why this approach leads to the right solutions. Being armed with this information makes it much easier to choose correctly.

Distributing Control Saves Money and Time

Although overall market statistics are difficult to come by, there's clearly a strong trend toward distributed motion control. A scan of vendors reveals a growing number of offerings in this area. Such automation intensive industries as automotive, packaging, semiconductor, and flat panel manufacturing, to name a few, increasingly make use of the technology. A key reason is that distributed control saves money and cuts installation time.

Before 2000, distributed motion control typically wasn't an option. Instead, automation designers had to contend with centralized control systems, in which a host, such as a PLC, managed the movement of motors. A downside of this approach was the number of wires required. A servo motor might need multiple wires to carry signals such as encoder feedback, sensor output, and other information back from the motor location to the central controller. Up to 25 wires per axis might be needed, according to published estimates. Constructing such a system could be challenging, with the outcome being thick cable bundles that were difficult to service and hard to make reliable.

Another downside was the length of the cable runs. Because wiring ran from the central controller to the motor, each added foot of distance upped the chance that a wiring break or other problem would occur. Likewise, each extra increment of length upped the price. Some automation systems in the material handling and manufacturing world have to span hundreds of feet. Given the number of wires needed, such runs could be extremely costly to build and service.

Somewhat less obvious problems in the centralized approach involve the controller and control cabinets. Extending system functions or upgrading the controller necessarily meant risking the entire automation system or stopping production. It was also possible that a system's determinism, its ability to respond to a request for action in a time-definite manner, could suffer if there were too many such requests. The latter would only happen when the automation system became too large or complex, thereby overwhelming the central controller. In such cases, the solution might be the implementation of multiple control cabinets.

In contrast, a distributed motion control system eliminates or reduces many of these problems. Because the control functions are in or near the drive, there's less wiring and cabling that runs from a central point to every motion axis. Instead of having a bundle of dozens of wires strung between a motor and a PLC, a distributed motion system may need only one running from the overall control point.

This reduction in wiring makes cabling easier and less expensive to install, while improving the reliability of the wiring. It also makes for easier maintenance and diagnosis, the latter because it's no longer the case that a long run has to be checked out when there is a problem. An added benefit is that control can be exercised over greater distances. The distributed system effectively has short control runs that together span a long distance.

The controller in the distributed case can be upgraded independently and there's less need for control cabinets. Indeed, it may be possible to eliminate them altogether, except for a single one for the overall system control point.

Finally, adding to a properly designed and implemented distributed system is less likely to impact its determinism. One of the main worries with putting motion control over a network is that the response time of the system may suffer due to network congestion or too many demands on a central controller. Traditionally, the first issue is overcome by the use of a high throughput or dedicated network. Neither of these approaches, though, deals with the problem of too many demands on a central controller. Another possible solution does. This is the use of a smart remote device which minimizes traffic on the network. Since the control of motion is spread throughout the system, there is an excellent match between motors and motion control resources. Thus, it is far harder to overtax the automation system with motion control requests.

A Variable Breakpoint

Given all of these benefits, should distributed control be implemented everywhere and in all circumstances? The answer is no, since there are some additional costs in a distributed motion control approach. In the simplest case – a single motor handling a simple motion with a short distance from motor to central cabinet – that extra cost will eat up any savings.

However, there are many cases where distributed motion control is clearly cost effective. Studies have compared the estimated cost of a standard centralized system to a distributed one for such typical scenarios as a conveyor. In a conveyor systems, multiple variable speed motors control individual segments, but there's only one axis of movement for each motor. Consequently, this is not the most complex of possible motion control projects.

Modeling this setup leads to the conclusion that potential total savings range from a low of about a fifth to a high of nearly three fifths. These savings show up in materials, installation labor costs, installation hours, and labor hours needed to build complex drive panels. Distributed control can potentially eliminate nearly every panel. Thus, in some cases, the savings in this area approach 100 percent.

The reason for this range of overall potential savings has to do with the assumptions as to a project's span and scope. The savings are higher for situations with more motors, the result of each added motor leading to added savings. At some point, this linear differential can accelerate and become quite pronounced.

In the distributed case, adding a motor typically means a fixed rise in cost and complexity, leading to a linear increase. For the centralized case, the same can be true. However, the slope is steeper, leading to increasing savings as motors are added. Continuing to add motors to a hypothetical project, though, can eventually mean a complexity threshold is passed, in which case the cost of adding a motor suddenly climbs.

Thus, distributed control isn't always, but often is, best. That fact naturally leads to a follow-on question. In what situations should a distributed motion control system be implemented?

The short answer is when roughly six motors are involved, but the long answer is that it depends. The six motor guideline is based upon the typical customer experience implementing solutions designed with Advantech's systems. For situations in which the total number of motors is less than six, distributed motion control typically doesn't pay. In cases where the total is greater than six, the cost savings in wiring, installation labor, and maintenance outweigh any added expense.

However, this guideline may not apply to all projects. In some cases, the savings in installation time and the ability to get a project up and running can justify added cost. The tipping point between centralized and distributed motion control would then be less than six. Alternatively, there could be a situation where a conveyor with a centralized control system is being replaced. If the wiring already exists, the cost benefits of the distributed approach could be lessened or even eliminated.

It's only after considering all of these factors that a determination about a given situation can be made. In general, the more complex a system and the greater the number of motors, the more it makes sense to implement a distributed approach.

Important Specifications

After the calculations are done and the decision is made to go with distributed motion control, the next question becomes which solution should be implemented. As is the case with the breakeven point, the answer depends upon the situation. However, there are some key specifications and attributes that should be considered. These can be used to differentiate one solution from the next.

These important specifications and characteristics include interpolation, time-determinism, synchronization, and integration of I/O and motion in the control loop. The amount of intelligence in the field devices is also important, as is the ability to integrate multiple types of motors in one loop and the openness, or lack thereof, of the software interface.

As for the first of these, interpolation enables a system to transform a complex shape or trajectory into something a motor can execute. While real world is full of curvilinear two and three dimensional shapes and paths, the motors responsible for motion control typically operate on one axis. With interpolation and the right motor arrangement, that one-dimensional response can be used to arrive at a destination along a curved or linear 2- or 3-D path. Clearly, then, any solution selected has to offer the needed interpolation.

Determinism can be a slippery concept. A real-time response, which is at the heart of deterministic systems, means the system takes action, without fail, within a specified time. That time is generally given in terms of a system clock tick. Thus, a system can be deterministic if it responds in a second, a millisecond, a microsecond, or any other unit of time.

Given that, the important question becomes how fast is fast enough. A system guaranteed to respond at too slow a speed is useless. One that responds very fast is likely to be expensive and in most situation inefficient, a result of the system sitting around waiting for something to respond to. Thus, it's important to have a system with the appropriate real-time response and corresponding determinism.

Synchronization, the ability of motors to stop and start at the same time, can also be important. Take the case of a conveyor belt. All of the one-axis motors that move the belt along need to start and stop together. Pulling off such coordination can involve controller to controller, motor to motor, or motor to I/O synchronization.

Successful synchronization requires that the motors respond promptly and in a uniform manner to any commands. It also requires that all of the motors receive the right sequence of commands at the right time, thus involving controller, motor, and I/O. Meeting this condition mandates there be deterministic communications to and from the motors for both commands and data.

The degree of integration is another key characteristic. This can show up in a number of areas, such as the ability to integrate motion and I/O in one control loop or the ability to combine multiple motor types in one loop. Both can cut down on the amount of wiring, contribute to flexibility and improve overall performance in the final automation system. What's more, integrating motion and I/O in one control loop can help with synchronization.

The intelligence of the field devices in any solution also needs to be considered. Smart field devices can share the load of the central controller, and this ability to offload computing could be important when upgrading the overall automation system or adding capabilities.

Finally, the nature of the software interface in the solution is an essential element. The interface can be based on standards and open solutions or it can be proprietary to a given system. The former minimizes training time for engineers and technicians because it doesn't require that they learn a new interface. It also can serve as a protection of the investment in training.

Advantech's Solution

The motion control market is highly fragmented and the requirements of the various groups of users are diverse. An extremely high powered, massively capable, and therefore expensive motion control solution will be overkill for many applications. The right approach, on the other hand, can solve the same problem and do so for less.

Advantech has, therefore, tailored its offerings to meet the needs of the majority of the market. An example of this approach can be seen in the AMAX-2000 series of embedded automation solutions, consisting of master and digital slave modules. These can handle runs of up to 100 meters and up to 64 motion devices in a single loop. They also integrate motion and I/O in one communication loop, simplifying cabling layout and system assembly.

Knowing that many automation systems don't require microsecond response times, Advantech designed its solution with a maximum cycle time of a millisecond. In a scant 40 microseconds more some 2048 digital I/O channels can be updated. With regard to synchronization, the 1-axis modules support coordinated start and stop.

The Advantech solution also offers interpolation, with an important resource-saving twist. The interpolation calculation is done in the slave module, freeing the central controller from the need to determine all the parameters. Instead, the central or machine controller selects a start and stop position, sending these to the slave motion module. This module then generates the velocity profile and controls the motors. A 4-axis motion module, for example, supports 2-D circular interpolation and 3-axis linear interpolation.

Safety and regulatory requirements haven't been forgotten in all of this. The motion modules have local I/O interrupt for either a limit switch or emergency stop, if that is desired or needed.
The programming environment is open and standard compliant, meeting the requirements of IEC-61131-3. The interface conforms to PLCOpen Function Blocks, the only worldwide motion control standard in the IEC-61131-3 environment. Because this standard is widely used by different

solution providers, it's likely that engineers and others in the technical staff already have been exposed to it, which minimizes the need for training.

Advantech has even extended this idea of an open architecture to the hardware level. The output of its motion controllers is generic. Thus, they can control cross-brand pulse type servo motor drives as well as managing stepper motor drives and linear motors. This capability frees the end user from the need to force a solution to use a particular type of motor. Instead, it allows the best motor for a given circumstance to be deployed.

With this generic hardware communication, different types of motors can be mixed in a machine or system. Thus, a machine needing 25-axis control can do so with linear, servo, and stepper motors. The result will be a system that more closely matches requirements than would otherwise be possible. Meanwhile, thanks to distribution motion control, the machine will meet specifications for less money and in less time.

Wrapping up

Going with a distributed motion control solution can save money and cut installation time if the application is complex enough. The breakeven point depends upon the specific circumstances. As a rule of thumb any system with more than a handful of motors is a good candidate.

When implementing a distributed control system, consider such specifications as interpolation, time-determinism, synchronization, degree of integration, software interface, and others. Make sure that the solution isn't overdesigned or specified too tightly. At the same time, check to be sure that the solution meets requirements.

Following these guidelines will help in making the right choice. With that, the potential savings from distributed motion control can be substantial.

###