

Drive & Control profile

PC vs. PLC: key factors in comparing control options



When deciding whether to choose a PLC or PC, analyze and compare characteristics that could differentiate the two technologies such as operation, robustness, serviceability, hardware integration, security, safety, programming and cost.

One of the most crucial decisions in the initial design phase of a machine is the selection of the control system. For years the PLC has been leading the way for advances in machine automation control. From small controls used in the automotive

industry to large-scale controllers running entire factories, the PLC was the primary controller of choice. Since PLCs were first introduced in the 1970s through today, they have dominated the process and automation markets.

PC or PLC: Key Factors

- **Operation:** Focus on how system will run and how tasks/ instructions are processed
- **Robustness:** Some PCs have moving parts (disk drives)—choose controls able to withstand high vibration environments
- **Serviceability:** Consider availability of replacement components and ease of repair/ replacement over controller life
- **Hardware:** Assess application and communication requirements, fieldbus interfaces (SERCOS etc.), memory and processor capacity, and HMI integration
- **Security:** PLCs have not been virus targets in the past; however hackers constantly seek new paths into industrial/corporate networks
- **Safety:** Integrated safety features are well-established on PLCs, only recently available on PC-based platforms
- **Programming:** Both programming environment (how control executes program) and language (IEC 61131 for PLCs, proprietary, C++) affect machine development time and operation
- **Cost:** Consider cost issues such as motion/logic performance, machine expandability, operating environment and total cost of development to final delivery

However, from the early 90s to present day, the PC has been successfully working its way into those markets as processor speed and range continue to increase and the cost of those components continues to decrease. The number of applications utilizing a PC is on the rise, causing an accelerated development that blurs the line between the two technologies. Ten years ago it might have been a clear choice what type of controller to use; today that is no longer the case.

The PLC was developed as a more streamlined, flexible and reliable alternative to switch boxes and relay panels. It was designed to be dedicated only to specific tasks in the factory and its language and structure were modeled around the switches and relay panel circuits it was replacing. Furthermore, it had to maintain its robustness and consistent performance in challenging environments that contained relatively

high levels of EMI, contamination, and vibration. As time passed, the PLC evolved to include the capabilities of motion control, advanced PID process control and integrated safety, while also adopting some PC features such as a web server and networking utilities.

The PC on the other hand, served a higher level role in the scope of the machine. It was used primarily for complex calculations, monitoring, measuring and factory networking as well as a user-interface to the PLC. It was usually housed in a more environmentally controlled location because it couldn't operate as reliably as a PLC in harsh factory conditions.

The PC eventually evolved to include PLC functionality while still containing its core capabilities. Also, the PC has become a more robust controller, allowing it to operate in the harsh environments that were once only possible for PLC operation. In

analyzing their control architectures we can see even more convergence of the technologies. With the addition of a PCI card, hardware drivers and software you can turn your PC into what is known as a soft-PLC. Furthermore you can add a real-time kernel to support more critical tasking and control algorithms. On the other hand, there are PLCs with a built-in PC where all you need is a keyboard and mouse to get started.

So, how do you choose one type of control over another? In trying to figure out whether to choose a PLC or PC, it is important to first try to simplify the decision-making process by analyzing and comparing some of the characteristics that could differentiate the two technologies. For this discussion, there are seven major areas to consider for comparison:

- Operation
- Robustness
- Serviceability
- Hardware Integration
- Security
- Safety
- Programming

Operation

When analyzing system operation, the focus should be on how the system will run and how instructions and tasks are processed. The standard PLC has an embedded real-time operating system (RTOS) with a dedicated processor that ensures a high degree of control system reliability. Furthermore, since the task of the PLC is only to handle automation and/or a process, it does not need to run other utilities such as antivirus programs or system updates.

In turn, the PC with the introduction of a real-time kernel or real-time OS can accomplish the same reliability



The PC has become a more robust controller allowing it to operate in the harsh environments that were once only possible for PLC operation.

of control as a PLC. From their experience with PCs at home and in the office, users might be wary of lock-ups of the PC (the so-called “blue screens”). It is important to note that these lock-ups could occur on any OS—including a PLC—if the software running it is not handling priorities properly. For industrial use, the software running the PC is dedicated to automation and will therefore have a minimal chance of lock-ups. Even if a lock-up would occur, the real-time kernel is not affected and continues operation.

Real-time operation is a relative concept that means any task is guaranteed to be handled within a certain time. Synchronized motion and/or advanced PID control requires a high level of real-time determinism while non-critical supervisory controller operations such as monitoring error messages or sending non-critical controller commands or queries would not.

Robustness

Robustness of the controller essentially refers to its durability in various environments. The standard off-the-shelf PLC has no moving parts so it can withstand harsh environments for millions of cycles. A standard PC contains some moving parts such as fans or hard disk drives, and is therefore less suited for environments with high vibration levels. However, industrial PCs offer options such as solid state drives, fanless cooling and in-cabinet mounting. With these options, a PC becomes just as durable as a PLC and can successfully withstand the toughest industrial or environmental conditions. Both the PLC and PC have

converged in this area but the PC requires additional options to equal a standard PLC.

Serviceability

Another factor to consider is the ease and cost of serviceability, i.e. the repair and replacement costs over the life of the controller. For a PLC, external devices can be replaced with ease while the system is in operation. Moreover, if you must replace the actual controller, the PLC’s compact modular design makes that a quick and painless job. This saves cost because it significantly reduces the down-time of the machine. In a PC you can also hot swap, but only USB or other external peripheral devices. If the PC has a more modular design, such as with a rack or panel-mount system, replacement time becomes closer to that of the PLC.

Not only is it beneficial to be able to effortlessly change out a system or its components, but to also have a constant resource pool of replacement parts with long-term availability.

In some industries *copy exact* policies require this long-term availability. It is easier to implement *copy exact* with a PLC since it doesn’t change as rapidly as a PC in terms of hardware and firmware. Trying to find parts for a PC even after a year or two is far more challenging than with a PLC. You can *copy exact* both, but it proves to be more difficult to do this for a PC.

Hardware

Every engineer appreciates a wide choice of options when selecting the control system hardware, as there is always a need for items such as additional peripherals, memory

and a user interface. Both the PLC and PC have the ability to control a multitude of devices using industrial communication networks. Some of the well-known networks used are SERCOS, Profibus, DeviceNet and CANbus as well as their more modern Ethernet-based counterparts such as SERCOS III, Profinet, Ethernet/IP and EtherCAT.

Although the PLC and PC can both offer an array of fieldbus options, the PLC has many of these options built in where the PC would need additional cards and drivers to provide a comparable offering. However, besides the typical fieldbus networks the PC is equipped with a more open and flexible array of interfaces such as USB, FireWire, Serial, wireless Ethernet, etc. This gives the user access to more off-the-shelf devices to handle tasks that a PLC could not usually handle. An example is the utilization of an advanced hi-res imaging system where the images would be stored, analyzed, compared and possibly archived. For this, a PC would be better suited because of the advanced nature of this task and the amount of memory required, while a PLC would have limited storage and processing capabilities.

For some applications, the user interface is a crucial component on the machine. The PC has one built in while the PLC would need one or all of the following for primary operation: switches, operator panel(s), or an industrial PC. So while the PLC can interface with devices over fieldbus and can do complex operations, it still needs the PC to handle more memory demanding tasks and to sometimes make connections to other devices in

the system that a PLC would have no direct connection to.

Security

Security mainly deals with protecting the file system and application. This has two aspects: preventing unauthorized access from the outside world (e.g. virus attacks, malware, etc.) and limiting user access (e.g. restricting user rights, hiding files, etc.).

Traditionally, a PLC is less exposed to unauthorized access from the outside world. In addition, because of its dedicated OS, very few instances are known of virus attacks on PLCs. Note that this does not mean that PLCs are immune to viruses! Moreover, since PLCs have enjoyed a virus-free status for years, there are no standard ways of detecting and eliminating a virus if it would occur. Although the PC is far more susceptible to virus attacks than a PLC, strict security measures can be invoked to significantly reduce potential threats, and standard software is available to detect and remove viruses.

Both the PLC and PC offer different levels of user access to keep the contents as secure or as open as desired.

Safety

Depending on the operating environment, safety can be a major concern for the user. It is vital in cases where human interaction with the machine could be potentially dangerous, and when safety standards in a fab or factory are crucial to machine acceptance.

The PLC with its long history in machine automation has designated

communication channels to slave devices to tightly monitor operation, and optionally has additional integrated circuits to allow for redundancies. Integrated safety has only recently become available on some PC-based platforms, and does not have a wide acceptance base yet.

Programming

The functionality of a device is only as good as the program running on it. Therefore the programming environment and language are crucial to optimal machine performance. One main difference between PLC- and PC-based solutions is how the code is executed. A PLC has a mixture of scan-based and event-driven program execution, whereas PC software typically operates as event-driven. The scan-based execution of a PLC program might take longer because the system needs to complete the higher priority actions in the cycle first. The difference in execution style requires a different programming

philosophy, and often users are committed to either one or the other.

The same holds for the programming language: PLCs are programmed using either one of the languages specified in IEC61131-3 (ladder logic, instruction list, etc.) or some proprietary vendor language, whereas PC-based controls can use programming languages such as C/C++/.NET. These languages offer unparalleled flexibility and easy integration.

One last item: the cost discussion

Even after analyzing the technical requirements and comparing the design options for your machine, there is one more topic that needs to be considered: cost.

Many applications can be solved by either a PC-based or PLC-based solution, but for some applications the cost of those solutions might be quite different. A comparison of the

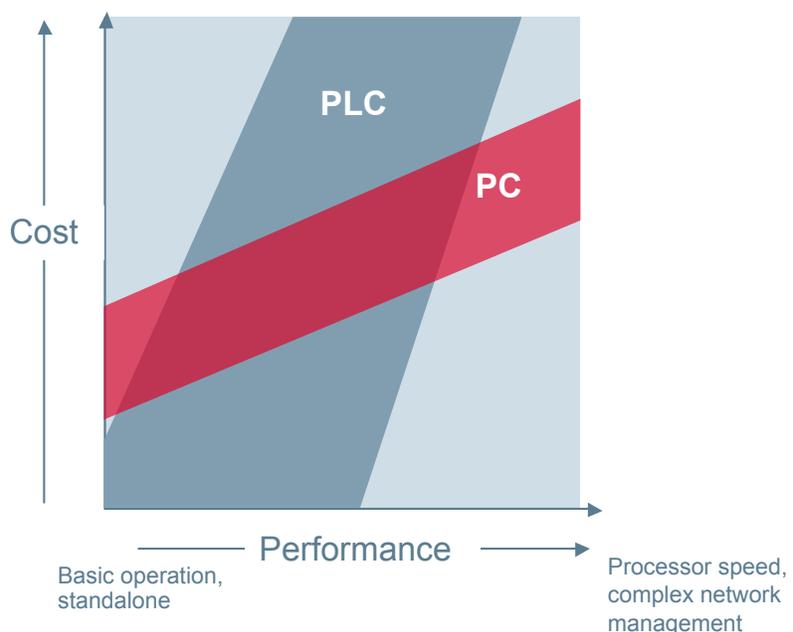


Figure 1

relative cost of PC- and PLC-based architectures centers on four different areas: performance, expandability, environment, and development effort.

Performance vs. cost

When your application requires the control system to fulfill demanding calculations, handle complex network loads and process a lot of data, look closely at the cost of a PC-based solution. Although the initial cost of a PC might be higher, it provides a very powerful system of which the cost only increases incrementally when more performance is required. PLCs, however, typically start at a lower price point, but can quickly get more expensive than a PC when high performance is needed (Fig. 1).

Expandability vs. cost

A similar scenario holds when considering expandability, for example the capability of your control system to add peripherals, increase calculation power or add data storage. PCs tend to have a higher cost when no additional functions are needed (because of the optimized nature of a PLC), but cost does not increase much when increasing expandability. The cost of a PLC is typically lower for applications where these items are not important, but the cost ramps up steeply when a lot of additional devices are needed (Fig 2).

Operating environment vs. cost

The operating environment can sometimes be the deciding factor for the type of control system, if solely based on cost. The standard PLC was designed to withstand the sometimes harsh conditions found in an industrial factory setting. It is no surprise that the standard PLC costs much less to operate in rough environments

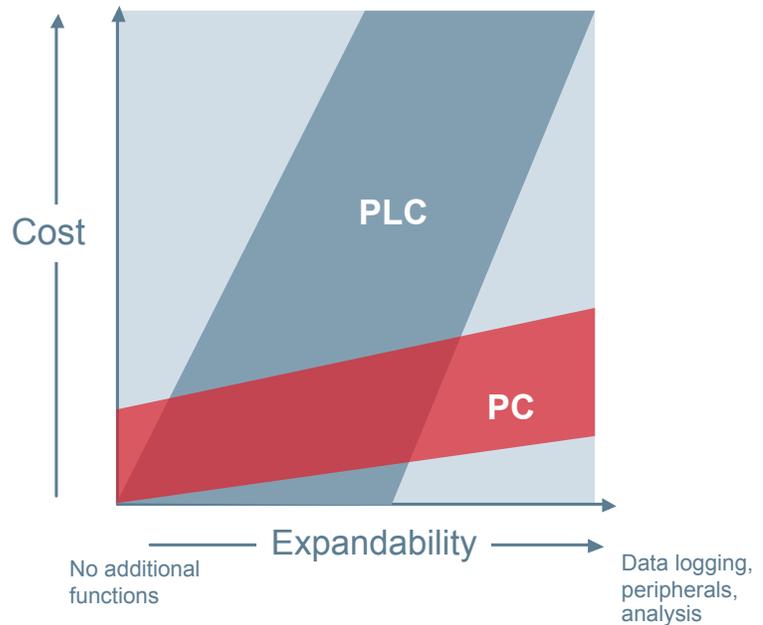


Figure 2

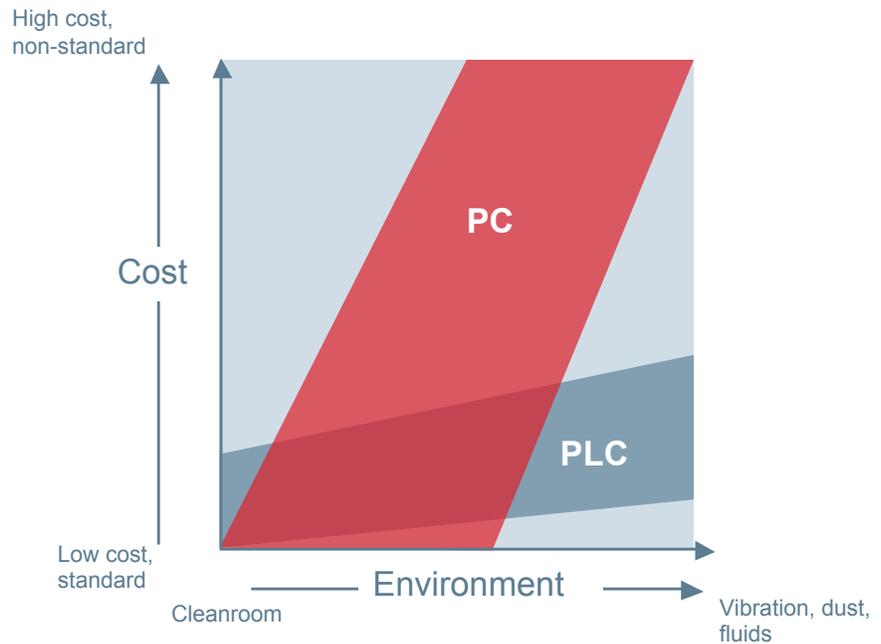


Figure 3

in comparison to the PC. Adding components to make a PC more robust significantly increases the cost (Fig. 3).

Development time and application level vs. cost

Software development time can add a significant cost to a machine design

project, but it often remains hidden when initially selecting the control system. If this is the first time the particular control system is applied, do you take the time to learn the new product or do you hire a third party with knowledge of the product to help integrate the new system?

With a PLC that utilizes a common format such as IEC61131-3, chances are that even if you are not familiar with the language, there is a large resource pool of those who are. Moreover, this platform has several built-in functions that are ready to integrate and solve certain problems quickly. Therefore the PLC can sometimes be a less expensive option than a PC-based controller from a development standpoint, for those who are first-time or intermediate users: the development time (or cost) decreases as the level and knowledge base of the user increases (Fig. 4). With extensive expertise and intimate knowledge of the system, there will be

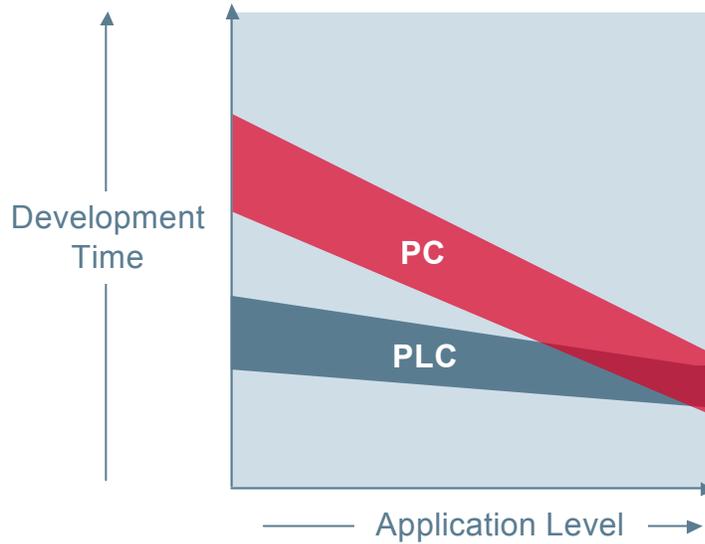


Figure 4

little or no difference between a PC and PLC-based solution.

Conclusion

The bottom line is that you will most likely choose either a PLC or PC or both for your control system. The main

issues to consider are performance, functionality and cost. However, looking deeper and considering factors such as programming needs, security and other criteria will ensure a smooth system integration process in the long run.