EtherNet/IP™ for Real-Time, Machine-to-Machine Control

With its huge commercial base and wide bandwidth, EtherNet/IP offers the potential to combine the required performance of a motion control network with the wider demands of a control network. This white paper describes an approach to EtherNet/IP that enables highly distributed and modular machine control architecture for motion control applications. This un-intrusive approach is easily layered across multi-vendor implementations – including purpose-built networks optimized for controller-to-drive communication.

This approach is the first of its kind to offer high performance, real-time machine-to-machine control over standard TCP/IP. As a result, it provides a unique, unifying real-time solution for many applications requiring multi-machine and multi-axis coordination, such as electronic line shafting, camming and transfer lines. The audience for this paper includes product managers at device vendors who are seeking to map out the product roadmaps to support high performance, motion-centric machine applications as well as users who apply machines to solve applications such as electronic line shafting, camming and transfer lines, among others.

Introduction

ODVA is currently defining extensions to EtherNet/IP that will enable a peer-to-peer connection and allow for the distribution of real-time motion control information among multiple controllers and device types. This capability will make the Common Industrial Protocol (CIP) the unifying technology in the industrial marketplace for machine-to-machine applications that require high performance motion control. The approach adapts the architecture of CIP Motion™, proven for controller-to-drive control, to new applications at the machine-to-machine and line control levels that can be independent of the underlying controller-to-drive technology.

The CIP Motion architecture itself has some unique advantages which have been extended to the approach described herein. As a part of the CIP protocol, the CIP Motion objects and services can coexist and cohabitate with services on other standard Ethernet solutions – and do so in a harmonized and non-disruptive manner.

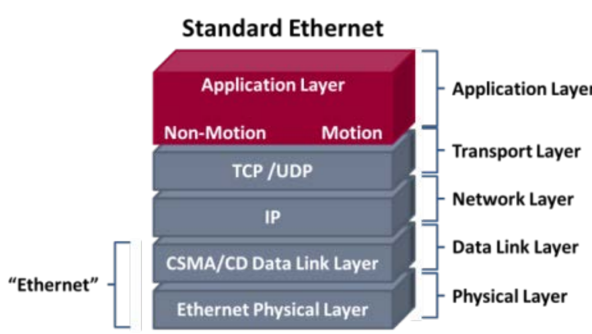

Figure 1: CIP Motion is built on a standard, Ethernet implementation

CIP Motion, as a part of CIP, takes advantage of the fact that it is built on a common, standard Ethernet stack.

This means that in addition to the Physical and Data Link layers defined in the IEEE-802.3 specification, CIP uses the standard Network and Transport layers typically deployed in general Ethernet applications. This allows for devices to easily interconnect using standard switches, routers and other infrastructure components.

One benefit of this single network solution is that there is no need for a dedicated motion network. Topologically speaking, motion control can be placed anywhere in the infrastructure without physically isolating the traffic from other forms of traffic on the wire. Also, by using the existing Ethernet infrastructure, motion control can be easily layered into brownfield installations where hardware already exists. This facilitates expansion to installed systems and takes full advantage of well-established and robust management and diagnostic tools.

So, what additional benefits does the CIP Motion peer-to-peer connection bring to multi-axis applications?

The distribution of motion control across multiple cells or machine sections occurs at the level in the architecture where the line control functionality is already being accomplished.

At this level, there is typically integration of machine-level HMI, recipe handling, historian data gathering, and engineering workstation functionality. The existing Ethernet wire that is being used for these disciplines can now be harnessed to share real-time motion control information for machine-to-machine coordination.

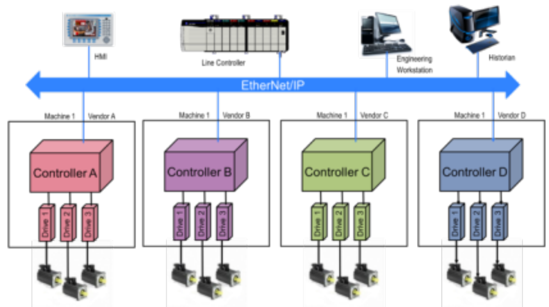

Figure 2: Harness existing Ethernet wire for sharing of real-time motion control

The CIP Motion peer-to-peer connection is a very simple, homogenous and non-intrusive mechanism. It is easily layered across multi-vendor implementations, creating a unifying real-time solution for electronic line shafting and camming applications.

The CIP Motion Peer-to-Peer Connection Definition

The CIP Motion peer-to-peer connection is designed to transmit high-speed motion data from a producing controller or device to multiple consuming controllers or devices over a single multicast connection.

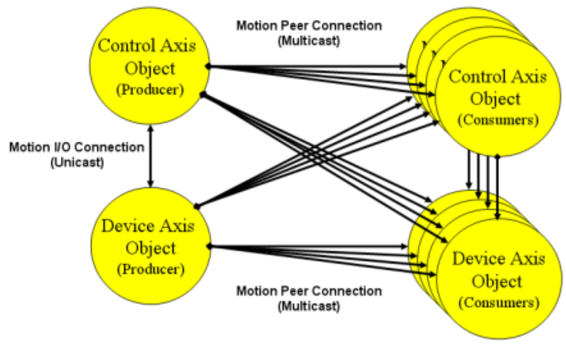

Figure 3: The CIP Motion peer-to-peer connection transmits motion data to devices

For example, time-stamped master axis position data distributed by the CIP Motion peer-to-peer connection allows consuming CIP Motion drives to precisely coordinate the motion of their motors to the produced master axis position according to programmed electronic gearing or camming relationships.

Use Cases

The CIP Motion peer-to-peer connection has many use cases. As discussed previously, one application is to communicate produced axis information for distributed camming and gearing purposes.

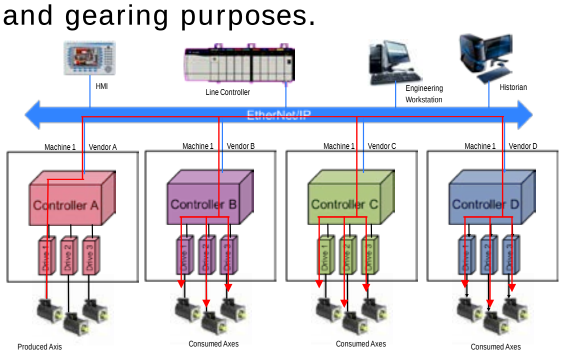

Figure 4: CIP Motion peer-to-peer connection *allows motors to communicate axis information for distributed camming and gearing purposes*

In this scenario, the position of a motor being driven in a machine cell by Vendor A may be produced as a master axis reference and sent out on the CIP Motion peer-to-peer connection. There it will be consumed by the other vendor controllers so that their respective axes can follow Vendor A's positioning information in a coordinated and synchronized manner.

A second use case is when the line controller itself generates a virtual axis to achieve full line synchronization and coordination. In this case, all the coordinated axes receive a common reference via the CIP Motion peer-to-peer connection from the line controller to maintain coordination and proper phasing from axis to axis. This allows for coordinated camming and gearing functionality, and complete line starting and stopping as all sections are brought up and down to speed in a synchronized manner.

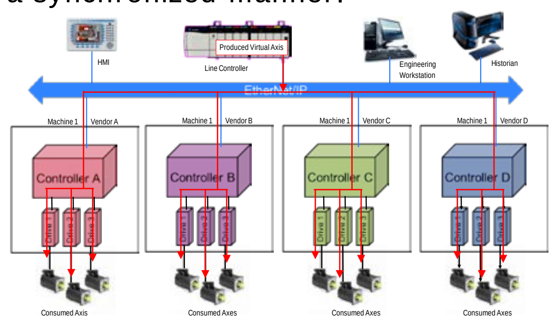

Figure 5: All the coordinated axes can receive a common reference via the CIP Motion peer-to-peer connection from the line controller

A third use case for the CIP Motion peer-to-peer connection is the sharing of axis information for the coordination of robotics control. In this case, a master reference could be produced over the CIP Motion peer-to-peer connection to properly coordinate the robot as the entire line increases and decreases in speed.

For example, the robot may be managing a "pick and place" application from one conveyor to another and needs to stay fully synchronized as the entire line is accelerated or decelerated.

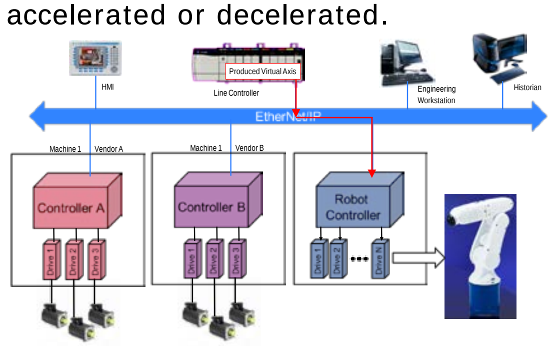

Figure 6: CIP Motion connection shares axis information to coordinate robotic control

Distributed Performance

As shown above, the CIP Motion peer-to-peer connection is designed to support controller-to-controller coordination. But it can also be used for controller-to-drive, controller-to-device, and for drive/device-to drive/device coordination.

Drives and other devices are now able to execute motion functions typically performed by the controller or by full-featured servo drives. In short, the CIP Motion peer-to-peer connection represents a major step towards a distributed Motion Control architecture – signaling a note-worthy paradigm shift in the industry.

There are many benefits to moving toward a distributed Motion Control architecture. One is its impact on overall system performance. Traditionally, a centralized controller executes the Motion Planner as part of its Motion Task. In this arrangement, the controller's Motion Planner is producing motion reference information to multiple drives in the system.

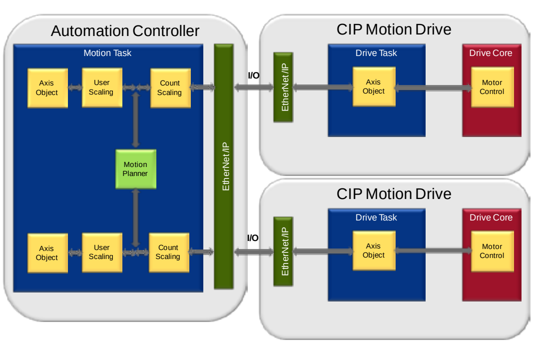

Figure 7: The Motion Planner is traditionally executed as part of a controller's tasks

By distributing the Motion Planner function to the end devices, the automation controller Motion Task execution time is significantly reduced, and the end drives are able to run their unique Motion Planner Update Periods at rates that are aligned with the response needed in the application. Furthermore, by eliminating the "one to many" relationship between the automation controller and the drives, real-time communications is greatly minimized due to reduced connection update rates. This model is much more flexible and translates to a dramatic increase in motion control system capacity and performance.

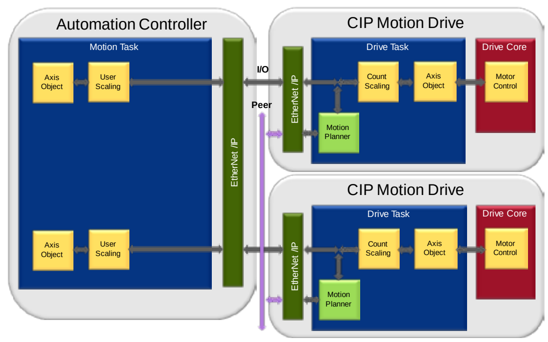

Figure 8: CIP Motion peer-to-peer connection *can off-load the Motion Planner function to the devices*

Motion control dynamic performance is also dependent on the Motion Planner update period, determining how frequently the Motion Planner function updates the command position applied by the drive.

The faster the command position update rate, the more points that are placed on the motion profile – reducing the interpolation error in following the commanded path.

The benefit of running the Motion Planner in the drive is that, since there is typically one planner instance per drive, the planner can be run at a very fast update rate, resulting in outstanding dynamic control. To appreciate the impact of the Distributed Motion Control feature, consider a control system running a high performance packaging machine producing 1,000 products per minute using a 1 millisecond Motion Task Update Period.

In this application, one product is being processed every 60 milliseconds. If there are 30 axes of control required for this machine, a controller-based planner would need to calculate all 30 axes every millisecond.

However, if the planner functionality is moved to the device instead, then each drive can execute its own path planning functionality while the peer-to-peer connection manages the information required for drive-to-drive coordination.

In this architecture, the drive-based planners can be run at a sub-millisecond update rate and the controller-to-drive cycle period can be increased to 10 milliseconds or more without impacting the motion quality. As a result, the estimated system capacity is easily increased.

Distributed Motion Functionality and Lean Device Design

With CIP Motion, the Distributed Motion Control architecture distributes the controller's Motion Planner function to CIP Motion drives. But that's not all it does – other motion functions may be distributed as well.

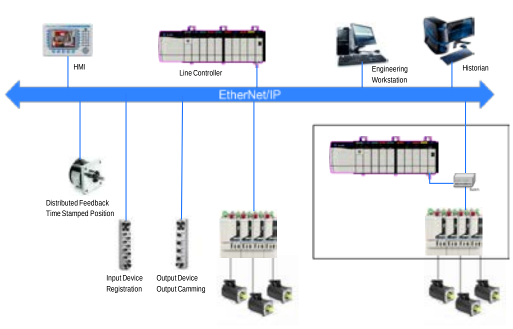

Figure 9: CIP Motion distributes more motion functions, creating new types of devices

The ability to distribute motion functions across CIP Motion devices and controllers addresses a recent trend in lean drive design in which drive vendors are off-loading certain functions of the traditional full-featured servo drive to other devices.

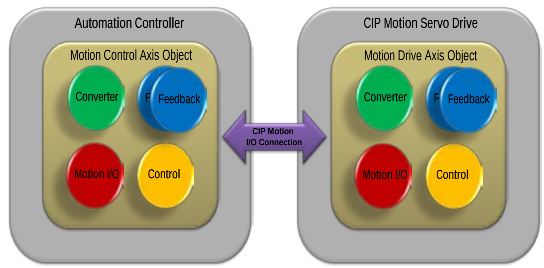

Figure 10: Functional components can be separated from the CIP Motion Drive Axis Object to create a distributed architecture

One function that has been packaged with the drive is the auxiliary feedback port.

Traditionally, this function is embedded in every drive on the chance that it will be needed. But in camming or line-shafting applications, a single encoder on the primary axis can be shared by all through a CIP Motion multicast connection. Since it is so infrequently used, it can be eliminated in future drive designs, replaced when needed by a dedicated standalone CIP Motion Encoder device.

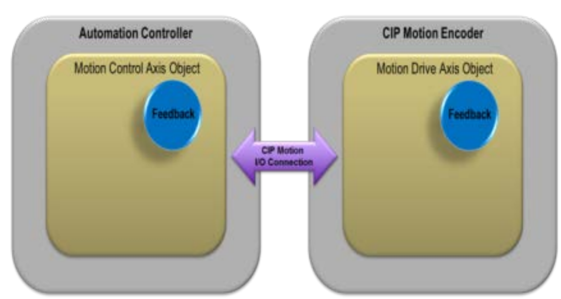

Figure 11: Illustrates the relationship between the Motion Control Axis Object and the Motion Drive Axis Object

In this case, the attributes for the CIP Motion Encoder Device already exist in the Motion Device Axis Object specification and consist of a relatively small subset of axis attributes that are applicable to the feedback function. These attributes are easily identified in the Motion Device Axis Object implementation table under the 'E' Device Control Code.

Instance Attribute			Implementation by Device Control Code					
Attr. ID	Acc. Rule	Attribute Name	E	F	P	V	T	Conditional Implementation
1351	Set	Induction Motor Rotor Leakage Reactance	-	R	R	R	R	Induction Motor only
1352	Set	Induction Motor Rated Slip Speed	-	O	O	O	O	Induction Motor only
1400 + o	Get	Feedback n Catalog Number	O	-	O	O	O	E
1401 + o	Get	Feedback n Serial Number	O	-	O	O	O	E
1402 + o	Get	Feedback n Position	R	-	R	R	R	E
1403 + o	Get	Feedback n Velocity	R	-	R	R	R	E
1404 + o	Get	Feedback n Acceleration	R	-	R	R	R	E
42	Set*	Feedback Mode	R	R	R	R	R	

Figure 12: This extract from the CIP Motion specification shows how attributes are identified in the device. E = CIP Motion Encoder device

Position data produced by the external CIP Motion Encoder can then be consumed by the associated drive via a CIP Motion peer-to-peer connection that provides auxiliary feedback functionality when needed.

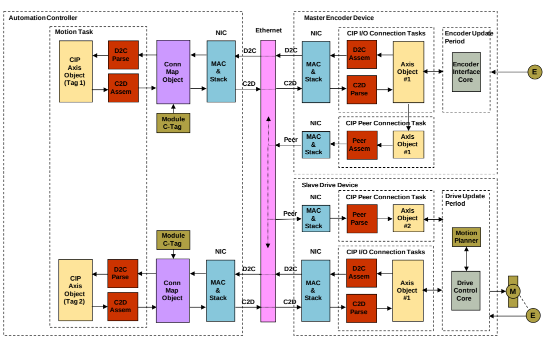

Figure 13: CIP Motion Encoder Device can produce position data to the Peer Connection

Another class of devices created by distributing motion functionality includes CIP Motion I/O devices -- including the CIP Motion Output Cam device and the CIP Motion Registration Input device.

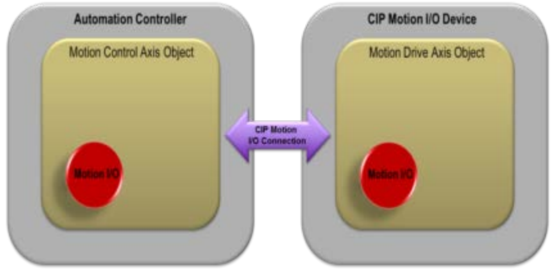

Figure 14: I/O devices are created by distributing motion functionality. This graphic depicts the object relationship model.

Similar to the distributed CIP Motion Encoder, the Motion Axis Object for this new CIP Motion I/O device would consist of a relatively small subset of axis attributes that are applicable to the CIP Motion Axis Object's I/O functionality.

These attributes are identified in the Motion Device Axis Object implementation table under the 'IO' Device Control Code.

Instance Attribute			Implementation by Device Control Code						
Attr. ID	Acc. Rule	Attribute Name	IO	E	F	P	V	T	Conditional Implementation
1434 + o	Set	Feedback n Velocity Filter Bandwidth	-	O	-	O	O	O	
1435 + o	Set	Feedback n Accel Filter Bandwidth	-	O	-	O	O	O	
60	Set*	Event Checking Control	R	R	-	R	O	O	
61	Get	Event Checking Status	R	R	-	R	O	O	
62	Get	Registration 1 Positive Edge Position	O	O	-	R	O	O	
63	Get	Registration 1 Negative Edge Position	O	O	-	R	O	O	
64	Get	Registration 2 Positive Edge Position	O	O	-	O	O	O	
65	Get	Registration 2 Negative Edge Position	O	O	-	O	O	O	
66	Get	Registration 1 Positive Edge Time	O	O	-	R	O	O	
67	Get	Registration 1 Negative Edge Time	O	O	-	R	O	O	
68	Get	Registration 2 Positive Edge Time	O	O	-	O	O	O	
69	Get	Registration 2 Negative Edge Time	O	O	-	O	O	O	

Figure 15: This extract from the CIP Motion specification shows how attributes are identified in the device. IO = CIP Motion I/O attributes.

A CIP Motion Output Cam device would consume real-time position data from a CIP Motion controller or device over the CIP Motion peer-to-peer connection.

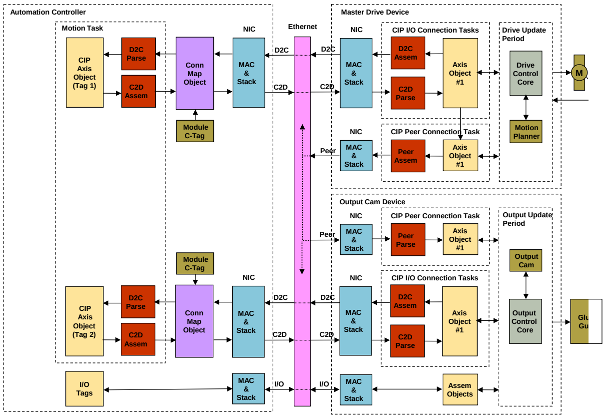

Figure 16: CIP Motion Output Cam device can consume real-time Position data from a controller or device

A CIP Motion Registration Input device would consume real-time position data from a producing CIP Motion controller or device over the CIP Motion peer-to-peer connection.

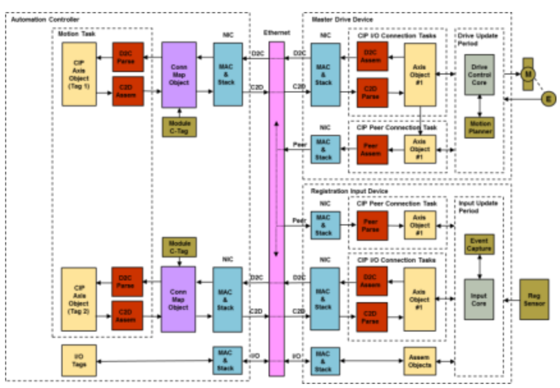

Figure 17: CIP Motion Registration Input device can consume real-time Position data from a CIP Motion controller or device

Converters (power supplies) previously built into every standalone drive are being eliminated in lean drive designs, replaced by a single, cost effective, standalone converter module supplying DC bus power to multiple inverter-only drives.

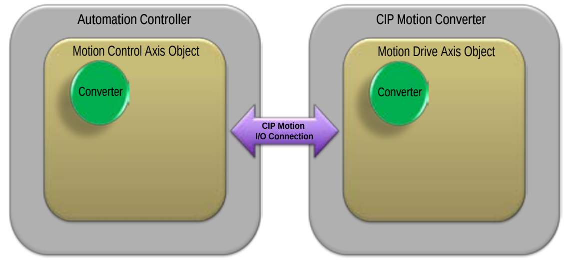

Figure 18: Converters (power supplies) previously built into drives are being replaced by standalone converter modules

The Motion Axis Object for this CIP Motion Converter device would consist of a relatively small subset of axis attributes that are applicable to the converter power supply function. These attributes may be easily identified in the Motion Device Axis Object implementation table under the 'B' Device Control Code.

Instance Attribute			Implementation by Device Control Code						
Attr. ID	Acc. Rule	Attribute Name	B	E	F	P	V	T	Conditional Implementation
614	Set	Mechanical Brake Control	-	-	O	O	O	O	
615	Set	Mechanical Brake Release Delay	-	-	O	O	O	O	
616	Set	Mechanical Brake Engage Delay	-	-	O	O	O	O	
620	Get	DC Bus Voltage	R	-	R	R	R	R	
621	Get	DC Bus Voltage - Nominal	R	-	R	R	R	R	
622	Set	Bus Configuration	O	-	O	O	O	O	
623	Set	Bus Voltage Select	-	-	R	R	R	R	
624	Set	Bus Regulator Action	R	-	R	R	R	R	
625	Set	Regenerative Power Limit	R	-	O	O	O	O	

Figure 19. *This shows how device attributes are identified in the specification. B = Device Control Code*

When the converter function is distributed to a standalone CIP Motion device, it still needs to communicate with the drives it supplies DC bus power to so as to coordinate their operation with the converter's state.

For example, if the converter becomes overheated, it needs to communicate an overload condition to the drives it is supplying power to so they can stop drawing from the DC Bus.

The Distributed Motion architecture provides a solution to this communication problem by allowing the converter to be a Peer Connection producer and associated power consuming drives to be Peer Connection consumers of converter data. In this way, the converter can communicate status information to the associated drives, directly coordinating their behavior with the converter, without controller intervention.

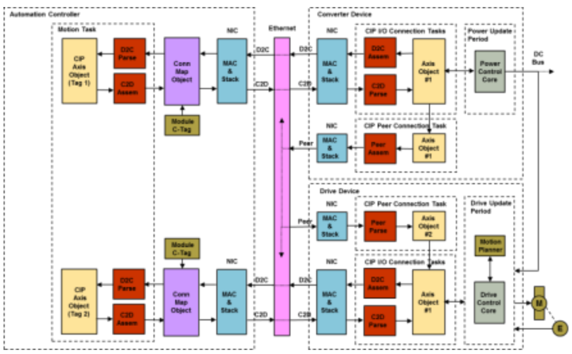

Figure 20: A converter can communicate status information to the associated drives

Conclusion

The CIP Motion Peer Connection enables a highly distributed and modular motion control architecture. This Peer Connection is un-intrusive in the system and easily layered across multi-vendor implementations - creating a unique, unifying real-time solution for electronic line shafting and camming applications. Today, no such solution exists in industry at this level of the network topology. The Peer Connection also allows motion planner execution to move from the controller to the drive, providing a major boost in overall system performance. Finally, the distribution of motion functions in the form of I/O devices, feedback sensors, and standalone converters results in a clean, efficient and modular control architecture that is easily deployed and easily scaled.

In closing, here are a few of the capabilities that the CIP Motion peer-to-peer connection might enable in a motion system:

- Master axis produced /consumed information for distributed camming and gearing synchronization
- Line controller machine motion coordination and synchronization via produced virtual axis
- Coordinated robot operation - e.g. synchronized interpolated move following a common virtual master axis
- Inverter/Converter control synchronization
- Distributed drive I/O resources - auxiliary feedback and registration
- Distributed motion planning

About the author

Steve Zuponcic chairs ODVA's Special Interest Group on Distributed Motion (SIG). Currently the SIG is engaged in working on enhancements to the ODVA specification for the technology described herein. Mr. Zuponcic is employed by ODVA principal and founding member Rockwell Automation, where he is an Applications Engineering Manager.

For more information
Contact ODVA via email at odva@odva.org or visit its web site at www.odva.org.