The Need for a Unifying Motion Control Standard

Peter C. Di Giulio, Pitney Bowes Inc.
Dr. Frank L. Severance, Western Michigan University

March 18, 1993

Abstract

Motion control technology is the backbone of our automation and robotics industries. This article identifies key challenges facing advances in motion control technology with a focus on the impact on domestic industries. Domestic adoption of a unifying motion control system architecture standard would be highly enabling towards accelerating industry growth and prosperity. In addition to enhancing productivity and competitiveness in worldwide markets, it would promote the emergence of high volume motion control based product offerings. Critical success factors are identified for a motion control architecture to be a highly effective and unifying standard. Two relevant communications based motion control architectures are discussed. One is SERCOS, which is currently receiving significant international attention. The other is IPCA, which is a highly versatile, yet effective domestic proposal. An overview of this latter architecture is presented.

Motion Control Industry Challenges

Emerging motion control industries (such as article processing products, factory automation, and robotics) face major challenges. Even though successful, the growth of this industry is heavily constrained by high costs, slow response to market needs and slow reliability improvements. While domestic developments have been largely underfunded and ad hoc, foreign investment in motion control technologies continues to be significantly higher and better coordinated to achieve a much stronger competitive position in the global economy. This is especially true of the European Community (EC) lead by a consortium of German companies. In this highly competitive environment, the life cycle cost pressures on products which employ sophisticated motion control is rapidly increasing.

The motion control industry is still young and applications tend to be highly customized with low production volumes. Vertical markets, whereby value added integrated solutions are reproduced and sold in high volumes, have not yet formed. This industry today carries similarities to the way that most of the Personal Computer (PC) industry was in 1980, where highly customized applications were being developed by consultants for those customers that could justify the heavy investment required. It was only after the (de facto) IBM and Macintosh standards evolved that the plethora of generic software that now exists was achievable.

Investment requirements to facilitate the rapid development of low cost, highly reliable motion control solutions are quite high due to the diversity of technologies employed. Beyond the often complex mechanics to be driven by the motion control system, Table 1 identifies some of the broad set of skills and capitalization required to produce effective motion control solutions.

Table 1: Requirements for Providing Low Life Cycle Cost Products

Development Requirements	**Other Life Cycle Cost Issues**
* System dynamics analysis	* Development Time, Cost, & Test
* Simulation & development tools	* Manufacturing Strategies, Costs, & Test
* System architecture	* VDE/FCC & UL agency compliance
* Data/Control communications	* Electrical Magnetic Compatibility
* Operator interface	* Reliability growth
* Upgrade/Expansion requirements	* Serviceability
* Re-use of modules and designs	* Part volumes
* Motion control software	* Ability to tool
* Analog/Digital hardware	* Part commonality
* Power distribution/Electronics Packaging	

Value of a Unifying Motion Control System Architecture

It is very difficult for individual companies to afford the resources required to develop an effective motion control system platform, extensive support tools, and a high level of expertise in all of the motion control technology areas. As a result, they can not generally achieve high productivity levels and low life cycle costs for their products. In addition, low motion control platform volumes resulting from typical low-volume, high-end applications and the multitude of unique platforms in use to support the needs of the overall market is acting to restrict high level tooling due to return on investment considerations. High level tooling investment is critical to achieving better development tools, lower recurring parts costs, and higher product reliability.

Standardization of a unifying motion control architecture will establish a common motion control technology platform for the development of turnkey motion control products and solutions. The enabling feature of a common platform is that it will provide a mechanism for leveraging the expertise of industry peers by promoting their development of effective application development tools and low cost, reliable "off-the-shelf" motion control building blocks. The architecture standard will act to reduce the cost of motion control intensive products throughout their life cycle. These include lower capitalization requirements, accelerated development, lower recurring product costs, higher product reliability, and more effective service support.

- Capital Requirements: Individual company capitalization requirements will be dramatically reduced since many of the components needed to build motion control solutions will be available as "off-the-shelf" modules. This translates into a reduction in development and tooling as these re-usable modules will incorporate tooling investments. Moreover, it is highly likely that manufacturing and service strategies supporting the common architecture will also be reusable and improve over time. The significant benefit is that companies can focus their investments on those development activities where they add value (i.e. In the use of platform modules to create application solutions or development of additional modules).

- Development Time: A unifying motion control architecture standard will greatly accelerate product development. The key drivers to providing a shorter, more predictable response to market needs are the (1) adoption of an effective existing system architecture, (2) availability of proven "Off-The-Shelf" modules (hardware and software assemblies and designs), (3) availability and continuous improvement of development tools tailored to the common architecture, and (4) opportunity to re-use manufacturing and service strategies and equipment.

- Product Cost: The higher part volumes resulting from a broadly accepted control architecture will greatly reduce recurring product costs by making it economical to employ high tooling levels. Note that this opportunity will be greatly enhanced to the extent that the architecture promotes modularity and part commonality within each motion control based product.

- Reliability: Product reliability is a major benefit of a maturing technology platform. Robust development tools, high level tooling, re-use of proven modules, and continuous improvement will all act to evolve the reliability of the platform building blocks and ease their integration.

- Service Support: Intrinsic product reliability will minimize service requirements. In addition, service effectiveness and productivity will be enhanced by the re-use of consistent service strategies among many products and the reduction in the number of unique parts.

The overall impact of an effective architecture standard will be to expand motion control market opportunities worldwide. Accelerated development of turnkey systems will facilitate more rapid penetration into existing markets. Lower system costs will act to promote higher product volumes and enable new product concepts that become feasible by virtue of the lower costs. Ultimately, domestic adoption of a motion control architecture standard will lead to our greater competitiveness and participation in the global marketplace.

Movement towards industry standardization will yield greatest benefit for motion control end-users and system integrators. System integrators can more rapidly provide effective, lower cost, reliable solutions to low volume end-users. High volume OEM users can achieve lower overall life cycle cost product offerings.

Existing motion control hardware and software vendors are the most challenged. Their profitability and future prosperity depends on their (1) continually tuning into evolving trends in the industry and (2) overcoming the inertia of their existing investments to implement the changes necessary to best position themselves relative to emerging economic opportunities. Note that vendors that do not join the movement towards more productive industry solutions and standards will probably not be at risk until after 1995.

The significant benefit offered to these vendors by a unifying motion control architecture is that they will not have to carry expertise in all motion control skill requirement areas, but can leverage their peer's expertise and efforts. Focusing on what their expertise is, they can add their own unique value to the evolving set of unified architecture offerings and achieve a much larger market for their products. It is highly likely that their bottom line profit will achieve a net gain as a result of growing industry prosperity.

Critical Success Factors for a Motion Control Architecture to be Unifying

A motion control architecture technology platform should contain the following features to be an effective, unifying standard :

- <u>Building block oriented</u>: The architecture should be fairly modular so that robust turnkey solutions can be rapidly developed using "Off-The-Shelf" architecture modules in a building block manner. Custom modules developed should be reusable as additional building blocks for future development.

- <u>Provide high performance at low cost</u>: The architecture should carry an effective performance versus cost curve. The system cost should be fairly low for low performance requirements. In addition, an incremental increase in the system performance requirement should result in a predictable incremental increase in cost (not a major jump). The architecture should carry an intrinsic upgrade philosophy to support high end system performance requirements.

- <u>Support a broad range of motion control applications</u>: The architecture should support a broad class of motion control requirements. It should be able to facilitate control of various types of motors, solenoids, and sensors. In addition, the architecture should offer a natural support for those functions which are integral to motion control based high volume product offerings. This includes user interface functions, external product communications, and software upgrade means.

- <u>Support both centralized and distributed control schemes</u>: The architecture should be flexible to allow centralized and/or distributed control to be employed to achieve a particular system's requirements. The selection of control scheme is driven by the needs of the system to be controlled and is strongly influenced by the inclination of the system designer. Both schemes have their advantages. Neither scheme taken alone will accommodate all applications.

- Provide effective development, manufacturing and service tools: Architecture ease of use will critically depend on the availability of effective development tools. Its effectiveness in creating highly manufacturable and serviceable products will strongly depend on the extent to which diagnostic and test capabilities are an integral part of the overall architecture concept.

- Carry intrinsic support to accommodate technology trends: For the architecture to have a life of greater than 10 years, it should be fairly technology independent. The architecture definition should not be tied to a specific CPU or communication media although it is highly likely that media solution(s) will emerge to provide system cost and reliability advantages while facilitating FCC/VDE & UL agency compliance. The life of the media solution will probably be less than 5-to-7 years due to media technology advances. The architecture should benefit from technology trends towards higher levels of electronics integration and provide natural support for new control techniques (i.e. DSP technology, Neural Networks, & Fuzzy Logic).

Pitney Bowes Intra-Product Communication Architecture Features

The Intra-Product Communication Architecture (IPCA), a concept evolving at Pitney Bowes, is a modular, high performance, motion control architecture. This communications based architecture offers cost effective networking of system data and control in a manner which is optimized for motion control applications. At the heart of the IPCA is a link level protocol which imposes very little restrictions to system application hardware and software. Key features of the IPCA are :

- Building block oriented: Any communications based architecture is intrinsically building block oriented due to the standard interface requirements for connecting to the communication network. In addition, IPCA offers an integrated development environment whereby "Off-The-Shelf" modules can be quickly integrated to test prototype system functionality.

- Provides high performance at low cost: The architecture offers a centralized control scheme whereby low cost, Smart I/O modules can be used to interface to peripheral devices (motors, solenoids, sensors, etc.). Where more sophisticated processing is required, a Smart I/O module can be upgraded to a CPU based module and system processing requirements would become more distributed.

- Supports a broad range of motion control applications: The IPCA is based on a hybrid centralized and distributed control scheme as illustrated in Figure 1. It is expandable to support any motion control requirements with the appropriate selection of Smart I/O modules and/or distributed processing modules. Multiple network systems are also supported.

- Supports both centralized and distributed control schemes: The IPCA is the only known architecture which will simultaneously support centralized and distributed control schemes at low cost. This is the key unifying feature of the IPCA.
- Provides effective development, manufacturing and service tools: Simulation tools have been developed to analyze IPCA performance for a given set of system requirements and a specification has been drafted for a network monitor to provide test support during system integration. Moreover, the IPCA incorporates integral manufacturing, service, and built-in test features which will allow system digital circuitry to be diagnosed down to the IC or board short/open failure from a single product connection.
- Carries intrinsic support to accommodate technology trends: The IPCA is completely processor and media independent although early media and power solutions are being investigated. It has been designed to take advantage of high level electronic integration trends. Employing IPCA communications with the centralized control scheme provides a natural Neural Network and Fuzzy Logic environment.

The IPCA is designed using "Very high scale integrated circuit Hardware Language" (VHDL) technology to provide workstation and silicon vendor independence. Much of the IPCA communications protocol is implemented in hardware to (1) provide high performance, (2) facilitate Application Specific Integrated Circuit (ASIC) integration with other functions, and (3) achieve independence of the application CPU and software.

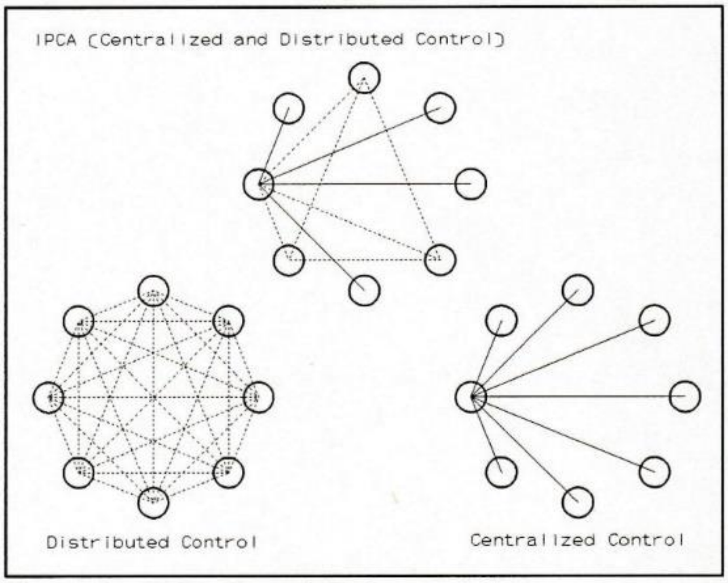

Figure 1: IPCA Control Options

What About SERCOS?

SERCOS (Serial, Real Time Communication System) is the result of a joint development between the VDW, the German Machine Tool Manufacturers Association, and the ZVEL, the German electrical engineering group. Based on a serial communications network using plastic fiber optics, SERCOS is a reliable reduced wiring alternative to employing numerous parallel wires to connect a system's central controller to peripheral devices (motors, solenoids, sensors, etc.). SERCOS is a "centralized-only" control scheme which communicates low-level motion control information at high data rates between a central controller and the peripheral devices. In typical applications, the central controller will send position information to the local peripheral device controllers on the network and these local controllers will return sensor information. SERCOS can be fairly effective for motion control applications where a centralized-only control scheme makes sense and where the cost pressures on the system are not great. However, there are two key deficiencies that prevent SERCOS from providing an effective, long term, universal control system for all automation and robotics applications.

The most significant deficiency is that SERCOS provides no intrinsic support for "distributed" or "mixed centralized and distributed" control requirements. In a distributed control environment, the system control function is shared among a number of intelligent modules connected to the communications network. These intelligent modules offer sophisticated control of local peripherals and communicate with other intelligent modules in a peer-to-peer fashion on an event driven basis. For high performance motion control applications, it is critical that, starting from the time that one of these intelligent modules wants to send a control message to another intelligent module, the time it takes for that message to reach the destination module (message latency time) be predictable and extremely short. SERCOS's communications protocol was not designed to support this requirement.

The second deficiency of SERCOS is that it is too expensive for high volume applications where a centralized-only control scheme is acceptable. Although centralized control allows the many peripheral control nodes of the system to be greatly simplified (and cost reduced) in favor of an increase in the single central controller cost, known SERCOS implementations do not fully exploit this opportunity. In typical SERCOS systems, low-level position commands are sent from the central controller to the peripheral control nodes and these peripheral controllers are responsible for getting the peripheral (i.e. a motor) to follow the position commands. In high performance systems, this requires that fairly sophisticated and costly CPUs be employed to implement these peripheral controller functions. In many applications, the opportunity will exist to eliminate CPUs in peripheral control nodes by passing lower level control data directly to the peripherals from the central controller. The other cost issues currently presented by SERCOS is that plastic optical fiber technology is still expensive for high volume, competitive product applications. Also, it is far from obvious as to how fiber technology can be cost effectively used in an integrated power and signal distribution technology which will likely be part of the evolution towards achieving lower motion control system costs into the future.

IPCA Architecture Overview

The Intra-Product Communication Architecture (IPCA) Topology is illustrated in Figure 2. The architecture supports three types of communication nodes. They are a Central Control Node (CCN), a Smart I/O Peripheral Control Node (PCN), and a Distributed Control Node (DCN). The protocol definition allows most protocol functions to be implemented cost effectively in hardware.

The CCN provides the network controller function and, if PCNs are present, performs motion control functions as well. The key network control functions are initializing the network and controlling DCN and PCN access to the network in a manner which avoids contention.

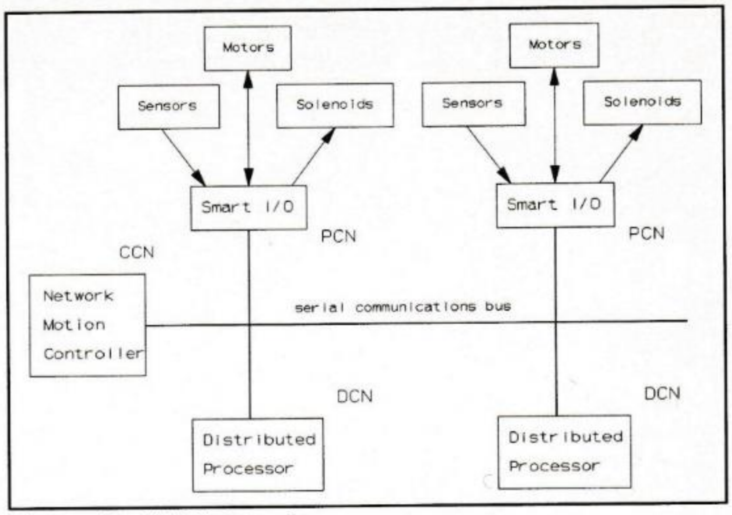

Figure 2: IPCA Topology

PCNs are Smart I/O nodes which provide a moderately sophisticated interface to motors, solenoids, sensors, status indicators, etc. PCNs generally do not include independent CPU environments but act only under control of the CCN. PCNs send sensor status messages to the CCN in response to CCN control messages directed at them. They can not send unsolicited messages on an event driven basis. As mentioned previously, the CCN provides a centralized motion control function for PCN Smart I/O nodes. In a manner similar to SERCOS, the CCN communicates low-level data to PCN motor/sensor controllers with a high refresh rate. Unlike SERCOS, this rate need not be the same for all PCN nodes, but can vary in accordance with scheduling needs of the CCN.

DCNs are usually CPU based nodes which can share the system control function in a distributed fashion. The IPCA architecture supports peer-to-peer communication amongst these nodes on an event driven basis. The CCN defines DCNs' access to the network and can communicate with them. Note that a DCN's application is unrestricted and can take on a broad range of functions such as a user interface, an intelligent peripheral controller, a system controller, a communications interface, and independent control of Motors, Sensors, and Solenoids.

Each physical network will support 1 CCN and a combination of up to 31 DCNs and PCNs. This is more than sufficient to support a product's local motion control environment. For large systems, multiple physical networks can be employed to achieve the overall system requirement.

IPCA Protocol Overview

The IPCA protocol is outlined schematically in Figure 3. A basic communication "TIC PERIOD" is defined to support the IPCA's sophisticated communication requirements. Each Tic Period is broken down into three phases; these are (1) the CCN Sync Message, (2) PCN Communications, and (3) DCN Communications. Each of these three phases are tightly coordinated by the CCN which contains the Tic Period timing function. The Tic Period time is user configurable and will typically be set between 0.5 and 2 msec.

The first phase consists of the CCN broadcasting a "Synchronization Message" to all of the DCNs and PCNs at the start of the Tic Period. This message, which can be thought of as the network "heart beat", is used to (1) synchronize PCN control/sensor activities, (2) provide a

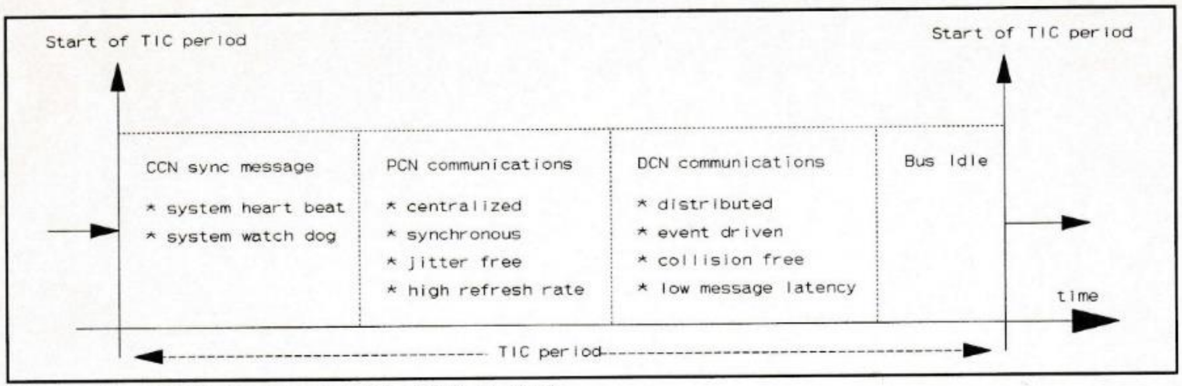

Figure 3: IPCA Protocol Basic TIC Period

slow clock to be used for various node timing functions, and (3) act as a system watchdog function. In this last case, DCN and PCN application functions can switch to safe state if a network failure prevents CCN Sync Messages from being communicated.

The second phase is reserved for CCN-to-PCN communications. This immediately follows the CCN Sync Message to facilitate the strict timing requirements for PCN communications. Here, the CCN communicates with selected PCNs at a high refresh rate and in a manner which provides very predictable and jitter free timing. A Cyclic Redundancy Check (CRC) is used to detect communication errors. If errors are detected, the message is ignored. Since this data is updated frequently, no retry is performed.

The time following PCN communications and until the start of the next Tic Period is reserved for DCN communications. Predicated on enable messages from the CCN, unsolicited DCN messages are sent directly between DCNs. The DCN access scheme is prioritized to eliminate the possibility of contention. A CRC and immediate acknowledge are used to detect communication errors. If a communication error occurs, a retry will be performed and a sequence bit is employed to ignore duplicate messages. More than one DCN message can be sent within a Tic Period. When the CCN determines that there is not enough time remaining in the Tic Period for another message to be sent, it will stop issuing the enable message. As shown in Figure 3, the bus will then be idle until the start of the next Tic Period.

Interfacing

The IPCA provides an efficient means of interfacing to motors, solenoids, sensors and indicators. By partitioning the CCN/PCN, communication and motion control requirements PCN circuitry requirements are greatly simplified. This is done by (1) Requiring that complex motion control decisions be located in the CCN, (2) Eliminating the need for a CPU environment with software download facilities in PCNs, and (3) Using the high PCN communication refresh rates to eliminate PCN communication error recovery requirements (PCNs have no retry capabilities). Figure 4 provides a block diagram for a typical low cost PCN resulting from this reduction in circuitry requirements.

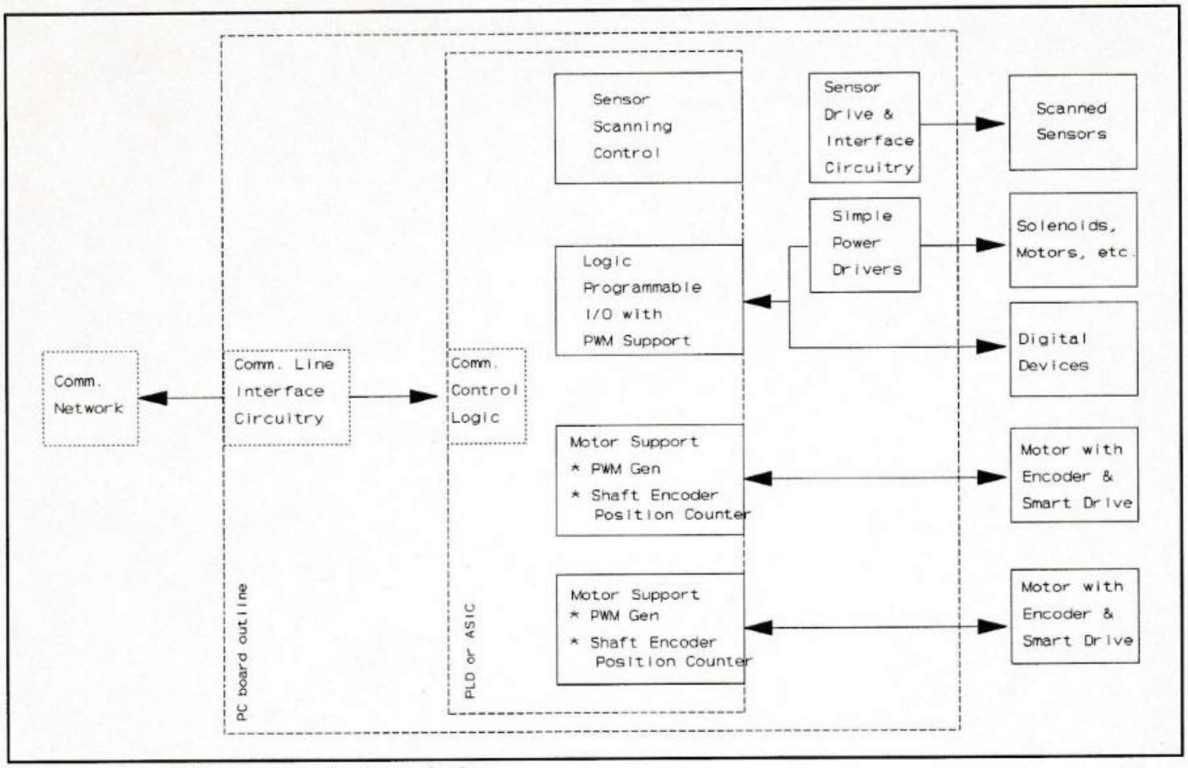

Figure 4: IPCA Smart I/O Interfacing

As shown in Figure 4, the PCN minimally requires communication interface and support circuitry, Smart I/O functions, and analog driver and interface circuitry. A PLD or small ASIC can be employed to implement the PCN's streamlined communication logic and Smart I/O logic requirements. Since the Smart I/O requirements are optimized to the application's requirements, all of the defined logic device functions are used to capacity.

At power up, the IPCA network data rate is set to 625 KBaud. During configuration set-up, the network data rate can be set as high as 10 MBaud, depending on the performance needs of the system and media bandwidth constraints. At 10 MBaud, the IPCA will support a 1 msec average data refresh for 31 PCNs. Where there is a mix of DCN and PCN nodes, the DCN maximum message latency will be less than 5 msec assuming that 30% of the network bandwidth is allocated to DCN communications. More typically, DCN maximum message latency will be less than 1-to-2 msec and any one of the DCNs can dynamically be assigned to the highest priority which would guarantee its messages a latency of less than 1 msec.

Flexibility has been provided for the IPCA protocol to be implemented in a multidrop or ring topology and with few restrictions on the selection of the communications media. This facilitates the media selection process in minimizing the IPCA's cost/performance.

The critical media parameter required to adjust to different communication media is the *Network Response Time*. This is the time for data to reach all of the nodes on the network and return to the sender. The Network Response Time is affected by (1) the communications data rate, (2) media delay time, (3) media interface circuitry response time, (4) clock resynchronization timing requirements, and (5) the IPCA communication logic response time (which is usually small since real time communication logic is implemented in hardware).

Board level Boundary Scan Logic Testing [1] is an emerging technology alternative to "Bed of Nails" testing for digital circuitry. Boundary Scan Logic allows test equipment to examine an electronics assembly through a single board connector and is used to detect board shorts, open circuits, and failed IC components. The percentage of digital IC components that conform to this standard has been growing at a dramatic rate.

The IPCA builds on this boundary scan technology by replacing this single board connector with a connection into the IPCA network. As a result, electronic assembly diagnostics can be conducted from a single communications network connect point to the system. Before and/or after product assembly, manufacturing test equipment can be used to diagnose system electronics to the failed IC level, thereby greatly reducing manufacturing costs. A simpler version of this test equipment can be used by customer service to identify failed boards in the field much faster and more accurately than can be achieved today. Boundary scan capabilities can be added to an IPCA node (probably the CCN) to provide built-in system test capabilities.

Support for the IPCA Architecture

SERCOS is a European idea created in an attempt to standardize communication to peripherals across machine tools. It was never designed for to support a distributed motion control environment. On the other hand, the IPCA architecture is versatile and lends itself to low cost motion control solutions. This lends support to manufacturers of motion products other than the limited subset used in industrial automation. By adopting such a common base, motion companies will surely enhance the ability of Original Equipment Manufacturers to produce high volume, low cost solutions, thus enhancing the industry over-all.

Efforts to stimulate interest using the IPCA approach are currently being pursued by several groups. Most obvious is the user community, which is organized as the American Institute of Motion Engineers (AIME), headquartered in Kalamazoo, Michigan. This group is actively looking to establish common solutions. While they have not yet specified IPCA, there is a degree of favorable consensus toward this approach and efforts to establish the IPCA as the architecture of choice continue.

Nor is the IPCA simply an industrial idea. Western Michigan University, also in Kalamazoo, Michigan, is doing and advocating more research on the standard. Currently this is in the form of graduate students working on formal specifications. These specifications are being written in State Architecture Notation (SAN), which is a state-based specification language which is hardware independent. Even though this effort has only just begun, it is hoped that these specifications will eventually lead to straight-forward and reliable promulgation of the IPCA architecture.

Conclusion

Traditionally the motion control industry has focused on low volume, custom applications. With the high mark-up afforded by these value added applications, there has been little incentive to bring down motion control costs. As a result, existing motion control technology costs have prohibited its use in high volume, commercial product offerings. The IPCA presents an excellent opportunity to reduce motion control costs in existing applications. Moreover, its requirements have been geared to enable high volume, commercial product offerings to appear leading to a broader and stronger motion control industry. Longer term, it is anticipated that the emergence of inexpensive motor technology products based on more modular control architectures and the maturing of artificially intelligent technologies (such as Fuzzy Logic Systems and Neural Networks) will precipitate a renaissance in the domestic robotics industry.

The authors welcome your interest in the IPCA and would appreciate proposals for the commercialization and proliferation of the technology by way of the standardization and development of motion control products (such as boards, tools, and chip sets). Contact Peter C. Di Giulio at (203) 924-3109 with your questions and comments about the IPCA technology and for a reprint of this paper. Dr. Frank L. Severance can be reached at (616) 387-4068 regarding Western Michigan University motion control and IPCA initiatives. A more complete description of the IPCA [2] was presented at the Motion Expo East'92 in Boston.

Acknowledgments

The authors would like to thank David Lee, David Riley, and Frederick W. Ryan for their valuable contributions to the IPCA development and William Berson, John Gomes, Ed Cornell, Paul Reece, Anne Pol, and Fred Z. Sitkins for their ardent support.

References

1. *IEEE Standard 1149.1 - 1990*, "IEEE Standard Test Access Port and Boundary-Scan Architecture", IEEE Inc., 345 East 47th Street, NY, NY, 10017.

2. *A Low Cost Motion Control Product Architecture* by Peter C. DiGiulio, Proceedings of the Motion Expo East'92, Boston Massachusetts, April 1992.