By Don K. Harrison, Vice-President,
James Honey,
Texas Micro.

System Management and CompactPCI Systems

The purpose of this paper is to give the reader an overview of hardware instrumentation extensions and standards being considered for CompactPCI boards and systems. These extensions will provide systems manufacturers a standardized method of deploying instrumentation, remote temperature sensors for example, in CompactPCI based computer systems. The System Management Subcommittee of PICMG (PCI Industrial Computer Manufacturers Group) is performing the work described here.

WHAT IS MEANT BY SYSTEM MANAGEMENT?

Manageability is a hot topic today for corporate system planners and IT managers. Recent surveys by Gartner Group and others have made the phrase "total cost of ownership" an essential part of any respectable marketing or sales presentation. What is meant is that the cost of the hardware and software is relatively insignificant compared to the cost of deploying, supporting and managing them. Consider the fact that just five years ago a hundred-megabyte disk was more than enough to handle the average desktop computer users needs. Today 2.4 gigabytes aren't always enough. Hardware has stayed on the complexity curve also. Consider the communications explosion, Internets and Intranets, higher speeds, switched hubs, 1GB Ethernet, etc. This tenfold or more increase in complexity has manifested itself to be a nightmare for systems administrators and IT managers. Factor in lost productivity, lost opportunity, etc. and it's clear what is meant by "total cost of ownership."

To combat this accelerating pace of technology, major industry players have begun initiatives to provide a framework for system management tools. These tools build upon the legacy standards such as SNMP (Simple Network Management Protocol) and CMIP (Common Management Information Protocol) with new standards such as DMI (Desktop Management Interface) and WMI (Windows Management Interface).

Corporate computing isn't the only group that has a need for improved management tools. Telecommunications companies have for years required equipment providers to monitor system operation and report failures. This is typically referred to as "NEBS (Network Equipment Building Standard) Alarming." A typical implementation in a telecommunications computer might consist of a service processor, typically a microcontroller, that monitors power supply voltages, fan rotation, temperature at various points, a heartbeat from the host application and other environmental data points and reports failures in the form of audible alarms, lit LEDs, relay closures, etc.

PICMG's vision is that these two forces are synergistic, and in fact, converging. The challenge for system vendors is to provide this increased level of instrumentation required by the new initiatives. The challenge as a specification body, in this case PICMG is to create a framework for interoperability both at the software and hardware layers. Since CompactPCI computer systems will find their way into both server and telecommunications applications; system management in some embodiment is an obvious requirement. Other applications, such as embedded computing, instrumentation, etc., will be able to take advantage of the built in features to add functionality

THE PREVAILING VISION AND THE OPPORTUNITY

Intel, in their literature, divides the task of controlling today's computing environment into four parts (paraphrasing):

- Asset Management: Instrumented components identify themselves to management software, and provide a wealth of data for inventory control and asset management.
- Off-hours and remote maintenance: Remote boot and remote control capabilities can automatically "wake up" a PC and install or upgrade software, without disrupting the user and without a technician's direct involvement.
- Initial system configuration: A service boot feature allows unattended installation of new systems, including installing the OS and user software.
- Remote problem resolution: Support technicians can take over a system remotely, increasing first-call resolution rates and minimizing the need for "house calls." Hardware monitoring, prediction of impending failures, and notification of problems before they bring down a system are an important component of problem resolution.

By providing support for these functions, IT managers have the tools they need to resolve many problems from their desk, insure that application and operating software is of the proper revision, etc.

The vision is extended by many of the initiatives to

include access to the World Wide Web with the idea that systems can now be managed on a global basis.

The opportunity with CompactPCI and PICMG standards is to provide better instrumentation of the system and it's modules. For the most part, hardware instrumentation deployment has been localized to single board computer modules (typically motherboards). This instrumentation might include a temperature sensor or two, possibly even one in the system power supply, but access to these sensors are implementation specific. For the most part, information transfer is "in-band" over traditional communications links such as the PCI bus, Ethernet, Internet, etc. Typically the system must be in running condition for these communications channels to convey any meaningful information. The in-band approach also makes the assumption that there is a bus connected to the module in question. Typically that isn't the case for components like power supplies or fan modules.

There is an "out of band" secondary bus, SMBus, defined to collect some of this localized data such as processor temperature or memory module parameters. This specification, which originated with Intel, is a step in the right direction, but deployment as a system wide data gathering network needs considerably more definition than that found in the SMBus specification. Address assignment using strict adherence to SMBus specifications won't work in a system with 14 CompactPCI slots, four processors, a CPU card, four power supplies, a service processor, etc. The goal is to provide extensions to the SMBus standard that would allow for the greatest degree of interoperability. The work of the PICMG System Management Subcommittee centers on defining a suitable secondary bus.

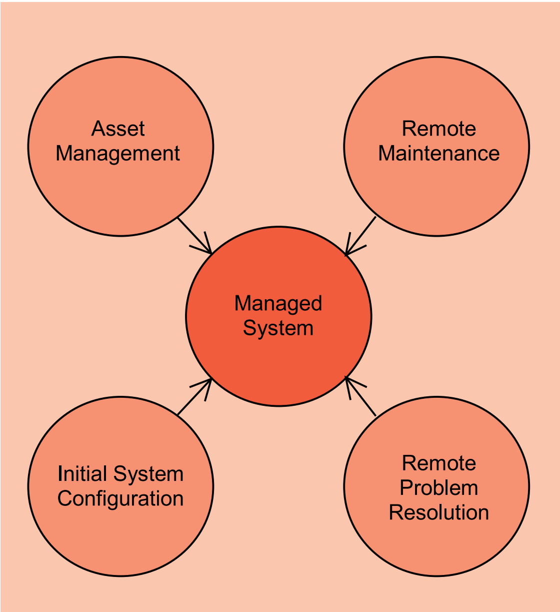

Figure 1. Intel divides the task of today's computing environment into four parts.

OVERVIEW OF THE SUBCOMMITTEE'S WORK

The original goal for CompactPCI was one of a high-performance, cost effective and highly maintainable system. The work of the CORE and HOT SWAP Subcommittees within PICMG has taken great strides to that end. With the ratification of the Telephony specification expected next month, Computer telephony applications will be able to easily utilize off-the-shelf CompactPCI systems for their applications. The System Management Subcommittee endeavors to add "highly manageable" to the list when its work is finally completed.

The goal of the System Management subcommittee (also known as the SMBus subcommittee) is to set instrumentation standards, both hardware and software, for manageable components in a CompactPCI based system. The on-going work includes:

- Analyzing existing secondary bus techniques such as I^2C, SMBus, USB, etc., and ultimately choosing the basic communications standard as the management bus.
- Defining AC and DC electrical specifications for the management bus.
- Deciding on a method for assigning addresses for nodes on the management bus.
- Choosing and/or defining communications protocol layers to insure reliable data transfer and interoperability with other standards.
- Writing prototype firmware to insure that concrete and interoperable standards will emerge.
- Writing prototype system level software to bridge the gap between instrumentation and higher levels of software such as DMI, SNMP, etc.

In addition to these key areas of concern, PICMG has something to offer the industry as a whole in sorting out the maze of secondary bus standards. The subcommittee will attempt to work with other specification bodies toward that end.

To date, the committee is just getting started, but is beginning to pick up momentum and the vision is becoming clearer. Originally, the group was only going to concern itself with an SMBus connection to the CompactPCI cards. When the group started to see an opportunity to provide a clear definition for system wide resources, the committee's scope was expanded to include those.

BUS DEFINITION AND SELECTION

Choosing the communications medium is a difficult and important task. The criteria have been:

- Must be low cost. For example an I^2C microcontroller capable of reporting information about a CompactPCI card costs $2 to $3 US.
- Must be rooted in a standard that promises to be around for a while.
- Should be harmonious with other standards. Example: I^2C is harmonious with SMBus.
- Must support a large (>64) number of addresses.

- Capable of sending unsolicited messages

Possible candidates include IEEE 1149.4 (JTAG), I^{2}C, SMBus, ACCESS.bus, CAN (Controller Area Network, an automotive favorite) and USB. Of these, 1149 and CAN aren't viable because of cost and the limited availability. SMBus and ACCESS.bus are derivatives of I^{2}C.

THE CASE FOR USB

USB has some advantages in that it definitely has a future and is being widely supported at low cost. There are two areas of concern when choosing a USB solution. USB utilizes a point-to-point physical interconnect which would require a centralized resource (hub). This has potential cost and single point of failure implications. Secondly, there is some concern that the complexity of a USB solution is overkill for what is needed by a management bus. Certainly there is no real need for ten megabit per second data rates.

I^{2}C, SMBUS AND ACCESS.BUS

The bus of choice, so far, is some derivative of I^{2}C. Over a decade ago, Philips invented an IC-to-IC interconnect they labeled I^{2}C. Since its inception, many ICs have been designed using I^{2}C as an interconnect, especially in the consumer electronics field. The attraction to I^{2}C can be summed up as follows:

- Simple 2 wire interface: just clock and data
- Ingenious true multi-master with built-in collision detection and arbitration
- Low cost
- Low power
- Proven technology
- Example software freely available
- Supported by many manufacturers (Philips, Hitachi, Harris, Intel, etc.)

An attempt was made to expand the role of I^{2}C to include an economical connection technique to peripheral devices such as keyboards, printers, mice, etc. This became ACCESS.bus. ACCESS.bus expanded I^{2}C to include a method of dynamic addressing and error checking protocol. ACCESS.bus unfortunately was dead on arrival when USB became the peripheral interconnect of choice. We are working with some of the members of this committee to see if there is anything that can be used (firmware, etc.). The dynamic addressing technique is interesting.

The other derivative of I^{2}C of interest is SMBus. SMBus was originally defined by a consortium of ten companies seeking to standardize the way host systems, mainly laptops and notebooks, communicated with their batteries and battery charging systems. The specification was released in February of 1995, and is widely used throughout the industry for communicating with smart battery subsystems. For example, Texas Micro's HardbodyTM uses SMBus to communicate between an on-board H8 microcontroller and the handheld computer's smart battery.

The SMBus is a specific implementation of I^{2}C that describes data protocols, device addressing and additional electrical requirements. The Smart Battery System (SBS) Implementation-Forum manages the "official" SMBus and is a clearinghouse for address assignments.

WHAT'S ALL THE FUSS ABOUT ADDRESSING ANYWAY?

One problem with I^{2}C and SMBus is that there are only 127 addresses available, and Philip's original vision was to assign addresses by device function. Several hundred addresses have been assigned in this way. When Philips began to sell microcontrollers for I^{2}C, the address could be programmed; eliminating the need for a Philips assigned address. Of course these micros couldn't interoperate very well without careful address choices. When the smart battery group decided to use I^{2}C, they wanted addressing to be under their control. Their plan was to assign addresses in a similar fashion as Philips. Obviously, with 30-something possible modules, conformance to either the I^{2}C or SMBus addressing regimens is difficult.

To complicate the addressing matter further, much of the SMBus work done to date has had to live with I^{2}C addresses assigned by Philips! National LM75 temperature sensors, LM78/9 system monitors and PC100 DIMM memory NVRAMS all boast fixed addresses assigned by Philips. It will be a challenge to define an addressing scheme that maintains interoperability with these devices, but it is a worthy goal.

The group is considering two possible approaches: use the dynamic addressing scheme abandoned by the ACCESS.bus group, or a liberal fixed addressing scheme with addresses assigned by module type (e.g., fans get 4 addresses, power supplies get 4, CompactPCI gets 32). The virtues of each scheme will be debated in the coming weeks.

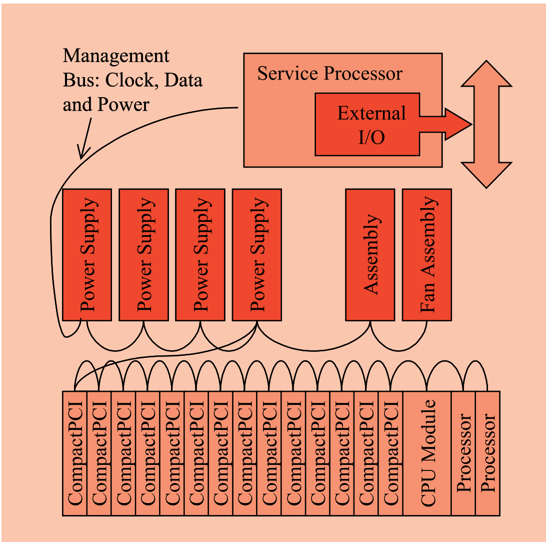

Figure 2. Block diagram of basic managed modules found in an enterprise server class system.

OTHER ISSUES

Each of the various I^2C derivatives defines a slightly different DC electrical interface. I^2C by definition is 5V. SMBus by definition is 3.3V. High and low thresholds, instead of being a fixed value, are a function of VCC for some parts. Part of the work of the committee will be to sort out these variations, and provide design guidelines, level shifter examples, etc.

CITIZENS OF THE MANAGEMENT BUS

Now that the groundwork and history have been introduced, let's see how the CompactPCI management bus can be used to instrument a system. Figure 2 shows a simple block diagram of basic managed modules found in an enterprise server class system. The mock system contains a single board computer, its processor modules (for instance Pentium II's), fourteen CompactPCI slots in two PCI segments, two fan modules, four power supplies in an "n+1" arrangement and a service processor (or alarm board). In this case, the service processor has a connection to an enterprise wide maintenance network.

Each of the modules monitor and report status to one of two hosts based on criteria programmed by the host. The modules have some things in common, but there may be specialized information as well. For instance, the fan module wouldn't report its output voltage levels like the power supply would, but it might report airflow measurements, RPM, temperature etc. The goal of the subcommittee is to build a framework for communicating the information, not to decide what type of information should be communicated. Some examples of the types of data that could be reported are:

- Serial Number and Revision Level of the module.
- Firmware revision levels
- Temperature of key components in the module.
- Fan RPM/Status
- Go/No-go Fault Indications
- Application Process Heartbeats from the Host
- Power supply voltages. (input and output)
- Ejector switch position.

The management bus isn't just for reporting status. When required, control can be exerted on various modules such as:

- Power Supply Inhibit
- Hot Swap Card Power Down or Reset
- Fault LED control.
- System Reset control.
- Engage Relays
- Fan speed-up

GETTING THE WORD OUT

The system as described has merit even if the information is contained within the local system. The service processor can close a fault relay and light a fault LED when problems are discovered. The real power of the system is only achieved, however, when the management system can communicate its information to higher levels. Today's technology centers on work done by the Desktop Management Task Force and the Desktop Management Interface (DMI). By mapping the management subsystem to DMI as a foundation, interfaces to legacy standards like SNMP as well as the latest object oriented standards like CIM are available. The group is just beginning to explore the requirements at this level.

NEXT STEPS

Much work still needs to be done before the entire system management vision can be realized. And of course, there is a review process, member voting, etc. That doesn't mean progress won't be measurable. The first step is to decide on the electrical characteristics, especially as they pertain to the CompactPCI backplane. That work should be rolled into the CORE specification in its next iteration. Once the physical interface is defined, board manufacturers can begin to think about how they can instrument their cards. Once fully deployed, the new specification will lead to a new level of interoperability for CompactPCI. ■

REFERENCES

- Desktop Management Task Force, http://www.dmtf.org/
- PICMG, http://www.picmg.org
- Intel Wired for Management, http://developer.intel.com/ial/wfm/index.htm
- Smart Battery System Implementers Forum, http://www.sbs-forum.org/
- SunSoft Management Topics, http://www.sun.com/webtone/roadmaps/sysmgt3.html
- This paper, other links and additional information can be found at the Texas Micro Website: http://www.texasmicro.com

Don Harrison, VP of Technology, Texas Micro Inc. Mr. Harrison held a variety of technical and engineering positions before joining Texas Micro in 1983, where he was instrumental in developing the first truly PC-compatible, passive backplane computer card, the beginning of the passive backplane industrial computer. Mr. Harrison and his team designed all of Texas Micro's passive backplane CPUs until his move in 1990 to head the company's SPARC/Telecom group. Recently, as Texas Micro focuses on CompactPCI as the architecture of choice for the future, Mr. Harrison has been working to define the next generation of high-availability, Intel-based telecom products.