

# PLC vs. PAC

These technologies continue to evolve, making differences harder to distinguish. Here are some thoughts on what does what, and how to choose between a PLC and a PAC for your next application.

Jeff Payne

02/04/2013

While PLCs (programmable logic controllers) have been around for more than 40 years, recent advances have greatly increased their capabilities, blurring the line between a PLC and PAC (programmable automation controller). What differences remain between these two categories? Is there a performance gap between PLCs and PACs that users should keep in mind when choosing the best solution for a particular application?

A brief bit of history can put the discussion in context. PLCs were created in the late 1960s to replace relay-based systems. Conceptually they were similar and used ladder logic that mimicked the appearance of wiring diagrams engineers used to represent physical relays and timers, and the connections among them. Early PLCs required dedicated proprietary terminals for programming, had very limited memory, and lacked remote I/O.

By the 1980s, PC-based software was introduced for programming PLCs, which had become faster and had added more features as years passed. Since then, many new technologies have been applied to PLCs, greatly expanding their capabilities on an almost continuous basis.

PACs are relatively new to the automation market, using the term coined by the market research firm ARC in 2001. Since then, there has been no specific agreement as to what differentiates a PAC from a PLC. Some users feel the term PAC is simply marketing jargon to describe highly advanced PLCs, while others believe there is a definite distinction between a PLC and a PAC. In any

case, defining exactly what constitutes a PAC isn't as important as having users understand the types of applications for which each is best suited.

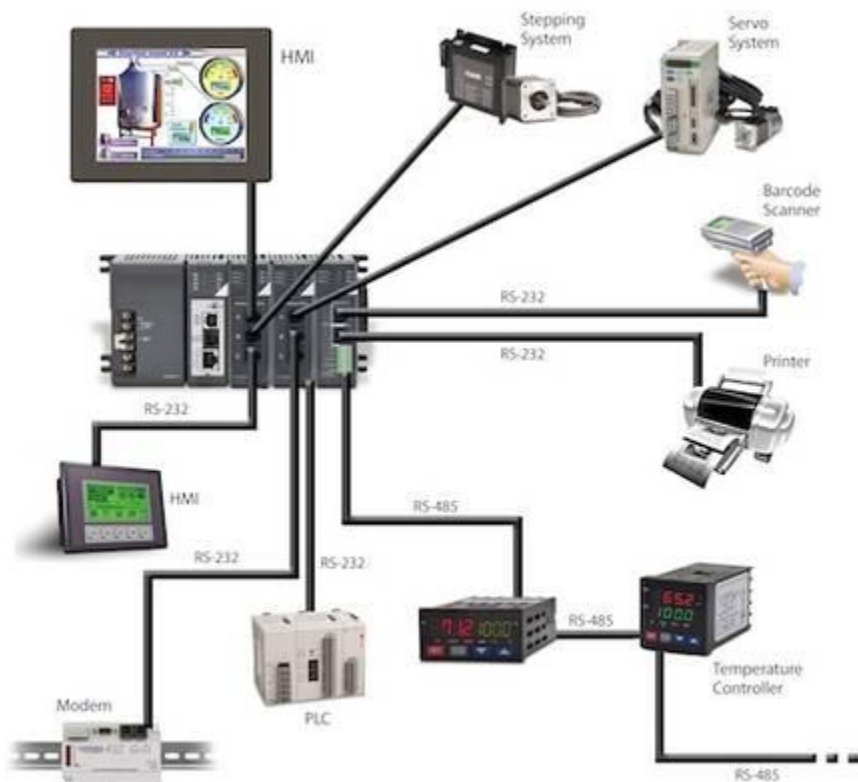
### **Determining users' needs**

Most suppliers carry a wide range of PLCs and PACs, which can make it difficult to choose the right product for a particular application.

Typically PLCs have been best suited for machine control, both simple and high speed. Common characteristics of these PLCs are simple program execution scans, limited memory, and a focus on discrete I/O with on/off control.

On the other hand, a PAC is geared more toward complex automation system architectures composed of a number of PC-based software applications, including HMI (human machine interface) functions, asset management, historian, advanced process control (APC), and others. A PAC is also generally a better fit for applications with extensive process control requirements, as PACs are better able to handle analog I/O and related control functions. A PAC tends to provide greater flexibility in programming, larger memory capacity, better interoperability, and more features and functions in general.

As a result of having an architecture based on ladder logic and a focus on discrete on-off control, expanding a PLC beyond its original capabilities—such as adding extensive analog control capabilities—has often proved difficult. In older or lower-end PLCs, separate hardware cards usually had to be added and programmed to accomplish functions outside the PLC's core focus. These functions included, but weren't limited to, networking multiple components, extensive process control, and sophisticated data manipulation.



To answer the demand for more PLC functionality, manufacturers have added features and capabilities. For example, older PLCs could only accommodate a relatively small number of PID loops, typically about 16, while new PLCs can handle thousands of such loops. Newer PLCs often feature multiple communication ports, and greatly increased memory as compared to older models (see Figure 1).

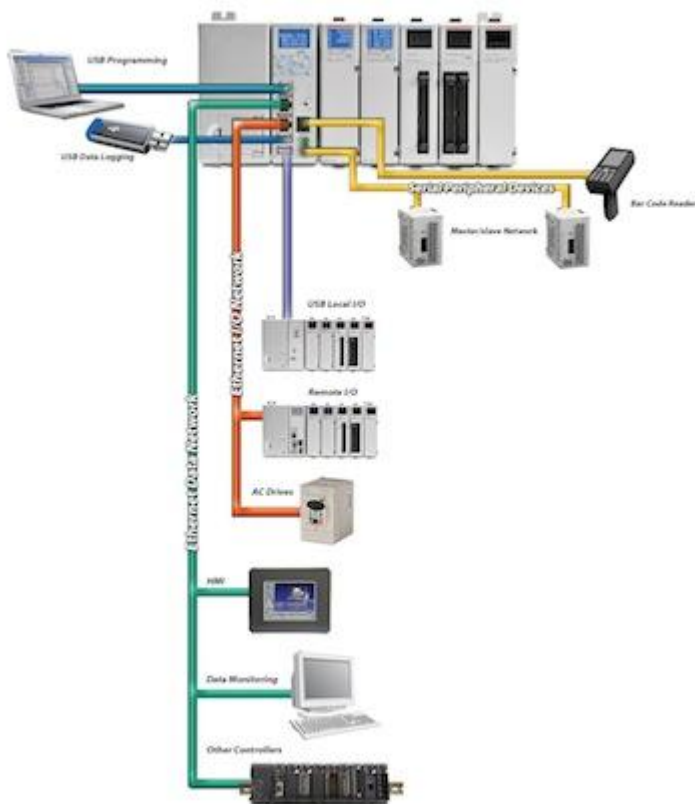
On the other hand, PACs provide a more open architecture and modular design to facilitate communication and interoperability with other devices, networks, and enterprise systems. They can be easily used for communicating, monitoring, and control across various networks and devices because they employ standard protocols and network technologies such as Ethernet, OPC, and SQL.

PACs also offer a single platform that operates in multiple domains such as motion, discrete, and process control. Moreover, the modular design of a PAC simplifies system expansion and makes adding and removing sensors and other devices easy, often eliminating the need to disconnect wiring. Their modular design makes it easy to add and effectively monitor and control thousands of I/O points, a task beyond the reach of most PLCs.

Another key differentiator between a PLC and a PAC is the tag-based programming offered by a PAC. With a PAC, a single tag-name database can be used for development, with one software package capable of programming multiple models. Tags, or descriptive names, can be assigned to functions before tying to specific I/O or memory addresses. This makes PAC programming highly flexible, with easy scalability to larger systems.

### The choice is yours

For simple applications, such as controlling a basic machine, a PLC is a better choice than a PAC. Likewise, for most applications that consist primarily of discrete I/O, a PLC is the best choice—unless there are other extraordinary requirements such as extensive data handling and manipulation.



If the application includes monitoring and control of a large number of analog I/O points, then a PAC is generally the better solution. This is also the case when the application encompasses an entire plant or factory floor, a situation that typically calls for

distributed I/O in large numbers, along with extensive loop control—functions better suited to a PAC than to a PLC.

The confusion arises when an application lies somewhere between simple and complex, and in these circumstances a high-end PLC or a low-end PAC platform will work. Ultimately, a choice between the two will be defined strictly by other factors outside of specific application requirements. These factors include, but aren't limited to, past experience with each platform, price, the level of local support, and anticipated future growth and changes.

Once a decision is made between a PLC or a PAC, users typically have a wide range of products from which to choose, even if only a single vendor is being considered. That's because PLCs and PACs are typically designed in systems of scale, meaning there is a family of controllers to choose from that range from lower I/O count to larger system capacity, with correspondingly more features and functions as I/O counts and prices increase.

**Table 1: PAC advantages over PLCs**

1.	<b>More open architecture</b>
2.	Modular design
3.	<b>More analog I/O options</b>
4.	Better process control capability
5.	<b>Designed for tight integration with SQL and other databases</b>
6.	Tag naming that lets users define data types
7.	<b>Greater I/O capacity</b>
8.	More built-in communications ports
9.	<b>USB data logging</b>
10.	Programming via USB port

### Functional differences

The demarcation line between PLCs and PACs has become less clear, but there are still some applications that clearly favor a PAC, due to its greater range of features, functions, and capabilities (Table 1). Here are a few observations:

- From a programming perspective, a PLC typically has a fixed memory map and addressing. In contrast, a PAC allows tag naming, letting users

define data types as they program. This provides more flexibility, especially when expanding the system.

- While many high-level PLCs have excellent execution speeds, PACs typically offer much greater I/O capacity and user memory size for larger projects and larger overall system sizes. This often makes them a better choice for large systems encompassing several areas of a plant.
- While advanced PLCs have increased communication and data handling options, PACs still offer more built-in features such as USB data logging ports, a web server to view system data and data log files, and an LCD screen for enhanced user interface and diagnostics (Figure 2).
- PACs are designed to be integrated more tightly with SQL and other databases. They often are still the choice for process control applications because they deliver other advantages such as standard 16-bit resolution analog for higher precision measurements.

Modern PLCs and PACs share many of the same features, and either will work in many applications.

The final selection will typically be determined by dozens of factors for any given application and company environment, including functional requirements, future expansion plans, company/vendor relationships, and past experience with specific automation platforms.

*Jeff Payne is product manager for the programmable controllers group at AutomationDirect, Inc.*

### **Key concepts**

- Differences between PLCs and PACs relate to functionality (or they should) and not just jargon
- Users making a selection for a new application should select on these differences when possible
- Applications should build in the ability to expand and incorporate technology improvements

Source:

<http://www.controleng.com/single-article/plc-vs-pac/44448cf771be09bff7115c621633bd94.html>