



## **PLCopen Motion Control: Reducing development time and cost with standardized motion programming.**

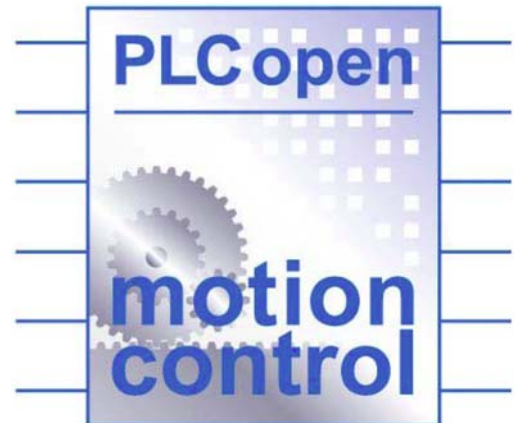
### **Abstract**

As industrial automation systems become increasingly complex, software development time and engineering expenses can exceed all other development hurdles. PLCopen offers standard, open motion control function blocks based on the IEC 61131 languages to create efficient, flexible code that is vendor- and product-independent. PLCopen Motion Control reduces software development time while maintaining a portable and easily upgradeable code base for future applications. The result is a shorter time to market with greater profits and future growth that is independent of the hardware platform.

### **Introduction**

Motion control is becoming increasingly complex in order to meet market demands for greater system throughput and improved output quality and precision. These objectives must be accomplished while decreasing overall machine costs. Conventional system designs must be replaced by innovative, high performance solutions.

Market demands, of course, start at the consumer level. In order to maintain and grow their consumer base, producers must continuously update their products, packaging, and process while reducing costs. In order to accomplish this, producers want machine builders to deliver smaller footprints, higher speeds, faster changeovers, reduced wastes, and improved reliability. Control suppliers must offer platforms capable of meeting these needs.



Motion control technology is often the target area to tackle these requirements. This area represents a high proportion of the overall machine cost and development and presents the greatest potential for performance and savings benefits. The PLCopen Motion Control Function Blocks support the user by providing vendor- and platform-independence and reducing the overall development time. This is especially true for machine builders in the packaging industry. A packaging plant may include machines for bagging, wrapping, bottling, labeling, weighing, inspecting, and palletizing just to name a few. In this industry, manufacturers must quickly react to consumer needs. This, of course, requires flexible machine designs, or existing equipment will become obsolete.

For example, a food and beverage company may need to quickly change a package concept to meet marketing demands. A larger product package and volume is more visible to consumers than simply applying a discounted price. However, this requires changes to both packaging materials and dispensing systems. This is a major production change. A strictly mechanical system would require a shut down of the production line and many hours, or even days, of change over. A mechatronic solution with flexible software interface can be changed over much faster. This translates into a quicker marketing response, decreased costs, and increased profits.



### **Why Standardize?**

Before the introduction of PLCopen and the IEC 61131 standard, the motion control industry was very fragmented. There were a variety of systems available, each with their own proprietary technology, languages, communication, and development tools. Machine builders often deliver multiple versions of a machine including a low-cost version, a medium-level machine, and a high performance solution. In addition to that, machine builders often have to release different brands of controllers for different regions.

The PLCopen Motion Task Force set out to create a library of function blocks which acts as a standard motion interface regardless of the architecture. This standard interface can be used across many different systems. This means applications can be developed independent of the platform, so the machine builder can more easily change architectures or support multiple platforms. Builders can wait until after the application stage of a project to finalize their choice of hardware. Maintenance and training costs are also greatly reduced.

The PLCopen Motion Control Function Blocks satisfy a long-awaited demand for a standardized programming method to control positioning tasks quickly, easily, and efficiently. These function blocks can be programmed in the IEC 61131 languages such as Ladder Diagram and Structured Text. Because these function blocks are hardware independent, they can be used to program many different types of motors, drives, feedback systems and even different topologies such as centralized or decentralized solutions. This hardware independence and programming flexibility also allows users to select hardware based on the requirements of the application instead of limiting their decision simply because of previous experience or level of training. The optimum hardware solution can be configured for the application at hand.

Standardizing on a motion interface also makes education much easier and more efficient. It is a great benefit to be able to train future programmers without having to directly refer to specific hardware or programming tools. Once trained, engineers can apply their knowledge to a broader range of motion control products.

### **Importance of Software**

According to the Managing Director of the PLCopen group, three decades ago machinery required almost no software, whereas now software development accounts for about half of the total cost of a production line. Flexible motion control technology paired with an open, standard industrial software interface can greatly reduce the complexity of all types of industrial machinery. The fragmented nature of the motion control industry generates considerable expenses by preventing the reuse of software, increasing engineering time, making different architectures incompatible, and limiting new developments. Motion control software standardization, along with IEC 61131 standard languages, addresses these issues. Standardized software interfaces mean:

- Greater reusability of software
- Less dependence on hardware
- Faster time to market
- Decreased time and costs associated with installation and maintenance
- Wider acceptance across the industry
- Reduced training time and costs



### **Architecture Independence**

The platform independent objective of PLCopen is a key feature that machine developers can leverage. For example, many developers are moving away from a centralized motion control approach because they recognize advantages of a decentralized solution. Centralized designs require an expensive motion controller to handle the hefty processing load of multi-axis systems. Even then, large axis counts may not be possible because of limitations of the controller and communication network. Decentralized control platforms with intelligent drives remove much of the load from the main motion controller and reduce the network limitations by closing their own positioning loops. This eliminates the need for an expensive motion controller. Larger numbers of synchronized axes are possible on a single network that provides a high speed, scalable motion solution. Motion control performance is not dependent on the processor performance or the number of axes being run on a network.

In order for machine builders to realize these performance and cost benefits, they must invest in software redevelopment and training. However, with standardized motion control interfaces across multiple platforms, machine builders can minimize this development time and cost. The intention of PLCopen is to allow developers to move from one platform to another as technology becomes available. The PLCopen Function Blocks operate independently of the underlying architecture. Machine builders can continue to develop their machines without fear of obsolescence or extraneous engineering costs.

Ethernet based motion control networks are another example of a technology machine builders can take advantage of. With its peer-to-peer communication capabilities, high communication bandwidth, and standard hardware availability, Ethernet is the perfect media to transfer large amounts of information within a minimum amount of time. And now, with the real-time, deterministic capabilities of networks such as Ethernet Powerlink, Ethernet is ideal for high performance motion applications. Machine builders have the ability to upgrade from existing architectures to a real-time Ethernet-based communication without wasting consuming time and money in development.

### **PLCopen Motion Control Function Blocks**

This standardization is accomplished through the efforts of the PLCopen organization. PLCopen was established in 1992 to harmonize the industrial control market across different platforms during development, installation, and maintenance in accordance with the IEC 61131 programming environment. Current topics include motion control, safety functionality, XML data exchange, and benchmarking standards. PLCopen also continues to promote the use and training of this standard around the world since worldwide acceptance is crucial to its open, independent goals. In order to produce a standard motion control specification, a set of reusable, hardware-independent control components has been defined based on the IEC 61131 function blocks. With this standard, application software can be reused. This is possible even across multiple platforms. Training and support costs are reduced as well. New developments can be implemented much more easily with shorter time to market.

In defining their motion control function blocks, the PLCopen Task Force wanted to achieve the following goals:

- Simple design allowing easy programming, installation, and maintenance
- Efficiency in the number of function blocks that are created
- Consistent protocol which conforms to the well-known IEC 61131 language standard
- Universal adaptation which allows hardware independent implementations
- Flexibility in the range of these function blocks and the future expansion
- Sufficient coverage of most motion control applications



These goals were achieved with a basic set of function blocks which allow both single axis motion and synchronized multiple axis motion. A state machine describes the behavior of an axis during a machine's sequence.

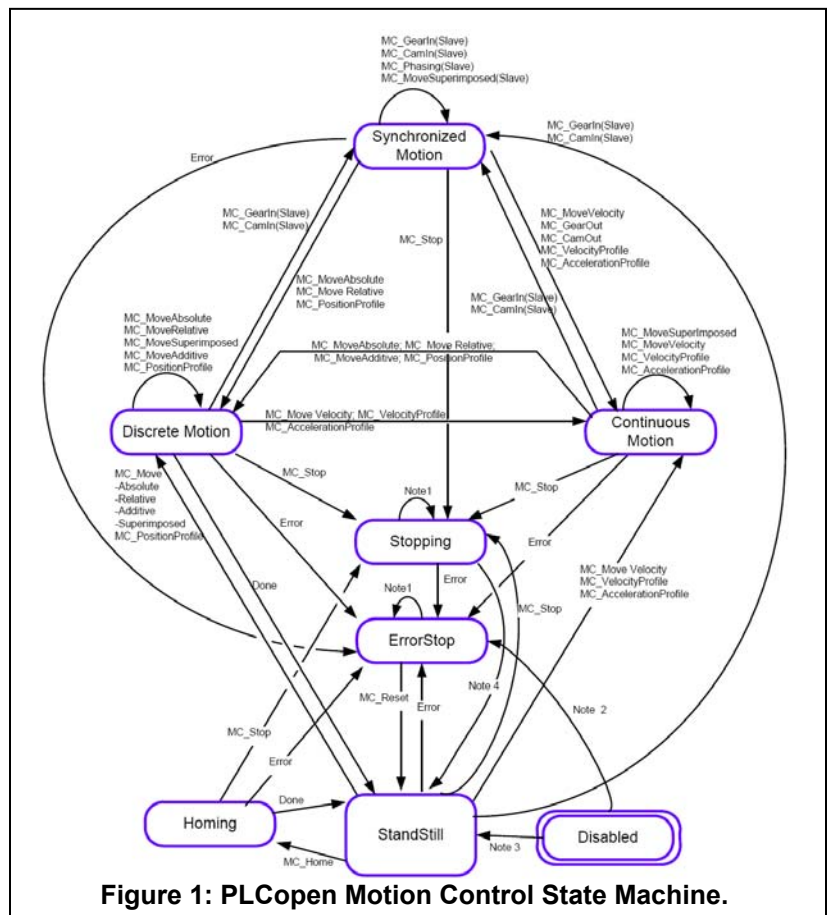
**Table 1: The Basic PLCopen Function Blocks for Motion Control.**

<i>Administrative Function Blocks</i>		<i>Motion Function Blocks</i>	
<i>Single Axis</i>	<i>Multiple Axis</i>	<i>Single Axis</i>	<i>Multiple Axis</i>
Power	CamTableSelect	MoveAbsolute	CamIn
ReadStatus		MoveRelative	CamOut
ReadAxisError		MoveAdditive	GearIn
ReadParameter		MoveSuperimposed	GearOut
ReadBoolParameter		MoveVelocity	Phasing
WriteParameter		Home	
WriteBoolParameter		Stop	
ReadActualPosition		PositionProfile	
Reset		VelocityProfile	
PassiveHome		AccelerationProfile	

The code behind these function blocks is hidden from the user and is the responsibility of the control software supplier. This code is dependent on the hardware and architecture of the system. The interface, which includes the inputs and outputs of these blocks, stays consistent over any platform. This differentiation of software levels is key to PLCopen's functionality. The motion of an application can be programmed in the same manor regardless of the hardware because the lower-level code is hidden from the user. The user does not have to have detailed knowledge of a drive or network architecture. The programming of an axis of motion is very logical. The axis needs to be powered, homed, and moved. In three steps, an axis is moving or even synchronized with another axis.

**Expandability of PLCopen**

After the task force outlined a basic set of function blocks to cover single and multiple axis control, it became clear that additional functionality was needed to reach a broader range of motion control systems, so the PLCopen Motion Control Extensions were developed. Beyond this, advanced function blocks can be created by machine builders and control suppliers as a supplement to the standard function blocks. These reusable libraries take the PLCopen standard even further. The PLCopen User Guideline provides details and examples on how these user specific libraries can be created



**Figure 1: PLCopen Motion Control State Machine.**



based on standard PLCopen function blocks and according to the same specifications. Machine builders can then apply and reuse the function blocks they have created for their area of competence. This task force is also working to expand programming further with interpolation of coordinated multi-axis motion in 3-dimensional space and extensions to the existing homing routines.

**Table 2: The Extended Function Blocks for PLCopen.**

<i>Administrative Function Blocks</i>		<i>Motion Function Blocks</i>	
<i>Single Axis</i>	<i>Multiple Axis</i>	<i>Single Axis</i>	<i>Multiple Axis</i>
TouchProbe		TorqueControl	GearInPos
AbortTrigger		MoveContinuous	
ReadDigitalInput		Halt	
ReadDigitalOutput			
WriteDigitalOutput			
SetPosition			
SetOverride			
ReadActualVelocity			
ReadActualTorque			
DigitalCamSwitch			

One example of a function block that a user might develop is a registration command. There is no specific registration function block defined in the PLCopen command set because registration operations can vary greatly from one application to the next. A standard version would not be effective. Instead, user's can create a registration function that matches their exact needs using existing function blocks. The TouchProbe function block was designed for this reason. It can be combined with some other motion function, such as the Phasing function block, to satisfy the registration task. Once this new registration function has been created, the user can reuse it on other axes or even other platforms. In this way, machine developers can build their own reusable, hardware-independent libraries.

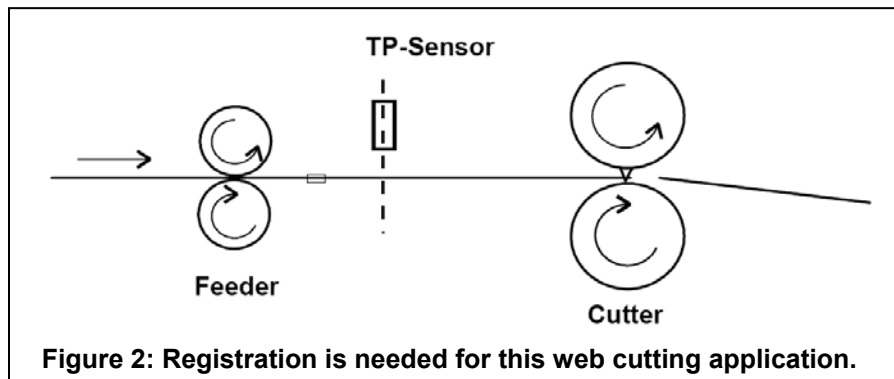


Figure 2: Registration is needed for this web cutting application.



In the example shown, a registration function is created to cut a web material based on a print mark. The PLCopen extended function block TouchProbe provides a very fast recording of an axis position at a trigger event, in this case, a print mark on the web material. The offset of the trigger position is passed to the Phasing function which advances or delays the cutter axis in regard to the feeder.

### Integrated Safety

Another development area that goes hand-in-hand with motion control is safety. In parallel with their motion control definitions, PLCopen is working to develop standardized safety functionality that can be integrated into logic and motion applications. The ability to incorporate these safety standards will achieve the same benefits as the motion control function blocks: greater reusability and portability, reduced engineering and training time, lower development and maintenance costs.

### The PLCopen Safety

standards also reduce certification time and costs. To accomplish this, PLCopen has defined programming guidelines, safety related datatypes, error handling and diagnostic concepts, representation of the software architecture, and certification guidelines. This combination of standardized motion and safety features provides users with a more complete solution from a single programming environment that is portable across multiple platforms.

### Conclusion

The work of PLCopen will continue to improve and unify the automation industry along with the cooperation of other organizations such as the OMAC Workgroup for Packaging Motion Control. The requirement for higher performance and more flexibility will further increase the importance of software in motion control applications. The growth of software intensive applications and mechatronic solutions will require consistency across varying platforms to ensure the success and future potential of machine builders. With this trend, PLCopen will continue to grow in the automation industry. The Function Blocks for Motion Control will create more efficient machine designs that are not dependent on specific vendor and product platforms. PLCopen Motion Control will bring greater efficiency and lower costs during development, maintenance, and training and will protect the software investments of machine builders to ensure future growth.

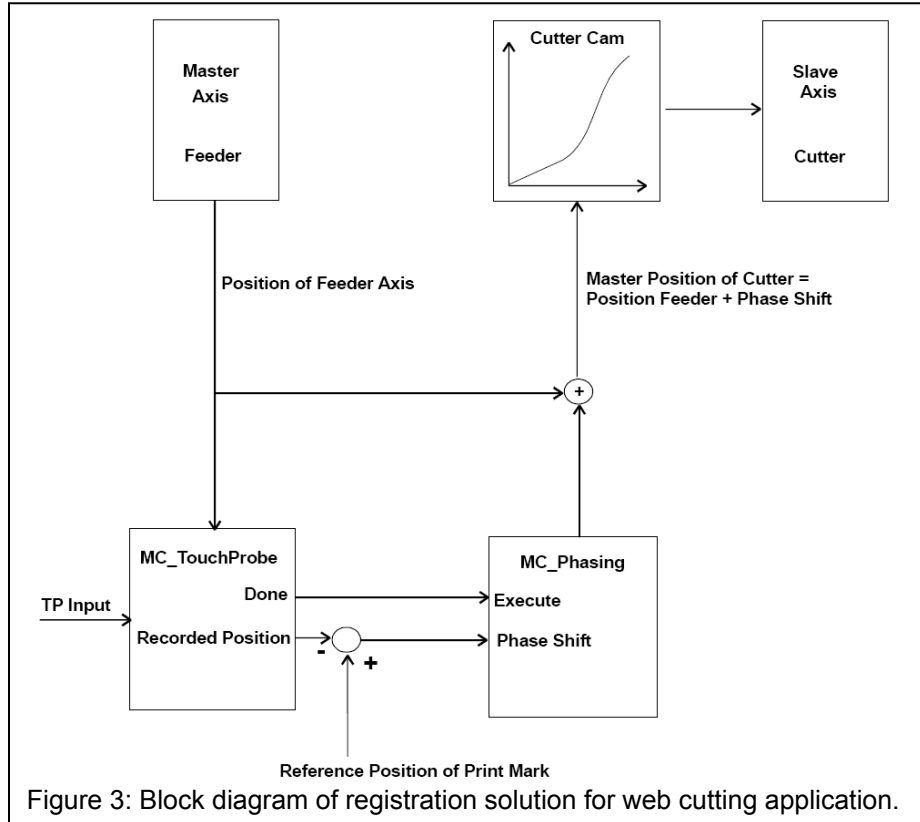


Figure 3: Block diagram of registration solution for web cutting application.



Perfection in Automation  
[www.br-automation.com](http://www.br-automation.com)



Nathan Massey  
Sales Engineer  
B&R Industrial Automation Corp.

For more information on PLCopen visit:  
[www.plcopen.org](http://www.plcopen.org)

For more information on B&R contact:  
B&R Industrial Automation Corp.  
1325 Northmeadow Parkway, S-130  
Roswell, GA 30076  
Phone: (770) 772-0400  
Website: [www.br-automation.com](http://www.br-automation.com)  
Email: [sales.us@br-automation.com](mailto:sales.us@br-automation.com)