By Peter Fröhlich,
Marketing + Consulting Embedded Solutions,
IBM Deutschland Informationssysteme GmbH.

New Trends in Automation: Concepts for PC-based Controllers

Process optimization poses new challenges on the IT structures in production environments. The industrial PC has become a standard platform and key element in the manufacturing field. The role of the PC is more and more being extended down to the machine control level, the traditional domain of PLC's. The article points out reasons for this trend and gives an overview of the different concepts of PC based controllers. An introduction is given to the work of the Open Control user group, which is determined to use the PC platform as basis of a modular and open automation architecture. Finally the approach IBM takes to implement PC based control is presented.

Today the PC performs a number of tasks as Human Machine Interface (HMI), communication, preparing, processing and storing manufacturing data or problem detection and identification. The industrial PC is very cost effective because its standardized hard- and software components participate in the mass market of office IT. The ability to use a PC also as machine controller puts a lot of market pressure on the traditional PLC. A PC taking over the tasks of a PLC not only saves cost, it also provides for an optimal integration of all functions within the PC.

FROM PLC TO PC

The move from the PLC to the PC based controller illustrates figure 1. The traditional PLC at the top was directly connected to the machine's actors and sensors. It's modular architecture made it easy to combine different types of I/O interfaces. Following its original purpose to substitute hard-wired logic circuits it was programmed using an external detachable programming unit and worked stand-alone afterwards. Today such static logic or simple control sequences can be implemented with the classic approach using cheap micro-PLC's.

For reasons of flexibility and cost effectiveness more complex automation systems build on field busses. The I/O interfaces that connect to actors and sensors are integrated in "intelligent clamps" (clamps with integrated field bus interface) which can be mounted close to the device. The main advantage of the traditional PLC to combine physical I/O interfaces and the process computer in one modular unit has lost its impact. Today in many applications traditional PLC's are reduced to the CPU and some communication and field bus interface modules. Typically an additional industrial PC serves visualization and operator interface purposes (HMI) using graphical software running on Windows or OS/2. This is shown in figure 1 in the

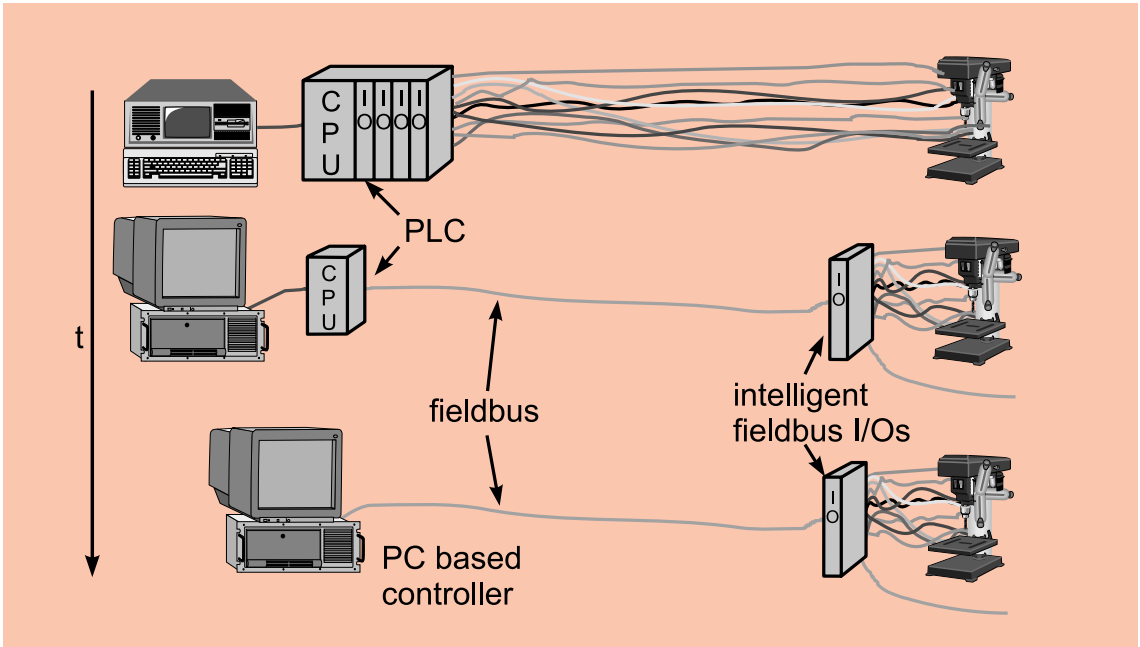

Figure 1. Steps in development from the PLC to a PC based controller.

middle.

Integrating the PLC functionality into the already existent PC results in a typical PC based controller as shown in figure 1 on the bottom. Either an additional CPU (coprocessor approach) or the main CPU of the PC executes PLC programs. Integration into the LAN and other functions are accomplished with cheap off-the-shelf PC slot cards instead of specialized PLC equipment.

Thus, on the hardware side the field bus paves the way for PC based automation because it takes away the need for direct peripheral access to field devices, which brings about some problems within a PC. On the software side the way is prepared by the standardization of the PLC programming languages within IEC-1131. A number of generic programming tools for IEC-1131 are available on the market. These tools generate code for PLC's of different manufacturers. It is even possible to combine them in one project. Therefore the development of control software is to a large degree independent of the choice of hardware. The standardization makes smooth transition to PC based controllers possible. The user doesn't notice any difference during engineering and runtime between a program running in an external PLC or directly in the PC.

CONCEPTS FOR PC BASED CONTROL

Four main concepts exist to execute control programs on the PC platform:

- Control software is executed directly within a standard multitasking operating system such as Windows NT or OS/2. Other than DOS or earlier versions of Windows these operating systems provide true multitasking and protected task execution, which should be seen as minimum requirements for machine control. However, they don't provide mechanisms to ensure timely reactions. Of course this is not directly a matter of the operating speed but rather of the synchronization between software execution and the technical process. General-purpose operating systems (GPOSs) are not built to do that.
- RTOSs (Real-time Operating Systems) provide these abilities: Using them on a PC is the traditional way of executing control programs on a PC. By using an RTOS the PC becomes a dedicated process controller. That buys only half the ticket: Cost reductions are provided on the hardware side because standard components can be used. Unfortunately main stream software for Windows or OS/2 cannot be executed on an RTOS. Flexibility and cost reduction of such solutions are therefore limited.
- Combination of RTOS and GPOS. Using a special slotcard, which generates periodic non-maskable interrupts (NMI), it is possible to switch back and forth between an RTOS and a GPOS combining the advantages of both. The maximum latency of real-time events depends on the switching frequency. The higher the switching rate is the smaller is the latency. Of course, also the overhead increases as the frequency goes up. Additional circuitry can provide extended interrupt management for fast reaction to interrupts issued by the periphery.

  Solutions for Windows 3.1 and Windows 95 have been around for a while. Recently also solutions for Windows NT have come on the market. Because of the far-reaching protection mechanisms in Windows NT it is much more difficult to combine it with another operating system.

  This architecture provides a cheap solution for PC-based control when medium performance and reliability are sufficient. Because the resources of the PC are shared by two operating systems, control programs and standard PC applications cannot be completely decoupled. Bugs in the PC applications, even more critical in device drivers (printer, screen etc.) can hurt the integrity of the control programs. If a specific setup of hard- and software is sufficiently tested and remains unchanged, however, the chance of running into reliability problems is small.
- RTOS on a coprocessor slot card in the PC. This solution is optimal in both performance and reliability. The coprocessor can be optimized for control applications running on dedicated real-time operating system. The PC performs all non time-critical tasks as HMI or data management. Both sections

	Reliability	Real-Time Function	Performance	Flexibility	Price
PLC + PC	high	good	high	medium	high
GPOS on PC	low	bad	medium	medium	low
RTOS on PC	high	good	medium	low	low
RTOS + GPOS	medium	medium	medium	medium	low...medium
Coprocessor	high	good	high	high	medium

Table 1. Comparison of different concepts of PC-based control regarding reliability, real-time reaction, performance, flexibility and price.

are well integrated across the fast bus communication between PC and coprocessor card. By using several coprocessor cards performance can be scaled up and absolute decoupling between different control tasks be achieved.

See Table 1 for a comparison of the different concepts for PC based controllers.

OPEN CONTROL

The advantages of PC-based automation systems go beyond pure hardware cost reduction. Well-established soft- and hardware standards of the PC environment offer new possibilities for the cooperation of various automation applications. To establish open interfaces and standards in the field of PC-based automation, IBM and Phoenix Contact founded the Open Control user group in spring 1996. The approx. 100 Open Control members are suppliers and users of automation soft- and hardware. The group has specifies interfaces and common data structures to share data between engineering environments and to communicate between runtime applications and with the fieldbus representing the periphery. The final specification for CALL (Control Application Link Layer) was presented at this year's Hanover industry fair. CALL consists of the three parts CALL-E (Engineering), CALL-R (Runtime) and CALL-P (Periphery) describing the corresponding interfaces and data structures. It is a basic principle of the Open Control group to use existing or emerging standards as far as possible. As an example, CALL-R contains OPC (OLE for Process Control) the communication interface specified by Microsoft and a number SCADA suppliers.

Currently the Open Control members implement the CALL interfaces into their software. A certification procedure ensures that Open Control hard- and software sticks to the specifications and thus is a building block, which can be combined with other Open Control components to construct a customer, optimized Open Control system.

A variety of Open Control compliant products will be offered late this year by a number of members. Of course IBM's own coprocessor solution described afterwards also meets the Open Control requirements. As one of IBM's core competencies are networks and network management, IBM technology will be used for an (optional) Open Control network platform which optimizes communication and eases management of large Open Control installations.

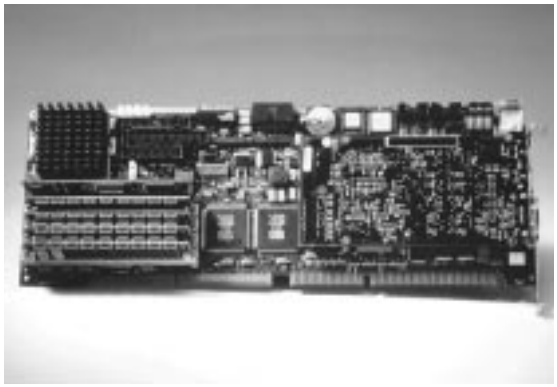

Figure 2. IBM 403 CoP Coprocessor and IBM INTERBUS PCI Mezzanine Card.

POWERFUL COPROCESSOR CARD WITH PMC SOCKET

As basic building block for PC-based automation systems IBM offers a powerful coprocessor solution. The IBM 403 CoP Coprocessor card is an ISA slotcard with the PowerPC 403 embedded CPU. Together with an IEC-1131 development system this card combines the advantages of a traditional PLC and a PC: very high reliability because of strict separation between control and visualization or other functions. In spite of its hot RISC performance the CPU stays cool because of its CMOS technology. The latter makes this CPU very reliable and allows putting a number of 403 CoP cards into one PC, replacing several external PLC's. Peripheral access is handled through the on-board PMC socket. Figure 2 shows the IBM 403 CoP Coprocessor card and the Interbus PMC module described afterwards.

INTERBUS GENERATION 4 ON A PMC MODULE

For industrial automation applications IBM has developed a PMC Interbus Generation 4 fieldbus master adapter which plugs right into the IBM 403 CoP. This card makes the large variety of sensors, actuators or intelligent field devices like axis controllers, identification systems or operator panels for Interbus accessible. The slim coprocessor/PMC sandwich only takes one ISA slot in the PC. Support for other fieldbusses is available.

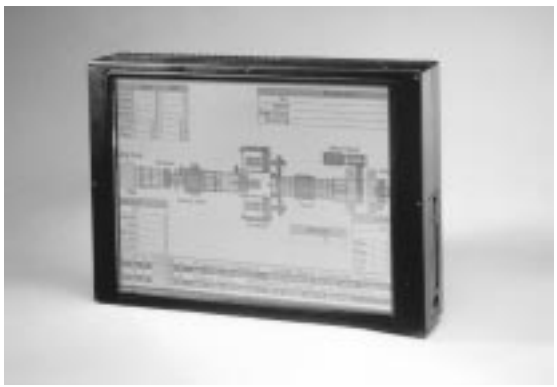

Figure 4: IBM 7592 Machine Interface Computer with Pentium and TFT touch display.

Figure 3. IBM 7587 Industrial Computer with Pentium single board slot computer.

PCS FOR INDUSTRIAL USE

The ideal PC to host one or several IBM 403 CoP cards for an integrated PC based automation system is ruggedized to handle rough machine environment conditions. The proven 19" IBM 7585 Industrial Computer convinces not merely with its technical data as the up-to-date Pentium performance, the extended temperature and shock resistance or elaborate air filtering system. It's the service-friendliness and the reliability of a carefully engineered piece of hardware that really makes the difference.

Very flexible are the passive backplane IBM 7587 and IBM 7588 Industrial Computers with up to 12 extension slots. They can be mounted externally eliminating the need for a 19" rack (Figure 3). As very compact yet powerful and flexible operator panel or any other HMI tasks serves the IBM 7592 Machine Interface Computer (Figure 4). The size of this Pentium PC is merely determined by its 12" TFT touch display. It uses IBM ThinkPad technology in a full-metal industrial housing.

Dr. Peter Fröhlich has been with IBM since 1996 and is responsible for the system architecture of IBM's Open Control solutions. He received his PHD in electrical engineering at the University of Stuttgart where he worked on object oriented concepts for real-time applications and qualitive modeling techniques.