Cut product development time and costs with PC/104 and microcontroller platforms

By Walter Calmette and Glenn de Caussin

**PC/104 offers an ideal platform for controlling the motors that run heavy equipment.** For simplicity and I/O convenience, commercial-off-the-shelf Microcontrollers (MCUs) with integrated I/O can be adapted to PC/104 to create extremely efficient controller designs. Additionally, consumer PDAs make excellent development tools for the controller and are useful in the end design as operator interface consoles.

Using MCUs as a basic building block of customized PC/104 boards reduces the time and cost of board development, allowing designers to focus on the value-added aspects of a motion control solution. Rather than investing six months and corresponding hard costs in developing a system that may or may not go into full production, a custom PC/104 solution can be developed in as little as four weeks, and costs can be substantially cut. In addition to cutting time and cost, PC/104 eases the production "buy/make" decision by transitioning to a PC/104 bus-less "make" product with a minimal amount of redesign.

Another consideration in motor and machine control is the operator interface. This interface provides valuable feedback to an engineer during development, and it provides critical machine information to the user, service personnel, and production floor while in service. Handheld PDAs have a high-tech look and feel and do not require a huge development investment. Using a PDA can eliminate the need to purchase and develop a separate handheld terminal system. A PDA provides a low-cost path for a high-resolution color touch screen, graphics, Wi-Fi connectivity, and nonvolatile memory, and its expansion slot(s) enable the addition of other high-tech options. Together, commercial off-the-shelf PC/104 modules and PDAs offer low-cost ways to control heavy equipment.

**Indexer example**

Calmotion is an industrial automation company focusing on delivering open standard solutions to engineering professionals. The company’s motor drives are designed to be part of an integral PC/104 bus, which in rugged versions provides a compact structure upon which to build a customized motor controller using industry standard software and hardware. PC/104 allows virtually any combination of digital/analog I/O with wired/wireless networks at a fraction of the usual cost compared to closed proprietary systems.

In conjunction with motor drives, the company uses PC/104 in a low-cost controller powered by the PIC18F8722 from Microchip (see Figure 1). The MCU offers I/O and real-world flexibility in control applications while leveraging the versatility and wide range of PC/104 I/O boards.

Recently, a customer discovered that parts for their old indexer control had been discontinued, and a redesign using longer life-cycle parts was necessary. The issue was whether they would run out of parts before the controller could be redesigned. The Calmation MC104P PC/104-based motor controller powered by a 40 MHz, 8-bit MCU handled the development and testing (see Figure 2 for a block diagram of the development version). After adding all the features and functionality to the front end and proving that it worked, the PC/104 bus proved no longer necessary to the end production system. It was eliminated by using a high-end MCU, thus stripping out additional costs (see Figure 3 for a block diagram of the production version).

**MCU interface**

Most MCUs are not designed as bus-based devices but rather as standalone integrated CPUs. For proof-of-concept during a development stage, interfacing an MCU to a PC/104 ISA bus can sometimes prove challenging. A Dual-Port RAM (DPR) memory can be used between the MCU and the ISA bus to greatly simplify the design effort.

The PC/104 address and data lines are straightforward. It may be tempting to select an MCU based on its ability to interface to the PC/104 bus, but the controller end product will most likely suffer if its peripherals are poorly suited for an application than if it has a clumsy PC/104 interface. When selecting an MCU, the designer has the choice of using I/O or memory space. Our experience in motor control leans towards choosing I/O, since it allows the most flexibility, ease of decoding, and less chance of conflicting with other system peripherals. The MCU can then be integrated with the appropriate read/write PC/104 bus signals.

Decoding is possible utilizing comparators or CPLD logic without much effort. In addition, most DPR units offer “mailbox” space that can be used to generate output interrupts on both sides of the memory and signal an interrupt to the MCU, or as an IRQ on the PC/104 side. The simultaneous, memory-access contention logic can add welcome-bus-wait states, eliminating the time spent designing and debugging a custom CPLD/FPGA solution. In addition, IDE software breakpoints can be used after reading a section of memory on the MCU side to debug drivers at runtime.

Another option for a motor controller is to use an MCU with an external memory interface because it makes connecting to the DPR straightforward. This is espe-
cially true if the MCU’s external memory interface has dynamic bus-wait-state capability. If not, logic can be added to alert the micro in cases of memory access contention, such as when it and the PC/104 controller simultaneously access the same DPR location. Input pins capable of generating interrupts on MCUs are common. Connecting them to the “mailbox” output of the DPR on the micro side can interrupt the MCU, indicating that the PC/104 controller has sent a data packet. This setup minimizes response time.

**Without a memory interface**

Using MCUs without an external memory interface requires a little more user code intervention. However, in this case, low-cost MCUs can be used. MCUs without external memory interface I/O pins can be set up to simply “bit bang” the DPR to read and write data. The data lines of the DPR are connected to general-purpose pins of an MCU. To use an MCU in this manner, its general-purpose pins must be able to dynamically configure data pins as inputs or outputs, or put them into high-impedance mode. To prevent a short condition, the micro and the DPR must not drive their respective data outputs at the same time.

The obvious disadvantage of this methodology is the additional pin configuration instructions required prior to a read/write cycle. Unless there is a tremendous amount of data or instruction traffic that must take place between the MCU and the PC/104 bus, the additional time delay when using this method is minimal. Although this is admittedly a crude manner in which to interface to the PC/104 bus, a low-cost flash MCU, such as the PIC18F8722 from Microchip, can offer surprisingly good performance.

If the microprocessor has a Parallel Slave Port (PSP) peripheral, the data direction configuration commands can be omitted from the user code. Typically, the PSP peripheral facilitates the use of a microcontroller in a data-bus interface application. As the name implies, they operate in a slave mode whereby read, write, and chip-enable input pins dynamically configure the direction of the data pins by a master processor. Mock read, write, and chip-enable signals also need to be bit banged as if coming from a master processor. In this manner, much like a null modem cable, these signals are fed back into the PSP enable and read/write input pins and thus eliminate port direction configuration code. The PSP on the Microchip PIC18F452 is an example of a low-cost, 44-pin TQFP MCU that we have used in this fashion.

**Visual user interface**

Since the goal is rapid development time while still designing a low-cost motor controller, it’s handy to use off-the-shelf development tools. It is even handier if those tools can be used for an operator interface in the final controller equipment. Consumer handheld PDAs offer an attractive platform to meet both of these goals (refer to Figures 4a, 4b, and 4c).

The most popular handheld devices use either the Palm or Windows OS. Windows devices tend to cost more than their Palm counterparts, but they usually offer more features when it comes to expansion and connectivity via Wi-Fi. The latest version of Microsoft Visual Studio .NET Professional supports the compact framework of .NET. This compact framework is the foundation of devices that use Windows CE, Pocket PC, and Smart phones.
The biggest advantage of using this development tool is the portability of applications from desktop/laptop computers to handheld devices. If portability of PDA/PC code is not a concern, other alternatives exist, such as the development system available from NS Basic Corporation. They have two similar packages that support application development for either Palm or Windows CE. As an example, a development system from NS Basic allows designers to create a software scope on a Palm OS PDA that’s useful for motor drive tuning.

Most PDAs have serial port capabilities either directly or through some type of interface cable. The trend with the newer PDAs leans toward a serial interface but with TTL-level signals. The cables used to convert TTL to RS-232 are typically parasitic and receive their power from the RTS signal via an RS-232 port. For designers planning to use this type of interface, it’s essential to ensure that the RTS has the appropriate voltage. Programming an application is fairly straightforward with one exception: The Palm OS and some Windows development systems ignore null characters. That is, they assume that nulls are white spaces and can be ignored. This result most often presents an issue when using protocols that use all the binary numbers 0 to 255. Choosing the correct method when reading the serial port will overcome this assumption.

There are obviously ways around this, but it’s critical to be aware of them at the outset. RS-232 is the easiest and lowest-cost interface to develop and accommodates a straightforward laptop connection as well. A number of off-the-shelf RS-232 converters allow connection to TCP/IP, Modbus, CAN, or Bluetooth, which can be an effective way to keep engineering development costs down. If a USB interface is desired, at least one device must be the host. USB-to-serial converters exist, but they will only work with a host such as a computer. A USB-only PDA acts as a slave device, not a master. The PDA can only communicate using USB if the product has been designed as a host.

Motor control: Putting it all together
Starting with off-the-shelf PC/104 boards for proof-of-concept and evolving to developing custom PC/104 hardware with MCUs enables the use of such integral features as:

- Many types of internal memory
- Serial ports
- Timers/counters
- Interrupts
- A/D converters
- Watchdog timers

This set of peripherals and capabilities simplifies circuit design and board layout. In turn, this configuration reduces development time and costs.

Additionally, most MCUs come with low-cost development tools, which allow in-circuit programming/debugging without an operating system. Single stepping, variable watch windows, and the software breakpoints that are standard in most IDE packages speed the debugging and coding process. The “hockey puck” MPLAB ICD2 from Microchip connects MCUs with the development computer’s USB connectors.

Other manufacturers offer similar devices in the same price range. C compilers are almost universally available, thereby reducing portability concerns. A number of MCUs have peripherals developed for such specific applications as motion and motor control, CAN, and Ethernet. Such peripherals minimize software development and eliminate additional hardware that might otherwise be required.

Walter Calmette cofounded Calmotion LLC after 5 years in sales management at Fadal Machining Centers and 11 years with Rockwell Automation in motor drive sales and application engineering. Walter received a BS in Physics from UCLA.

Glenn de Caussin cofounded Calmotion LLC after 20 years in electronics and controls at Fadal Machining Centers where his last position was director of software/controls. Glenn received a BSEE from Cal State Northridge (CSUN).

For more information, contact Walter and Glenn at:

Calmotion, LLC
9909 Topanga Canyon Blvd., #322
Chatsworth, CA 91311
Tel: 818-357-5826
Fax: 818-357-5827
E-mail: sales@calmotion.com or support@calmotion.com
Website: www.calmotion.com