

EMBEDDED PC SPECIAL BONUS SECTION: EMBEDDED PC

CIRCUIT CELLAR[®]

THE COMPUTER APPLICATIONS JOURNAL

#69 APRIL 1996

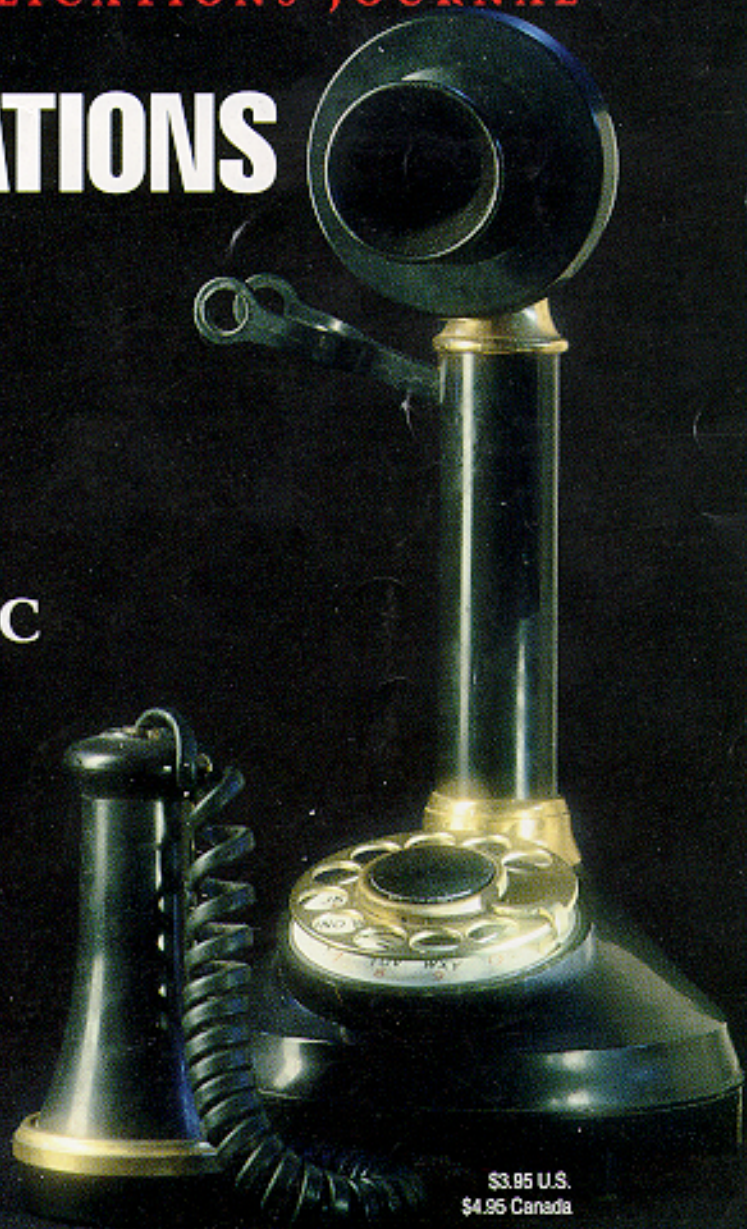
COMMUNICATIONS

Caller ID
Fundamentals

Networking Inside
an Automobile

Embedding the SPARC

EPC: Designing PCs
for Harsh
Environments



\$3.95 U.S.
\$4.95 Canada



TASK MANAGER

Political Arachnids



With the campaign season firmly upon us, I just finished perusing several Republican candidate Web sites, looking for someone to vote for in the primaries tomorrow. While the traditional candidate's communications media have been TV and radio spots, it's interesting to see what they're doing with the newest mass communications medium: the World Wide Web.

You might think a Web site to be a convenient cross between the quick sound bites of a TV commercial and the more in-depth coverage of a print medium. What I found was an interesting range of coverage spanning the two.

I'm not really interested in a candidate's life history or pictures of his family (though you'll certainly find both). Tell me where you stand on the issues. Several of the sites made it easy to find out where the sponsor stood. Others were nothing more than feel-good extended television commercials with very little hard content. While the negative campaigning that has dominated television these days was minimal, it was still there on a few sites.

I shouldn't be surprised at what I found, but I was hoping for more. A Web site should afford the user the chance to do some in-depth research into the candidates. The textual nature of the medium allows the campaign ample room to discuss as much as they like without paying a lot. The graphical nature also enables them to make it attractive. Such a mix was, unfortunately, the exception rather than the rule.

Switching from political to technical communications, we start with a look at how the SPARC processor can be used in telco applications. Next, we take a quick look at how Caller ID works and is decoded. Moving from telephone communications to intravehicle communications, the next article gives an introduction to the latest in networking technologies used in today's (and tomorrow's) automobiles.

On the subject of software communications, we present a technique to allow tasks within a multitasking system to more effectively exchange information while making the code more reusable. Wrapping up our feature section is an article and a pair of sidebars that present some new SPARC-compatible processors designed specifically for embedded applications.

In our columns, Ed reveals the mysteries behind memory caches, Jeff uses some colorful language when describing the newest in the world of LEDs, and Tom presents the latest contender in his PID-Pong challenge.

In *Embedded PC*, we start with a look at a new technology: a complete PC in a module. Next, we explore the issues related to designing embedded PCs for use in harsh environments. In *PC/104 Quarter*, Rick overviews what's available for developing software for use on PC/104 processor boards. Finally, Russ concludes his reign in *Applied PCs* by looking at what's available for packaging embedded PCs for different environments.

editor@circellar.com

CIRCUIT CELLAR®

THE COMPUTER APPLICATIONS JOURNAL

FOUNDER/EDITORIAL DIRECTOR
Steve Garcia

PUBLISHER
Daniel Rodrigues

EDITOR-IN-CHIEF
Ken Davidson

PUBLISHER'S ASSISTANT
Sue Hodge

EPC MANAGING EDITOR
Janice Marinelli

CIRCULATION MANAGER
Rose Mansella

TECHNICAL EDITOR
Carole Boster

CIRCULATION ASSISTANT
Barbara Maleski

ENGINEERING STAFF
Jeff Bachiochi & Ed Nisley

CIRCULATION CONSULTANT
Gregory Spitzfaden

WEST COAST EDITOR
Tom Cantrell

BUSINESS MANAGER
Jeannette Walters

CONTRIBUTING EDITORS
Rick Lehrbaum
Russ Reiss

ADVERTISING COORDINATOR
Dan Gorsky

NEW PRODUCTS EDITOR
Harv Weiner

ART DIRECTOR
Lisa Ferry

PRODUCTION STAFF
John Gorsky
James Soussounis

CONTRIBUTORS:
Jon Elson
Tim McDonough
Frank Kuechmann
Pellervo Kaskinen

CIRCUIT CELLAR INK®, THE COMPUTER APPLICATIONS JOURNAL (ISSN 0896-8985) is published monthly by Circuit Cellar Incorporated, 4 Park Street, Suite 20, Vernon, CT 06066 (860) 875-2751. Second class postage paid at Vernon, CT and additional offices. One-year (12 issues) subscription rate U.S.A. and possessions \$21.95, Canada/Mexico \$31.95, all other countries \$49.95. All subscription orders payable in U.S. funds only, via international postal money order or check drawn on U.S. bank. Direct subscription orders and subscription related questions to Circuit Cellar INK Subscriptions, P.O. Box 698, Holmes, PA 19043-9613 or call (800) 269-6301.

POSTMASTER: Please send address changes to Circuit Cellar INK, Circulation Dept., P.O. Box 698, Holmes, PA 19043-9613.

Cover photography by Barbara Swenson
PRINTED IN THE UNITED STATES

For information on authorized reprints of articles, contact Jeannette Walters (860) 875-2199.

HAJAR ASSOCIATES NATIONAL ADVERTISING REPRESENTATIVES

NORTHEAST & MID-ATLANTIC
Barbara Best
(908) 741-7744
Fax: (908) 741-6823

SOUTHEAST
Christa Collins
(305) 966-3939
Fax: (305) 985-8457

MIDWEST
Nanette Traetow
(708) 357-0010
Fax: (708) 357-0452

WEST COAST
Barbara Jones & Shelley Rainey
(714) 540-3554
Fax: (714) 540-7103

Circuit Cellar BBS—24 Hrs. 300/1200/2400/9600/14.4kbps, 8 bits, no parity, 1 stop bit, (860) 871-1988; 2400/9600 bps Courier HST, (860) 871-0549. World Wide Web: <http://www.circellar.com/>

All programs and schematics in *Circuit Cellar INK®* have been carefully reviewed to ensure their performance is in accordance with the specifications described, and programs are posted on the Circuit Cellar BBS for electronic transfer by subscribers.

Circuit Cellar INK® makes no warranties and assumes no responsibility or liability of any kind for errors in these programs or schematics or for the consequences of any such errors. Furthermore, because of possible variation in the quality and condition of materials and workmanship of reader-assembled projects, *Circuit Cellar INK®* disclaims any responsibility for the safe and proper function of reader-assembled projects based upon or from plans, descriptions, or information published in *Circuit Cellar INK®*.

Entire contents copyright © 1996 by Circuit Cellar Incorporated. All rights reserved. Circuit Cellar INK is a registered trademark of Circuit Cellar Inc. Reproduction of this publication in whole or in part without written consent from Circuit Cellar Inc. is prohibited.

1 2 **SPARC Telco Applications**
by *Richard Pedreau*

1 8 **Caller ID Fundamentals**
by *Richard Newman*

2 2 **Vehicular Control Multiplexing with CAN and J1850**
Part 1: Vehicular Multiplexing Fundamentals
by *Willard Dickerson*

2 8 **The Embedded Sun**
Part 1: Introduction to the Hardware
by *Anindya Ray & Lee Hanson*

DSP in RISC Embedded Processors
by *Richard Pedreau*

Fujitsu's SPARClite Alternatives
by *John Burns*

3 8 **Embedding a Message-based System**
by *Pat Baird*

78 **Firmware Furnace**
80x86 Performance
Cache Crazyiness Redux
by *Ed Nisley*

84 **From the Bench**
LEDs Finally Fill the Rainbow
by *Jeff Bachiochi*

9 0 **Silicon Update**
Fuzzy PID-Pong
by *Tom Cantrell*

EMBEDDED PC

See **Pages 43-77** for our
Special Bonus Section

2 **Task Manager**
Ken Davidson
Political Arachnids

6 **Reader I/O**
Letters to the Editor

8 **New Product News**
edited by Harv Weiner

INSIDE ISSUE 69

98 **ConnecTime**
Excerpts from
the Circuit Cellar BBS
conducted by
Ken Davidson

112 **Priority Interrupt**
Not Just
Ferengi Values

81 **Advertiser's Index**

READER I/O

FREEDOM OF SPEECH

Steve's *INK 67* editorial reminded me of the astuteness of a tagline I saw recently: "Law is not always just, and justice is not always legal."

Since this issue went to press, I'm sure you've heard the brouhaha over the book *Hit Man*. I'm not in favor of murder for hire. But, I am incensed that some would deny me the right to obtain the knowledge of how to do it! Right now, the author is a pariah because his book was seemingly used as instruction for an actual murder.

But, if circumstances were different... say, for instance, that a foreign power took over our government and the book was used for a successful assassination of the usurpers; the author would be a hero.

Context is all. My right to know should not be taken away just because someone used that knowledge to do evil.

I was surfing the Internet the other day and managed to land on a Web page sponsored by legitimate, honest-to-God (pardon the pun) Satanists. I investigated, and when I found out that it was too wickedly serious for my taste and sensibilities, you know what I did? I refused to investigate further, got out, and left it behind.

In my opinion, that was censorship at its finest. Heck, I'm not even an elderly grandmother, and what I saw shocked me (but I do have breasts). I defined it as obscene. I had the freedom of choice and I exercised it, as I do every day.

You are so right that "unless we collectively head off the dim-witted thinking that government intervention and censorship are tools to preserve a free society, we are destined to lose a society and freedom worth preserving." I can decide for myself, thank you, and I can train up my children (well, he is grown now, but I did my best) in the way they should go. I don't need Big Brother's help.

Too bad I have to go to work and am rushed or I could have written you a letter worthy of another editorial. I just wanted you to know that "It Just Frosts My Chops" was greatly appreciated.

Pat Shields
via the Internet

LETTERS TO LETTERS

I would like to reply to a letter in *INK 67* from Jim Chaney regarding the Engine-Control System series (*INK 62-64*).

Mr. Chaney wrote, "Although Ed's two-coil ignition system is fine for drag racing, for a street application,

the increased plug wear over a four-coil system would be unacceptable. There's a need for a feedback of resistance at the spark plugs during various RPMs and load, which should also provide cylinder pressure calculations."

Buick has been using a distributorless ignition system with two cylinders per coil since the mid '80s. Spark plug wear was actually less in my Buick GN (turbocharged V6) than previous cars with traditional distributors. Now, practically all manufacturers use a similar system.

About measuring plug resistance, Chrysler uses a two-coil DIS system in the four-cylinder Neon. With the advent of OBD_II regulations, monitoring the plug resistance was required for misfire detection. The waste spark system has a negligible amount of resistance from the plug not under load. So, total secondary resistance is dominated by the actively firing (compression stroke) plug. The waste spark helps emissions, as the extra spark on the exhaust stroke can ignite the remaining hydrocarbons.

Keep up the good work, *INK!*

Dave Cooley
Wendell, NC

Contacting Circuit Cellar

We at Circuit Cellar *INK* encourage communication between our readers and our staff, so have made every effort to make contacting us easy. We prefer electronic communications, but feel free to use any of the following:

Mail: Letters to the Editor may be sent to: Editor, Circuit Cellar *INK*, 4 Park St., Vernon, CT 06066.

Phone: Direct all subscription inquiries to (800) 269-6301.

Contact our editorial offices at (860) 8752199.

Fax: All faxes may be sent to (860) 872-2204.

BBS: All of our editors and regular authors frequent the Circuit Cellar BBS and are available to answer questions. Call (860) 871-1988 with your modem (300-14.4k bps, 8N1).

Internet: Letters to the editor may be sent to editor@circellar.com. Send new subscription orders, renewals, and address changes to subscribe@circellar.com. Be sure to include your complete mailing address and return E-mail address in all correspondence. Author E-mail addresses (when available) may be found at the end of each article. For more information, send E-mail to info@circellar.com.

WWW: Point your browser to <http://www.circellar.com/>.

FTP: Files are available at <ftp://ftp.circellar.com/pub/circellar/>.

NEW PRODUCT NEWS

Edited by Harv Weiner

ADD-ON SERIAL PORT FOR SBC

memPORT provides that extra serial port so often needed for debugging single-board computers during program development or on site. Prior to memPORT, developers had to temporarily relinquish a port or build an extra port into every product.

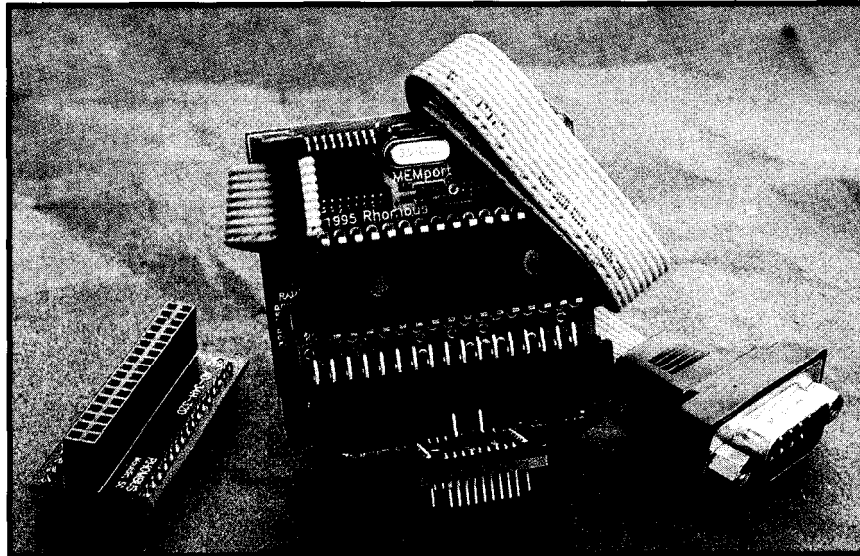
The 2" x 1.8" memPORT PC board is installed through an adapter into a 28- or 32-pin DIP or 32-pin PLCC memory socket. It contains a buffered UART, RS-232 level conversion, memory-mapping logic, and a DIP memory-replacement socket.

Through the replacement socket, the system continues to use the displaced memory IC, except for a small block of eight bytewise addresses.

In return, the system gains a memory-mapped RS-232 serial port with full-duplex operation at up to 115.2 kbps, double-buffer transmit, and a quadruple-buffer receive. Power, typically 35mA over and above that needed by the memory IC, is taken from the memory socket.

A flat cable assembly makes the transition from memPORT's 10-pin header to an AT-compatible DE-9 connector. Software guidelines are included for setting up the UART and for typical polled transfer routines. If the user's software-development program can't be customized, the port can be used for the application itself.

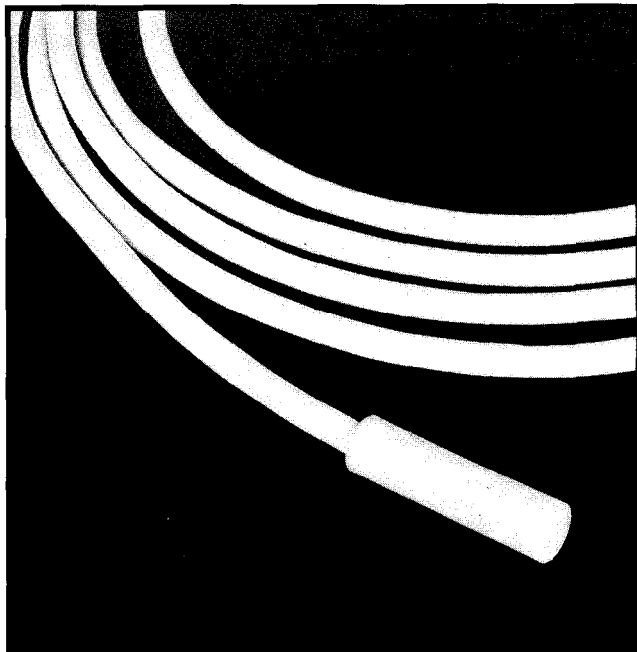
memPORT comes complete with transition cable for \$139. Memory socket adapters are available as a DIP for \$25 or as an economy PLCC version for \$30.



Rhombus

P.O. Box 871 • Mauldin, SC 29662 • (803) 676-0012 • Fax: (803) 676-0015

#500



LOW-TEMPERATURE SENSOR

The M2020 NTC Thermistor replaces traditional electromechanical devices with an electronic component that provides more sensitive temperature regulation. When the M2020 is used with a microprocessor interface, actual temperatures can be displayed. The sensor was designed specifically for refrigerators and freezers.

Encapsulated in a molded plastic case, the M2020 withstands harsh temperature conditions. Under test conditions of 1,000 h at room temperature in water, the change in the resistance value is less than 1%. The M2020 also achieves fast temperature cycling. At temperatures changing 100 times from -40°C to $+60^{\circ}\text{C}$, the change in resistance is less than 1%.

Siemens Components, Inc.
186 Wood Ave. S. • Iselin, NJ 08830
(908) 906-4300 • Fax: (908) 632-2830

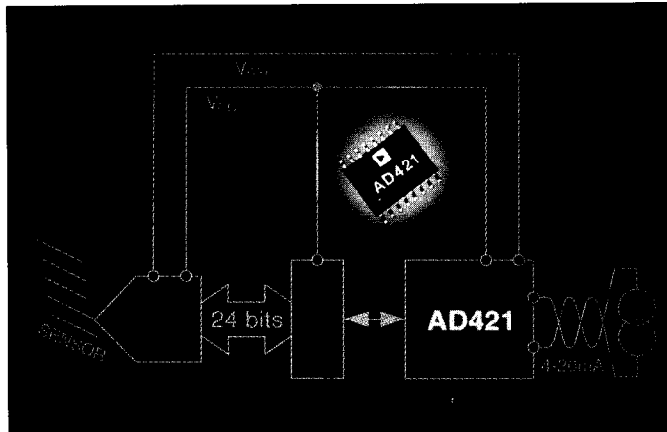
#501

NEW PRODUCT NEWS

LOW-COST DAC

To control a variety of digital-processing functions, the AD421 loop-powered D/A converter sends 4–20-mA signals to a microcontroller. It offers a zero-scale 4-mA output current with $\pm 0.1\%$ offset error and a 20-mA full-scale output current with $\pm 0.2\%$ gain error. Full-scale settling time to $\pm 0.1\%$ occurs within 5 ms.

The DAC is a high-precision, fully integrated, low-cost solution housed in a 16-pin package. It includes an on-board voltage regulator which provides +5-, +3.3-, or +3-V outputs as well as onboard +1.25- and



+2.5-V precision reference voltages. The device also features a high-speed 2-Mbps serial interface, a clock oscillator circuit, and a programmable alarm-current capability which lets the transmitter send out-of-range currents to indicate a transducer fault.

The AD421 is fully compatible with the standard Highway Addressable Remote Transducer (HART) protocol or other similar Frequency Shift Keying (FSK) serial-communications methods. This communication protocol allows simultaneous analog and

digital communication with intelligent field instruments and transmitters.

The AD421 is available in 16-pin DIP, 16-lead SOIC, and 16-lead SSOP packages. The part is specified over the standard industrial temperature range of -40°C to $+85^{\circ}\text{C}$ and costs \$6.95 in quantity.

Analog Devices, Inc.
One Technology Way
P.O. Box 9106
Norwood, MA 02062-9106
(617) 329-4700
Fax: (617) 821-4273

#502

RS-232-TO-V.35 CONVERTER

The Model 240 Universal RS-232-to-V.35 Interface Converter efficiently steals power from the RS-232 interface and provides a DTE- or DCE-switchable configuration. It includes a unique LCD display, called DataSpy, that informs the user of the status of the data and handshake signals included in the RS-232 interface.

Supporting data rates that range from DC to 100 kbps, Model 240 includes conversion circuitry for 13 signals. The power to drive the unit is derived from the interface signals (data, control, and clocks) on the RS-232 port. At a minimum, TD and one control signal are required. The Model 240 incorporates an externally accessible switch to configure the unit as a DTE or DCE device.

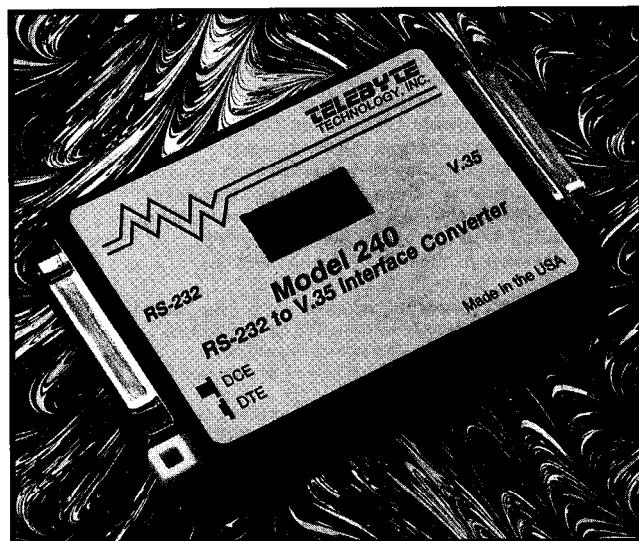
The DataSpy LCD display provides status information about the interface signals being processed by the converter. It operates from less than 1 mW of power and does not affect the operation of the Model 240. The graphic display presents the user with live status of the transmit and receive data signals (TD and RD) and control signals (CTS, RTS, DSR, DCD, and DTR).

The electrical interface for both ports is implemented in DB-25 female connectors. Each unit is supplied with a 10', male-to-male DB-25 extension cable which can be used on either the RS-232 or V.35 port. Additional cables are available.

Model 240 is packaged in a rugged metal case measuring 3.3" x 2.86" x 0.76" and sells for \$220.

Telebyte Technology, Inc.
270 Pulaski Rd.
Greenlawn, NY 11740-1616
(516) 423-3232 • Fax: (516) 385-8184

#503



NEW PRODUCT NEWS

STEPPER CHIPSET

The MC1451A intelligent-motion chipset supports up to four axes of electronic gearing. The master input for each axis is provided by quadrature encoder input signals. The output consists of pulse and direction signals. The chipset is programmed using any standard microprocessor by sending high-level motion instructions which are then interpreted by the chipset.

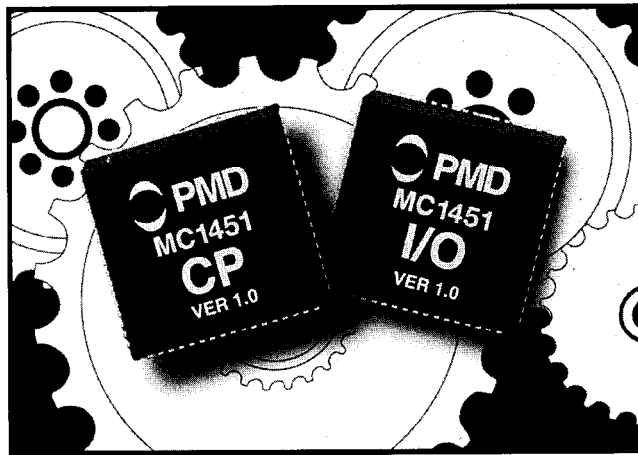
Other standard features of the chipset include 32-bit motion registers, programmable breakpoints, host interrupts, and three user-selectable profiling modes: S-curve, trapezoidal, and velocity contouring.

In addition, a special motor-stall-detection capability has been added. This feature uses the encoder feedback signal to determine when the motor has lost steps, even while the motor is in motion. When the chipset detects a stall condition, it safely shuts down the ongoing move to avoid system damage.

The MC1451A is packaged in two 68-pin PLCCs, with an optional 44-pin PLCC used for encoder feedback. Pricing for the four-axis version with encoder feedback is \$129 in quantity.

Performance Motion Devices, Inc.
97 Lowell Rd.
Concord, MA 01742
(508) 369-3302 • Fax: (508) 369-3819

#504



PRECISION, THREE-TERMINAL REFERENCE

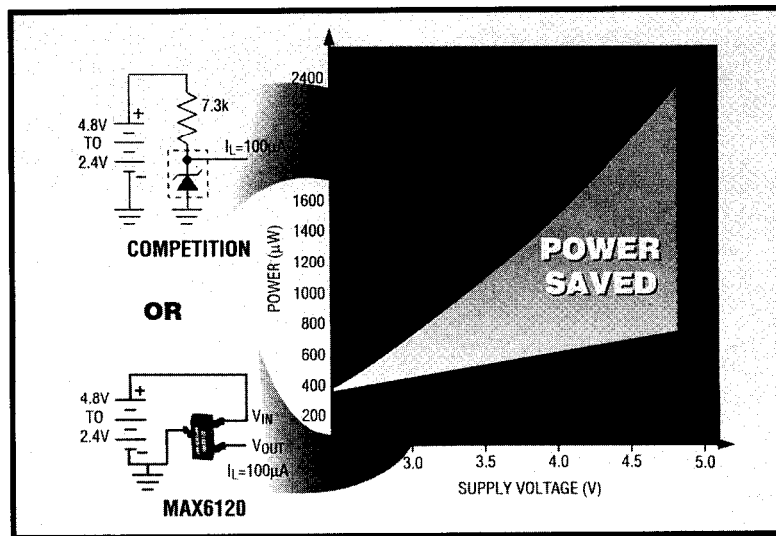
Maxim Integrated Products has released the MAX1620, the first 1.2-V micropower, precision three-terminal voltage reference offered in an SOT-23 package. Ideal for 3-V battery-powered equipment where power conservation is critical, the MAX1620 is a low-power alternative to existing two-terminal shunt references.

Unlike two-terminal references that throw away battery current and require an external series resistor, the MAX1620's 70- μ A maximum supply current (typically only 42 μ A) is independent of the input voltage, which means maximum efficiency at all battery voltages. In addition, it operates from a supply voltage as low as 2.4 V, and initial accuracy is $\pm 1\%$. The MAX1620's temperature drift is 100% tested in the SOT-23 package and guaranteed to be less than 100 ppm/ $^{\circ}$ C (typically only 50 ppm/ $^{\circ}$ C).

The MAX1620 is available in a 3-pin SOT-23 package, as well as an 8-pin SO package in the extended-industrial (-40° C to $+85^{\circ}$ C) temperature range. Prices start at 80¢ (1000s).

Maxim integrated Products
120 San Gabriel Dr.
Sunnyvale, CA 94086
(408) 737-7600
Fax: (408) 737-7194

#505



FEATURES

12

SPARC Telco Applications

18

Caller ID Fundamentals

22

Vehicular Control Multiplexing with CAN and J1850

28

The Embedded Sun

DSP in RISC Embedded Processors

Fujitsu's SPARClite Alternatives

38

Embedding a Message-based System

SPARC Telco Applications

In telecommunications, controllers are traditionally hardware based. However, Temic proposes a judicious mix of hardware and software. You gain the same performance for a fraction of the cost.

FEATURE ARTICLE

Richard Pedreau

Otemic's TSC701 Advanced Communication Controller, pictured in Photo 1, initiates a new challenge in embedded telecommunication applications. This innovative concept is built on three premises: increased flexibility, combined online signal processing and protocol handling, and maintaining performance at the system level.

New multistandards (e.g., for base stations or handsets), proprietary protocols, and rapid product evolution bring with them the need for increased flexibility which outperforms the usual hardware-based solutions. A software approach, to the extent that it reaches the same overall performance level, brings much more flexibility.

Emerging communication applications need signal-processing features (e.g., compression capability) combined with real-time protocol handling and the usual control skills on a single chip. The ideal is to include only the necessary DSP functions, thus achieving the best performance-to-cost ratio from a component standpoint.

The performance-to-cost ratio extends to the system level. Subsequently, the challenge is to preserve the performance level at the highest level while using low-cost peripherals like PC-type DRAM.

MULTISTANDARDS

The new telecommunication standards or multistandards originate in part from local regulatory organizations (especially in Europe) which force different rules from one country to the next. Because telecom equipment manufacturers are usually indus-

try giants, the advantages of a flexible approach—keeping the customization operation as late as possible in the process—are obvious.

Most OEMs want to start with globally usable hardware (including wireless base stations, PABX, ISDN adaptors, and even telephone sets) and simply customize it with different line-interface daughterboards and software. This approach reduces manufacturing costs while improving overall quality.

A second reason for the new multi-standards is the need to adapt the same equipment to different uses. An obvious example is the telephone handset.

The handset is slowly evolving to become a personal piece of equipment and not a community one. From one phone at home, one wireless, and one portable, the trend is toward one handset per individual, which can function both in GSM/PCN (portable) mode and DECT (wireless) mode for home or office.

This emerging general-purpose wireless handset must remain low cost while becoming at least five times more capable. In this area, these challenges pend:

- size constraints prohibit multicomponent solutions
- the silicon-area enlargement required for a comprehensive processor with full DSP implementation leads to an unacceptable cost increase for consumer electronics

Since a processor needs to control the equipment, it makes most sense to integrate chosen DSP features as coprocessors in a powerful but physically small RISC core.

Thus, the large computing power of the RISC CPU enables it to handle the integrated DSP coprocessors as another execution unit. Basic equipment management uses only about 10% of its CPU load, which is perhaps the only way such a versatile handset can be conceived.

This approach brings DSP functions back to where they left off over 10 years ago. Then, the lack of computing power in the main processor core of the CISC processors at the time of the

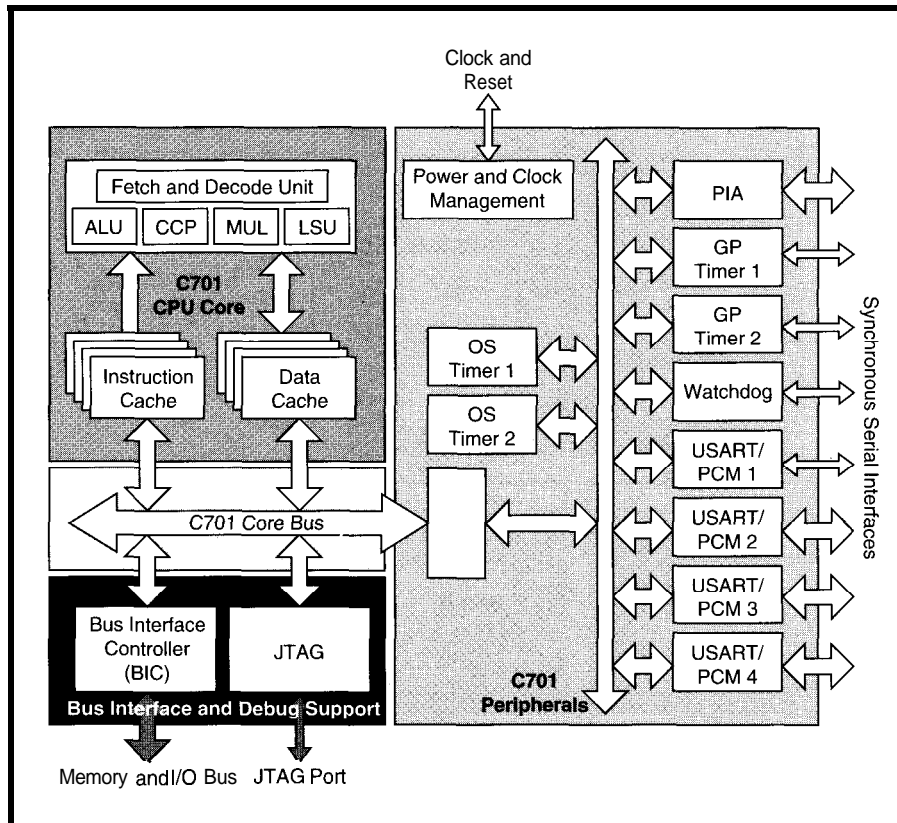


Figure 1—TSC701 combines online signal processing and protocol handling.

68k, 28000, and similar processors paved the way for specialized DSPs.

I don't mean to imply that DSP processors will become obsolete. On the leading edge of technology, there's always a place for high-end specialized DSPs. They're the only way to solve certain calculation problems. But, CPU architectures can usually come back a few years later with a "normalized" solution.

To enlarge the scope of this discussion, the tendency to actually integrate so-called "visual" instructions into the new processors (like Sun or Intel) is another example of the same evolution. Although until now this field has been restricted to highly specialized processors, in my opinion it will soon return to the standard processor area.

PROPRIETARY PROTOCOLS

Proprietary protocols are the norm in this field. Communications is an area where one finds very important multinational companies such as Alcatel, Nortel, Nokia, or Ericson. These companies logically have to preserve their know-how by using proprietary protocols. On the other hand, they

must also remain close to the standard to preserve the industry compatibility.

In dealing with proprietary protocols, there are some very practical advantages to a software approach like Temic's TSC701. First, this approach makes it easy to respond to slight variations in protocol parameters like CRC polynom, recognition flags, channel filtering, and so on.

Also, competitive software solutions should come with a comprehensive library of drivers in documented source format as well as directly implementable binary files.

Thus, the user has two advantages: fast time to market using a provided turnkey solution, and the ability to later customize the drivers to react to application upgrades.

SPARClet ARCHITECTURAL CONCEPT

The TSC701, whose components are diagrammed in Figure 1, combines online signal processing and protocol handling.

As a principle, SPARClet maximizes use of all hardware resources when it's logically possible to do so.

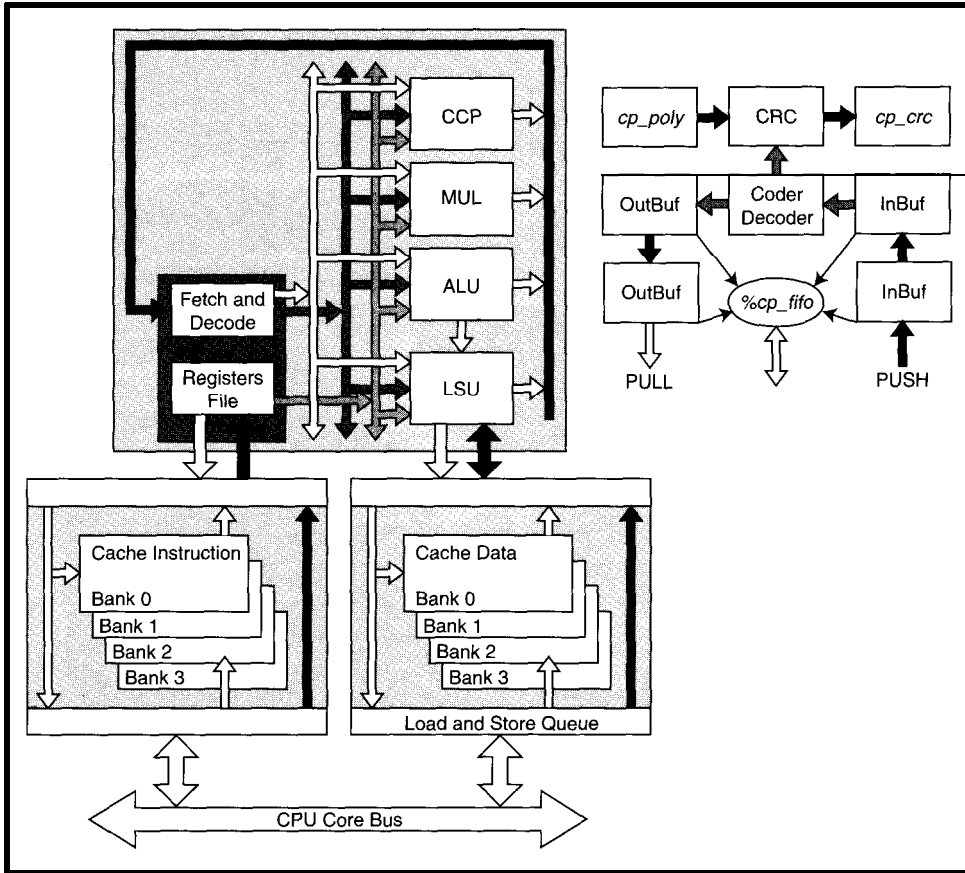


Figure 2—Full message coding and sending is accomplished by the TSC701 SPARClet core on the left using the process outlined on the right.

The first target, however, is not performance, but rather price to performance. In particular, resources are not duplicated to increase performance [i.e., it's not a superscalar design].

A resource (e.g., the adder) is used as soon as it is free and the operands are available. There are two reasons for an instruction to stall. The first is data dependency.

Because we want to proceed with an instruction as soon as possible and because operands often don't depend on the immediately preceding results, the general case to consider is that there is no data dependency.

Actual dependencies are checked to maintain compatibility with SPARC. When such a dependency is detected, one of the following behaviors is chosen depending on the situation:

- the dependent instruction is stalled, waiting for its operands to be available
- the operand is bypassed from a pipeline stage where it is available when it exists

The other reason an instruction can stall is resource dependency. Resources for most instructions are available every cycle. However, this is not always true.

For example, the multiplier is used for several cycles in order to multiply. When one multiplication follows another, the second one waits until the first is completed before proceeding to the multiplier.

Parallelism between instructions is especially visible for multicycle instructions (since SPARClet is not superscalar, single-cycle instructions are not executed in parallel). Long instructions include:

- integer multiplication (minimizes hardware)
- memory accesses on cache miss

Integer division and floating-point operations are not yet implemented but will be multicycle instructions when they are.

The same principles apply to the whole design, not just the SPARClet

core. For example, when the cache processes a miss, it's still available to process other requests from the core, especially in the case of a hit. Of course, the same hazard-detection mechanism is implemented to ensure data integrity.

Because SPARClet is coded in such a way that it can be parameterized, architectural enhancement can lead to gain in various areas, including:

- RAM speed
- power dissipation (it allows a lower frequency for the same processing power)
- chip area by reducing hardware resources (cache size, multiplier, etc.)

DSP CAPABILITIES

The emerging advanced communications systems require high-performance embedded devices to support new features such as real-time speech recognition or image and data compression. The half-rate GSM protocol, which requires an overall computing power of 40 MOPS, is an example.

In these new systems, special-purpose devices (DSPs) or coprocessors frequently act in conjunction with microcontrollers. However, due to increasing application complexity and system constraints, reduction in components is an important issue in system cost.

SPARClet is a general-purpose, modular architecture combining superscalar techniques, digital signal functions, and on-chip peripherals specially designed to address these requirements.

DSPs traditionally have few unique architectural features that set them apart from general-purpose processors. In fact, most of those functions can be handled by a general-purpose architecture.

SPARClet extends the general-purpose SPARC architecture to match these capabilities with a low-cost implementation target, which is a fundamental factor in embedded systems.

SPARCllet is a SPARC V8e-compliant architecture. The concept of this new implementation is based on a parallel but nonsuperscalar architecture which allows several operations to run in parallel. However, some superscalar techniques, such as out-of-order mechanisms, synchronize the infinite-input datastream.

Traditional processor architectures operate memory accesses, scalar instructions, and multiplication sequentially. So, they require fast computing and/or fastest access elements to achieve a high level of performance.

SPARCllet parallelizes these operations to relax the constraints of the operational units' intrinsic performance and the speed of external devices such as memory.

The best way to give a clear picture of the architecture is to browse the main features of a DSP and see how the SPARCllet architecture and the TSC701 address each one. DSP features include:

- single-cycle multiplier

- multiple operations per processor cycle
- zero overhead looping and circular buffers, implemented in specific instructions

Most of today's DSPs propose a single-cycle multiplier as a mandatory operator for digital signal processing. In reality, a software-pipelining approach can be used because such algorithms use a low percentage of multiplication (1030%).

Also, the necessary memory load-and-store operations can be inserted transparently during the multiplier latency cycles. In this way, high performance can be reached with a low-cost multiplier, which is linked to the silicon cost.

SPARCllet offers a parameterizable multiplier speed that is set based on the application's requirements. In the TSC701 implementation, the chosen configuration is a 32 x 32-bit, 4-cycle multiplier. It offers accumulation on a regular window register (64 bits which can be double length if necessary)

concatenated with 8 bits of the y register. An extra cycle is used in case the accumulation register changes from the last executed multiply-and-accumulate instruction.

As mentioned, SPARCllet is a non-superscalar implementation which fetches one instruction per cycle. However, the architecture allows multiple operations (like memory accesses, multiplication, and addition) to run in parallel.

A general-purpose architecture like SPARCllet can actually achieve zero-overhead features by unrolling loops and software pipelining.

In the case of SPARCllet, no specific instructions are required because the execution time for general-purpose instructions is masked by parallel execution flow. Circular buffers are already part of the original SPARC architecture.

On top of all these advantages, development is easier with a SPARC than a DSP chip because of the completeness of SPARC's development tools.

CIRCUIT CELLAR PROJECT FILE, VOL. II

\$17.95
SAVE 28% OFF
COVER PRICE OF \$24.95

All NEW Projects to SPARK Your Imagination!

GPZ8 Audio Sampling System
Wiring Your House for the 21st Century
Multiprocessor Architecture using DSP
ANDMUCHMORE!!!

VISA, MasterCard, or International Postal Money
Order (U.S. funds drawn on U.S. bank only)

Circuit Cellar Project File

4 Park Street

Vernon, CT06066

Tel: (860) 875-2199

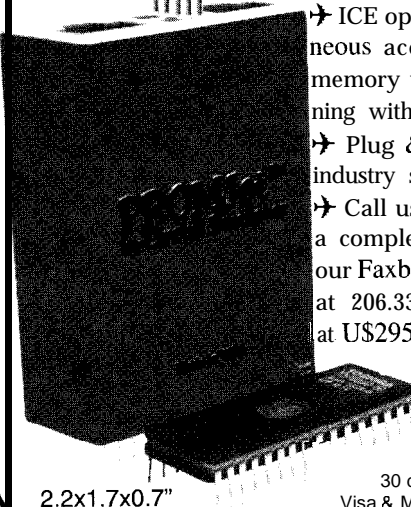
Fax: (860) 872-2204

*includes domestic delivery. Please add \$6 per copy for delivery to Canada & Mexico. add \$8 per copy for delivery to other non-U.S. addresses.

PROMJet[®]-ICE

→ Ultra compact EPROM and FLASH emulator with highest download speed (1-4 Mb/S), largest memory capacity (1-32Mb) and fastest access time (85~25ns) in the industry
→ Other features include 3V target support, jumperless configuration, battery backup, 128 bit bus support and external power supply. → Fits directly into memory socket or use extension cable for flexibility. → Compact design based on high density FPGAs and double-sided surface-mounted 1 layer PCB for added reliable operation

→ ICE option allows simultaneous access to PROMJet memory while target is running without waitstate signal
→ Plug & Play drivers for industry standard debuggers.
→ Call us at 206.337.0857 for a complete data sheet from our Faxback service or fax us at 206.337.3283. Price starts at US\$295 for a one Mbit unit



2.2x1.7x0.7"

EmuTec Inc

Everett Mutual Tower
2707 Colby Av, Suite 90
Everett, WA 98201, USA

30 day money-back policy
Visa & Mastercard accepted

TSC701 PROTOCOL HANDLING

The TSTSC701 microcontroller handles the HDLC protocol with a software-based method that is both innovative and flexible. There are three main benefits to the user:

- there's no physical limitation to the number of possible HDLC channels
- the CRC calculation can be customized, providing the ability to adapt the computation to proprietary protocols
- data transfer between USARTs, external memory, and communication coprocessor uses software DMA instead of a regular DMA channel

Thus, adaptation and filtering can be done locally in the software DMA routine, enabling the user to maximize the performance-to-memory tradeoff.

For example, if the transaction involves only one channel in a PCM frame, frame buffer size can be limited by filtering the data stream inside the transfer routine. A regular DMA channel would transfer the whole data stream, thus forcing a large amount of external memory to be used. The necessary filtering would be possible only as a postprocess.

The full message coding-and-sending chain is detailed in Figure 2. The first step involves the coding of the data stream and is performed by the internal HDLC coprocessor. This coprocessor performs the HDLC coding and CRC calculation at a rate of one bit per cycle. The TSC701 then stores the coded data by 32-bit words in an external frame buffer located in DRAM.

This background process is interrupted by a trap initiated from the PCM/USART transmitter. This trap

signals that its internal-transmission FIFO (16 bytes) has reached a level lower than the programmed limit.

As the trap has to be served with a highly deterministic behavior (to avoid any gaps in the transmission), the corresponding trap handler ought to be locked in the internal instruction cache.

This trap handler performs the software DMA function of transferring words from the external frame buffer to the transmitter FIFO.

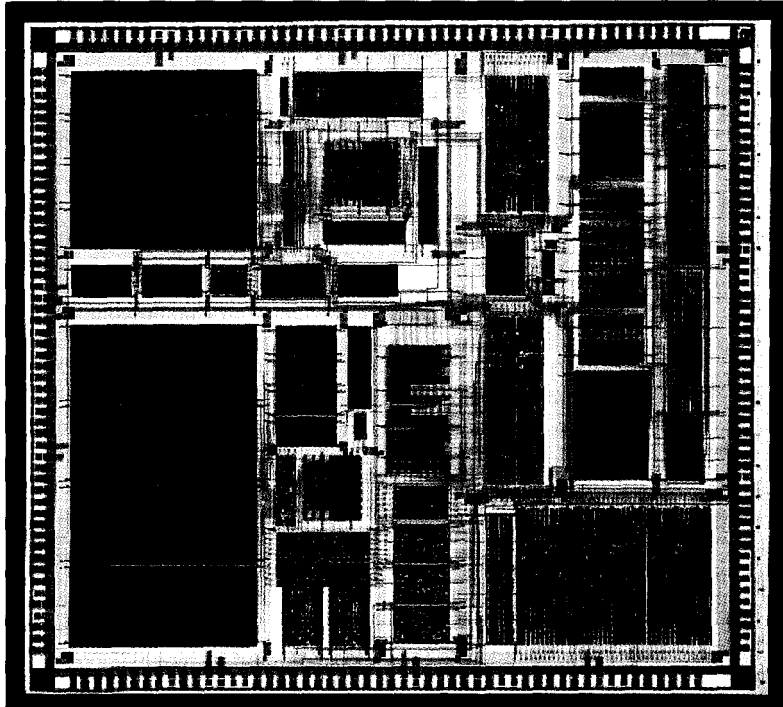


Photo 1—Temic's TSC701 Advanced Communications Controller shifts signal-processing features out of the hardware and into software in order to provide equivalent performance, greater flexibility and lower costs.

It must switch to a special 32-register window disconnected from the regular SPARC circular windowing structure to spare save-and-restore time loss. In this case, the window permanently contains the pointers necessary to access the frame buffer in DRAM and temporarily stores the transferred data.

Reception mirrors the transmission process. This mechanism provides maximum flexibility in CRC computation (the polynomial register is programmable by the user).

The CPU load used to handle the HDLC protocol remains low. Two full-duplex 2-Mbps links induce an overall consumption of about 10 MIPS out of

the roughly 50 MIPS available on the TSC701 working at 50 MHz.

BYPASSING SLOWDOWNS

SPARClet supports high throughput of the data stream in the user's system. Due to the parallel architecture, memory and I/O device access time has little impact on the computing performance of the processor.

The SPARClet I/O stream exploits internal features such as load-and-store buffers, which decouple the internal computing flow and external I/O accesses.

Data and instruction caches are sized quite high on the TSC701 to provide the best hit rate (16 KB of instruction cache and 8 KB of data cache). When there's a cache-memory hit, access time from the core is limited to one cycle. External memory consistency can be managed in Write Through as well as Copy Back mode.

SPARClet qualifies as a dataflow architecture in the sense that pipelining is extended up to the core bus and bus-interface controller. The core bus is based on a split-cycle mechanism (i.e., the

request and its completion are split).

Thus, the core is able to pipe requirements to the bus controller without waiting for the first access to complete.

SPARClet's core stalls only if a real data or resource dependency occurs. For instance, assume the program performs a load instruction from a memory location not present in the cache at the moment (i.e., a cache miss), then

- the transaction request is posted to the bus interface controller
- during the waiting time for the data to be available, subsequent instructions are executed unless

they use the data expected from memory. Only in this case does a stall occur.

These three features work together to minimize the impact of external-memory and I/Q-device access times. The embedded market drains high production volumes, so it's especially sensitive to manufacturing cost-not just the processor's cost but the cost of the whole system.

These parameters have been taken into account as major constraints when developing the SPARClet architecture and the TSC701 Advanced Communications Controller.

GLOBAL CONCLUSION

The three features highlighted in this article-software approach, DSP integration, and system cost reduction-provide processor manufacturers with a new challenge over the next few years. In addition to the industrial requirement for standardization, current trends are motivated by a strong market appeal.

Cellular-phone manufacturers expect a dramatic increase in sales in conjunction with a big price reduction. Staying competitive in this market means changes in design methods.

The Internet boom brings with it communication cards (fast modems) or ISDN adaptors in almost every individual PC. This equipment will soon move from the office and factory into the home office, which means production will increase by four or five times in less than a decade.

Inside a worldwide company, LAN interconnections were just an advantage a few years ago. More and more, they are becoming a necessity.

These market factors will force OEMs and silicon providers to solve the triadic cost, performance, and flexibility equation.

Integrating signal-processing functions, adapting to low-cost peripherals, and implementing a software-driven approach for processors address this challenge. It also maintains an acceptable path from the wafer-fab technologies' standpoint&

After receiving his Master in Electrical Engineering, Richard Pedreau worked as a test engineer for Philips. He joined Temic 11 years ago, where he has managed the microcontroller products engineering and technical marketing for the SPARC products division. Richard may be reached at richard.pedreau@matramhs.fr.

SOURCE

TSC701 Advanced Communication Controller
Temic
Matra MHS SA
"Les Quadrants"
3, avenue du Centre
78054 St-Quentin-en-Yvelynes Cedex
France
(33) 40 18 18 18
Fax: (33) 40 18 19 20

IRS

401 Very Useful
402 Moderately Useful
403 Not Useful

E4 EPROM EMULATOR

AMAZING Product.
AMAZING Price. Only \$249.
Supports EPROMs to 5 12Kx8.
Nothing extra to buy.

\$249

Supports all devices from
27C64 to 27C040 (4 MEG)

- *Powerful PC software tools.
- ◆ Full screen and command line modes.
- ◆ Supports all data formats.
- *Hardware error checking & correction.
- ◆ Software configurable.
- ◆ 100nS access time.
- *Power-off data retention.
- ◆ High speed downloading (LPT1-3).
- ◆ Non-intrusive CMOS LP design.
- ◆ Chain up to 8 units together - any configuration.
- ◆ Compact size with hard protective case.
- ◆ 1 year warranty & lifetime support.
- *Discounts on 2+ units.



Powerful tools, reasonably priced.

SDI

SCANLON DESIGN INC.

TEL: (800) 352-9770 Toll Free in
(902) 425-3938 North America
FAX: (902) 425-4098 Prices are in US Dollars.
EMAIL: 71303.1435@COMPUSERVE.COM
5224 BLOWERS STREET, HALIFAX, NS, CANADA B3J 1J7

Order today or Call or FAX for details.

(800) 352-9770

FEATURE ARTICLE

Richard Newman

Caller ID Fundamentals

As Caller ID has become popular, people want to know how to make the data available to computing devices. This introduction describes how information is transmitted, formatted, and decoded.



Caller ID service as provided by the regional telephone companies has been described by telephony types as being *the* application which will enable small home and office Windows applications.

While Caller ID boxes are available for about \$19 from national discount chains, there is not a single inexpensive Caller ID interface for the PC. Most implementations are multiline or add storage features which increase the price significantly.

In this article, I present an inexpensive, straightforward, and simple Caller ID decoder. You can connect a telephone line to one side of it and out the other to get standard serial data just as if it were coming out of an off-the-shelf modem.

Since data is delivered serially, you can handle it with a standard modem program set to hex decode mode or with a custom program that decodes the data into uniformly formatted fields.

For this project, I'm using the Motorola MC145447P Calling Line Identification (CID) Receiver with Ring Detector and colorburst crystal FOX036S. The Motorola CID chip is inexpensive (\$2.60) and has integrated ring detection. You don't need any external cir-

cuitry to determine when a call is arriving.

THE BASICS

Caller ID data is transmitted from your local telephone company office to your telephone line directly after the first ring. During this time, your telephone line is on hook, and there is no DC current flow.

The data is transmitted onto the high-impedance line, which is only AC terminated by the phone's bell in your home.

Since the telephone company's equipment is expecting to see a high-impedance state on your phone line, the interface of the Caller ID receiver must pick the AC audio signal off the telephone line without terminating the line and answering the inbound call.

All telephone company exchanges operate slightly differently because of the make of the physical equipment and version of the software running on the switch. It is therefore possible for unique incompatibilities to surface.

For example, information is transmitted right after the first ring and is complete before the second ring starts. If you answer the telephone after the first ring, you might still receive the Caller ID data. However, if you answer during the first ring, the exchange usually aborts the transmission of the CID data, losing the information for the call.

CIRCUIT DETAILS

The Motorola MC145447 CID chip has an analog front end, which interfaces to the telephone line with two 47- μ F, 200-V nonpolarized capacitors in series with two 10-k Ω resistors.

The input to the CID chip is differential. Because of this, it attempts to decode any differential (AC) voltages seen on the line, which results in occasional periods of unintelligible garbage.

The data transmitted from the telephone company is in standard Bell 202 format, similar to the format used by the old 1200-bps modems we all had a few years ago [Bell 212]. The data is transmitted at 1200 bps with 8 data bits, no parity checking, and 1 stop bit. It is asynchronous, serial, and binary.

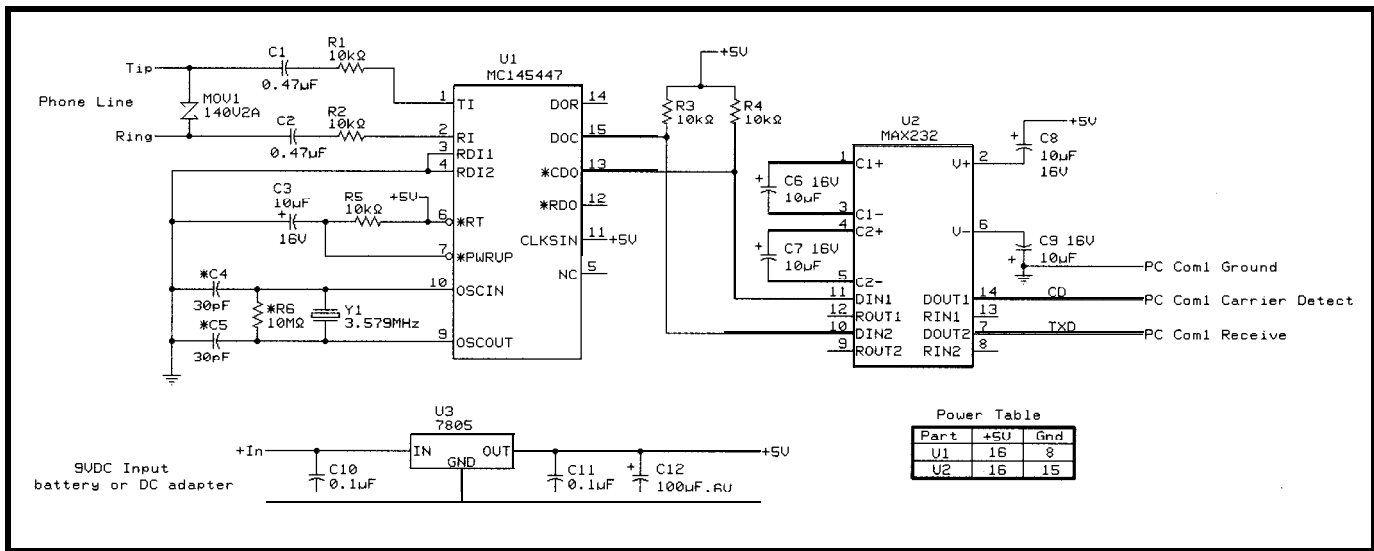


Figure 1-1 The complete circuit brings the phone line in one side and delivers PC-compatible serial data out the other

A logical 0 [called a space] is sent as 2200 Hz, and a logical 1 (called a mark) is 1200 Hz. You can see there is nothing special or proprietary about the data. It comes to you exactly as your PC would send it out its serial port.

If you're using DSP to decode the CID signal, the typical worst-case amplitude of the transmitted signal is **-13.5 dBm** from the telephone company. The facilities (the wires traveling though your town and ending at your house) introduce another **-20 dB** of attenuation.

This interference means that the CID receiver demodulator should be capable of decoding a worst-case signal of **-34.5 dBm**. I chose the Motorola device because it meets this worst-case specification.

If you look at the schematic presented in Figure 1, you see that the telephone line goes into one side of the CID receiver chip and data comes out the other side in inverted physical format.

If you take this inverted data and feed it into a MAX232 RS-232 driver/receiver, it inverts the data and translates it into RS-232 standard voltages.

At this point, you're ready to attach the RS-232 data to a PC or embedded controller and make a call to the chip. When the CID data is decoded, it is presented to the PC in an almost readable format. You can see the caller name, telephone number, time, and date along with some garbage data.

Since we aren't making a stand-alone box, but one that connects to a PC, I disregarded any features for power saving or ring detection. You should power the project from an isolated AC adaptor.

RING SIGNALS

The CID receiver is forced to stay active always because RD11 and RD12 are held low and *RT is held high. A standard delay-type reset circuit made from C3 and R5 makes *PWRUP go low shortly after power is applied to the circuit.

In this always-active mode, the data out of the DOC pin attempts to decode any signals on the telephone line, including ringing voltages and occasional DTMF signaling, during the dialing of outgoing calls. This data appears as garbage.

The circuit takes another signal out of the CID chip on 'CDO, which is only high when valid CID data is being received. The software of your system ignores and flushes all characters received when *CDO is low.

As soon as *CDO goes high, the software buffers all received characters in a queue. This technique ensures that the queue always contains usable data.

One discrepancy between the data-book description of the device and my real-world prototype is the drive capability of the DOC and *CDO pins. I found it necessary to apply a 10-k Ω pullup resistor for the MAX232 to receive the data correctly.

When you receive the Caller ID message, it can be up to 80 characters long and in one of two standardized formats called *fixed* and *variable*. The variable format is standard in North America and is what I will discuss here.

VARIABLE FORMAT

The variable-format service data is one long data package divided into subpackages. The first two characters received in the data package start with an 80 hex. The next char-

[0x80][0x27][0x01][0x08] 04301212 [0x02][0x0a] 2145554141 [0x07][0x0f] Caller's Name [0x01][0x01][0x00]

- 0x80—indicates the start of package indication
- 0x27—indicates the total number of characters (hex) to be transmitted in the main package
- 0x01 0x08—means that the date and time are coming next and are 8 characters
- 0x02 0x0a—signals that the calling number is next and is 10 characters
- 0x07 0x0f—indicates that the calling name is next and is 16 characters
- 0x01 0x01—is the checksum
- 0x00—marks the end of transmission

Table 1—An example of a complete caller ID stream from the telephone company includes name and number.

acter is the number of characters total to be transmitted.

The subpackages follow, starting with a character that indicates the type of subpackage (name, number, date, time, service) and the total number of characters in the subpackage. The subpackage types include:

- 0x01—date and time in DDDDTTTT format
- 0x02—calling number
- 0x07—calling name

Table 1 offers an example of a package and indicates what the separate components of the package stand for.

Your software should sync on the 0x80 character and be able to accept any package type next. There is no guarantee that the subpackages will arrive in a certain order nor that all subpackages will be sent in a particular package.

There are other subpackage types I won't elaborate on but which might indicate private or blocked calls. Typically, even if a subpackage meaning

private or **blocked** is sent, a number package is also sent with an ASCII "P" or "B" in the first character of the called number field. Your software should not always expect numeric data for the number field.

If you decide to apply this circuit to an application which doesn't have differential input, you should be able to couple the signal directly into the tip pin. Since you're not using a differential input, this signal needs to be twice the recommended amplitude to activate the demodulator section of the Motorola device.

If you find this hard to do in a single supply system, you could add an inverting op-amp to the tip pin and apply your signal to the input of the op-amp and the ring pin. This modification simulates a differential input to the chip from an externally provided single-ended input.

EXPECTATIONS & APPLICATIONS

If you expected this article to be deeply technical, you're probably thinking, "Gosh! This is really easy!"

Yes, it is. So, when you apply this circuit or specification to your system, if you use the Motorola chip as a caller ID decoding block, it should be almost plug-and-play.

All that's left is the application. You could have a window pop up a caller's name and number onscreen. This read-out could be juxtaposed with another window that holds notes about the caller from a database and include details such as account status.

Heh! Before you know it, you're enabled! You've found a perfect application for Caller ID and you. □

Richard Newman is an electrical engineer living in Dallas, TX. He designs specialized communications and industrial automation equipment either in partnership or on contract. He may be reached at ricardo@netcom.com.

IRS

- 404 Very Useful
- 405 Moderately Useful
- 406 Not Useful

HOME AUTOMATION

Software for Windows

Wireless Sensors

Rule-based Logic

Wireless Controls

CYBERHOUSE

Visit our web site for a complete product description and to get a working demo

<http://www.savoysoft.com>

or call 800-527-2853

SAVOY
Software Development

386 SBC \$83

OEM (1K) PRICE INCLUDES:

- 5 SER (8250 USART)
- 3 PAR (32 BITS MAX)
- 32K RAM, EXP 64M
- STANDARD PC BUS
- LCD, KBD PORT
- BATT. BACK. RTC
- IRQ0-15 (8259 X2)
- 8237 DMA 8253 TMR
- BUILT-IN LED DISP
- UP TO 8 MEG ROM
- CMOS NVRAM

USE TURBO C, BASIC, MASM RUNS DOS AND WINDOWS EVAL KIT \$295

\$95 SINGLE PIECE PRICE

UNIVERSAL PROGRAMMER

- DOES 8 MEG EPROMS
- CMOS, EE, FLASH, NVRAM
- EASIER TO USE THAN MOST
- POWERFUL SCRIPT ABILITY
- MICROCONT. ADAPTERS
- PLCC, MINI-DIP ADAPTERS
- SUPER FAST ALGORITHMS

OTHER PRODUCTS:

8088 SINGLE BOARD COMPUTER OEM \$27	• Q5
PC FLASH/ROM DISKS (128K-16M) 21 75
16 BIT 16 CHAN ADC-DAC CARD *	55 195
WATCHDOG (REBOOTS PC ON HANGUP) 27 95

*EVAL KITS INCLUDE MANUAL BRACKET AND SOFTWARE. 5 YR LIMITED WARRANTY FREE SHIPPING HRS: MON-FRI 10AM-6PM EST

MVS MVS BOX 850 MERRIMACK, NH (508) 792 9507

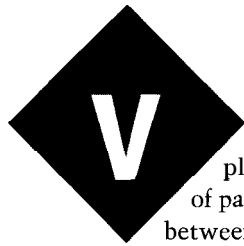
FEATURE ARTICLE

Willard Dickerson

Vehicular Control Multiplexing with CAN and J1850

Part 1: Vehicular Multiplexing Fundamentals

In this article, Willard lets engineers in on what vehicular multiplexing is, how systems interact, and how protocol is established. Why? So your automatic windows don't take precedence over your ABS.



Vehicular multiplexing is a means of passing information between vehicle control modules and/or subsystems through a serial data link. The link is typically one or two wires shared among several modules [also called nodes].

Link sharing is facilitated by placing a special vehicle multiplexing control unit at the interface of each node. The main computer-controller communicates with several distributed nodes through the same port.

The nodes are automobile modules such as sensors, ABS, audio system, traction control, multi- or single-point injection (gasoline engines), diesel injection (diesel engines), cellular telephone, cruise control, and so on.

This series overviews vehicular control multiplexing and evaluates the Motorola embedded controllers (the MC68HC708AS20 in J1850 and the MC68HC05X16 in CAN) in vehicular multiplex devices. Part 1 describes vehicle multiplexing as well as the J1850 and CAN protocols.

In part 2, I'll overview both the MC68HC708AS20 and '05X16. I'll conclude with how these controllers are implemented in their respective J1850 and CAN networks.

VEHICLE MULTIPLEXING ORIGINS

The concept of vehicle multiplexing comes from the computer-architecture technique of local area networks. In this concept, different nodes or modules share the same connection(s) for data communication.

Each node in a distributed system does not require a separate port into a main computer. As a result, fewer wires are needed to communicate between units. This concept has been used extensively in military aircraft, heavy-duty trucks, and factories.

Since the increase in vehicle electronics resulted in excessive, bulk-wiring harnesses measuring several kilometers, automobile manufacturers recently standardized reduced-wire multiplexing in passenger vehicles.

DEFINING THE PROTOCOLS

J1850 or CAN can be described from three vantages:

- as a class of multiplex system
- in its layers
- as fields of information in its message structure

There are three main classes of vehicle-multiplexing systems: A, B, and C. Class D is currently being defined. The Society of Automotive Engineers (SAE) characterizes classes by transfer rates, recommended uses, and intent.

Class A defines vehicle-multiplexing protocols that support transfer rates up to 10 kbps. This protocol is typically used in trip or mileage computers, electric windows, solenoid-driven switches, stepper-motor driven devices, entertainment modules, and so on. It primarily reduces cost, power, CPU throughput, and EMI.

Class B protocols accommodate transfer rates in the range of 10–125 kbps. They are typically used in engine and transmission control functions and cluster data passing. They are also used for general-purpose applications and legislated diagnostics (in accordance with California regulations expected to become law across the U.S. by the year 2000).

Class C protocols support transfer rates from 125 kbps to 1 Mbps. These

rates are typically used in advanced engine-control functions (e.g., variable valve timing and fine-gear correction), ABS, and suspension damping. Class C is intended for systems requiring a higher level of speed, intelligence, and safety than Classes A and B.

If you look at vehicle multiplexing protocols in terms of layers, each layer describes a predefined set of physical, electrical, or software characteristics.

For example, a physical layer describes the number and length of lines needed to communicate data at a certain speed. With the CAN protocol, to transmit 1 Mbps, you have a maximum line length of 100 m.

The type of transmission buffers required for a protocol is also part of the physical layer. In contrast, the size-of-message field is described in another layer. Each protocol has unique requirements for message-field sizes and arrangements.

Finally, the vehicle multiplex system can be described in terms of its message structure. This structure defines the number and size of each field, the type of information in the fields, and how they are recognized in a message.

For example, the message structure of Class A and B protocols is not as complex as Class C since Class C protocols facilitate control over more tasks in a shorter duration. Class A protocols have lean message schemes.

Conversely, the complexity of Classes B and C depends on their respective applications. Class B tends to communicate with a wider variety of modules and is typically the most complex message scheme.

Notably, the truck and bus vehicle multiplex committee has taken the J1850 (Class B) and put it on top of CAN (Class C), thereby making a more complex version of these protocols.

Class A and B direct functions like automatic window motors, switches, and simple LCD displays rather than functions requiring substantially faster bit rates and more diagnostic data. However, Class C data link controllers are capable of such simple applications, too.

MULTIPLEXING SCHEMES

Vehicle multiplex schemes typically provide more protection against noise and signal corruption than older serial protocols such as the SCI and SPI. As a result, vehicle owners have lighter weight cars with more reliable communication links.

Typically, vehicle-multiplex data-link controllers use fewer connections and provide lower susceptibility to automotive-related interference than

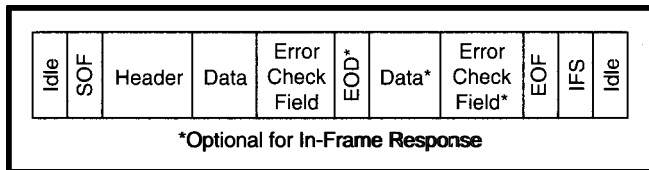


Figure 1—The *in-frame response* is required for the J1850 message-frame format for the 41.6 kbps PWM encoding.

the simpler, more traditional data links.

A multiplexing scheme identifies which node can communicate on the shared link at a given time. There can be seven nodes on a single link. Vehicle multiplexing can take place via frequency division, time division, token slot, or token ring.

A scheme's protocol specifies how to implement the vehicle-multiplex model. It provides a set of rules for transmitter and receiver communication. Frequently, it includes error checking, acknowledge methods, signal rate, and signal encoding.

The protocol is implemented by a protocol handler, which consists of both hardware and software, depending on the task's complexity. Less complex tasks are often done in software. Sometimes, when vehicle-multiplex circuitry is found on embedded con-

trollers, internal hardware and software resources implement vehicle-multiplex protocols.

The partitioning of the multiplex protocol is determined by available hardware and software resources. The partitions provide an overall project structure. It determines whether the multiplex controller is implemented in an embedded controller or is stand-alone, what application to load in the CPU, what incremental loading is necessary on the CPU from each protocol layer, and what the cost goals are. In many cases, partitioning tradeoffs are made to meet cost goals.

Additionally, simulation can help determine the most suitable partitioning for a problem. For instance, a verilog simulation examines CPU use in a protocol-layer application. If the layer's throughput exceeds projected goals, then alternate strategies are examined.

Error-checking schemes involve parity checks, cyclic redundancy, noise sampling, as well as simple or complex error-handler routines.

The timing methods used in vehicle multiplexing are either synchronous or asynchronous. Data is transmitted serially by one of a variety of methods, which can include either communication mode.

Data flow for these methods can be simplex, half duplex, or full duplex. Whichever method is chosen, the actual bit-rate clock is not transmitted on a separate line but is embedded in the data transmitted.

Synchronous timing provides a known timing relationship between

application layer—where legislative diagnostics are found. Standard messages provide information about the condition of systems affecting vehicle emissions.
 presentation layer—consists of the addressing strategy, diagnostic codes, and their parameters. The addressing strategy has both physical and functional modes. Diagnostic codes and parameters are determined by legislative requirements.
 session layer—places the system in an idle or sleep power-saving mode, or it can wake up and alert the system when signals are present.
 transport layer—includes message screening and filtering for hardware and software and buffering for the bit, byte, or message level.
 network layer—involves nondestructive arbitration in which the priority of a node is determined by the message.
 data link layer—involves bus communication, message format, synchronization, and requirements for response to the message or message errors. See the text for more information on the J1850's message frame format.
 physical layer—describes hardware performance aspects of the system such as bit rate, bit encoding, drive type, redundancy, and media.

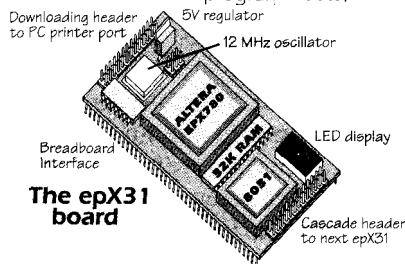
Table 1—The seven expanded layers for the J1850 can be embedded into three layers which are used in practice

Flexibility X2 UC + FPGA

Microcontrollers let you adapt your designs to changing requirements. Now, Field Programmable Gate Arrays can double your flexibility by letting you change your product's circuitry on-the-fly. But how do you quickly learn the tricks of combining FPGAs with microcontrollers?

The epX31 is your answer! It has everything you need to design and test microcontroller+FPGA applications:

- ★ epX31 – the microcontroller used in thousands of embedded designs,
- ★ EPX780 – a RAM-based FPGA with 3,500 reconfigurable logic gates, 32KB RAM – for programs+data;



The epX31 board

The FPGA amplifies the capabilities of the 8031. Program the EPX780 to create custom address decoders, new Interrupt sources, additional timers, specialized coprocessors, real-time bus monitors and more! The only limit is your ingenuity!

It's easy to use the epX31. You get a complete tutorial on FPGA design with our *FPGA Workout* text. Or modify one of our design examples for your own application using the PLDSHELL FPGA programming tools and 8031 assembler included with the epX31. Next, load the FPGA configuration + 8031 object code into the epX31 through the PC parallel port. Then apply real-world test signals to your design under the control of your PC.

The epX31 grows with your designs. You can cascade several epX31s to build multiprocessor or multi-FPGA systems. Or connect special-purpose chips to the epX31 through the breadboard interface.

The epX31 lets you try out more design ideas with less effort. No more untangling wire-wrap or patching printed circuit boards. Just reprogram the static RAM and you're ready to try again. That will put a smile back on your face from 9-to-5! So will the price!

epX31 \$249

- ✓ epX31 board
- ✓ PLDSHELL
- ✓ runtime modules
- ✓ FPGA Workout text
- ✓ download cable
- ✓ 8031 assembler
- ✓ runtime utilities
- ✓ design examples

XESS Corp.

USA: (800) 549-9377
INTL: (919) 387-0076



Check our Web site for FPGA tutorials!
<http://www.xess.com>

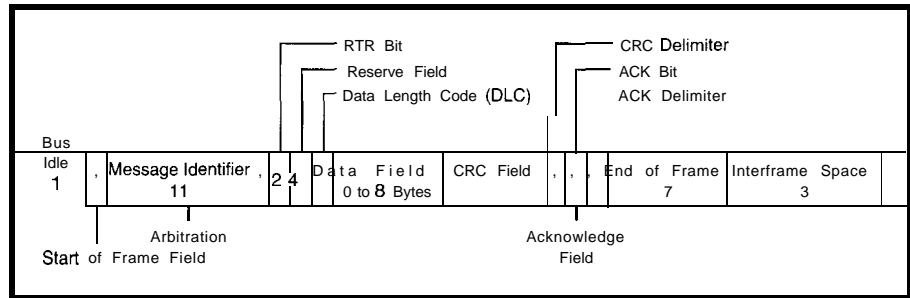


Figure 2—The CAN message frame format includes a parity bit in the CRC field.

the bits and bytes. The bytes are transmitted continuously until a condition halts or suspends their transmission.

The receiver recovers the clocking from transitions in the data. This technique allows fewer lines to be used than standard SPI controllers, which usually provide a separate line for the receiver clock.

Conversely, asynchronous controllers have a variable relationship between bits and bytes. That is, each bit can be transmitted separately or in groups. Typically, a start bit synchronizes the transmission.

ENCODING

Vehicle-multiplex messages are usually encoded to:

- reduce emissions in a harsh under-the-hood environment
- improve recognition of bits during arbitration for single-wire systems
- reduce physical media costs by facilitating single- or dual-wire systems

The common encoding types include NonReturn to Zero (NRZ), Pulse Width Modulation (PWM), Variable Pulse Width Modulation (VPWM), Frequency Modulation (FM), Modified Frequency Modulation (MFM), and Manchester.

The method of encoding is determined by cost, bandwidth efficiency, and the EM1 imposed on a given system. Cost increases with the more complex devices and circuits of certain schemes. Bandwidth efficiency becomes critical for higher-speed multiplex protocols such as CAN. The EM1 is essentially affected by the number of transitions per unit time a waveform generates and data encoding.

NRZ encoding represents data with unipolar (above ground) or bipolar

transitions. The width of a bit representing one or zero is the same, which allows infinite bits per transition. No edges are present if the same bit is asserted continuously.

The NRZ scheme offers a low number of transitions per bit (hence low emissions) and fixed transition or sample points. The bipolar version requires twice the voltage swing as the unipolar scheme.

On the other hand, NRZ accumulates clock errors among nodes, making it harder to synchronize the bits. This characteristic is a major nuisance in vehicle multiplexing since the clock is not transmitted on a line separate from the information. This problem can be solved by bit stuffing.

Bit stuffing guarantees at least a 1-bit transition every 5 bits. A bit of the opposite polarity is inserted each time five consecutive 1s or 0s are detected. The receiver understands and uses the same rules by deleting the inserted bits from the stream.

PWM encoding represents each bit by varying the pulse width of periodic signals from one-third to two-thirds. The shape of the pulse and approximate locations of the edges are fixed during each periodic signal. It can use two edges on the data link for each bit transmitted. This encoding is commonly found in J1850 equipment with data rates of about 10.4 kbps.

PWM offers defined sample points, fixed bit lengths, the ability to arbitrate wired or contention buses, and the ability to resynchronize all receivers on a rising edge of each bit.

Conversely, PWM is less cost effective than an automotive data link (i.e., single wire). The dual edges increase radiated emissions. It's therefore more difficult recognizing bits, especially with extreme ground offset.

VPWM represents a binary signal by varying a pulse within periodic boundaries. Unlike PWM, however, the variations in the pulse and edges can alter within each periodic pulse.

FM is represented by periodic clock pulses signifying a one if intervening time slots are pulsed or a zero if no change occurs. This technique adversely affects radiated emissions since several edges can potentially be generated at higher frequencies.

MFM encoding is similar to the FM, except that it eliminates the clock pulses unless the data remains constant for more than two consecutive bits. The pulses can be replaced with transitions. That is, a one is signified by a transition, and a zero by no transition. Unfortunately, there is more complexity distinguishing between clock pulses and logical transitions. This problem can be resolved due to the 41.6- μ s bit times for J1850.

Additionally, arbitration is more complex with MFM than FM because for any given bit position, either a one or a zero has higher priority depending on the previous bit stream.

Manchester encoding is represented with transitions. It defines various fixed bit times (e.g., 96 μ s for 10.4 kbps) and forces a transition at each defined boundary. A one is signified by an additional transition triggered in the middle of a bit time, and a zero by an unchanged pulse during a bit time.

51850 OVERVIEW

J1850 is a one- or two-wire serial protocol for low- to medium-speed vehicle-multiplex applications. The lower transmission speed (10.4 kbps using VPWM) is implemented in a one-wire system and the higher transmission speed (41.6 kbps using PWM) in a two-wire system [1]. It falls under the Class B protocol, and emissions are in-between those of NRZ and PWM.

This protocol or variations of it— is used by domestic automotive companies. It controls such devices as window motors and lock solenoids, digital instrument display, antilock brakes, and fault communication [2].

As with other vehicle multiplex schemes, this protocol consists of a synchronous multimaster bus system. Multiple units connect to the same bus, and any unit can request control of the bus. Through arbitration, one unit is selected to master the bus.

This protocol offers:

- open architecture
- moderate complexity
- single-level bus topology
- multimaster peer-to-peer
- legislated diagnostics

As an open architecture, J1850 allows prioritization of frames and is compatible with CSMA/CR.

Moderate complexity reduces cost because it requires less hardware and software, which in turn decreases development and maintenance costs.

A single-level bus topology provides one link so all nodes transmit and receive from a single path. It receives all frames simultaneously.

A multimaster system enables multiple nodes to request access of a bus (i.e., any node can potentially be the master). This approach also reduces hardware and development costs since no special hardware and software is required for a separate or additional master node.

Legislated diagnostics consist of automated tests for vehicle emissions or any other environmental test required by law. These tests go federal in 1997.

Layers provide a standard means to categorize and describe fundamental hardware and software architectural characteristics of a vehicle-multiplex

physical layer-consists of its drive capability, bit level, and format and transmission medium.
transfer layer-includes message framing and arbitration, error detection and report, and fault confinement.
object layer-includes message buffering, acceptance filtering, and prioritized message handling.
application layer-presents the hardware and software trade-offs which are dictated by the specific application.

Table 2—The CAN layer information includes four basic layers.

Running under Windows?

Using a standard data acquisition

board is like using an old

typewriter.

They both get the job done, but ...

there is a better way.

Standard data acquisition boards can unknowingly sabotage your data. Ensure the integrity of your results.

ADAC's Windows Optimized 5800 Series gives you the resources you need: FIFOs, Channel Gain RAM, Dual DMA, aggressive prices, and some of the best noise performance in the industry!

5801MF: 16 channel 12-bit A/D, 333KHz, 2 16-bit D/A, 40 digital I/O

5803HR: 16 channel 16-bit A/D, 100KHz, 2 16-bit D/A, 40 digital I/O

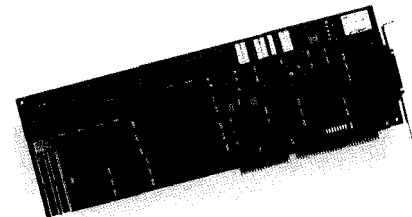
learn more -

voice 800-648-6589

fax 617-938-6553

web <http://www.adac.com>

email info@adac.com



ADAC

American Data Acquisition Corporation
70 Tower Office Park, Woburn, MA 01801 USA

#112

Circuit Cellar INK® Issue #69 April 1996

25

system. Table 1 illustrates the seven layers of the typical J1850 system can include.

In practice, however, these seven layers reduce to three. The presentation layer embeds in the application layer, just as the session, transport, and network layers are a part of the data link layer. Only the physical layer stands alone.

The message-frame format is required for the 41.6-kbps PWM and not for the 10.4-kbps VPWM [1]. As you can see in Figure 1, the in-frame response format consists of a start-of-frame signal (SOF); header, data, and error-check field; end of data (EOD); more data; and end-of-data-frame (EOF) check field [3]. Notably, the error field is optional with the in-frame response format for 10.4-kbps VPWM [1].

A standard J1850 system requires the examination of four types of errors: cyclic redundancy check (CRC), frame length, out of range, and invalid bit, byte, and symbol detection [4].

The first two errors are typically decoded by examining the number of bits or bytes within partitions of a transmission. However, the latter two error types are detected through a form of digital or analog filter.

The message structure of the J1850 protocol is partitioned into header and data bytes. Standard J1850 allows a maximum of 101 bit times or 112 bytes per message, excluding the SOF, EOD, NB, and EOF fields. The header consists of three bytes which include an introduction to the type of message, the functional or physical address of the receiver, and the physical address of the sender.

The final partition consists of three sections of data information. The first two sections present 0-8 bytes while the last two sections consist of three bytes. The first two sections of data provide information about the type of message ID, the message ID number, and control signals that point to the proximity or vehicle section a message

is sent to (i.e., under the hood, mid-vehicle, rear trunk, etc.)

The final section of data pertains to three bytes of control and status information. The control information is acted on by the parameter owner, and the status is used by the parameter-monitoring or controlling nodes.

The common arbitration method used for the J1850 bus is nondestructive. The message with the highest priority is transmitted, while transmitters losing arbitration simply cease transmitting until they receive an idle bus transmission.

The maximum number of nodes on the J1850 bus depends on bus speed, wire length, and drive strength. For example, 32 nodes can be driven on a 41.6-kbps, 40-m bus.

CAN OVERVIEW

The CAN protocol originated over 14 years ago in Germany by Bosch GmbH. This fairly complex, high-speed protocol offered a solution to reduce the size of wiring harnesses and power for distributed loads while improving the noise susceptibility between these nodes.

The CAN protocol can be described by the same characteristics as the J1850: speed-performance, advantages, layer information, message format, and so on. The CAN protocol is considered suitable for, but not restricted to, a Class C controller.

Automotive manufacturers use this complex protocol because it's suitable for relatively complex operations (e.g.,

engine or traction control). It handles the number of messages required at a latency sufficient for the algorithms of these applications.

At times, however, automotive manufacturers use a simplified subset of CAN for less stringent applications like controlling window or seat adjustment motors, or door-lock solenoids.

CAN is commonly found in European vehicles. Typically, it is more expensive to implement than J1850 because of its complexity. It can be used for applications such as low-level laptop computer communication, engine control, ABS communication, and the applications common to J1850.

The CAN protocol provides fault-tolerant requirements at transmission rates of 125 kbps-1 Mbps.

The CAN protocol can also be described in terms of layers [5]. Table 2 defines its four basic layers: physical, transfer, object, and application.

CAN messages are communicated through four frame types: data, remote, error, and overload.

The data frame communicates data between nodes. It consists of a start bit or signal, 1 1-bit message identifier, remote transmit request (RTR) bit, reserved bit, 4 bits for data-length code (DLC), data field consisting of 0-8 bytes, 16-bit CRC field, two acknowledge bits, and an end-of-frame field. Figure 2 depicts a data frame.

The message identifier represents an address that can be a physical, functional, or combination. The DLC represents the length of the data field.

Features	J1850	CAN
Common bus waveforms	VPWM, PWM	NRZ
Number of basic frame formats	1	4
Number of bus wires	1 (VPWM) or 2 (PWM)	2 wire, twisted pair
Maximum frame length	101 bits or 12 bytes	<100 bits
Bus rate	10.4-41.67 kbps	126 kbps-1 Mbps
Arbitration method	priority ID encoding	priority ID encoding
Error checking (8 bits basic)	CRC (15 bits)	CRC (15 bits plus parity)
Low-power modes (implementation dependent)	suspends clock and tristates the bus	suspends clock and tristates the bus
Number of descriptive layers	3 (7 expanded)	4
Methods to initiate transmissions	bus wakes device from low-power mode, receive or send request for transfer, or external control asserts control signal, e.g., a CPU writes to a control register	bus wakes device from low-power mode, receive or send request for transfer, or external control asserts control signal, e.g., a CPU writes to a control register
Limits on bus length	40 m (35 ft for tester) at 41.6 kbps	130 m for 50 kbps or 20 m for 1 Mbps
Method of selecting receivers on bus	message IDs broadcast and arbitrated across bus	message IDs are broadcast and arbitrated across bus
Maximum number of nodes on bus	32 at 41.6 kbps	100 at 50 kbps

Table 3—J1850 and CAN are similar in some respects and quite different in others.

The acknowledge bits inform the transmitter that at least one node received the data correctly.

The remote frame is the same as the data frame, except that it does not include the data field. The frame is used by the receiver to request data from the transmitter. This data is sent in the subsequent data frame. The message ID of this data frame matches the requesting remote frame's ID.

The RTR bit determines whether a frame is data or remote. Data is indicated by the dominant RTR bit, and remote by a passive RTR bit.

The error frame indicates an error if an undesired condition is detected by a node. The detected errors from data or remote frames are transmitted after the respective frame.

Finally, the overload frame invokes additional delay between adjacent data frames and remote frames.

The common arbitration methods used for the CAN protocol establish a polling or priority scheme between the various ID fields found within the data or remote frames.

The number of the units or nodes found on a CAN bus are determined by the drive strength of the CAN nodes since each can be a master and by the wire length that attributes to loading.

Typically, CAN is on two-wire bus connections. Sometimes, a one-wire scheme is used for limited applications. The two wires are twisted pair without a separate transmitted clock or with one wire transmitting the data and the other a synchronous clock.

J1850 AND CAN PROTOCOLS

The major aspects of the J1850 and CAN protocols are in Table 3.

Next month, I'll overview the processors embedded in J1850 and CAN to see just how they're implemented in their multiplexing schemes. □

Willard Dickerson is a design project leader for 5xx RISC-embedded controllers in Motorola's Advanced Microcontroller Division. He develops hardware and firmware for 8- and 32-bit embedded controllers. He may be reached at willd@nambe.sps.mot.com.

REFERENCES

- [1] "SAE Recommended Practice J1850 Class B Data Communication Network Interface," 1993.
- [2] M. Nagao, et. al., "Bus Driver IC for Use in Vehicle Multiplexing Communications," *Proceedings of 5th Annual IEEE, 79-82, 1992.*
- [3] Mark P. Zachos and A.J. Pohlmeier, "Message Structure and Strategy to Drive SAE J1850 Networks: An Introduction to SAEJ2178," SAE International Congress and Exposition. Detroit, MI. Feb. 24-28, 1992.
- [4] Motorola, *HC08 Central Processing Reference Manual, Manual CPU08RM/AD, 1993.*
- [5] Sarjay Gupta, "CAN Facilities In-Vehicle," Society of Automotive Engineers, 1990.

IRS

- 407 Very Useful
- 408 Moderately Useful
- 409 Not Useful

CONTROL




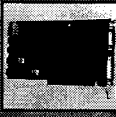


RELAYS • LIGHTS • MOTORS

MEASURE

TEMPERATURE • PRESSURE • LIGHT LEVELS • HUMIDITY

INPUT

SWITCH POSITIONS • THERMOSTATS • LIQUID LEVELS

MODEL 30 \$79 	MODEL 45 \$189 
MODEL 100 \$279 	MODEL 60 \$179 
MODEL 40 \$99 	MODEL 70 \$239 

NEED A CUSTOM PCB? TRY US.

PRAIRIE DIGITAL, INC.

PHONE 608-643-8599 • FAX 608-643-6754

846 SEVENTEENTH STREET • PRAIRIE DU SAC, WISCONSIN 53177

#113

Products...

- AVT-1850-1: J1850 VPW Development System
- AVT-713: J1850 VPW & PWM Dual Interface
- AVT-921 & 922: PC Based & Hand Held Automotive Diagnostic Tools (Under Development)

Engineering Support...

- Custom Software Development
- Custom Hardware Development
- On-site & Off-site Software & Hardware Support



STOP

We can provide you with vehicle network expertise, products, and resources.

Contact us today



Advanced Vehicle Technologies, Inc.

1509 Manor View Road
Davidsonville, Maryland 21035

410-798-4038 (Voice)
410-798-4308 (Fax)

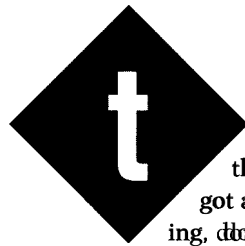
#114

FEATURE ARTICLE

Anindya Ray & Lee Hanson

The Embedded Sun Part 1: Introduction to the Hardware

Although renowned as a workstation giant, Sun has been rather quiet in the embedded field—until now. With microSPARCIIe, Sun hopes to make major inroads into high-end embedded applications.



here's an adage that says if you've got a good thing going, don't change it—or

if you do, change it minimally.

Over the years, Sun Microsystems SPARC Technology Business has been successful with its high-performance SPARC microprocessor architecture. It recently announced its newest chip, the UltraSPARC 64-bit microprocessor, which includes an integrated multimedia instruction set.

Therefore, it only makes good engineering and business sense to parlay Sun SPARC's previous processor architecture, technology strengths, and development efforts to help design and develop the new microSPARCIIe. This chip (see Table 1 for complete specifications) targets three high-end embedded-system applications: networking, telecommunications, and printer/copier.

The MicroSPARCIIe will be announced in April '96 with production-quantity shipments scheduled for the third quarter of 1996. The chip is a derivative of the microSPARCII, Sun's low-end workstation processor (released March 1995).

MicroSPARCII has an integer unit (IU) with an eight-window SPARC register file, high-performance floating-point unit (FPU), 16-KB instruction

cache, 8-KB data cache, 64-entry memory-management unit (MMU), DRAM interface, and SBus controller.

This article explains how these and newer functional blocks and features are used in the design of the microSPARCIIe. In designing this microprocessor, our engineers focused on four main design goals. They wanted to develop:

- a microprocessor that is cost-effective for high-end embedded-systems markets, which could be reduced in cost over time to meet evolving embedded-system requirements
- a flash memory interface that enables customers to run real-time operating systems (RTOS) and load and run their own code out of ROM (see sidebar "Streamlining User Development")
- PCI bus-interface capability for customers currently using Sun products. This feature provides a way to move from SPARC legacy code and SBus capability to PCI at their own pace.
- a high-performance direct-memory interface that permits data to move back and forth quickly

The key to a large part of microSPARCIIe's performance is its ability to implement a system with few external interface components. In this case, you don't need external memory controller, I/O device controller, and ROM controller chips. The chip's performance is 85 SPECint92 and 70 SPECfp92 160k Dhrystones with 125-MHz CPU clock, a 33-MHz PCI clock, and 25-MHz SBus clock.

MicroSPARCIIe ARCHITECTURE

As Figure 1a illustrates, the microSPARCIIe microprocessor contains eight basic sections: IU, FPU, 16-KB I-cache, 8-KB D-cache, 64-entry MMU, memory interface, flash memory interface, and SBus interface. The processor also includes an interface to the PCI bus via a bridge chip since the 'IIe's interface pins are not currently designed for direct PCI-bus communication.

The bridge chip, known as Falcon, is composed of a PCI device controller,

DMA transfer controller, and configuration registers that define the PCI addressing space (see Figure 1 b). The microSPARCIIe supports up to four SBus slots via its direct interface to the SBus and up to four PCI slots through the bridge chip.

The IU executes the SPARC integer instructions defined in the *SPARC Architecture Manual V. 8*. It contains 136 registers comprising 8 windows of 16 register sets and 8 global registers. The IU operates on prefetched instructions, using a five-stage pipeline. Branch folding and single-cycle load-

and-store instructions improve its throughput.

Branch folding is a technique used for reducing the delay of branch processing. Branch instructions are detected early in the microSPARCIIe's pipeline, and the instruction flow follows the predicted outcome of the branch.

Since branches are removed from the instruction stream before they execute, the integer unit can handle two instructions at once if one is a branch and the outcome of the branch is correctly predicted.

By predicting that all branches will be taken, the microSPARCIIe uses simple prediction algorithms. If the prediction is correct, branch folding occurs, and no bubble exists in the execution pipeline. If the branch is not taken, execution continues with the next sequential instruction after a one-cycle delay.

The FPU fully executes all single- and double-precision floating-point instructions [1]. Quad-precision instructions trap in software and are implemented there. The FPU contains 32 x 32 f registers, a general-purpose execution unit, and an FP multiplier.

In most instances, these architectural features enable the parallel execution of an F P M U L and another FP instruction. A three-deep queue of FP instructions is provided to increase the efficiency of concurrent floating-point and integer execution.

The MMU translates the 32-bit virtual addresses of each running process to 31-bit physical addresses. The three high-order bits of the physical address are maintained to support memory mapping into eight different address spaces.

The MMU also serves as an I/O MMU and controls arbitration between I/O, D-cache, I-cache, and TLB references to memory. It contains a 64-entry fully associative TLB, supports 256 contexts, and protects memory so that a process is prohibited from reading and writing the address space of another process.

The 16-KB I-cache is direct mapped, virtually indexed, and virtually tagged. It's organized as 512 lines of 32 bytes plus 32 tag bits. Cache refill is performed four 64-bit words at a time. Cache streaming and bypass are each supported for both the I-cache and D-cache.

The 8-KB D-cache is a direct-mapped, virtually indexed, virtually tagged write-through cache with no write allocate. Data store is organized as 512 lines of 16 bytes, plus 32 tag bits. Single-word integer and double-word FP read- and write-cache hits take only one clock cycle.

A four-deep store buffer can hold data stored from the IU or FPU to memory or other physical devices. The

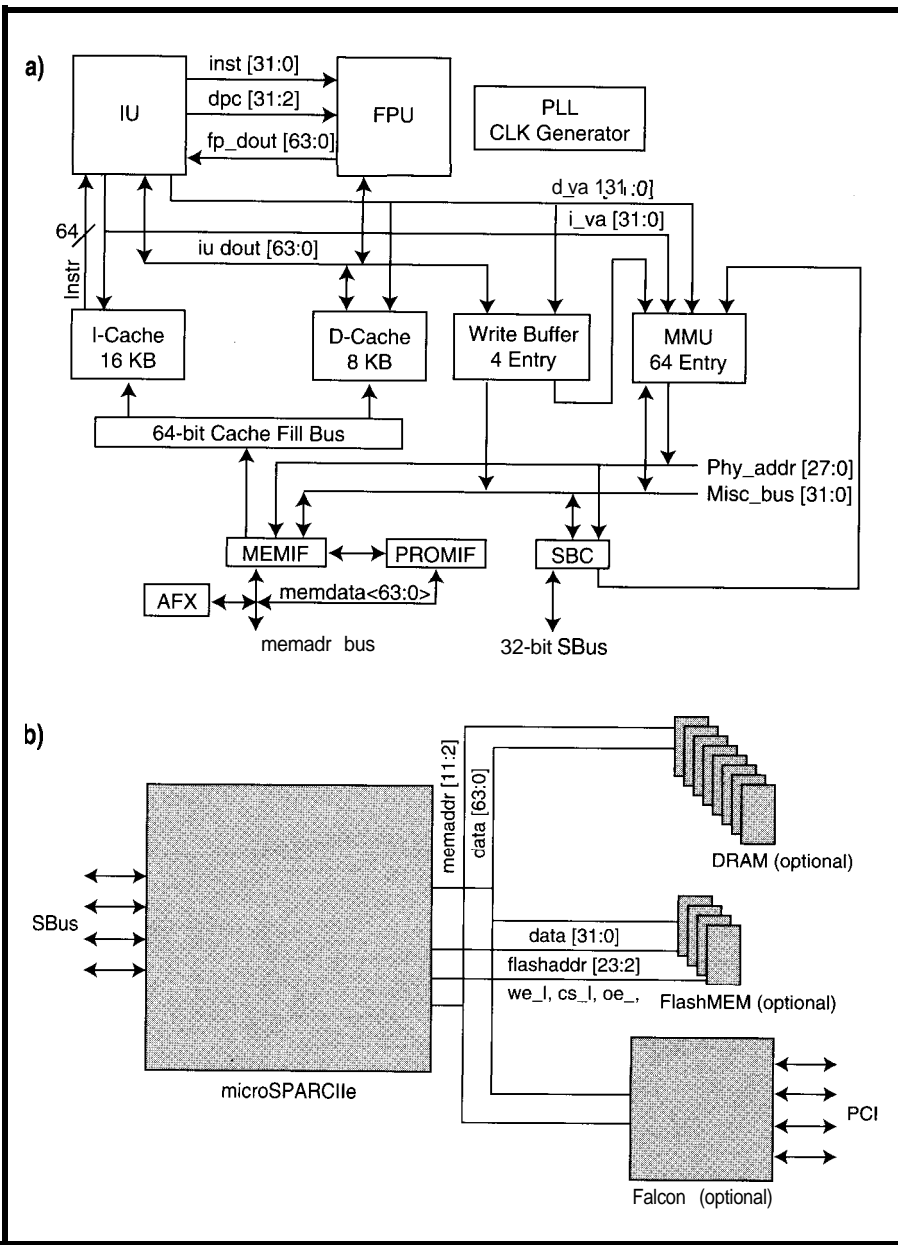


Figure 1-(a) The microSPARCIIe has an integer unit, floating-point unit, 16-KB f-cache, 8-KB D-cache, 64-entry MMU, memory interface, flash memory interface, SBus interface, and PCI bus interface via a bridge chip. (b) The bridge chip's function is to interface to the PCI bus on one side and to the *IIe* on the other.

Streamlining User Development

As a major part of our second goal, we analyzed the needs of embedded-products users. We found that currently used printer, telecom, and networking systems are based on general-purpose high-end microprocessors. We therefore looked for ways to provide embedded-systems engineers a clear advantage when using microSPARCIIe.

Here's one distinct advantage. A considerable amount of code is now readily available to support microSPARC-IIe as well as other existing SPARC processors. More than 50% of all embedded-applications software is developed on Sun workstations and can be directly ported to the microSPARCIIe.

By porting code to the 'IIe, both software and hardware engineers save a great deal of time-time otherwise needed for emulation and cross-compilation. Debugging code via in-circuit emulation can consume weeks of

engineering resources. As a conservative estimate, it often takes as much as 20% of the overall development cycle.

A number of third-party development tool companies support this concept of development and deployment on the same platform. Such vendors as Wind River, Force, Lynx, Sun, and Microtech offer both operating systems and debuggers which run on Sun desktop machines and port directly to the embedded application.

It's prudent to maintain engineering familiarity with the same tools and avoid cross-compiling. By developing and deploying on the same architecture, you avoid the problems associated with the classic routine of developing on a platform, cross-compiling from the target microprocessor, setting up the emulator, and debugging.

store buffers are 64-bit registers. As in the I-cache, cache refill is done two 64-bit words at a time.

The DRAM memory-interface controller generates all the signals necessary to support up to 256 MB of system memory. The DRAM bus is 64 bits wide with two parity bits, each one covering 32 bits of data. System DRAM is organized as eight banks, each of which may be 2, 8, or 32 MB, depending on the size of the DRAM used.

RAM-refresh-control logic provides complete DRAM refresh control, and the refresh controller performs CAS-before-RAS refresh. The onchip refresh control is programmable, and self-refreshed DRAMs are also supported.

The SBus interface performs all functions necessary to connect the microSPARCIIe to the SBus, including dynamic bus sizing, cycle rerun control, burst-cycle reordering, arbitration, and general SBus control. This inter-

face controls SBus devices sharing the bus and supports the following transactions:

- Programmed Input/Output (PIO) between the CPU and SBus devices
- Direct Virtual Memory Access (DVMA) between SBus masters and local resources (Local DVMA)
- Direct Virtual Memory Access (DVMA) between SBus masters and other SBus slave devices (SBus DVMA)

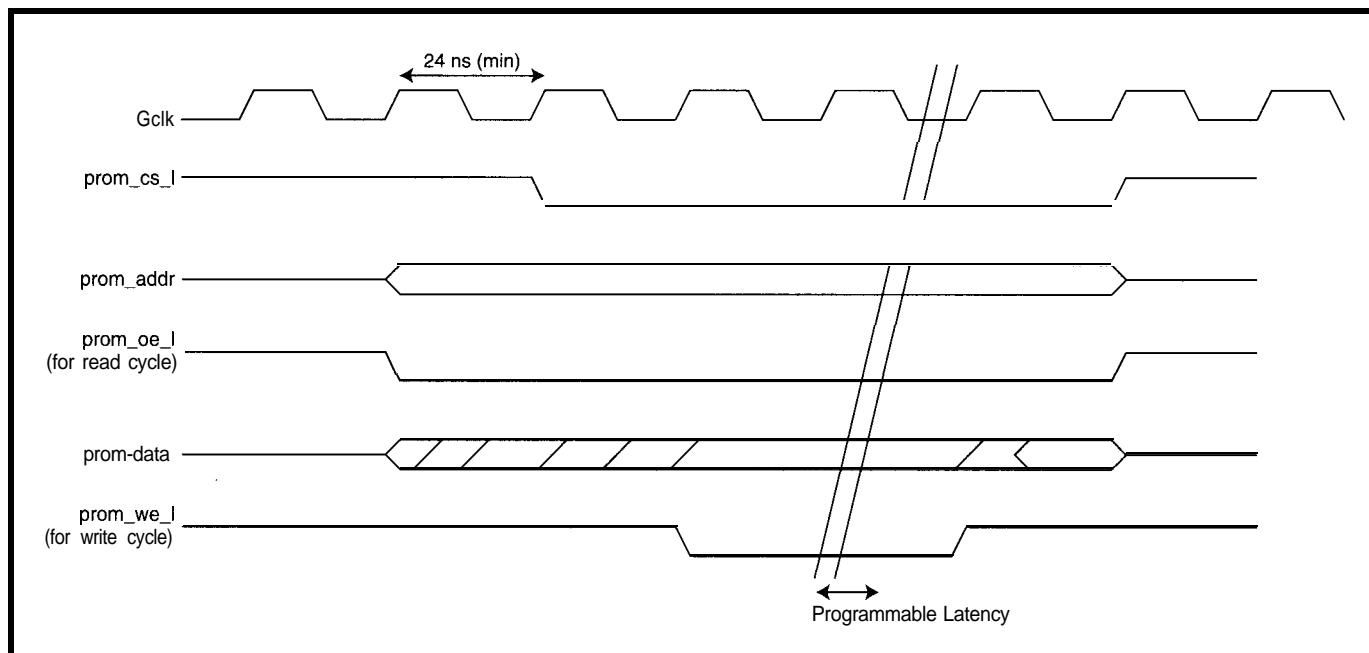


Figure 2—The microSPARCIIe flash-memory interface provides glueless connection to 28FxxxXX-compatible flash-memory devices. The read/write cycle shows programmable latency, which defaults to 15 Gclk if desired (1 Gclk = 0–41.67 MHz). After powerup, this latency can be reprogrammed to any multiple of Gclk between 2 and 15. For example, if the 'IIe's internal clock is 10 ns, Gclk is 30 ns, and if flash-memory latency is programmed to 4, it becomes 120 ns. This latency is based on a 32-bit data bus, with an address range of 4 megawords (=16MB). Only 32-bit word accesses are allowed, so subword accesses require a read-modify-write sequence.



DSP in RISC Embedded Processors

by Richard Pedreau

TEMIC's new SPARClet family of 32-bit SPARC-compatible microcontrollers comes with integrated DSP functions.

To optimize the price-versus-functionality tradeoff, SPARClet's approach is to find out which DSP functions are necessary in the dedi-

cated field targeted by each of the family's products. It then integrates those functions as hardware coprocessors during product definition.

This methodology offers two advantages:

- DSP functions are deeply integrated in the CPU and use the regular register file and instruction set. They are considered part of the CPU core rather than an add-on.
- it follows SPARClet's basic rule-maintain the development chain consistently and avoid a cross-development or dual-processor environment. CPU and DSP cores usually have to be programmed separately, which generates extra code-writing and debugging.

The TSC701, the first member of the SPARClet product line launched by TEMIC in 1996, is a good illustration of this strategy. Its dedicated target is the low-end communication market [hubs, routers, ISDN adapters, cellular base stations, etc.).

Most emerging applications use an additional DSP core to implement filtering algorithms based mainly on iterative multiplication. The TSC701 addresses this issue by including a 32 x 32-bit, 4- or 5-cycle, fully parallel Multiply-and-Accumulate (MAC) unit.

In contrast, the highest-performing DSP cores perform one-cycle multiplication. Despite this, the TSC701 achieves the same or even higher overall performance for a lower cost.

What are the challenges?

- The DSPs have x- and y-buffer input queues (previously stored by the main processor) that must be loaded from external memory. The TSC701's MAC unit uses regular window registers as operands and result. So, bus traffic is reduced by two.
- The TSC701 multiplier takes four or five cycles, depending on whether the accumulation register is the same as before. But, during this time, the Load/Store Unit can load the next two operands.

Thus, the TSC701 remains very close to the one-CPI (Cycle Per Instruction) rate on the whole loop. Such a rate is unreachable by most controller/DSP associations (which use dual-port RAM most of the time) because of the loss in main-processor-to-memory and memory-to-DSP buffer transactions.

Use of a secondary DSP remains valid in configurations employing all the DSP functions. However, in embedded applications requiring intensive use of just a few DSP functions, it makes much more sense to integrate them into the main controller.

For other products being introduced in the same product line, TEMIC follows the same approach: it fits the DSP capabilities of each derivative to its targeted application field.

RELAY INTERFACE

PROVIDES SOFTWARE
CONTROL OF RELAYS

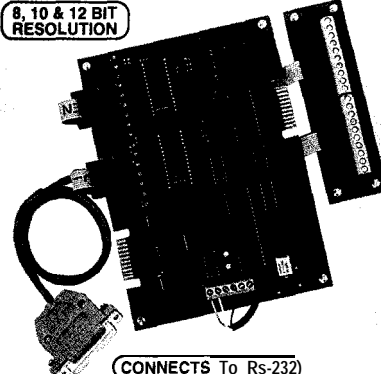


CONNECTS TO RS-232

AR-16 RELAY INTERFACE (16 channel) \$ 89.99
Two 8 channel (TTL level) outputs are provided for connection to relay cards or other devices (expandable to 128 relays using EX-16 expansion cards). A variety of relays cards and relays are stocked. Call for more info.
AR-2 RELAY INTERFACE (2 relays, 10 amp).....\$ 44.99
RP9 REED RELAY CARD (8 relays, 10 VA).....\$ 49.95
RH-8 RELAY CARD (10 amp SPDT, 277 VAC).....\$ 69.95

ANALOG TO DIGITAL

8, 10 & 12 BIT
RESOLUTION



CONNECTS TO RS-232

ADC-16 A/D CONVERTER* (16 channel/8 bit).....\$ 99.95
ADC-8G AD CONVERTER* (9 channel/10 bit).....\$124.90
Input voltage, amperage, pressure, energy usage, light, joysticks and a wide variety of other types of analog signals. RS-422/RS-485 available (lengths to 4,000'). Call for info on other A/D configurations and 12 bit converters (terminal block and cable sold separately). Includes Data Acquisition software for Windows 95 or 3.1
ADC-8E TEMPERATURE INTERFACE* (8 ch).....\$ 139.95
Includes term. block & 8 temp. sensors (-40' to 146' F).
STA-9 DIGITAL INTERFACE* (8 channel).....\$ 99.95
Input on/off status of relays, switches, HVAC equipment, security devices, keypads, and other devices.
PS-4 PORT SELECTOR (4 channels RS-422).....\$ 79.95
Converts an RS-232 port into 4 selectable RS-422 ports.
CO422 (RS-232 to RS-422 converter) \$ 39.98

● EXPANDABLE...expand your interface to control and monitor up to 512 relays, up to 576 digital inputs, up to 128 analog inputs or up to 128 temperature inputs using the PS-4, EX-16, ST-32 & AD-16 expansion cards.

● FULL TECHNICAL SUPPORT...provided over the telephone by our staff. Technical reference & disk including test software & programming examples in QuickBasic, GW Basic, Visual Basic, Visual C++, Turbo C, Assembly and others are provided.

● HIGH RELIABILITY...engineered for continuous 24 hour industrial applications with 10 years of proven performance in the energy management field.

● CONNECTS TO RS-232, RS-422 or RS-485...use with IBM and compatibles, Mac and most computers. All standard baud rates and protocols (50 to 19,200 baud).

FREE INFORMATION PACKET...use our 800 number, Fax or E-mail to order, or visit our Internet on-line catalog.
URL: <http://www.eeci.com>
Technical Support (614) 464-4470

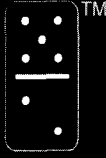
24 HOUR ORDER LINE (800) 842-7714
Visa-Mastercard-American Express-COD

Internet E-mail: eeci1@ibm.net
International & Domestic FAX: (614) 464-4656
Use for information, technical support & orders.

ELECTRONIC ENERGY CONTROL, INC.
380 South Fifth Street, Suite 604
Columbus, Ohio 43215-5491

BIG
THINGS
COME

DOMINO
Microcontroller



DOMINO
Microcontroller
MODEL SN

Starting at
\$79

Micromint's Domino-52 microcontroller is a "supercomputer" in less than 0.75 cubic inches. We've packed the most essential elements into one tiny package. Domino is a plug-and-go module, just attach +5V and a terminal or network. A simple keyed sequence saves an autostarting program in nonvolatile memory.

SPECIAL FEATURES

- 80C52 with ROM-resident, full floating-point BASIC
- 32K bytes SRAM and 32K bytes EEPROM
- Two PWM outputs, I²C bus
- Serial I/O: (up to 19,200 bps) RS-422, RS-485 & RS-232A
- Two interrupts and three timers
- Parallel I/O: 12 bits, 3 shared with ADC and I²C
- Power: +5V @ 15 mA;
- Size: 1.75"×1.062"×0.4" potted
- A/D converter: 2 channels, 12 bits, 10k samples/sec.
- Connections: via 2×10, 0.1" dual-row header
- -20°C to 75°C operating temperature
- Industrial temperature available



4 Park Street • Vernon, CT 06066

Call 1-800-635-3355

(860) 871-6170 • Fax (860) 872-2204

Table 1—The new microSPARCIIe targets three high-end embedded-system applications: networking, telecommunications, and printer/copier.

The SBus interface works with the MMU to arbitrate the system and memory resources and for I/O address translations.

Integrated on the 'Ile is a 32-bit flash-memory interface which allows glueless connection to Intel 28FxxxXX-compatible flash-memory devices. This interface has a software-programmable latency (described later). It's suitable for embedded systems which take advantage of diskless and DRAMless small form-factor systems.

Also new to the 'Ile is direct hook-up to the Falcon PCI bridge chip. This feature gives a transition path to accommodate the growing industry acceptance of PCI. Falcon provides two-way XI-to-DRAM and PCI-to-PCI DMA capability similar to SBus-to-DRAM and SBus-to-SBus DVMA.

MODULARITY

As one of our major strategies in designing microSPARCIIe, we wanted to make the functional blocks as modular as possible.

For example, modular design enables future elimination of the FPU if desired. Also, the SBus interface can be omitted to comply with certain other embedded-systems design requirements.

In derivatives of the 'Ile, the PCI interface can be brought on-chip, replacing the SBus interface. Cache sizes can also be modified or, in applications where data caching is unnecessary, eliminated entirely.

MicroSPARCIIe Specifications

Process:	0.5 micron, 5-layer metal CMOS
Transistors:	One million
Die Size:	10x 10 mm
Power:	Fully static, 3.3-V core 5.013.3-v I/O Power-down mode < 10% of nominal Peak power 7 W
System Level Architecture:	(1) Direct programmable DRAM interface (2) Direct interface to flash memory (3) Direct interface to SBus—support for up to 4 SBus slots (4) Interface to PCI bus via bridge chip—supports up to 4 PCI slots
Clock Speeds:	33-MHz 32-bit PCI bus 25-MHz SBus
CPU Speed:	125 MHz
Performance:	85 SPECint92, 70 SPECfp92, 160 Dhrystones
On-Chip Memory:	16-KB instruction cache 8-KB data cache

To save device pins, the flash memory interface block shares the data bus with the DRAM interface and has extraneous control for writing to flash memory. Supported flash memory devices include Intel parts in the 28F series.

These devices have specified access times of approximately 100–150-ns access period, and data widths are typically 8 or 16 bits. These flash memory accesses are about one-third to one-half as fast as DRAM ones.

In the microSPARCIIe, the flash access is kept to 32 bits at all times. From a software perspective, the flash-memory access behaves the same as the DRAM access, even though it is physically addressed in a different address space.

All flash-memory accesses can be cached similarly to DRAM accesses. Thus, you can run tight-looped code from within cache. Also, the microSPARCIIe's flash-memory interface has programmable latency. This feature can be important to the embedded-system designer who wants to

Fujitsu's SPARClite Alternatives

by John Bums

Fujitsu's latest entries in the embedded market, the MB86933H and MB86936, have met with success in imaging and communications applications.

The newest parts carry on the tradition Fujitsu began in 1990, when they adapted the SPARC architecture specifically for embedded applications where price and performance are paramount.

Both new processors include the popular features of previous family members and maintain code compatibility. Both integrate peripherals to minimize glue logic.

The MB86933H is intended for applications which require excellent performance at the lowest possible cost. The MB86936, with on-chip FPU, three DMA channels, and a video-rasterization interface for printer applications, is geared for high-performance applications.

To achieve superior performance, the MB86933H is built around the SPARClite integer core. The core features a five-stage pipeline which operates at a sustained rate of 1.08 CPI.

Additions to the SPARC architecture, pioneered by Fujitsu to enhance embedded performance, include hardware `MULTIPLY`, `DIVIDE STEP`, and `SCAN` instructions.

To show the benefit of the SPARClite `SCAN` instruction, consider a code segment that compresses long binary strings by looking for runs of all ones or zeros and coding these so lossless reconstruction is possible. Each instruction runs one cycle out of the instruction cache if it is in the active path for a particular case.

Without a hardware implementation of `SCAN`, which takes one cycle, an additional software routine requiring 43-52 cycles would be needed. This routine would also consume instruction-cache space. Designers of communications applications find `SCAN` useful where there is a large amount of I/O management.

The `DIVIDE STEP` instruction was also implemented for embedded applications without deterministic interrupts. `DIVIDE STEP` executes a 32×32 -bit division in less than 40 clock cycles and efficiently handles an interrupt at any point without losing the division in progress. The multiplier provides 64-bit results in only 1-5 clock cycles.

The MB86933H is manufactured in 0.8-micron CMOS process and runs at 25 MHz. It features a 1-KB direct-mapped instruction cache. Evaluation of standard and customer benchmarks shows that instruction cache contributes more than data cache to performance. As with previous members of the family, cache lines can be individually locked.

To facilitate low-cost system design, the MB86933H features:

- a configurable data bus which supports 8-, 16-, or 32-bit memory
- a bus interface which provides programmable chip selects and programmable wait-state circuitry
- an on-chip interrupt controller which handles up to 15 interrupts with external priority encoder or 4 external uncoded interrupt requests
- a complete DRAM controller which supports 256-KB, 512-KB, 1-MB, and 2-MB configurations in up to two banks
- a fully static circuit design which enables the processor clock to be slowed or stopped to conserve power with no loss of internal state. In normal operation, power consumption is 1.16 W

IN SMALL PACKAGES
BLACKJACK  TM
TELECONTROLLER

BLACKJACK 
TELECONTROLLER
MODE: _____ SN: _____

Starting at **\$139**

Micromint's BlackJack-552 is a true "telecontroller" incorporating both microcomputer and FCC-certified Xecom modem in a single package. BlackJack comes preprogrammed with firmware utilities which are optimized for assembly language, C, and BASIC programs. Like Domino, BlackJack is easy to use. Attach power and a terminal, then upload, store, and execute your program.

SPECIAL FEATURES

- 80C552 processor with firmware monitor, 14.7456 MHz
- Serial I/O; (up to 38,400 bps); full-duplex TTL, I²C bus
- Connection: 1.2" dual-row 48-pin DIP header
- Parallel I/O: 20 bits, 4 bits shared with RTC
- Two interrupts: three timers, watchdog timer, two PWM outputs
- Hardware real-time clock
- A/D converter; 8 channels, 10 bits, 20k samples/sec.
- 2400 bps data modem (data/fax available)
- Power: +5V @ 55 mA;
Size: 2.75" x 1.375" x 0.5" potted
- 32K bytes SRAM, 32K bytes ROM and 128K bytes EEPROM
- Industrial temperature available



4 Park Street • Vernon, CT 06066

Call 1-800-635-3355

(860) 871-6170 • Fax (860) 872-2204

#119

Circuit Cellar INK®

Issue #69 April 1996

35

As a result of these features and a price of less than \$15 in high volume, the '933H has been designed into a new generation of digital cameras that provide document-ready images and bar-code readers that use two-dimensional bar codes.

The MB86936 is designed for high-performance applications and offers many more features integrated on-chip. The part is manufactured in OS-micron CMOS, and its SPARClite core runs at up to 50 MHz in clock-doubled mode. Less-expensive memory systems can be used with the bus running at 25 MHz.

To minimize system cost and reduce glue logic, the MB86936 has an integrated DRAM controller, providing an address multiplexer, refresh timer, and page comparator. The controller's programmable state machine governs the timing relationships of the multiplexed row and column addresses and the DRAM control signals.

Memory space can be blocked off into sections which can be marked as noncacheable. High-speed SRAM can also be programmed as noncacheable to further reduce glue logic. The MB86936 has been designed into laser-printer applications, in which this capability is important.

In addition, the MB86936 features a 4-KB instruction cache, 2-KB data cache, two 24-bit timers, three DMA channels (one for video), and an onchip interrupt controller. The caches are organized as two-way set associative. As with all SPARClite caches, individual lines can be locked.

The set-associative organization means one bank of cache continues to operate as fully functional direct-mapped cache, no matter how many entries in the other bank are locked. In addition, the integer unit continues processing without waiting for an entire cache line to be filled.

A final feature of the MB86936 which has proven valuable to designers is an integrated ANSI/IEEE754-compliant FPU with a three-stage pipeline. All single-

precision operations except DIVIDE and SQUARE ROOT are executed in one clock cycle, as are double-precision ADD and SUBTRACT. This capability has proven attractive for both color laser printers and machine-vision applications.

All SPARClite processors comply with both V. 8 and V. 8E (the embedded extension of V. 8) of the *SPARC Architecture Manual* and execute all the SPARC integer instructions it defines. Typically, SPARClite processors feature a register window scheme consisting of 136 registers configured in eight windows of 16 register sets and 8 global registers. The MB86933H reduces the number of windows to six.

All SPARClite processors except the MB86933H feature a debug support unit (DSU). The DSU provides a direct connection to an in-circuit emulator from STEP Engineering. So, even with the caches in operation, bus transactions can be monitored without interrupting system operation at speeds up to 50 MHz. The DSU makes possible hardware breakpoints, single-step operation for debug, and full instruction trace. JTAG boundary scan provides further testability.

These parts meet the needs of today's applications in imaging and communications. Future parts will migrate to Fujitsu's 0.35micron CMOS process to provide even more cost-effective solutions at the low end of the market. Application-specific accelerators currently under development will offer greatly enhanced performance throughout the entire range of potential applications.

John Burns received a Ph.D. from the University of Minnesota in 1975. He taught mathematics and computer science for 13 years. Prior to joining Fujitsu 7 years ago, John was a systems design consultant. He is now the marketing manager for Embedded Control Products at Fujitsu. John may be reached at jburns@fmi.fujitsu.com.

migrate to faster or slower flash-memory devices.

Programmable latency for flash memory is done in software rather than via pins. Software-programmable flash latency allows device access anywhere from 50 to 450 ns in 25-ns increments. The latency on reset is 450 ns. The addressing range is 0-16 MB.

Figure 2 shows a typical flash-memory read/write cycle. The clock being referenced is the G clock (Gclk) coming out of the processor. It runs at one-third the CPU clock speed. If the CPU clock is running at 100 MHz,

which is a 10-ns period, the Gclk is running at a 30-ns period.

The second signal is the flash-memory chip-select signal (prom_cs_1), which goes active when flash memory is accessed. The third bus is the flash-memory address bus (prom_addr).

The fourth bus (or signal) is the PROM or flash-memory output-enable signal (prom_oe_1 for read cycle), which goes active when the flash memory is accessed for reads. When flash memory is accessed for writes, this signal stays inactive.

The fifth signal is the flash-memory data bus (prom-data), which is the

same data bus as the DRAM bus. The last signal is the write-enable (prom_we_1 for write cycle) for the flash-memory device, which goes active on writes.

Flash memory offers nonvolatile storage when power is off. This capability is important in many embedded applications in which power-saving measures are critical and power is controlled by relays, environmental sensors, or even manual interrupts.

CONCLUSION

The original microSPARC product provided a cost-effective system solu-

tion to the workstation. MicroSPARC-IIe provides the functionality and cost necessary to truly put SPARC in the embedded arena.

The embedded community is looking for low-cost solutions to I/O. The 'IIe provides such a solution via PCI capability, while enabling the existing SPARC user to maintain compatibility with his previous implementation via SBUS.

Direct control of flash memory and resident code cacheability provide high-performance code execution. In benchmarking the 'IIe, you'll see that it provides high sustained memory bandwidth at a reasonable cost.

You'll also find that the execution of embedded code is extremely fast due to the cacheable ROM space. Many products provide ROM space as memory-mapped I/O that is not cacheable.

For the first time, the embedded engineering community will have a true develop-and-deliver capability with the 'IIe. Even prior to silicon, the embedded community can start to develop on Sun workstations. When the device becomes available, the code moves directly to the embedded product.

The impact of thousands of applications now available to the embedded developer can significantly improve his productivity.

And, last but not least, true compatibility-from the embedded product up through the system that does database management and billing-is now a reality. □

Anindya (Andy) Ray is a staff engineer and computer architect at Sun Microsystems SPARC Technology Business. He has previously been engaged in Sun projects including SuperSPARC, SuperSPARCII, and microSPARC derivatives. Prior to coming to Sun, Andy worked in CPU development at Intel, Prime Computer, Hyundai, and Philips. He may be reached at andy.ray@eng.sun.com.

Lee Hanson is Director of Engineering for Embedded Products at Sun Microsystems SPARC Technology Business.

Prior to joining Sun, he was director of engineering for Intergraph and was responsible for microprocessor development. Lee has also worked at Gould Computers, Amdahl, National Advanced Systems, and NCR. He may be reached at lee.hanson@eng.sun.com.

REFERENCE

[1] Karen Gettman, Ed., **SPARC Architecture Manual, 8**, Prentice-Hall Publishing, Division of Simon-Schuster, Upper Saddle River, NJ, **15-17**, September 1991, ISBN 0-13-825001-4.

SOURCES

microSPARCIIe
Sun Microsystems, Inc.
SPARC Technology Business
MS USUN03-305
2550 Garcia Ave.
Mountain View, CA 94043-1 100
(800) 681-8845
Fax: (408) 774-8769

SPARClet microcontroller
TEMIC
Matra MHS SA
"Les Quadrants"
3, avenue du Centre
78054 St-Quentin-en-Yvelynes
Cedex, France
(33) 40 18 18 18
Fax: (33) 40 18 19 20

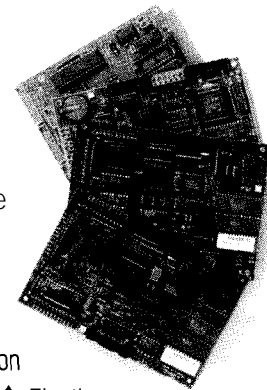
MB86933H, MB86936
Fujitsu Microelectronics, Inc.
3545 N. First St.
San Jose, CA 95134-1804
(800) 866-8608
Fax: (408) 943-1417

In-circuit emulator
STEP Engineering, Inc.
661 E. Arques Ave.
Sunnyvale, CA 94088-3166
(408) 733-7837
Fax: (408) 773-1073

410 Very Useful
411 Moderately Useful
412 Not Useful

**JAM
Packed**

**Embedded
Controllers**



◆ A/D inputs, 12-bit accuracy ◆ Analog outputs ◆ Relay control ◆ Counter/Quadrature encoder inputs ◆ Buffered RS-232/485 serial ports ◆ Operator interface via keypad and LCD display ◆ Program using a PC ◆ 512K program, 512K data memory ◆ 5V only operation ◆ Built-in BASIC supports all on-card hardware ◆ Floating point math ◆ From \$195 in 1's

REMOTE™
PROCESSING
The embedded control company

Call for more information and **FREE**
Catalog of embedded controllers:

Ph: 303-690-1588, Fx: 303-690-1875

FEATURE ARTICLE

Pat Baird

Embedding a Message-based System

Adding event-driven messaging code to embedded projects offers flexibility for evolving products. Pat shows how easy it is to implement such a scheme.



As embedded systems designers, we use a variety of techniques to get a job done. Usually, some off-the-shelf design is changed, modified, and sent off. It's not built from the ground up. Instead, old projects are dug up and patched to meet new specifications.

Designs are pushed to the limit. Decisions made in an afternoon turn out to be critical in future product development. Frustrating afternoons are spent circumventing basic design decisions made long ago.

For example, I work in the packaging industry on an embedded PC controller with EGA EL touchscreen. Software includes a homespun multi-threaded multitasking system, interrupt-driven I/O, state machines, an embedded compiler, high-resolution timer routines, and a message-based control system.

These structures were added over several years, as customers clamored for new and improved functionality. The message-based system proved to be a great way to add robustness to a product. Let me show you how to build one into your system.

ENHANCED ROBUSTNESS

Someone once defined robustness as the ability for a system to succeed in a

situation it wasn't designed for. Incorporating a messaging system makes it easier to implement unforeseen changes later on.

Here's a case in point. The touchscreen controller I work with has no keyboard. Instead, all inputs are performed by pressing on-screen buttons. An IR matrix crisscrosses the screen and presents each touch as a mouse click. The underlying messaging system handles this "mouse" I/O. Each button press sends a message, and a dispatch routine calls the subroutine for that button.

Normally, each touchscreen controller is a stand-alone system. However, last month a customer requested a proposal for a network consisting of four machines with four standard controllers and one central controller.

Everything had to be run by the central controller as the individual machines would never be touched. I had to figure out how to take an existing stand-alone design and graft onto it some sort of remote control code.

It turned out the retrofit was easy. A button press on the central machine sends a message down a serial cable, instead of calling a local control routine. On the remote machine, an interrupt-driven I/O handler receives this message and stuffs it into the messaging queue. The dispatch routine then grabs this message and performs some action on the machine.

Here's the interesting part: to show button presses, button colors are simply inverted. This inverting is done not by the act of pressing a button, but rather by receiving a button-press message. Thus, when a message is received from the central controller, the buttons flash on the remote machine as if someone were there pushing buttons.

With a message-based system, an old (and proven) design was pushed in a new and unusual way with little fuss and muss. And, the manuals stayed the same!

CONCEPT FUNDAMENTALS

With a messaging system, instead of there being one thread of execution, several events can occur. Each event has an associated handler routine. Key


```

#define MAXMSG 50
#define MAXQUEUE 50

typedef struct {
    int nMsg;           /* message number */
    void (*handler) (void); /* message function */
} MESSAGE;

typedef struct {
    int head;
    int tail;
    int msgQ[MAXQUEUE];
} QUEUE;

typedef struct {
    MESSAGE mt[MAXMSG]; /* message table */
    int mtTop;          /* next available position */
    QUEUE mq;
} MTABLE;

/* global variables */
MTABLE tbl;

/* InstallMessageHandler—given a table and message #,
 * puts function into the message table.
 * Sorts function table by message #.
 * Returns:
 * 0: if message already exists
 * 1: if installed correctly */

int InstallMessageHandler(MTABLE *nTable, int nMsg, void
                          (*pfnHandler)(void))

    int i, top;

/* search for previous entry on table */
for (i=0; i<nTable->mtTop; i++)
    if (nTable->mt[i].nMsg == nMsg){
        verbose("Installation Collision")
        return 0;
    }

top = nTable->mtTop;
nTable->mt[top].nMsg = nMsg;
nTable->mt[top].handler = pfnHandler;
nTable->mtTop++;
sort(nTable);
return 1;
}

```

presses, mouse clicks, and serial I/O can all send messages. Different events (or sources of events) can send the same message. Thus, a mouse click functions the same way as a hot key. Windows programmers should recognize this mechanism.

The device handler is responsible for sending messages. A dispatch routine calls the appropriate subroutine. To the subroutines, it doesn't matter who originally sent the message—they execute just the same. The messaging code mediates between the caller and callee.

MESSAGE SYSTEM CORE

The concept of this system seems simple, but building a messaging system is simpler. The code examples here are written in C, but these structures could be done in any language.

To build a message-based system, just a few routines are involved:

- InstallMessageHandler0
- DispatchMessage()
- SendMessage()

These and a few other support routines (see Circuit Cellar BBS) are spun to-

gether. The code doesn't do anything very useful (e.g., the message DoA() prints an "A" to the screen). It's provided as a skeletal structure on which to build your program.

InstallMessageHandler0 can be found in Listing 1. This routine first checks to make sure that the message you're trying to install doesn't already have a handler and returns an error if it does. If not, the routine simply adds the ID and function to the message table, then sorts the table.

DispatchMessage() is given in Listing 2. This function dequeues a message, searches the function lookup table for the message ID, and calls the appropriate function.

Listing 3's SendMessage() simply enqueues the message ID into the message queue.

Another routine, not reproduced here, bypasses the message queue and calls the appropriate routine right away. It's called SendPriorityMessage0 and is handy for those ABORT NOW! messages.

Although the program is small [it knows just a few messages), I chose to keep the message table sorted to reduce message-lookup time. This sort is not critical and could be removed. You could enhance it by using a binary search instead of the simple linear search Dispatch() currently uses.

The code also includes an unused message table, which illustrates the syntax of a predefined table. Instead of going through all the InstallMessageHandler() calls at run time, a message table can be built at compile time. This method is easier to read and debug since it avoids tracking every InstallMessageHandler0 call.

At work, I use both a predefined message table and messages installed at run time. The dual system means that I can maintain a simple table of standard functions that remains the same for most of my customers, but I can also add new functions and customize them as required.

ENHANCEMENTS

If you look at the program, you'll see that it has macro capabilities. While putting together this article, I realized just how simple macros would

Listing 2—The *Dispatch* function dequeues a message and calls the associated function.

```
/* Dispatch—given a message table. calls the function
 * at the head of the queue.
 * Returns:
 * 0: if message handle not found
 * 1: if successful */

int Dispatch(MTABLE *nTable)
{
    int i, nMsg;
    int tail = nTable->mq.tail;

    /* is queue empty? */
    if (nTable->mq.head == tail)
        return 0;

    tail++;
    tail = tail % MAXQUEUE; /* wraparound at end of queue */
    nMsg = nTable->mq.msgQ[tail];
    nTable->mq.tail = tail;

    /* look up nMsg in table */
    for (i=0; i<nTable->mtTop; i++)
        if (nTable->mt[i].nMsg == nMsg){
            if (bMacroOn) /* are we recording a macro? */
                macrotable[macroindex++] = nMsg;
            nTable->mt[i].handler(); /* call the function */
            return 1;
        }
    verbose("DISPATCH: Func. not found");
    return 0;
}
```

be to implement. The code takes only 25 lines. Here's how it works.

The *Dispatch* routine looks at a global variable called *bMacroOn*. If *bMacroOn* is set, the routine records the message number to an integer

array called *MacroTable* as it dispatches. Messages *MacroOn* and *MacroOff* (Record and Stop respectively) set and reset *bMacroOn*. The message *PlayMacro* (Play) is a simple for loop, calling *SendPriorityMessage()* for

each message stored in *MacroTable*.

So, if you record a macro by sending an *Record*, *Dispatch* starts copying message IDs to *MacroTable* as well as dispatching the message. If *Dispatch* sees an *S* message, it stops recording macros. If *Dispatch* sees a *P*, the *PlayMacro* routine is called and cycles through the *MacroTable*, calling routines via *SendPriorityMessage*.

This skeleton could easily be the base for a simple calculator program. For instance, draw a calculator on the screen. A mouse driver would detect and send button-press messages. A keyboard driver consisting of *if* (*kbhit()*) *SendMessage*(*getch()*); would also send messages. A serial I/O routine could grab characters from the serial port and send messages as well. The three different input methods all give the same output.

CONCLUSION

Message-based systems are implemented straightforwardly and are an easy way to extend a project.

It's hard to predict a product's future. Designs are often pushed in unusual ways. Adding a messaging structure won't make a program run faster, improve the graphics, or jazz up the user interface of the final product.

What it will do, however, is keep future expansion readily available and painless. □

Pat Baird works as an engineer in the packaging industry and is an undergraduate at the University of Wisconsin-Parkside. He may be reached at 71233.2565@compuserve.com.

Listing 3—*SendMessage* simply enqueues a message.

```
/* SendMessage—given a message table and message handle,
 * stuffs the message into the queue.
 * Does not check if it's a valid message!
 * Returns:
 * 0: message table full
 * 1: if successful */

int SendMessage(MTABLE *nTable, int nMsg)

    int head = nTable->mq.head;

    head++;
    head = head % MAXQUEUE; /* wrap around queue */

    if (head == nTable->mq.tail){
        verbose("SendMessage: Table Full");
        return 0;

    nTable->mq.msgQ[head] = nMsg;
    nTable->mq.head = head;

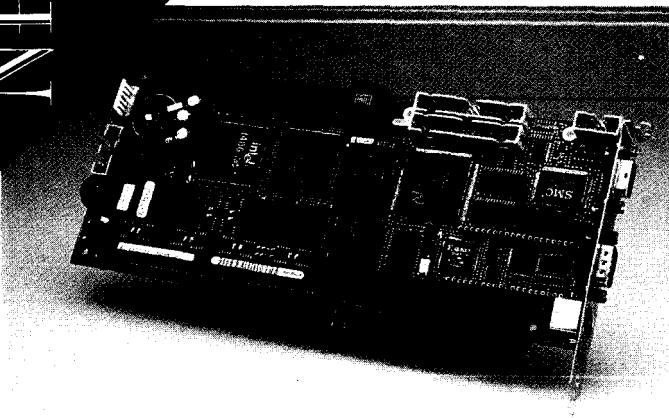
    return 1;
```

SOFTWARE

Software for this article is available from the Circuit Cellar BBS, the Circuit Cellar Web site, and on Software on Disk for this issue. See the end of "ConnecTime" for downloading and ordering information.

IRS

413 Very Useful
414 Moderately Useful
415 Not Useful



CUSTOMIZABLE EMBEDDED COMPUTER

The EPC-33 and EPC-34 single-board computers, based on the Intel 486SL-enhanced processor, are unique. They offer a high level of integration and graphics options including flat-panel and SVGA monitor support and integrated network interfaces. With optional 1 O-Base-T Ethernet, both products can operate as a network server or network node running applications remotely. The units are suitable for medical instrumentation, telecommunication equipment, and other portable, space-conscious environments.

Users can customize both units to meet specific needs by choosing from several options. Either board can be ordered with optional onboard Ethernet interface and SVGA video graphics. Other options include 4-MB flash and 128 KB of battery-backed SRAM. The flash/SRAM option enables users to develop diskless systems.

The EPC-33 and EPC-34 use the Intel DX2 processor running at 50 MHz and the Intel DX4 processor running at 100 MHz, respectively. Features common to both include one

72-pin SIMM socket providing 4, 8, 16, or 32 MB of DRAM; one RS-232C serial port (COM1); one serial port (COM2) as RS-232 or RS-422/485; IEEE 1284-1-compatible parallel printer port (LPT1); enhanced local-bus IDE harddrive interface; PC-PS/2 compatible keyboard interface; and a standard ISA-bus card-edge connector.

The base price for the EPC-33 is \$550 and for the EPC-34 is \$740, both in quantity. These prices are for standard configurations and do not include DRAM, flash, video, or network-interface options.

RadiSys Corp.
15025 S.W. Koll Pkwy.
Beaverton, OR
97006-6056
(503) **646-1800**
Fax: (503) 646-1850

#510

EMBEDDED MOUSE SOLUTION

The **HulaPoint** offers system integrators and OEMs a turnkey embedded-mouse solution. This compact, lightweight, ergonomic device combines a mouse's functionality with a joystick's ease of use to offer both accurate and effortless cursor control.

The **HulaPoint** uses a proprietary sensor technology pioneered by Fujitsu—a magnetic-field-detection method that uses Hall-effect devices. The sensor outputs coordinate data according to the direction and degree of travel of the mobile section. It offers 10" motion in all directions for positive cursor control and user feedback. Because the sensors use magnetic technology, they never wear out. This feature, coupled with a low-profile design that contains no parts to break or moving elements to collect dust, makes the **HulaPoint** sensor extremely durable.

The **HulaPoint's** controller IC translates the motion of the sensor with an Advanced Motion Algorithm. This algorithm, optimized for the physiological abilities of humans, provides a heightened level of control. Because little resistance is placed against the user's finger, the sensor responds to a feather touch, making exact cursor positioning easy to attain. The IC provides autoselectable RS-232 and PS/2 output and supports both the IBM and Microsoft two-button and Logitech's three-button mouse protocols. Various configurations are available for off-the-shelf design implementation.

The **HulaPoint** assembly, including the sensor, IC board, and driver is available from USAR Systems. An evaluation kit, comprising the assembly, cables, and documentation, sells for \$95. A *User Input Device Databook* is also available.

USAR Systems
568 Broadway • New York, NY 10012
(212) **226-2042** • Fax: (212) 226-3215
<http://www.usar.com/>

#511

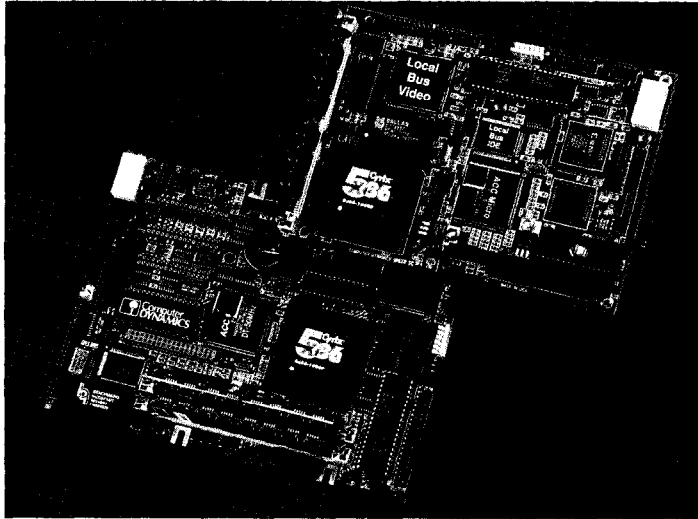


Nouveau **IPC**

edited by Harry Weiner

SINGLE-BOARD 5x86 COMPUTER

Computer Dynamics has released a pair of single-board computers based on the 5x86 CPU: the PC/104-expandable **SBC104-586** and the ISA-expandable SBC-586. The 5x86 processor offers a 64-bit internal data path, an 80-bit floating-point unit, branch prediction (which allows multiple instructions in a single clock cycle), and speeds faster than a 75-MHz Pentium. This processing power, combined with the local bus video controller, makes each board a GUI engine ideal for driving flat panel displays and running demanding software.



Pentium-class performance creates an unequalled PC/104-expandable platform for graphics-intensive applications. Data never slows down, from CPU to video and out to the hard drive. The local bus IDE port supports up to two hard drives, each up to 8.4 GB. The 5" x 6.5" board also features up to 32 MB of DRAM, speaker and keyboard interfaces, a real-time clock, and the PC/104's 8- to 16-bit interface. The SBC104-586 is rated at 0-70°C, operates at +5 V only, and consumes less than 10 W.

The SBC-586 board local-bus video controller runs at 32 bits for maximum throughput and features built-in Windows accelerators. The 5.75" x 7.75" board drives XGA (1024 x 768) color TFTLCDs

in 256 simultaneous colors, and VGA color TFTs or passive matrix color LCDs in 64,000 colors. Up to 64 MB of DRAM is offered onboard. Three sockets support 512-KB EPROMs, 512-KB SRAMs, and 256-KB flash memory for a total RAM and ROM capacity of 1.5 MB.

The SBC-586 has an IDE hard-disk controller which supports up to two 8.4-GB hard disks. Other SBC-586 features include two COM ports, a printer port, watchdog timer, real-time clock, speaker and keyboard interfaces, and a standard Phoenix BIOS. An ISA PC bus connector and an SBX connector are included for expansion. This board is also rated at 0-70°C and draws 9.7 W.

Pricing for the **SBC104-586** is \$1520 and the SBC-586 is \$1620 (in quantities of 100, 4-MB DRAM).

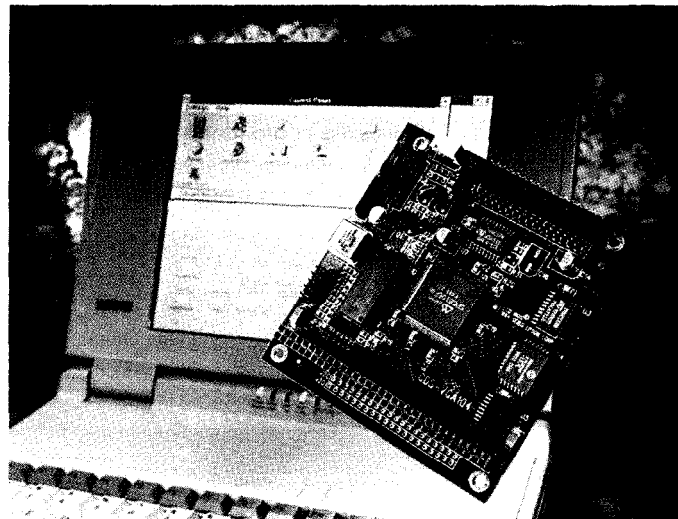
Computer Dynamics
7640 Pelham Rd. • Greenville, SC 29615
(864) **627-8800** • Fax: (864) 675-0106
sales@cdynamics.com

#512

PC/104 LCD CONTROLLER

Communication and Display Systems has introduced the **VGA-104** controller to interface LCD flat panels to PC/104 embedded systems. The controller is compatible with both color and monochrome passive-matrix LCD panels from 320 x 240 to 640 x 480 resolution. An embedded PC/104 system with a flat-panel display is useful in industrial, commercial, scientific, and medical applications.

The VGA-104 features a standard PC/104 format and hardware VGA compatibility. Colors are converted to 64 gray shades for monochrome panels and 4096 colors for passive, dual-scan color panels. The board includes 512-KB display memory, vertical centering, and onboard generation of all display voltages. It measures 3.775" x 3.550" and draws less than 180 mA at 5 V.



Communication & Display Systems, Inc.
194-22 Morris Ave. • Holtsville, NY 11742
(516) **654-1143** • Fax: (516) **654-1496**

#513

Nouveau PC

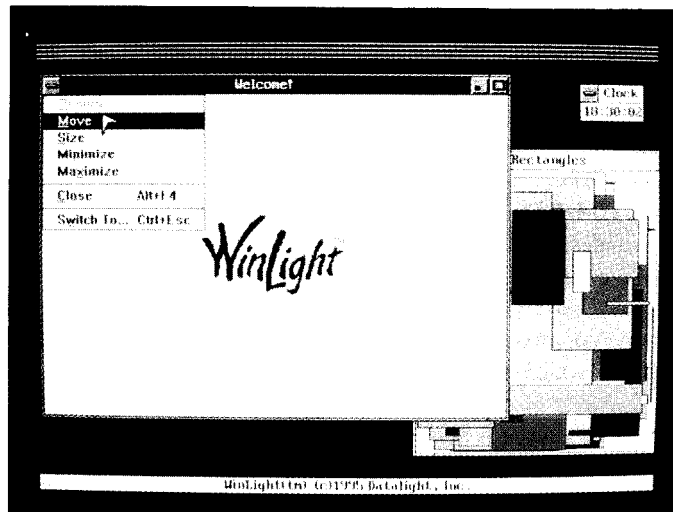
EMBEDDED OPERATING SYSTEM

Winlight, a Windows-like operating system, is designed specifically for embedded and mobile systems. It enables developers to use standard Windows desktop development tools for creating applications that operate with less than 256 KB of code space.

WinLight supports only the calls needed for an embedded system, so it requires smaller ROM and RAM. Source code for device drivers is included in the **WinLight Software Developer's Kit**, so **WinLight** can be customized for nonstandard hardware such as unusual screen sizes, shapes, and types. The **WinLight SDK** also includes software utilities to create a ROM disk for placing applications and the operating system in ROM. It can even run the code portions of **WinLight** and most applications directly out of ROM. **WinLight** source code is also available.

WinLight features a Graphical User Interface (GUI) and cooperative multitasking, and runs in protected mode like Windows. The GUI enables developers to create user-friendly screens with pop-up menus, buttons, and windows. An end user selects from among buttons representing valid selections on each screen, rather than having to use a keyboard and know proper syntax. It also supports Windows-compatible Dynamic Link Libraries (DLLs) and resources that enable easy application customization.

The **WinLight Software Developer's Kit** sells for \$595. The kit provides the software tools and **accessories** necessary to configure



WinLight for various hardware environments and includes a certificate for 20 duplication licenses. License fees vary from \$12 per copy for 500 to \$6 per copy for 10,000.

Datalight

188 10 59th Ave. NE • Arlington, WA 98223
 (360) 435-8086 • Fax: (360) 435-0253
sales@datalight.com

#514

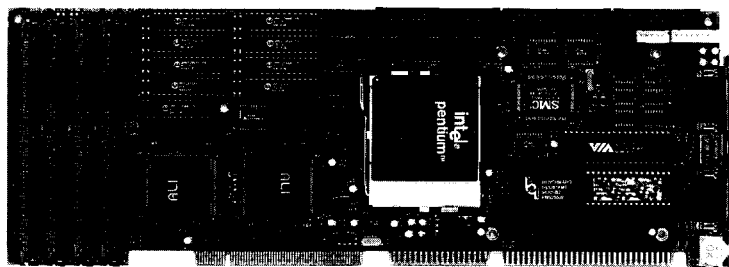
PENTIUM-BASED CPU BOARD

Operating at speeds up to 150 MHz, the ISP-586 ISA/PCI Pentium board combines a full-featured passive-backplane CPU with a high-speed **PCI-compatible** bus and a PC/I 04 expansion port. This mix makes it ideal for high-speed embedded applications.

The **ISP-586/150 MHz** performs at a Landmark **V2.0** rating of 863 MHz. The PCI expansion bus is compatible with the PCI Industrial Computer Manufacturers Group (PICMG) specification and supports up to four PCI master peripheral boards. The board includes two serial ports, a bidirectional parallel port, a dual floppy-disk port, an IDE hard-disk port, a **PS/2** keyboard port, **PS/2-compatible** mouse port, **onboard** speaker, watchdog timer, and up to 128 MB of DRAM.

In addition, each board has a standard PC/I 04 expansion port for adding optional boards such as an EPROM/RAM disk-emulator board, video controller, or digital I/O. Since the ISP-586 was designed for embedded and industrial applications, the BIOS boots without either a keyboard or monitor.

The watchdog timer makes the board ideally suited for controlling critical processes where unattended operation is essential. It can be programmed to generate a nonmaskable interrupt or system reset in the event of an I/O timeout delay or external failure. The timeout delay is adjustable from 1 to 220 s.



The ISP-586 SBC board comes complete with a user's manual and carries a two-year warranty. It is priced from **\$795**.

Micro Computer Specialists, Inc.
2598 Fortune Way
Vista, CA 92083
(619) 598-2177 • Fax: (619) 598-2450

#515

Nouveau NPC

PROGRAMMABLE CONTROLLER

The **PIC14000** mixed-signal controller targets a variety of applications where analog signals are measured and digital control and communications are implemented. Typical applications include smart batteries, battery chargers, portable computers, instrumentation, embedded temperature sensors, and systems needing high-resolution data acquisition and processing.

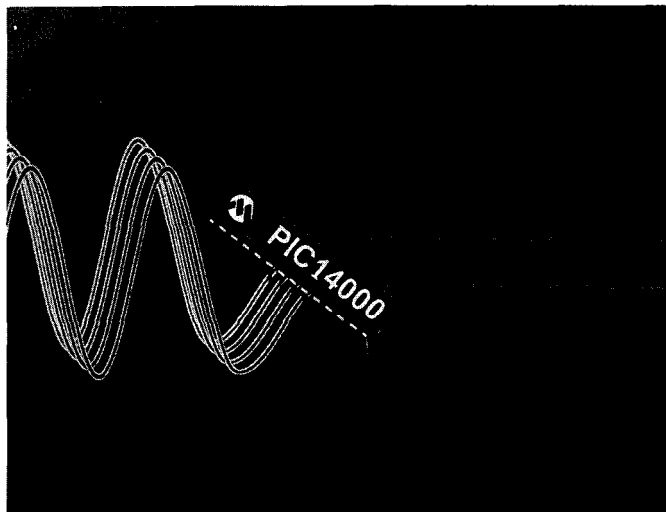
The PIC14000 offers 4-KB x 14 on-chip EPROM program memory and 192 bytes RAM based on an 8-bit RISC core, enabling it to support algorithmic-intensive calculations at up to 5 MIPS. Other features are 35 single-word instructions, up to 20-MHz operating speed, 6 internal and 5 external interrupt sources, 11 interrupts, 8 levels of hardware stack, and 38 special-function hardware registers.

The A/D and D/A converters measure and generate analog signals enabling the design of inexpensive, single-chip, closed-loop systems requiring few external components. A slope A/D converter offers programmable resolution up to 16 bits, enabling high-precision measurement of real-world signals. Two multirange D/A converters can be used for precise control applications such as charging batteries. To reduce the need for external components, the device features an on-chip low-voltage detector, temperature sensor, voltage-regulator control, and internal 4-MHz clock oscillator.

Twenty I/O pins with individual direction control are available for interfacing with displays, interrupts, data input, and communications interfaces to support any three-wire, two-wire, or single-wire interfaces. The two-channel I²C port enables high-speed serial communications between other microcontrollers as well as data storage with nonvolatile memories. The PIC 14000 supports the ACCESS.bus, Intel/Duracell Systems Management Bus (SMBus), and Standard Battery Data standards.

To address low-power concerns, particularly in portable systems, the chip supports both a Sleep mode that dissipates no more than 200 µA and a new Hibernate mode that dissipates no more than 5 µA. The device can be programmed to wake from Sleep on sensing a specific level of current flow, within a range of current flow, and outside a given range.

The PIC14000 sells for \$6.85 in quantity and is available in DIP, SSOP, and CERDIP packages.



Microchip Technology, Inc.

2355 W. Chandler Blvd. • Chandler, AZ 85224-6199 • (602) 786-7200 • Fax: (602) 899-9210

#516

PC/104 RESOURCE GUIDE

The PC/104 Consortium announces the Eighth Edition (November 1995) of its popular **PC/104 Resource Guide**. Like its predecessors, the 200-page booklet is available free to engineers and companies developing embedded systems. It includes an overview of the PC/104 standard, a cross-reference of available PC/104-related products and functions, and product listings from over 130 of the Consortium's member companies which describe the companies' PC/104 modules and related boards, peripherals, and software.

The PC/104 standard, which has also become the basis of a new IEEE draft standard (P996.1), defines a compact, self-stacking, modular form factor for embedding IBM PC and PC/AT-compatible system functions within embedded microcomputer applications. The PC/104 modules' small size (3.6" x 3.8") and

low-power requirements (typically 1-2 W per module) make them ideally suited to embedded-control applications. Such applications include vending machines, test equipment, medical instruments, communications devices, vehicular systems, data loggers, and industrial control subsystems.

PC/104 Consortium
 P.O. Box 4303
 Mountain View, CA 94040
 (415) 903-8304
 Fax: (415) 967-0995

#517

Nouveau IPC

Embedded Computer Design

Rethinking Embedded PC System Integration

Is there room for a new form factor in the embedded systems market? New chip-based plug-in modules with ISA compatibility offer designers another way to speed embedded designs to market.

Embedded systems are so widely used in the appliances of everyday life that we largely take them for granted.

From the microwave ovens, coffee makers, cellular telephones, and fax machines to the electronic systems of complex medical instruments, embedded applications form the heart of many consumer products.

Nonetheless, designing a new embedded system poses great challenges to system integrators. The search for smaller, faster, and less expensive products remains critical.

While most embedded-PC vendors seek higher levels of integration by redesigning the form factors of board-level products, others rethink the way a PC is integrated into an embedded system.

An option that increases the integration of embedded systems and single-board

computers is chip-level integration. ZF Microsystems recently introduced the Single-Device Personal Computer (SDPC), a line of ultraminiature, PC-compatible modules.

The modules incorporate an Intel 80386SX microprocessor and all associated logic chips in a single module. The company's SMX/386 OEModule, a chip-like module, complete with AT-compatible BIOS, embedded DOS, and a PC ISA bus standard interface, works well in a broad range of embedded systems requiring PC compatibility.

SPEEDING DEVELOPMENT

While creating smaller, faster, cheaper, and lighter products, designers are looking for the best and fastest way to get their

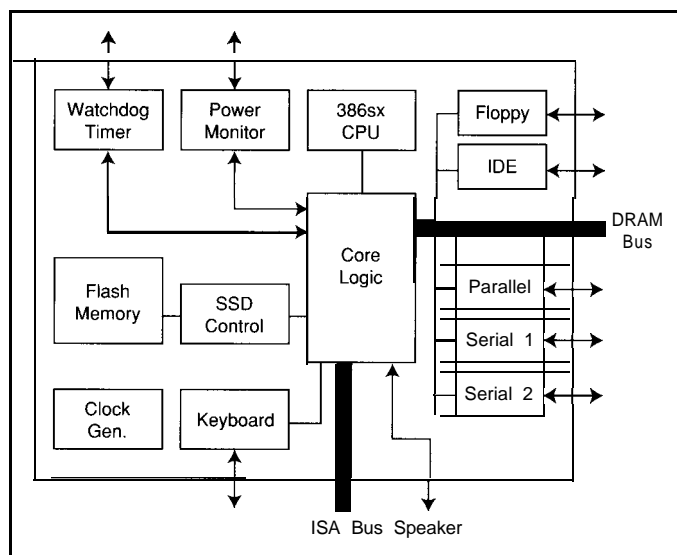


Figure 1: All the functions of a PC/AT motherboard are integrated into the SMX/386 OEModule. Embedded DOS delivers the C: prompt when the system is powered up.

embedded products to market.

With its readily available and cost-effective development tools, GUIs, and operating systems, the x86 architecture has achieved global dominance. As a result, the embedded-PC market has become highly competitive. There's a need to provide solutions other than the standard board-level offerings currently available.

SDPCs provide complete PC/AT-compatible subsystems in a semiconductor-like module that provide CPU, I/O, and flash disk functions based on Intel's x86 architecture. They're application independent computer subsystems that can be incorporated directly into a customer's proprietary circuitry.

But, before I elaborate about the SDPC, let's check out what problems engineers currently experience in designing and building products that are PC compatible.

IN-HOUSE DESIGN

This approach is most common when product volumes are in the mid- to high-level range (1,000–20,000+ units per year). However, designing from scratch has both advantages and disadvantages.

The obvious advantages are that in-house design typically offers the lowest hardware cost and greatest integration. You therefore gain the smallest, most reliable overall solution.

However, these OEMs have significant obstacles to overcome with in-house design. To semiconductor suppliers, 1,000–20,000 components per year is quite small. At today's prices, Intel or Advanced Micro Devices stand to gain only \$15,000–\$150,000 of annual revenue from the sale of 80386SX processors to such accounts.

For the large chip suppliers, the amount of revenue is too low to merit the sales and support costs. Thus, such customers are rarely given direct attention. Instead, they

are typically pushed through an industrial distributor.

Component pricing from industrial distributors is quite high compared to direct, higher-volume pricing. Distribution margins rarely support the high level of technical support required by the OEM's engineering staff. As a result, consultants are frequently used, which drives development costs even higher.

The OEMs in this segment are thus caught in the middle. Board-level systems cannot deliver the best solution in terms of cost, size, or reliability. And, semiconductor manufacturers don't want to deal with them directly since their business potential is simply too small.

In addition, in-house design, which is based on discrete components, is particularly vulnerable since any one component designed into the board could become obsolete. As the desktop market leaves older technologies behind, chipset manu-

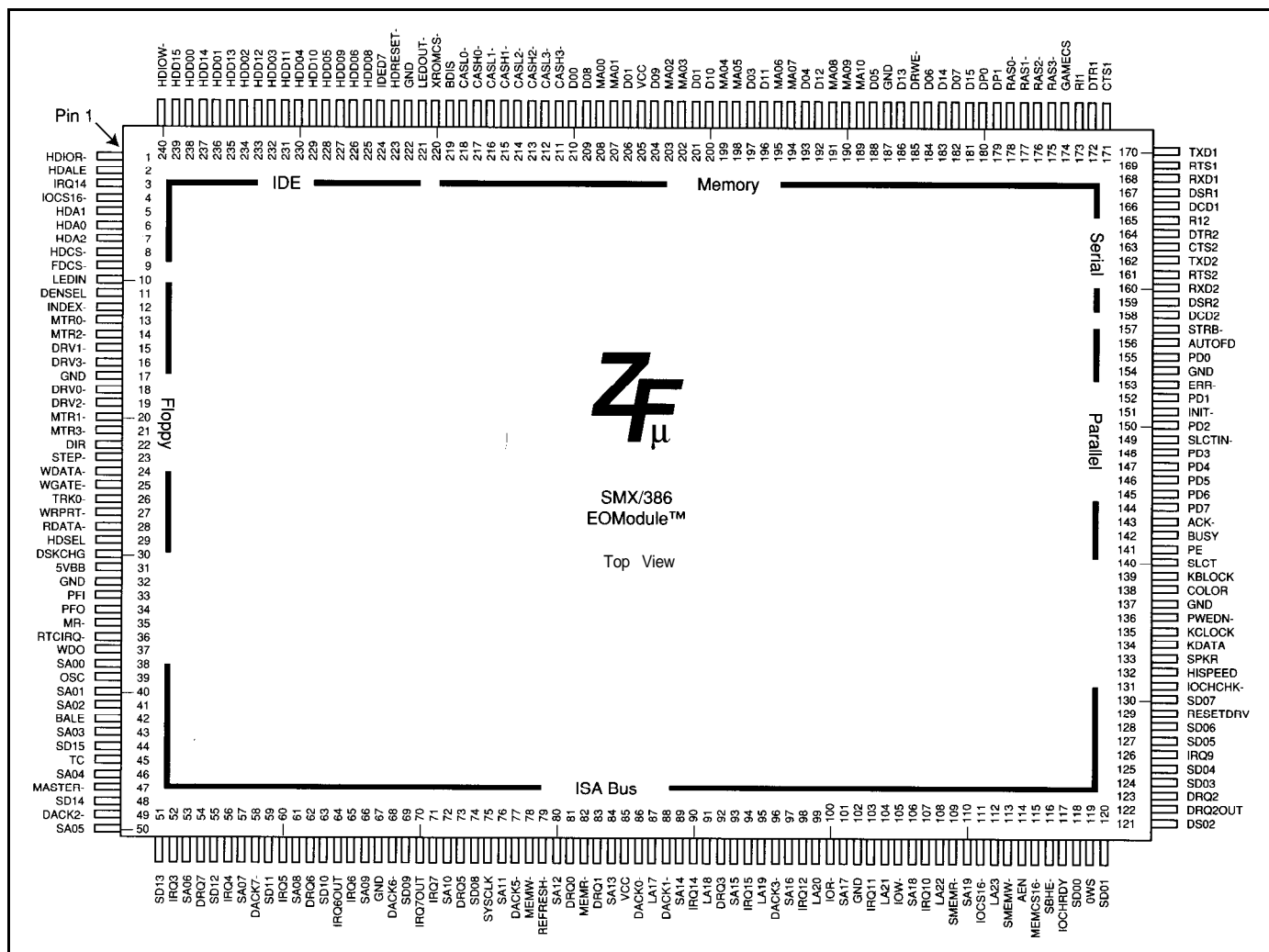


Figure 2: The pin assignments on the SMX/386 OEModule enable designers to locate the module so that connections to peripherals—memory, video, and other external functions—are convenient and simple.

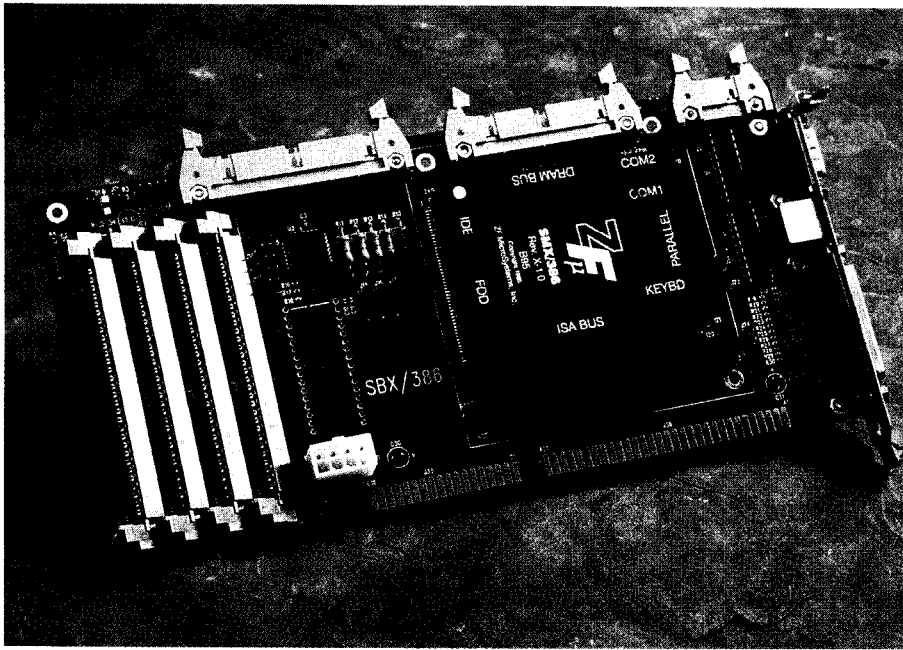


Photo 1: **SMX/386** plug-in development boards offer **OEMs** the fastest possible time to market. Developers gain an easy development environment that includes the Intel **386-based OEModule** on a PC card along with PC AT-compatible BIOS and embedded DOS.

facturers often discard components which no longer have mass appeal. This vulnerability potentially costs an entire project. The greater the number of discrete components in the design, the higher this risk becomes.

Finally, board solutions involve the ongoing cost of procurement and inventory management of the up to 100 individual components required to produce the PC-compatible portion of an in-house design.

OFF-THE-SHELF BOARDS

Engineers can also use an off-the-shelf, board-level computer from companies like Megatel, Radisys, or Ampro. The single-board computer, connected via a series of cables, then becomes the engine driving the OEM's proprietary electronics.

Although these boards have found widespread use in applications requiring 1,000 or fewer boards per year, they don't satisfy the requirements of higher-volume applications where size, power consumption, or harsh environments are a major concern.

The main drawback of this type of embedded solution lies in the inherent decrease in reliability. Shock and vibration can cause the interconnecting cables and mounting hardware to loosen.

Additionally, many board-level suppliers build bus-oriented products, requiring large backplanes and card cages. The form factors of these products are typically

too large and costly for most high-volume applications. The business structures of these suppliers are oriented toward the small to medium OEM and industrial end-user markets (100-1,000 units). The products are built in relatively small volumes and carry relatively high price tags (e.g., a \$100,000 anesthesiology machine).

The higher cost of these solutions and the additional overhead of power, space, and assembly time limit the OEM designer to use these solutions in similar low-volume, high-end products.

SUBSYSTEM INTEGRATION

Engineers can also integrate a subsystem with a single-board computer in their equipment either with their own team or an outside VAR doing the integration. However, this is just a bolt-on approach.

The advantage of the subsystem approach is that the subsystem becomes a "standard" unit that can be easily integrated into additional projects. It also represents a convenient method for easy field service swap-out.

The penalties of large size and higher cost remain, thus again limiting solutions to high-end, low-volume applications.

SINGLE DEVICE PC

SDPCs represent a new alternative for the OEM designer. The modules offer many

IF YOU DO FUNCTIONAL TESTS

YOU NEED HOT SWAPPING ELECTRONIC EXTENDERS

For PCI, ISA, VXI, EISA, VESA,
Micro Channel & NuBus.
Custom Designs available.

Electronic ISA Extender /
PC Mini Extender



Electronic PCI Extender /
PCI Mini Extender



Insert/Remove Cards With PC Power On!!

- Save Time Testing And Developing Cards
- Save Wear On Your PC From Rebooting
- Adjustable Overcurrent Sensing Circuitry
- NO Fuses, All Electronic For Reliability
- Single Switch Operation W/Auto RESET
- Optional Software Control Of All Features
- Breadboard Area For Custom Circuitry
- And More...

Full line of passive extenders:
PCI, ISA, VXI, EISA, VESA,
Micro Channel & NuBus

Passive PCI Extender

Passive EISA Extender



Passive MC32 Extender

Passive ISA Extender



AZ-COM, INC.

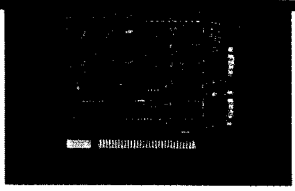
3343 VINCENT RD., #D
PLEASANT HILL, CA 94523 U.S.A.
TEL: 510-947-1000
FAX: 510-947-1900

24-Hour Fax on Demand:

51 0-947-1 000 Ext.7

Email: sales@az-com.com


VISIT OUR HOME PAGE AT:
<http://az-com.com/>



Enhanced Windows & OS/2 Communications

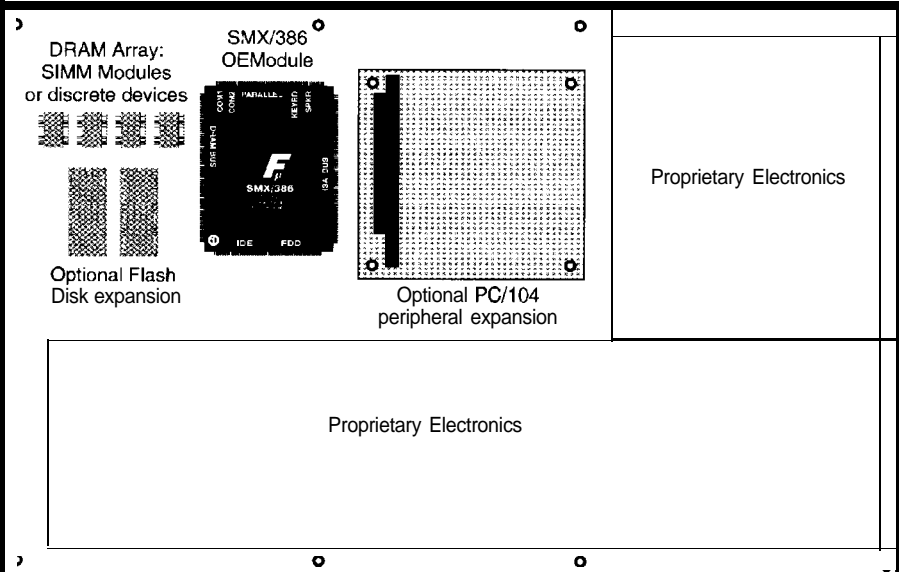
- Two RS-232 Ports
- Independently Selectable Address
 - IRQ(2-5, 10, 11, 12 & 15)
- 16550 UART Standard
- 16 Bit Addressing
- Free Technical Support

Part #3088 Price: \$89.00



Sealevel Systems, Inc.
P.O. Box 830 • Liberty, SC 29657
(864) 843-4343

#203



DRAM Array: SIMM Modules or discrete devices

SMX/386 OEModule

Optional Flash Disk expansion

Optional PC/104 peripheral expansion

Proprietary Electronics

Proprietary Electronics

Figure 3: PC-compatible **OEModules** can control a wide array of **PC/104** peripherals. Designers therefore not only reduce overall system costs but also speed up their time-to-market.

SBCIEEE-488 AND RS-232

SINGLE BOARD COMPUTER WITH C SBC488 APPLICATIONS

Why tie up a PC/Workstation? The SBC488 is a stand alone test controller, an OEM GPIB 8 RS-232 interface and an RS-232 to IEEE 488 converter.

SBC488 FEATURES

- IEEE 488.2 interface/RS-232 interface.
- Z-World(tm) SmartCore with DYNAMIC C(tm).
- Optically Iso. 12 Bit D/A, A/D and 24 Bit Digital I/O
- Source code available for OEM applications.
- WINDOWS(tm) control program available-SBCwin

STARTING AT \$395 EA. IN SINGLE PIECE QUANTITIES

AVAILABLE EXCLUSIVELY FROM:

AMERICAN DISTRIBUTORS INC.
8 WHIPPANY RD
WHIPPANY NJ
07981



TEL (201)-428-0510
FAX (201)-887-0923
CompuServe 74863.3435 Designed by: **TIDAL ENGINEERING**
WWW.TIDAL-ENG.COM SmartCore and Z-World are trademarks of zworld Engineering

#204

AMX The Real-Time Multitasking Kernel

680x0, 683xx 80x86/88 real mode
80386 protected mode i960® family
R3000, LR330x0 Z80, HD64180

NEW • DOS Compatible File System
• TCP/IP • 29K support

- Compact, ROMable, fast interrupt response
- Preemptive, priority based task scheduler
- Mailbox, semaphore, resource, event, list, buffer and memory managers
- Configuration Builder utility
- InSight™ Debug Tool
- Comprehensive documentation
- No royalties, source code included

For a **free Demo Disk and product description**,
Phone: (604) 734-2796
Fax: (604) 734-8114



KADAK Products Ltd.
206 - 1847 West Broadway
Vancouver, BC, Canada V6J1Y5

#205

benefits of an in-house design. As a discrete component in a proprietary design (see Photo 1), the end product has a high level of integration, reduced component count, and increased reliability.

The attendant headaches of developing or licensing and modifying a BIOS and operating system are eliminated. Also, since everything is within the one chip, the procurement and inventory of numerous discrete components becomes minimal.

Using this approach, engineers can use an SDPC's full PC/AT compatibility to build cost-effective embedded systems. They gain the benefits of the huge base of PC software and hardware while dramatically reducing development overhead.

For instance, when a company builds a medical device, such as a blood analyzer, that is PC compatible, the challenge is not to build a better PC, but a better blood analyzer. By using an SDPC with an already developed BIOS, blood analysis, not PC BIOS implementation, gets greater attention. The end result: a product is introduced more quickly and at lower cost.

Rather than outsourcing portions, most OEMs prefer designing products completely because it's expedient and volumes are low enough to handle internally. But, with higher product volumes, other problems come such as reliability and testing. These issues can make full in-house designs less attractive or unfeasible.

For those companies, an SDPC represents a fast way to incorporate a fully PC-compatible system controller into their prod-

uct. The module represents the lowest possible development risk and greatest reduction in time-to-market.

As you can see in Figure 1, the SMX/386 OEModule includes all standard motherboard functions. It has a 33-MHz 80386SX CPU, full PC core logic, full 8- or 16-bit ISA bus, DRAM controller, serial and parallel I/O, floppy and IDE disk controllers, embedded AT-compatible BIOS, and 256 KB of user-available flash disk. Embedded DOS delivers the C: prompt when the system is switched on.

In Figure 2, you can see that the module provides 240 pins with a full 40-pin interface to a standard IDE hard disk drive. The pins interfacing to the ISA bus are conveniently located to ease board layout.

The common denominator for the multichip modules appearing in the embedded market is the attempt to provide close to PC functionality in a single device. ZF Microsystems OEModules, S-MOS Systems Cardio, or MicroModule Systems' Northstar series of Pentium processor modules are all examples of products that could be classified as SDPCs.

Although each of these products targets a different segment of the embedded market, they all bring a high level of integration. The embedded designer is able to shorten development times and eliminate the overhead associated with incorporating a single-board computer into an embedded application.

These SDPC modules represent a layer of functionality ideal for midrange produc-

tion volume levels. They're optimally suited for the middle ground between high-volume embedded markets such as Intel's '386EX and the lower-volume single-board computer (including PC/104 CPUs). Notably, with lower volume, additional cabling and hardware mounting are acceptable alternatives to designing from scratch.

The most significant advantage of the SDPC's component-like construction appears when designing for harsh environments. In a stressful environment, the issues of mounting hardware and system size rise to the forefront of system design decisions.

The integrated SDPC fits an entire AT motherboard into a space that's roughly equivalent to that required for two Intel 486DX chips. Designed for embedded systems, it needs minimal external components. The chip-like design of the surface-mount device ideally suits SDPCs to designs demanding low power consumption, high reliability, and small product size.

As Photo 2 illustrates, an OEModule fits easily in a hand. While many single-board computers of equivalent technology require a 5.75" x 8" footprint (46 square inches), this module only takes up 2.2" x 3" or 6.6 square inches. It replaces many proprietary solutions that add more cables and wiring to embedded systems.

By SDPCs using the industry-standard PC/AT ISA bus, designers can leverage the most cost-effective and widely available selection of hardware and software. This flexibility allows product designers to start with a proven architecture that is widely serviced and takes advantage of low-component prices.

With PC compatibility, OEMs have access to many cost-effective development tools and operating systems. ZF Microsystems also provides CAD files so designers can integrate the module into board layouts like any other chip.

ISA AND INTEGRATION

Many designers looking for ease of integration want to use the vast array of PC-compatible devices and

Photo 2: ZF Microsystems' new SMX/386 chip module is tiny enough to fit in the palm of your hand, yet powerful enough to run many embedded PC applications. The OEModule's overall footprint measures a scant 2.344" x 3.125" (59.54 mm x 79.88 mm).



peripheral products available for the ISA standard.

The SDPC's fully compatible ISA bus interface makes this possible. By adding either an ISA PC-type slot or a PC/104 connector for expansion, hundreds of standard PC peripherals are available. The OEM can then enjoy the best of both worlds: proprietary in-house design benefits and industry-standard peripherals.

In a sense, SDPC is the step after PC/104. At the time of its conception, PC/104 helped OEMs bring highly integrated embedded devices to market as quickly as possible. Now backed by about 130 member companies in the PC/104 Consortium, the standard substantially reduces development costs, risks, and development time.

Embedded-systems designers also have widespread use of PC standard hardware and software. Components for this widely available architecture are more economical than for traditional architectures, such as STD, VME, and Multibus.

However, PC/104 has some inherent limitations as a CPU platform. As embedded-PC system software becomes more complex and requires greater amounts of memory, an embedded CPU design becomes less cost-effective within the constraints of the PC/104 board outline.

By incorporating an SDPC directly within a proprietary design and adding a PC/104 connector for expansion, overall system cost is greatly reduced without sacrificing the benefits of the wide variety of PC/104 peripherals available (see Figure 3).



Complete x86 Solution!

Multitasking in Real and Protected Mode

smx Full-featured, high-performance, preemptive kernel. Customized to x86 processors. Ideal for demanding applications. Real mode works stand-alone or with DOS. 16-bit, segmented protected mode works with *pmEasy* 16 or 286 I DOS Extender. 32-bit, flat protected mode works with *pmEasy32* or TNT.

Protected Mode Environment

pmEasy 16- or 32-bit protected mode entry, DPMI services, application loaders, *Periscope/32* and *Soft-Scope* debugger support

DOS-Compatible File System

smxFile Full-featured file manager. IDE, floppy, flash, RAMdisk, and PCMCIA drivers available.

Loadably Modules

smxDLM Runs independent executables as tasks which may be downloaded or loaded from disk. floppy, flash, etc.

Networking

smxNet TCP/IP stack. Fast UDP. Packet driver interface. Ethernet, SLIP, and PPP drivers. FTP, SNMP, NFS. Net server, other protocols

Task-Level Debugging

smxProbe Provides tracing and symbolic debugging. Works with or without code debuggers. Local or remote operation.

C++ Classes

smx++ Class library built upon smx. Provides fully OOP-compatible kernel interface.

Extended Memory, Real Mode

smxEMS Allows copying data between real memory and extended memory buffers or accessing extended memory via a window.

User Interface

smxWindows Text windowing. Dresses up user interfaces. *Meta-Window* & *Zinc* graphics support.

Low Cost & Easy to Use

Royalty-free licenses. Supports common, low-cost C compilers, debuggers, and locators. Great manuals, Source code available

Reliable

On the market 7 years. 100's of applications. Extensive error checking. 30-day free trial.

1-800-366-2491

mdi@earthlink.net

MICRO DIGITAL, INC

fax 714-891-2363 int'l 714-373-6862

<http://www.earthlink.net/~mdi>

ZF MICROSYSTEMS' SMX/386 OModule

Key features:

- Chip-like design in very small form factor (2.2" x 3")
- PC compatibility
- Built-in AT-compatible BIOS and embedded DOS PC/I 04 standard interface
- Fast, easy development environment

Body Size:

- 2.20" x 3.00" x 0.450"

Overall Footprint:

- 2.344" x 3.125"

leads:

- 240 (50 x 70) gull wing pins at 0.030"

Pinout:

- See Figure 2

Power requirement:

- +5 V \pm 5% at 750 mA

Operating environment:

- 14°F to 158°F
- -10°C to +70°C
- 5-95% relative humidity (noncondensing)

Storage temperature:

- -67°F to 185°F
- -55°C to +85°C

Weight:

- 5 oz.
- 140 g

development, but also a CAD file containing the component outline for the module.

Compact, rugged, easy. Perhaps it's time to seek another solution. EPC

David L. Feldman is president and chief executive officer of ZF Microsystems, a startup embedded systems developer founded in April 1995. Dave formerly served as the founder and former chief executive of Ampro Computers, the creator of the PC/I 04 concept, and has more than 25 years of experience in business management and the embedded systems market. He may be reached at dfeldman@zfmicro.com.

CONTACT

SMX/386 OModule (\$260 in 1000s)
ZF Microsystems, Inc.
1052 Elwell ct.
Palo Alto, CA 94303
(415) 965-3800
Fox: (415) 965-4050
sales@zfmicro.com

A BETTER WAY?

Off-the-shelf hardware and development systems give embedded system designers tremendous flexibility. They potentially save development time and purchasing costs.

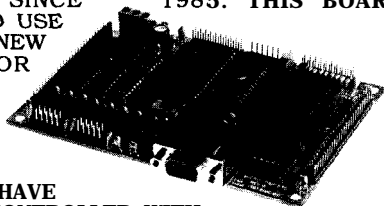
Development done on a PC/AT system is easily transferred to the embedded system with little or no modification. In the case of OModules, customers receive not just an AT BIOS and embedded DOS for

I R S

- 4 16 Very Useful
- 4 1 17 Moderately Useful
- 4 18 Not Useful

SINGLE BOARD COMPUTER

THAT'S RIGHT! \$129.95 FOR A FULL FEATURED SINGLE BOARD COMPUTER FROM THE COMPANY THAT'S BEEN BUILDING SBC'S SINCE 1985. THIS BOARD COMES READY TO USE FEATURING THE NEW 80535 PROCESSOR WHICH IS 8051 CODE COMPATIBLE. ADD A KEYPAD AND AN LCD DISPLAY AND YOU HAVE A STAND ALONE CONTROLLER WITH ANALOG AND DIGITAL I/O. OTHER FEATURES INCLUDE:



- * UP TO 24 PROGRAMMABLE DIGITAL I/O LINES
- * 8 CHANNELS OF FAST 8/ 10 BIT A/D
- * OPTIONAL 4 CHANNEL, 8 BIT D/A
- * UP TO 4, 16 BIT TIMER/COUNTERS WITH PWM
- * UP TO 3 RS232/485 SERIAL PORTS
- * BACKLIT CAPABLE LCD INTERFACE
- * OPTIONAL 16 KEY KEYPAD & INTERFACE
- * 192K OF MEMORY SPACE, 64K INCLUDED
- * 8051 ASSEMBLER & MONITOR INCL., BASIC OPT.

1985-1995
10
YEAR
ANNIVERSARY

EMAC, inc.

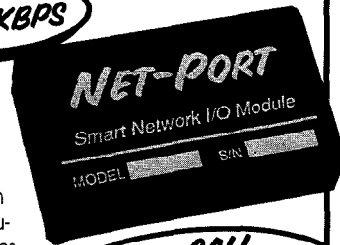
618-529-4525 Fax 457-0110 BBS 529-5708
P.O. BOX 2042, CARBONDALE, IL 62902

NET-PORT

\$69.00

115 KBPS

Net-Port is a complete serial data acquisition and control system in a 3/4-cubic-inch package. The potted Net-Port contains a variety of digital and analog I/O along with power supply regulation and communication line drivers. Net-Port requires no programming. A simple ASCII command protocol sets and reads all I/O.



CALL
1-800-635-3355
TO ORDER

FEATURES

- RS-232A, RS422, and RS-485 at 300 bps to 115 kbps
- Sixteen parallel I/O lines and PC bus
- I-channel, B-bit ADC (Net-Port B)
- P-channel, 12-bit ADC and 2-channel, 12-bit DAC (Net-Port E)
- PWM output: 2Hz to 3.5 kHz, 5-95% duty cycle
- Simple ASCII command set, requires no programming!
- High-performance, built-in functions: parallel I/O buffering, LCD and keypad control, analog data averaging, data logging
- Sixteen-character ID allows hundreds of Net-Ports
- Small size, encapsulated construction
- Wide power supply input range

NET-PORT B \$69.00 NET-PORT E \$99.00
NET-PORT carrier board w/power supply \$49.00

Prices do not include shipping
Features subject to change

MICROMINT, INC.

4 Park Street • Vernon, CT 06066 • (860) 871-6170 • Fax (860) 872-2204

Designing and Building PCs for Harsh Environments

For computers to work in harsh environments, extra care must go into the design and manufacturing of the systems. Vinit characterizes harsh environments, examining what design criteria are needed for specific stresses.

The widely accepted Wintel PC architecture is increasingly found in inhospitable habitats. Unfortunately, the hardware designers of PC chips never intended their products to be operated in anything but comfortable human habitats.

So, how does one design and manufacture PC-compatible computers that operate reliably in environments like vehicles, industrial plants, and outdoors?

Designing electronic systems for harsh environments is well understood in aerospace applications. The Department of Defense (DoD), NASA, and commercial airplane manufacturers have designed customized computers, and mechanical structures with operating lifetimes of over 20 years.

Besides the clever packaging of electronic and mechanical components, they use components specifically designed for extreme environments. As a result, the component technology used in aerospace systems has generally lagged commercial computer technology by 3-5 years.

The effect of harsh environments on electronic components is also reasonably well understood. Based on this information, DoD and NASA have developed qualification procedures and approved parts lists.

These approved lists are organized according to different harshness categories for a variety of environmental conditions. Some parts survive the radiation environments of low-earth orbit, some endure atmospheric flight, and so on.

Most harsh environments are also characterized and defined in accordance with environmental design standards. (There are many such standards maintained by standards organizations worldwide.) In addition many companies, particularly vehicle manufacturers, develop their own proprietary environmental specifications.

After looking at the characteristics of harsh environments and their affects on components, I'll provide some techniques for designing and developing PCs to operate in harsh environments.

CHARACTERIZING HARSH ENVIRONMENTS

Computers are used in many harsh environments: industrial plants, ground vehicles (on and off road), airborne vehicles, geophysical, traffic monitoring, and so on. Each of these environments has its own thermal, vibration, shock, humidity, power, and EMI characteristics.

Computers have to be designed and manufactured to operate reliably in these adverse conditions. To do so, we need to understand the effect of extreme conditions on rugged computers. Characterizing and then simulating these extreme conditions is required to design and test a rugged computer for proper operation in the field.

Let me spend some time characterizing the environmental conditions that rugged computers need to operate in.

Temperature extremes range from the cold of Canada to the heat of Saudi Arabia and are caused by extreme ambient temperatures. These extremes are also man-made. For example, there's the cold of a

food freezer or the heat from an internal combustion engine.

Generally, commercial products are designed to operate from 0°C to +45°C, industrial products from -40°C to +85°C, and military avionics from -55°C to +125°C. However, there are as many exceptions to these standards as there are computers using them. Notably, for most applications, the industrial temperature range is sufficient.

All computers, rugged or not, are subjected to shipping-and-handling shock during the trip from their place of manufacture to the place of operation. Computers in environments with rotating or translating machinery are subjected to additional shock and vibration during their operational life.

The severity of the shock and vibration depends on the role of the machine and its condition. A computer mounted on an injection-molding machine is subjected to repeated shocks as the mold is compressed. A forklift-mounted computer, on the other hand, experiences continuous vibration and occasional collision shock.

Barring lightning surges, AC power from national power grids is generally pretty stable. On vehicles, however, DC and AC power is noisy and full of nasty transients and voltage surges. Rugged computers must deal with power fluctuations without data loss and their power supplies must survive the power aberrations.

In a heavy-duty vehicle with a 12-VDC power system, the power normally varies from 11 to 16 VDC and dips down to as low as 4.5-6 VDC during a cold crank (engine starting at -40°C). Jump starts can cause voltage surges of -12 to +24 VDC. Voltage-regulator failure causes the battery electrolyte to boil off with resulting voltage surges of 75-130 VDC.

Load-dump transients occur when the vehicle's alternator current load abruptly reduces. This fallout happens, for example, when headlights are switched off. This simple action can cause a voltage spike of up to 150 VDC with a risetime of 100 µs and a decay time of 100 µs to 4.5 s.

Rugged PCs are subject to electrostatic shocks similar to desktop computers. Humans generate charges in normal daily activities of more than 10,000 V. When we touch a computer, the faceplate, or an I/O connector, the electrostatic discharge (ESD) can damage the computer.

Standards Organization	Description
National Electrical Manufacturers Association (NEMA) Washington, DC	NEMA has standards for practically every electrical component or system used in electrical products. The most commonly used specifications for rugged computers are the NEMA I-1 2, which outline how enclosures have to be sealed. NEMA 3 is for outdoor use (e.g., light rain), NEMA 4 is for indoor hosedown, and NEMA 12 is for indoor dripping liquid.
International Electrotechnical Commission (IEC) Geneva, Switzerland	IEC gives comprehensive standards for all environments that an electronic device may operate in. IEC 801 defines electromagnetic compatibility for industrial-process measurement and control equipment. IEC 529 refers to IP Codes, which offer sealing levels and are similar to NEMA.
International Standards Organization (ISO) Geneva, Switzerland	ISO has over 700 standards for components and systems used in vehicle applications. It also defines standards for other rugged applications.
European Commission (EC) Brussels, Belgium	The EC defines the European Norms (EN) directives that are published in the Official Journal (OJ) of the European Commission. EN55022 defines emissions (Class A is for industrial applications) and EN50082-1 defines immunity levels for information technology equipment.
Electronic Industries Association (EIA) Washington, DC	EIA defines electrical and communications standards that apply to rugged computers.
Institute of Electrical and Electronics Engineers (IEEE) Piscataway, NJ	IEEE defines electrical and communications standards that apply to rugged computers.
American National Standards Institute New York, NY	ANSI defines electrical and communications standards that apply to rugged computers.
Federal Communication Commission (FCC) Washington, DC	FCC outlines many standards for RF compatibility. Part 1.5, subpart B defines emission limits for unintentional radiators such as rugged computers. Class A is for industrial usage.
Society of Automotive Engineers (SAE) Warrendale, PA	SAE defines standards for operating equipment on vehicles. The most commonly used standards are the <i>SAE Handbook</i> and the <i>SAE J1455</i> , which recommend environmental practices for electronic equipment design (e.g., heavy-duty trucks).
U.S. Department of Defense Washington, DC	U.S. DoD offers comprehensive standards for all environments an electronic device may operate in.
American Association of Railroads Washington, DC	AAR defines standards for operating equipment on locomotives and rail cars. For rugged computers, the ATCS Specification 110 defines environmental requirements for electronic equipment on locomotives.
Aeronautical Radio Incorporated (ARINC) Annapolis, MD	ARINC defines standards for operating computer equipment on aircraft. Enclosure standards are referred to frequently for rugged computer applications.

Table 1: Many industry environmental standards exist and can be used in designing and manufacturing rugged-PCs.

Although ESD has a very short period, the ESD waveform is fast rising. It is this fast risetime that harms ICs. Typically, an ESD event lasts 1 ns. Its fast risetime is what causes parasitic-inductive effects (conductive ESD) and induced voltages (radiated ESD).

ESD can cause three types of failures: catastrophic (e.g., a dead I/O port), reset of computer [e.g., a soft failure], and latent (i.e., component lifetime reduces).

Rugged computers usually operate in noisier RF environments than desktop computers. In vehicles, power-supply transients can easily be induced into I/O cabling that is routed around the vehicle.

Electrical noise as a result of engine operation can add 3 Vp-p and 50-Hz to 10-kHz noise into the power system. Rugged computers themselves can emit debilitating RF, disrupting other equipment such as wireless radios that are in proximity.

Rugged computers may need to survive submersion, salt spray, wash-down, and high-humidity (often combined with high temperature) environments.

There are many industry standards that characterize the rugged environments PCs operate in. Some of these are presented in Table 1.

HARSH ENVIRONMENT EFFECTS

Not surprisingly each harsh environmental stress causes different damage to electrical components. I'll go through each stress, outlining what damage can occur.

Parameter	Measure of no appreciable degradation in low-level contact resistance in connector pins.
Durability:	
• mating	Durability over a number of mating cycles. Look for wear in gold plating.
• thermal shock	Changes in resistance of connectors after thermal cycling and thermal shock.
• humidity	Resistance to high humidity and temperature, including moisture penetration and migration.
Electrical:	
• dielectric	Ability to withstand overvoltages after thermal cycling.
• insulation	Resistance of insulation to DC leakage current in normal operation and after humidity test.
Shock & Vibration	Effect of shock test on connector. Short and long term vibration effects.
Gas Tightness	Effect of hostile gas atmosphere on connector.

Table 2: Connectors are a key source of component failure in rugged computers. When choosing connectors for a rugged PC design, these parameters should be provided by the connector manufacturer.

Thermal stress, caused by power on/off cycles and ambient thermal fluctuations, causes alternating stresses in the computer assemblies. These stresses lead to cracks in the structural elements as the fatigue life of the component is used up. Elements such as solder joints, PCB plated through-holes, and crimped wires in connectors are the most likely to develop fatigue cracks and result in failures.

Each stress cycle a system is subjected to uses up a part of its fatigue life. Materials can fracture when subjected to repeated stresses below their rated static strength. Submicroscopic cracks are caused by these stresses. Eventually, these minuscule cracks join to form visible cracks.

Steady-state extreme temperatures also affect component life. The effect of junction temperature on electronic components is directly related to component life. Elevated junction temperatures of 120°C results in 30% more end-of-life failures (lifetime defined as 90,000 power-on hours) than operating at junction temperatures of 100°C.

To put it another way, for every 10°C rise in junction-op-

erating temperature, the life expectancy of a semiconductor halves. Atypical semiconductor device has a MTBF of 200,000 hours at 50°C or 23 years operating continuously. At 80°C operating temperature, the MTBF reduces to 3 years.

Vibration and shock stresses also cause component fatigue. Occasional vibration and shock, such as those caused during shipping, can be debilitating, but the cyclic vibration and shock caused by moving vehicles or rotating machinery causes similar fatigue cracks to those of thermal cycling stress.

Humidity corrodes metal parts because of galvanic and electrolytic action. It also

changes electrical properties, condensation (resulting in electrical shorting), and decomposition of organic components due to attacking organisms such as mildew.

Common experience shows that the most common failure as a result of relative motion caused by thermal cycling, vibration, shock, and transportation is in interconnect systems. About 30% of all failures are in connectors, cables, and PCB interconnects. Other components in a rugged computer system also can fail during stress in different ways.

Electronic components are rapidly shrinking in size and increasing in complexity. No other component represents this trend more than microprocessors. The Intel Pentium 5-V CPU running at 75 MHz generates 10 W of heat in a surface area of 4

square inches. Generally, the hottest component in a rugged PC is the CPU, and it is the most challenging component to thermally manage.

Though memory densities are increasing almost as fast as CPU processing power, heat dissipation is not a major problem for memory.

Hard-disk drives are the most fragile subsystem in rugged PCs, mostly because they have a poor tolerance for shock, vibration, and temperature extremes. Smaller HDDs, particularly the 2.5" and 1.8" form factor, have become more tolerant to occasional shock, but are still quite vulnerable to repetitive shock and continuous vibration.

Many rugged computers now offer solid-state flashdisk storage instead of HDD storage. Flash prices are decreasing rapidly and densities are creeping up.

Flash disk drives tend not to be used in regular operation of the

rugged PC, except to add or save data from the hard drive. As a result, FDDs are seldom the weak point in a rugged system.

Batteries have limited lifetimes and are generally difficult to repair in rugged computer in-field applications. Commonly, batteries power static RAMs during power shutdown. These SRAMs use more current at higher temperatures, thereby shortening battery life.

Rugged PCs generally use graphic displays in place of the alphanumeric displays of field instrumentation. There are several graphic flatpanel technologies: electroluminescent (EL), liquid crystal display (LCD), plasma, and new ones under development.

These displays have lower operating temperature boundaries and are susceptible to shock and vibration, though not as sensitive as hard disks. For LCDs, backlight longevity is also a problem. The fluorescent tubes (cold and hot cathode) degrade in brightness with use.

As mentioned, the most common failure modes for rugged computers are in the system interconnects. The choice of con-

CPU	Maximum Power(W)	Maximum Corn. T _{case} (°C)	Maximum Ind. T _{case} (°C)
486SX-25	2.0	85	N/A
486DX-33	4.6	85	110
486DX2-66	6.0	85	110
486DX4-100	4.3	85	N/A
Pentium P54C	10.0	70	N/A

Table 3: Wintel CPUs are a leading source of "hot" components in rugged PC designs. This table shows the thermal parameters of Wintel CPUs.

ACTIVE COOLING SYSTEMS

Rugged computers operating in elevated ambient temperatures benefit from two types of active cooling systems: liquid cooling and thermoelectric cooling.

Liquid Cooling Systems

A water-cooled cold plate with copper tubing is an inexpensive way to cool a rugged PC. These liquid cooling systems are rated by thermal resistance (HSR) versus water flow.

Typically, copper-tubing cold plates have a thermal resistance of 0.005-0.03 °C/W at a water flow rate of 1-1.5 gal./min. Multiplying thermal resistance by the heat load gives the temperature differential between the computer enclosure and ambient air.

These plates can be attached to a computer enclosure surface, which acts as a heat-sinking surface for internal components.

Thermoelectric Systems

A thermoelectric module is a small solid-state device that acts as a heat pump. Heat is removed from one surface and

dissipates to the surrounding air using the Peltier Effect.

The Peltier Effect, along with discoveries by Seebeck and Lord Kelvin, shows that current flowing through two types of conductors causes a temperature change.

Solid-state thermoelectric coolers are constructed as a sandwich of two ceramic plates with n- and p-type semiconductor pairs. When DC voltage is applied, this device transfers heat from one ceramic plate to the other.

If the current flow is reversed by reversing the voltage, the heat transfer occurs in the opposite direction. Hence, a thermoelectric device cools in high-ambient temperatures and heats in cold ambient temperatures.

Several thermoelectric devices can be used in parallel to produce distributed cooling. In addition to the thermoelectric device, a heat-sink unit and fan is needed to dissipate the heat pumped by the thermoelectric module.

Also, most thermoelectric devices operate on DC voltage and require a power supply and thermal control circuit. These components have to be carefully mounted to the computer enclosure to avoid adding further active heat load.

nectors, cables, and proper integration of cable harnesses in a rugged system are imperative. The connector and cable parameters to be considered are presented in Table 2.

mercial PC CPU card. The Kinetic 8905 Kool is a conduction-cooled card that's integrated with a Ziatech ZT8905 Pentium CPU board. It conducts heat from the

Pentium chip to the rugged card cage they are plugged into.

Fluorinert liquid cooling gel-paks offer another easily integrated method to achieve

DESIGNING RUGGED PCs

Knowing the environmental conditions a rugged PC has to operate in is one thing. Designing one that operates reliably for that particular environment using off-the-shelf components and subsystems within often stringent budgets is tricky. Let me give you some techniques for designing rugged PCs.

With thermal stresses, the components of primary concern include the CPU, hard drive, and LCD. Intel 5-V CPUs (3-V CPUs dissipate less heat) need considerable cooling as shown in Table 3.

The CPU can be cooled with airflow. However, most rugged computers do not use fans because they are unreliable, and in the case of sealed computer systems, cool air is unavailable.

Thermoelectric cooling is another form of active cooling that removes heat from the CPU surface. But, with this method, you face the problem of dissipating the heat from its hot surface (see "Active Cooling" sidebar).

Passive cooling is the ideal method to remove heat from CPUs. Photo 1 offers an example of the passive cooling of a com-



Your 1st Choice for C Compilers

Byte Craft's optimizing C compilers are fast and efficient. C extensions provide control over bit manipulations, I/O port and memory definitions, as well as support for direct register access and interrupts.

We respond to your C compiler needs.

Byte Craft Limited

421 King Street N., Waterloo, Ontario CANADA
Tel: (519) 888-6911 Fax: (519) 746-6751
BBS: (519) 888-7626 Email: info@bytecraft.com

PIC16/17Cxx

MELPS740

COP8

MC68HC08

Z8

MC68HC05



HEAT LOAD TO BE DISSIPATED

The total heat load (Q) to be dissipated from a rugged computer is calculated as follows:

$$Q = Q_{\text{Active}} + Q_{\text{rad}} + Q_{\text{conv}} + Q_{\text{cond}}$$

1. Active Heat load (Q_{active})

$$Q_{\text{Active}} = VI$$

where V is the voltage supplied to rugged computer (V) and I represents the current consumption of rugged computer (A).

2. Radiation heat load (Q_{rad})

$$Q_{\text{rad}} = F\epsilon sA(T_{\text{amb}} - T_c)$$

where F is the shape factor (worst case value = 1), ϵ is the emissivity (worst case value = 1), s is the Stefan-Boltzman constant ($5.667 \times 10^{-8} \text{ W/m}^2\text{K}^4$), A is the area of cooled surface (m^2), T_{amb} is the ambient temperature (K), and T_c is the cold-plate temperature (K).

3. Convection heat load (Q_{conv})

$$Q_{\text{conv}} = hA(T_{\text{amb}} - T_c)$$

where h is the convective heat-transfer coefficient ($\text{W/W/m}^2\text{ }^\circ\text{C}$) (a typical value is 21.7 for a flat-horizontal plate in air at 1 atmosphere), A is the area of cooled surface (m^2), T_{amb} is ambient temperature ($^\circ\text{C}$), and T_c is cold-plate temperature ($^\circ\text{C}$).

4. Conduction heat load (Q_{cond})

$$Q_{\text{cond}} = \frac{(kA)\Delta T}{L}$$

where k is the thermal conductivity of the computer enclosure ($\text{W/(m }^\circ\text{C)}$), A is the area of conducting surface (m^2), L is the length of conducting path, and ΔT is the temperature difference from hot to cold side (usually $T_{\text{amb}} - T_c$).

passive cooling. The fluorinert liquid is contained in hardy pouches that can be placed between a heat-generating component and a heatsink. The fluorinert convects heat from the hot component to the cold heatsink. It also spreads heat over other components, reducing thermal gradients.

Designing PCs with extended-temperature CPU cards is also recommended. The extended-temperature cards use off-the-shelf chips, but augment them with:

- *industrial- or military-temperature-range CPUs
- other wider-temperature-range components
- slower clock speeds which put a safety margin into chip timing

- PCB material with a lower thermal coefficient of expansion
- a screening process that exposes the systems to elevated temperature burn-in

In many custom aerospace computers and MIL-VME designs, computer

Photo 1: Kinetic Computer designed the **8905-Kool** cooling card to conductively remove heat from the pentium chip on the Ziotech **STD 32** Pentium CPU card and dissipate it into a **Kinetic RCC-32 sealed, rugged STD 32** enclosure.

boards are constructed with aluminum core. This layer of aluminum is sandwiched into PCB material that conducts heat from component leads to the edge of the board and then to the capturing enclosure.

Once heat is removed from PCB components, it needs to be dissipated from the computer enclosure to the ambient air. If the ambient air temperature is sufficiently lower than the computer enclosure temperature, computer-generated heat can be dissipated via convection and radiation.

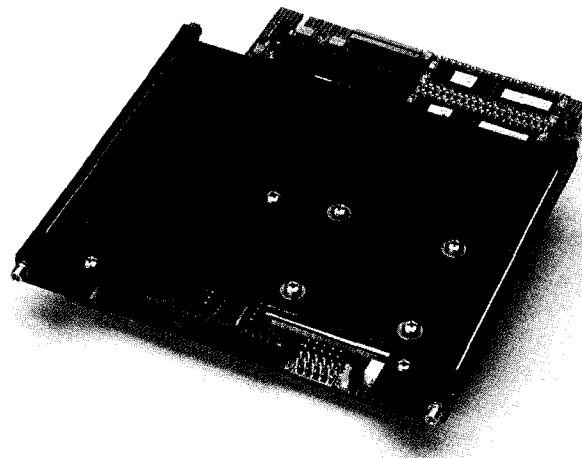
Enclosures depending on convection and radiation generally use cooling fins to increase thermal-transfer surface. These fins need to be oriented in the same direction as gravity so air can rise through the fins.

In higher-temperature ambient environments, supplemental active cooling may be required. This additional cooling can be in the form of liquid cooling (e.g., cold water loops) or thermoelectric cooling.

When looking for a rugged PC to operate in a high-ambient-temperature environment, the designer should determine the total heat load of the computer in the mounting location (see "Heat Dissipation" sidebar).

To deal with vibration and shock, increasing the harmonic frequency of rugged computer components, particularly PCBs, is essential. For example, a PCB mounted in a card cage with plastic card guides could have a vibration harmonic frequency below 20 Hz. When this same card is held in with a wedge-lok securing system, the harmonic typically increases to 150 Hz. The former card will likely experience component or structural failure before the latter card.

Photo 2 shows how Kinetic Computer's Card Module secures an off-the-shelf STD



32 card so it can operate in high-shock and -vibration environments (see Photo 3).

Elastomer shock mounts are the most common method for attenuating a high-shock and vibration environment to protect sensitive components. They are often used on the hard drives of rugged computers.

You can also specifically seek components and subsystems geared for vibration and shock (e.g., hard drives). Unfortunately, harddrive specifications are generally not useful in determining whether a HDD will survive a certain environment.

Instead, you must qualify several HDD vendors by simulating the characterized environment in a vibration test system. Mount the HDD as it would be in the final product. Impart vibration and shock in all three axes with and without the disk being read from and written to.

Power surges and transients are common in severe environment applications, particularly on vehicles. Large transients, caused when loads switch off, are the norm on heavy-duty vehicles. Known as load dumps, these transients can be protected against with tranzorbs and transient suppressors. They also protect from voltage surges.

PTC thermal fuses protect against voltage surges. Battery backup, especially of system RAM, provides graceful software shutdown in the event of power brown-outs and interruptions.

Protecting against ESD occurring at the enclosure or at I/O connectors is not difficult. Anyconductivecomponents, like internal PCBs, should have a 0.5" air gap to the enclosure.

All I/O connector pins can have ESD protection devices, which are essentially low-power, fast-acting transient-voltagesup pressors. These devices need to be located as close to the ESD entry point as possible. Using a ground plane layer in PCBs and diverting ESD to the ground plane via these devices is recommended.

Membrane keyboards and other input devices such as touchpads are especially vulnerable to ESD. Membrane keyboards should have a large border that is free of printed conductors. Membrane keyboards can also have a guard ring around its perimeter to divert ESD to the enclosure the keyboard is mounted into.

For EMI and EMC, most rugged computers must comply with FCC and CE Mark emissions and immunity (vehicle computers

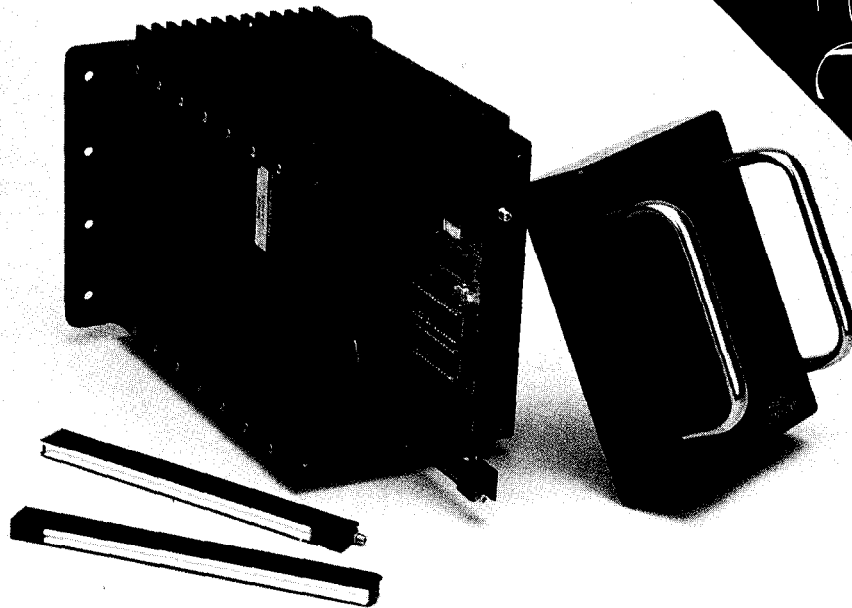


photo 2: Kinetic Computers' **STD** 32 Card Module allows an off-the-shelf **STD** 32 board to be **easily** captured in a **RCC-32 enclosure** while providing a vibration resistant friction mount using industry-standard **Wedge-Loks**.

are exempt from these regulations). Since most rugged PCs use off-the-shelf boards, tweaking board design and layout to meet EMI and EMC requirements is difficult. The

only method available therefore is to block emissions at the enclosure.

Fortunately, most rugged computer enclosures are constructed with metal, which

For Borland C/C++, Microsoft C/C++, Borland Pascal

RTKernel

Real-Time Multitasking for DOS

RTKernel is a professional, high-performance, real-time multitasking system for MS-DOS and Embedded Systems. It can use DOS device drivers and BIOS, and runs other DOS applications as a task - even Windows!

RTKernel is loaded with features: an unlimited number of tasks, excellent performance, a full set of inter-task communication functions (semaphores, mailboxes, synchronous message-passing), real and protected mode support, drivers for up to 38 COM ports and Novell's IPX services, and lots more...

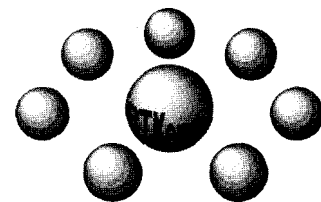
It's ROMable and very compact (about 16K code, 6K data), making it ideally suited for Embedded Systems.

RTKernel is well-documented and easy to use. All hardware drivers always come with source code; kernel source code available at extra charge. No run-time royalties.

Join thousands of satisfied customers!

In North America, please contact:
 On Time Marketing
 88 Christian Avenue Setauket, New York 11733 USA
 Phone (516) 689-6654 Fax (516) 689-1172
 BBS (516) 689-6285 FaxFacts (516) 689-6315 CIS 733 13,3177

From other countries, please contact:
 On Time Marketing
 Karolinenstrasse 32 · 20357 Hamburg, GERMANY
 Phone +49-40-437472 Fax +49-40-435 196
 CompuServe 100140,633



Use RTKernel for:

- ▶ process control
- data acquisition
- ▶ real-time simulations
- ↗ background processing

Libraries: \$495
 Source Code: add \$445

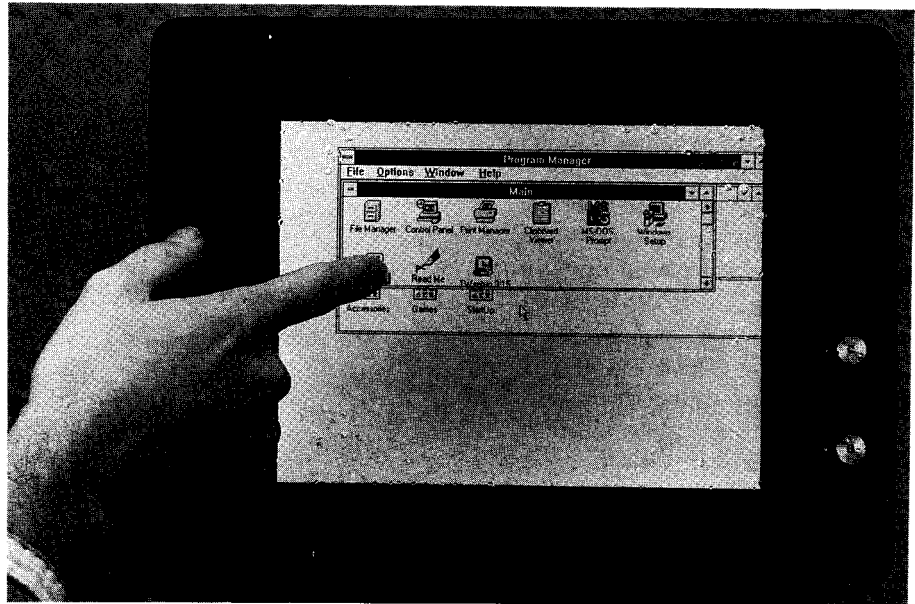


On Time
 MARKETING

Professional Programming Tools

Free demo disk! Request Info Kit!
 Internet: <http://www.on-time.com>
ftp@on-time.com
info@on-time.com

Photo 3: Kinetic Computer provides rugged LCD displays for vibration-prone environments such as those on **heavy-duty** vehicles. These displays are available as a dumb monitor driven by a rugged PC or as stand-alone rugged smart displays with PCs built into them.



is an excellent shield. You can also create additional shielding through using foil, metal plates, and EMI gaskets.

Proper design of power and ground cabling can dramatically reduce EMI. Filters on I/O and power-input lines are usually necessary to meet regulations. Plastic enclosures can be either impregnated or coated with metal to provide conduction.

If board designs can be altered, softening system clock edges reduces emissions at critical frequencies. Adding bypass capacitors, minimizing lead lengths, and keeping I/O lines away from clock signals further reduce EMI.

Ground and power plane layers in PCBs are **highly** recommended. Filtering of power and I/O signals by adding damp ing resistors, inductors, and ferrite beads also reduces EMI from a PCB.

The ideal way to protect electronic components and subsystems from condensation resulting from high humidity is to apply conformal coating on the components. There are many different types of conformal coatings. The designer should use a coating that doesn't harm the components being protected.

CONCLUSIONS

The Wintel PC has become synonymous with automation worldwide. The relentless increase in the price/performance ratio of desktop PCs is driving these computers into diverse applications.

PCs can be designed to operate in tough environments, but it requires appropriately designed electronics and packaging, design qualification testing, and proper manufacturing. Despite the pressure on rugged PCs to match the price and performance of desktop PCs, suppliers of rugged PCs always suffers from lower production volumes and additional packaging costs. Customers have to be convinced that the price difference is worth it.

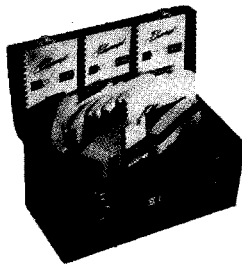
There is sufficient experience that desktop PCs and laptop PCs do not survive in rough environments. As this experience becomes widespread, customers will truly begin to understand the value of ruggedized PCs. EPC

Vinit Nijhawan is president and founder of Kinetic Computer Corp. in Cambridge, MA. Kinetic Computer is a leading *manufacturer of rugged* PCs for vehicles, outdoors, and *industrial plants*. Vinit was previous/y a principal of Payload Systems, a space hardware manufacturer. He may be reached at vinit@kin.com.

TIRED OF SHOPPING AROUND?

Phoenix Technologies Ltd., the over 10 years, has your complete software solution for the

Embedded BIOS Extensions enable unique functions required in the versatile environments of Special Purpose PCs.



BIOS software for the widest variety of core logic and CPUs.

Phoenix PicoCard™ is the only PCMCIA solution that enables full compatibility with hundreds of PC cards in resource-limited systems.

phoenix Power Management software is the industry's first and best with system and device power control for extended battery life and system stability.

The PhoenixPICO™ OEM Adaptation Kit includes all of the PhoenixPICO™ software in a kit that reduces development times, is easy to use and **cost** effective!



Call **1-800-677-7305** or 1-408-654-9000 to contact Phoenix or a Distribution Partner or Certified Developer.

P/CO Innovative system-enabling software solutions for Special Purpose PCs. <http://www.ptltd.com/pico>

IRS

4 19 Very Useful
420 Moderately Useful
42 1 Not Useful

Rick Lehrbaum

The soft Side of PC/104

Rick takes a step away from hardware to zero in on what's available on the software side of PC/104. After a brief overview on embedded DOS applications, he focuses on real-time software, Windows, and the BIOS.

“PC/104 software?” you ask.
Why not?

I've talked before about how PC/104 modules represent Lego-like microcomputer-system building blocks. It's similar to the object-oriented approaches used in structuring software using C or C++.

When you use PC/104 hardware along with object-oriented software, your whole system—hardware and software—becomes a collection of object-oriented building blocks. Looked at this way, it's easy to see why software is every bit as important as hardware in a PC/104-based system.

Take a look at your last embedded-PC solution. Did you decide to use an embedded PC because of the PC's raw computational horsepower? For its sophisticated hardware architecture? Probably not!

However, what the PC lacks in hardware elegance, it more than makes up for in software. Think about the number of operating systems, drivers, function libraries, and development tools. Poll the growing number of designers using embedded

PCs. You'll quickly see it's not PC hardware but software that drives the trend.

This is especially true, since software represents the main cost and risk in developing embedded systems.

SOFTWARE ARCHITECTURE

In theory, applications should follow the “well-behaved” software model of Figure 1. However, the modest performance (I'm being kind!) of PC hardware and MS-DOS has prompted programmers to circumvent MS-DOS. Instead, they interface directly with the BIOS or device drivers (see Figure 2) or, in performance-sensitive situations (such as screen updates), directly control the PC's hardware (see Figure 3).

From these approaches, you can see there are lots of ways to skin the embedded-PC software cat!

CHOICES, CHOICES, CHOICES

In deciding how to structure the software side of your PC/104 application, there are lots of options available:

- Standalone applications—perhaps you don't need an operating system at all! Did you know a standard PC-compatible BIOS lets you run an application directly out of an EPROM without DOS, device drivers, or anything else?
- Normal DOS-based applications—this, of course, is the most obvious and straightforward approach. Write your application as a DOS EXE file and include a bootable DOS drive in the target system.
- Real-time applications—if your application demands multitasking or high-performance software features, you may need real-time support software. Depending on your system's needs, there are several suitable approaches.

Now, for a more detailed look...

KEEP IT SIMPLE

Like a lot of PC/104-based system designers, you've probably worked with single-chip microcontrollers in the past. There, the standard approach is to burn

code into an EPROM, after assembling or compiling it into separate code and data address spaces. If the application is relatively simple, you may feel inclined to do the same thing with your PC/104 project.

It so happens that the PC architecture has a handy mechanism called a **BIOS** extension that lets you execute code within an EPROM on system startup. The extension is a standard feature of the PC BIOS. In fact, your PC's video BIOS software is loaded from a BIOS extension EPROM located on your VGA display controller card. Here's how it works.

After system initialization and self test, but before booting the operating system from a floppy or hard drive, the BIOS scans upper system memory on 2-KB boundaries, looking for specially formatted blocks of code to execute. These blocks must start with the data pattern **55AAh**.

If this pattern is found, the BIOS reads a length byte and performs a checksum on the remainder of the block. If the checksum is correct (it must equal 0), the BIOS performs a **FAR CALL** to the fourth byte of the BIOS extension. In this manner, you can run a simple embedded-PC application out of an appropriately formatted EPROM.

However, you may want to think about the limitations of this approach before using it. Specifically, since BIOS extensions run before DOS boots, you can't take advantage of DOS services or DOS-loaded **TSRs** and **drivers** (download **BIOS** services).

Also, don't count on using your favorite compiler since many compilers require a DOS run-time environment to execute their compiled code. Listing 1 provides an example of a BIOS extension that simply outputs a message to the screen and returns control to the BIOS.

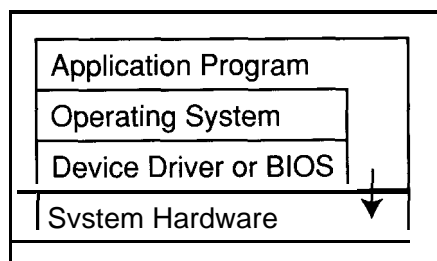


Figure 3: By bypassing the **operating** system, **BIOS**, and device drivers, the user gets to directly control system hardware, but loses the advantages of an object-oriented approach.

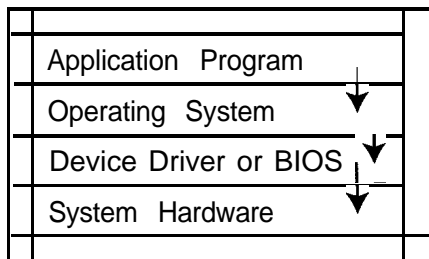


Figure 1: A well-behaved software is characterized by a layered software architecture.

For more information on how to create a BIOS extension, check **Ampro's** application note **AAN-8702**.

DOS-BASED APPLICATIONS

If you're like most embedded-PC designers, you probably write your application in C, compile it into an EXE file, and run it under DOS. Plain old DOS.

But, why use DOS?

Even if your application has no need for any operating system services whatsoever, DOS may offer some important benefits:

- DOS lets you use your favorite **desktop**-PC compiler and debug tools by providing the required run-time environment. As a result, you can program an embedded application without any special embedded tricks (or compilers)!
- With DOS in your embedded system, the target and development environments are exactly the same. This parallelism simplifies the task of porting your application from the development computer to the embedded target computer. Life becomes easier in a lot of ways.
- Even if your embedded application doesn't require DOS service, DOS or DOS-based utilities ease system installation and maintenance. For example, an MS-DOS Interlink utility transfers data to and from the system.

On the other hand, it's not unusual for a PC/104-based embedded application to log data on or read parameter files from a PC-compatible floppy.

For example, imagine an intelligent PC/104-based paint-mixing machine that reads color matching parameters from a floppy. It could easily be programmed to support an endless variety of paint brands or customer-defined requirements.

Another example might be a PC/104-based environmental monitoring instrument

which logs its data onto a hard drive for subsequent analysis.

Where do you get DOS licenses for a PC/104-based embedded system?

It depends on what kind of DOS you need and how many copies you require. For real MS-DOS, buy a few copies from the local computer store or a few thousand from Microsoft. For needs somewhere in between, try Annabooks. They've been designated by Microsoft as an MS-DOS distributor for the PC/104 market.

There are also a couple of PC/104-oriented DOS alternatives. Datalight offers ROM-DOS and General Software has Embedded DOS. These products offer some interesting enhancements peculiar to embedded-system requirements. Embedded DOS includes normal DOS functions as well as real-time, multitasking capabilities.

GET REAL!

"But, real-world applications can't run on a PC," you say?

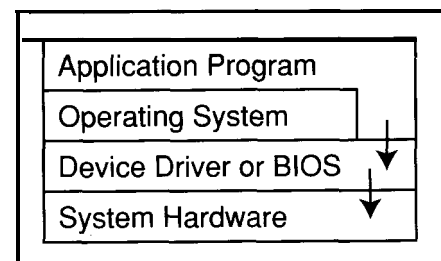


Figure 2: Some application programs **bypass** the operating system, communicating directly with the drivers or even the hardware.

What? Are you worried that DOS is too "soft" for your rugged, perhaps mission-critical application? What should you do?

For sure, there are many embedded applications that must continue running around the clock without operator intervention. Perhaps, your embedded PC is the heart of an automated toll booth.

What if it stops running? Everyone rides free until the technician arrives to reboot it! This is nice for the commuter, but a disaster for the Transit Authority. And for you, it might mean you lose the next contract!

No doubt, you've seen this message on your desktop PC: "Press F1 to continue.." What happens when DOS outputs this message to a PC/104-based system without a screen or F1 key? Someone has to press reset, that's what!

Don't despair! Turning a PC/104-based system into a real-time machine is the sole

purpose of a sizable group of real-time software companies. And, the result is a wide variety of excellent alternatives.

I like to divide these real-time software products into three categories: libraries, executives, and operating systems.

REAL-TIME LIBRARIES

A minimalist approach to adding real-time support to a PC/104 application is through the use of a real-time, multitasking function library. There are several potential benefits to this approach:

- you don't need to deal with the complexities of a more full-featured real-time executive or operating system
- function libraries often include full source code, which may prove invaluable when it's time to debug or tune the application or functions included in the library
- you usually aren't required to pay per-copy royalties
- initial out-of-pocket expenses are likely to be low

On the other hand, the functions provided by the library are usually limited to basic things like task management, task switching, queue management, and event synchronization. You won't get higher-order services such as memory management, console I/O, serial communications, networking, or a file system.

In short, if you need basic multitasking support for a real-time application and are prepared to do the hardware management work yourself, this approach may be reasonable. Two typical real-time function libraries are Interwork from Block Island Technologies and DIVVY from the Drumlin division of Micro/sys.

Table 1 shows what is offered by a typical real-time library. As you can see, the library's functions are quite primitive in comparison to what you'd expect from DOS or Windows! Listing 2 shows how you might use these services to manage a multitasking application.

REAL-TIME EXECUTIVES

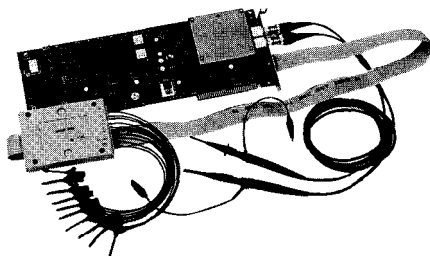
Next in order of complexity are the real-time executives. These packages provide more system services than the multitasking real-time libraries.

Despite additional vigor, real-time executives are usually "lean and mean." They use system resources—hardware and soft-

PC-Based Instruments

200 MSa/s DIGITAL OSCILLOSCOPE

**HUGE BUFFER
FAST SAMPLING
SCOPE AND LOGIC ANALYZER
C LIBRARY W/SOURCE AVAILABLE
POWERFUL FRONT PANEL SOFTWARE**



\$1799 - DSO-28204 (4K)
\$2285 - DSO-28264 (64K)

DSO Channels

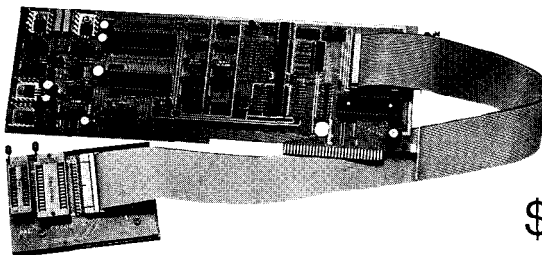
2 Ch. up to 100 MSa/s
or
1 Ch. at 200 MSa/s
4K or 64K Samples/Ch
Cross Trigger with L A
125 MHz Bandwidth

Logic Analyzer Channels

8 Ch. up to 100 MHz
4K or 64K Samples/Ch
Cross Trigger with DSO

Universal Device Programmer

PAL
GAL
EPROM
EEPROM
FLASH
MICRO
PIC
etc..

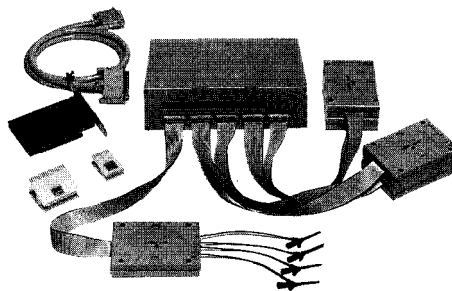


\$475

Free software updates on BBS
Powerful menu driven software

400 MHz Logic Analyzer

- up to 128 Channels
- up to 400 MHz
- up to 16K Samples/Channel
- Variable Threshold Levels
- 8 External Clocks
- 16 Level Triggering
- Pattern Generator Option



\$799 - LA121 00 (100 MHz, 24 Ch)
\$1299 - LA32200 (200 MHz, 32 Ch)
\$1899 - LA32400 (400 MHz, 32 Ch)
\$2750 - LA64400 (400 MHz, 64 Ch)

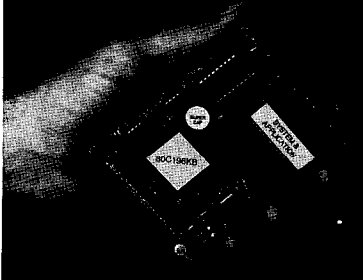
Call (201) 808-8990



Link Instruments

369 Passaic Ave, Suite 100, Fairfield, NJ 07004 fax: 808-8786

SBC196



- Low Power
- 8x10b A/D
- I²C/Wire
- R-time Clk
- Small, 3"x4"
- 2 Ser Chan
- LCD/Key Support
- Low Cost
- Forth-83196 w/80K ROM Apps
- EEPROM to 2K

vesta technology, inc.

(303) 422-8088 FAX: (303) 422-9800

#213

A NOTICE TO OUR READERS:

Circuit Cellar INK® will occasionally provide a listing of subscribers to vendors with offers of substantial interest to our readers. If you would prefer not to be part of this listing, send the mailing label from the front cover of Circuit Cellar INK along with your request to:

**Circuit Cellar INK
Subscriber Service
Department
P.O. Box 698
Holmes, PA 19043-9613**

listing 1: Here's an assembly language program structured as a **BIOS** extension.

```

; This initialization code is for a tiny model with code,
; data, and stack in the same segment.
BIOS-END equ 00060h ; end of RAM used by BIOS

DGROUP GROUP PROG
PROG SEGMENT BYTE PUBLIC 'PROG'
ASSUME CS:PROG, DS:PROG

; Initialize run-time code. place stack in RAM all else in ROM
Init PROC FAR
db 055h,0aah ID pattern
Blocks db 1 ; # of 512 blocks code and data

Entry
int 012h ; Get number of Kbytes in AX
mov cl,6 ; Divide by 64
shl ax,cl ; to get paragraph address
sub ax,01000h ; 64 Kbytes for stack
mov bx,ss ; Get caller's FAR return
mov cx,sp ; address
mov ss,ax ; Now set up
mov sp,0ffffh ; the real stack
push cx ; Save the caller's return
push bx ; address on my stack

call MainCode ; execute your program

; if program exits (via a FAR return). control returns to the BIOS
pop bx ; Get the BIOS
pop cx ; stack back
mov ss,bx ; Restore
mov sp,cx ; them
ret ; and ret to BIOS and boot
Init ENDP

; MainCode is a simple sample program It signs on and returns.
MainCode PROC NEAR
lea si,Message ; Point to the message
MainCodeLoop:
cld ; Ensure forward direction
lodsb ; Get the next character
or al,al ; Is it the end?
jz MainCodeExit ; Yes, exit
mov ah,14 ; No, get TTY output parm
mov bx,7 ; Screen attribute
int 010h ; Output the character
jmp SHORT MainCodeLoop ; and loop for more
MainCodeExit:
ret ; Return to BIOS

Message db 'Main Code',13,10,0; Signon message
MainCode ENDP
PROG ENDS
END

```

ware--thrifty, and are performance-oriented. Also, most real-time executives are offered in a modular format. You only use (and pay for) the components you need.

Depending on your real-time executive, you can get just about any function you need. The core component, called a *kernel*, provides basic system and task management and control including resource initialization, task queuing and switching, memory management, and so forth.

Optional software components support functions like console I/O, serial communi-

cations, networking, disk read/write, and so on. Figure 4 illustrates the key services offered by a typical real-time executive.

Depending on the vendor, source for both the kernel and device drivers may be available, so you can troubleshoot sticky problems yourself. Also, real-time executives are commonly (though not always) sold on a one-time-payment basis, which eliminates the cost and overhead of paying royalties each time you build a system.

An interesting feature of most PC/104-oriented real-time executives is their ability

listing 2: A sample C application uses the DIVVY real-time functions presented in Table 1.

```
#include <mt.h>
#define onesecond 18L
#define quartersecond 4L
#define thirdsecond 6L

void task1(void)

    static unsigned int count;
    for(;;) {
        _settextposition(20,1);
        print("Task 1 has run %06d times",count++);
        mt_delay(quartersecond);
    }
}

void task2(void)

    static unsigned int count;
    for(;;) {
        _settextposition(21,1);
        print("Task 2 has run %06d times",count++);
        mt_delay(thirdsecond);
    }
}

void task3(void)

    char buffer[9];
    for(;;) {
        _settextposition(1,55);
        _strtime(buffer);
        printf("%s",buffer);
        mt_delay(onesecond);
    }
}

main()

    /* 3 tasks, no flags, standard tick rate */
    MT_setup(3,0,0);
    create_task(task1,"task1", 5, 2048, NULL);
    create_task(task2,"task2", 5, 2048, NULL);
    create_task(task3,"task3", 5, 2048, NULL);
    _clearscreen(_GCLEARSCREEN);
    mt_start();

    /* system is multitasking here until ^C */
    printf("Program exiting.\n");
}
```

to be used alongside DOS. Your system boots from DOS and then loads the executive and your application from an EXE file.

Real-time executives with this capability usually provide a function call so your application can use DOS and BIOS services such as file or console I/O. You can therefore use DOS device drivers and TSRs.

Assume that the appropriate DOS network drivers are present to convert the network drive into a local DOS resource. Your application-running under a real-time executive-can then access a remote disk drive across an Ethernet LAN using DOS file read/write services.

Keep in mind that DOS and BIOS are not reentrant, so only a single task can be

permitted to access BIOS or DOS functions at a time. To prevent problems, the real-time executive must provide a suitable "locking" mechanism.

Although this restriction may be unacceptable in some systems, quite a few PC/104 applications don't need shared access to peripheral resources such as a disk, display, network, or keyboard. They therefore benefit from this approach.

Another problem exists if your application needs to run in protected mode. To take advantage of BIOS or DOS services, a protected-mode application needs to switch in and out of real mode, which would result in unacceptable inefficiencies.

Kodak's AMX, U.S. Software's SuperTask!, and Microtec Research's VRTX are good examples of real-time executives.

REAL-TIME OPERATING SYSTEMS

At the high end of the scale is the full-featured real-time operating system (RTOS).

"What's the difference between a real-time executive and a real-time OS?" Since there's no universally accepted answer to this question, I'll give you my version.

An executive manages the computer's basic resources-CPU, memory, interrupts, timers, and DMA. An operating system does all this while managing the system's peripheral devices, which typically include disk drives, keyboard and display, serial communications, and networking.

In truth, a lot of RTOS offerings are modular, consisting of a real-time kernel and an assortment of device managers. As a result, it may be difficult to decide whether a particular package is a real-time executive or a real-time OS.

In a minimal configuration, it may fit the definition of an executive. Yet, when its peripheral device managers are included, it becomes a full-fledged RTOS.

In fact, it doesn't matter what you call it! It's a full-fledged RTOS if:

- it includes all the basic services of a real-time executive (described earlier)
- it includes file system, console I/O, communications, and networking
- once running, it completely eliminates the use of the normal PC BIOS
- all these functions support full, multi-threaded (multitasking) operation
- it can do all this in protected mode (on '386 CPUs and up)

Increasingly, PC-compatible RTOSs provide GUI support with built-in drivers for VGA display, a standard PC keyboard, and a standard mouse. Look at Figure 5 to get an idea of what you can easily accomplish using this kind of RTOS GUI.

Some popular PC/104-oriented products appearing to meet all of these requirements are QNX from QNX Software Systems, Lynx from Lynx Real Time Systems, OS-9000 from Microware, pSOS from Integrated Systems, VxWorks from Wind River Systems, and VENIX from Venturcom.

It wouldn't be right to leave the subject of RTOSs without talking about device drivers.

Assuming your RTOS and application runs in protected mode, device drivers for all hardware must be 32-bit, reentrant code. The RTOS can't use the PC-compatible BIOS since the BIOS doesn't meet this condition. So, the RTOS you use must replace the BIOS with its own version of the functions within the BIOS.

This is a blessing and a curse. You can expect improved speed and robustness. However, any non-standard hardware functions in your PC/I O&based system require protected-mode drivers for your particular RTOS and hardware.

An RTOS claiming to support PC/I 04 applications should provide a complete set of drivers for all standard PC-compatible controllers and peripherals. What you need to worry about is support for embedded-PC extensions such as watchdog timers and solid-state disks or interfaces like PCMCIA, SCSI, and LAN adapters since these haven't attained chip-level standardization with desktop PCs.

Don't be surprised to find yourself haggling with both your RTOS and hardware suppliers over who should supply the drivers! You may end up creating some yourself.

In some cases, you can circumvent the RTOS driver problem by using a real-time executive, instead of an RTOS, along with DOS. However, this only works in applications that don't run in protected mode or need multithreaded access to the resources in question.

While many PC/I 04 applications meet these constraints, others do not. Beware!

DO YOU DO WINDOWS?

Despite all this talk about the exotic world of real-time libraries, executives, and operating systems, you may be wondering about Windows. It has the ability to manage system memory and protected mode, run multiple tasks, support graphics functions, and simplify user interfaces.

So, why not use Windows (or Windows 95) in a PC/I 04-based embedded application?

You're certainly not alone in this thought. There are quite a few PC/I04 applications, both completed and under development, that use Windows!

But, you may object that you can't implement a real-time system using Windows.

However, this depends on your definition of "real time." Or, to be more practical, it depends on your real-world application's performance requirements.

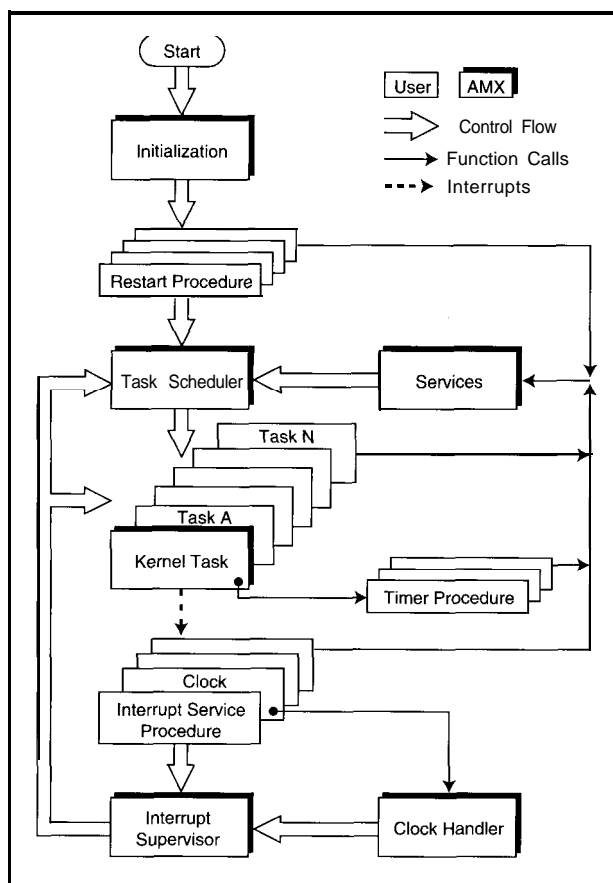


Figure 4: The functions offered by AMX are representative of typical real-time executives.

Officially, a real-time system is one that behaves deterministically. But, I prefer to think of it as any system that needs to interact with external real-world stimuli during its operation.

With my definition, a real-time system provide enough performance to successfully perform its defined task. Some applications require millisecond precision while others respond to a single event per hour.

Under this interpretation, even Windows can support some real-time requirements! (Please don't be offended, anyone!)

What's an example of a Windows-based PC/I 04 embedded system?

Consider an information kiosk at an airport. It offers information about shopping, local entertainment, and community services.

A '486-based PC/I04 CPU module runs Windows 95. A touchscreen emulates a standard serial mouse and is connected to COM1 while the display is a VGA-compatible color LCD. A SoundBlaster-compatible PC/I 04 module adds speech and musical audio output, and the data-

base is stored on a hard drive. A CD-ROM provides animation and entertainment, and a small thermal printer on LPT1 generates hardcopy of selected information and sale coupons. The database is periodically updated from a central office over a modem connected to COM2.

You could easily run this entire application within Windows or Windows 95. If you found the system too slow at accessing the hard disk, reading the CD-ROM, or updating the display, you'd simply do what you would with your desktop PC—upgrade the CPU!

For this application, a '386SX or '486SLC is probably too slow, but a '486DX-100 would be fine. Fortunately, these are now available in PC/I04 modules. Next year, you'll be able to upgrade to a Pentium-based PC/I 04 module!

My point—if millisecond response times aren't needed, you may be pleasantly surprised with the capabilities of the world's most popular multitasking operating system (yes, that's Windows!). Don't forget to provide an externally accessible reset button or a watchdog-timer.

But if your application demands quick response to external stimuli, efficient use of system resources, or high confidence in uninterrupted system operation, you'd be better off with a full-fledged RTOS.

DON'T FORGET THE BIOS

With all this discussion of application and operating system software, don't overlook a very intimate part of every PC/I 04 system: the CPU module's BIOS. After all, it's responsible for some important stuff:

- initializing the CPU, memory, and system/peripheral controllers

<p>Startup and Exit mt_setup / MTsetup% mt_start / MTstart% mt_stop / MTstop</p> <p>Dispatcher and Timing yield / MTyield% mt_delay / MTdelay%</p> <p>Task Control create-task / MTctask% kill / MTkill% die / MTdie% sleep / MTsleep% wake / MTwake% suspend / MTsuspend%</p> <p>Priority and Tags get_my_pty / MTgetmypri% get_your_pty / MTgetyourpri% chg_priority / MTchgpri%</p>	<p>Flags set-flag / MTsetflag% clr_flag / MTclrflag% set-flag-wait / MTsetflagwait% clr_flag_wait / MTclrflagwait% set_flag_int / _set_flag_int (assy) clr_flag_int / _clr_flag_int (assy)</p> <p>Queues mt_makequeue / MTmakeQ% mt_del_queue / MTdelQ% send_msg / MTsendmsg% rcv_msg / MTrcvmsg% send_msg_wait / MTsendmsgwait% rcv_msg_wait / MTrcvmsgwait% num_msgs / MTnummsg%</p> <p>Standard I/O tgetch / MTinkeywait\$ tgetche / — tungetch / —</p>	<p>— / MTsetinputcolor% tgets / MTlineinput tgets / — lptputc / — lptputs / MTlprint% mt_bEEP / MTbEEP mt_tone / MTsound mt_quiet / MTquiet</p> <p>Status mt_str_err / Mterrorstr\$ status / MTstatus% get_my_id / MTgetmyid% get-your-id / MTgetyourid% get-tag / MTgettag\$ mt_my_stkleft / MTmyskleft% task_stkleft / MTtaskstkleft%</p>
--	---	--

Table 1: The functions offered by a typical real-time library don't compare with what you can get in DOS or Windows, but can get a lot of simple jobs done.

- performing the system power-on-self-test
- loading BIOS extensions (if present)
- booting the operating system (if present)
- providing hardware interface support during system operation for keyboard, disk, serial, parallel, and timer functions

These are important things to get right! What if, on powerup, a system performing critical control or monitoring functions doesn't successfully boot its operating software? What if that system offers security surveillance or temperature or pressure monitoring? System failure could endanger lives or property. In an intelligent vending machine, down time costs its owners substantial revenue.

What if a hard disk drive spins up a bit too slowly after a power brownout? When the BIOS fails to boot the operating system from a not-quite-ready disk drive, a normal desktop PC hangs until someone presses F1 or reset. Such a failure mode is undesirable—more likely, unacceptable—in many (most?) embedded applications.

Your PC/104-based application may need to perform its task around the clock without failure. With its reliance on "Press F1 to Continue," the normal desktop-PC BIOS doesn't make it easy for you to sleep!

Fortunately, some PC/104 CPUs now include BIOS enhancements that address the special concerns of embedded applications. Here are some features you should look for in an embedded-PC BIOS:

- Battery-free startup
- Did you know your desktop PC won't power up and boot when the battery that

backs up the CMOS setup data (in the real-time clock chip) is dead? Eliminate this risk by keeping a redundant copy of the setup data in a tiny inexpensive EEPROM. In fact, you can eliminate the battery entirely (an actual requirement for certain applications), provided you don't need a battery-backed time-of-day clock function.

- Fail-safe boot
- What if the hard drive spins up slowly or there is a soft read error when the BIOS attempts to boot? An embedded-PC BIOS accommodates this condition by looping (indefinitely) in its attempt to boot from the designated boot device. It won't just give up and display, "Disk Read Error, Press F1 to Continue..."

- Solid-state disk support
- Conventional disk drives may simply be too "soft" for reliable operation in your PC/104-based embedded application. Your concerns may include shock and vibration, operating temperature, mechanical failure, and speed.

You may need a solid-state disk (SSD). SSDs can be nonvolatile RAM (NVRAM), EPROM, or flash memory. Whatever SSD you choose, the BIOS probably needs to support it.

Do you wanna be famous? Are you or your company using PC/104 technology in an interesting or unusual way? Tell us about it.

PC/104 DESIGN CONTEST

You are invited to submit unique PC/104 projects or applications to our design contest. Be sure to include functional block diagrams with descriptions of the hardware, software, and peripherals used. Contest entries will be judged for technical merit, applicability, and originality. The judges: Circuit Cellar INK's Steve Ciarcia, Ampro's Rick Lehrbaum, and Embedded PC's Managing Editor Janice Marinelli. We'll highlight winning applications in Embedded PC, plus designers will be invited to submit a design write-up for a future PC/104 Quarter. And there's more! Winners will receive **prizes**.

- 1st prize **Ampro CoreModule™/486 Development Kit**
- 2nd prize **Ampro CoreModule/386 Development Kit**
- 3rd prize **Ampro CoreModule/PC Development Kit**

All entries must be received no later than August 15, 1996. Winners will be announced at September's Embedded Systems Conference and the winning project descriptions will appear in December's issue of Circuit Cellar INK.

Contact us today for your entry form and then mail your contest entry to:
Janice Marinelli, PC/104 Quarter Contest
Circuit Cellar INK 4 Park Street Vernon, CT06066
Tel: (860) 8752199 Fax: (860) 872-2204
www: <http://www.circellar.com/> E-mail: PC104.contest@circellar.com



Co-sponsored by Ampro Computers, Inc., the originator of PC/104, and Circuit Cellar INK, home of Embedded PC

- Watchdog timer

Has your desktop PC ever crashed? (Dumb question, right?) You reach for the power switch or reset button. No great harm is done. You lose ten minutes of editing that you can usually recreate.

On the other hand, if your embedded PC crashes, no one may know anything is wrong-let alone be able to restart it.

These occasions mark the times a simple embedded-PC enhancement saves the day. It's inexpensive and easy to include a watchdog timer on a PC/104 CPU. But, for best results, the embedded-PC BIOS needs to support the watchdog timer function.

- Fast boot

Some embedded systems can't tolerate the lengthy self-test and startup delays typical of a desktop PC. The system may need to be up and running within a few seconds, not tens of seconds. Your embedded-PC BIOS should let you shorten initialization.

- Serial console

Your PC/104-based embedded system may not include normal desktop-PC user-interface hardware like a PC keyboard, VGA controller, or CRT monitor. Yet, there may be times when you need to see what's going on at the DOS (or application) prompt.

Your BIOS should offer you the option of routing the DOS console (keyboard and display) functions to a serial port.

- Serial loader

Another useful enhancement comes from a BIOS function that loads executable code through a serial port prior to system boot. This feature can load a test program or reformat an internal disk drive (or SSD) that lost data or requires an update.

- System customization hooks

Your embedded system probably includes specialized hardware for keypad input, data display, or real-time control. Often, such devices need to be initialized immediately on powerup (i.e., before the system loads the OS and application).

You might think you need BIOS source code to customize the BIOS. But, once you find out how much it costs, you'll change your mind. (You'd rather buy a BMW!)

Fortunately, some PC/104 CPUs include a BIOS enhancement that lets you

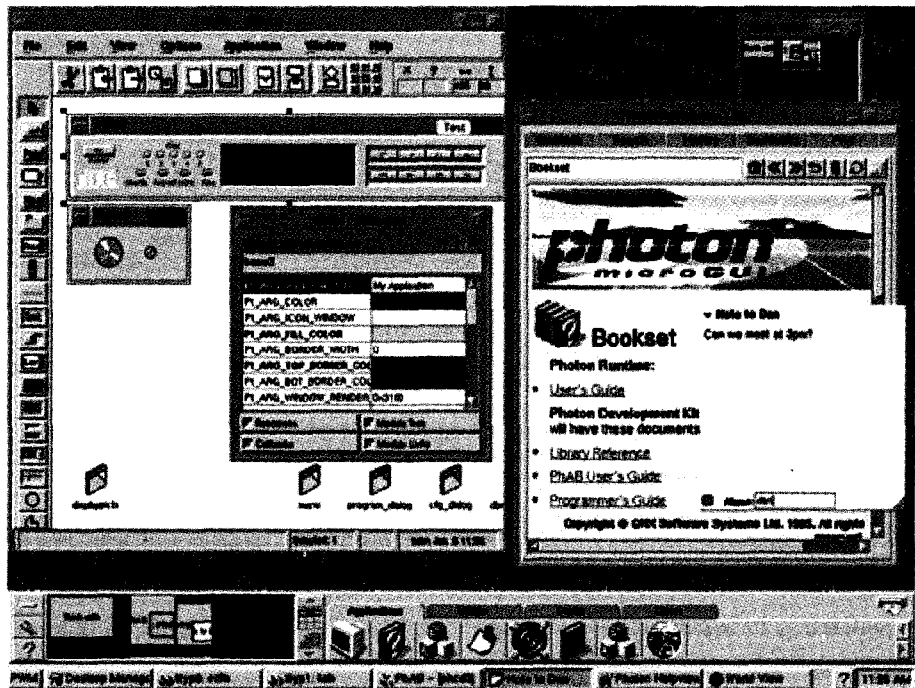


Figure 5: A typical user interface implemented using services provided by photon, the QNX RTOS GUI, includes elements familiar to most GUI users.

patch in custom initialization routines. You don't need to buy and modify BIOS source.

NOW, FOR MY FAVORITE APP

A couple of years ago, following one of my PC/104 seminars, a member of the audience described what has become my all-time favorite embedded PC/104 software story.

His project for UC Santa Barbara was a PC/104-based atmospheric datacollection system carried aloft by a weather balloon. The data was collected and transmitted by two-way radio to a ground station. He also used the two-way radio to modify the application program from the ground while the balloon was aloft.

Initially, he ran into a problem. The radio transmission data rate was too slow to upload (literally!) the entire application program. So instead, he used an embedded PC/104 CPU's special features.

He put the application source code, DOS editor, and C compiler directly on the system's disk drive. He enabled the serial console option of the PC/104 CPU, which routed the DOS console (keyboard and display) functions through the radio modem (via the COM1 serial port).

This way, when he modified the application, he could perform the same steps as on a desktop PC: exit the program, start the editor, edit the source code, compile the application, and run the recompiled code.

I call this "Compiling on the Fly." Some programmers sure do have guts. I'd definitely recommend the CPU's watchdog timer function in a system like this!

Think what happens if the recompiled program crashes! Do you think he tested the code on the ground first before performing the edit, compile, and execute process in the air? I bet he did!

YOUR FAVORITE APP?

I'd like to find out what your favorite application is. Check the advertisement on page 69. No doubt, you've come up with a splendid PC/104 application that we all need to hear about. Write it up and send it in. Not only might you win one of three prizes, but you just may end up being one of PC/104 Quarter's greats.

So, what is it? Do ya' wanna be famous? PCQ.EPC

Special thanks to Drumlin for Table 1 and to Kadak Products and QNX for the Figures 4 and 5, respectively. Your support is greatly appreciated.

Rick Lehrbaum cofounded Ampro Computers where he served as vice president of engineering from 1983 to 1991. Now, in addition to his duties as Ampro's president, Rick chairs the PC/104 Consortium. He may be reached at lehrbaum@ampro.com.

CONTACTS
 Application Note "AAN-8702"
 Ampro Computers, Inc.
 990 Almanor Ave.
 Sunnyvale, CA 94086
 (408) 522-2 100
 Fax: (408) 720-1 305

Embedded MS-DOS
 Annabooks
 1 1838 Bernardo Plaza Ct., Ste. 102
 San Diego, CA 92 128.24 14
 (6 19) 673.0870
 Fax: (619) 673-1 432

DOS Alternatives
 ROM-DOS
 Dotalight
 307 N. Olympic Ave., Ste. 201
 Arlington, WA 98223
 (360) 4358086
 Fax: (360) 4350253

Embedded DOS
 General Software, Inc.
 320 108th Ave. NE, Ste. 400
 Bellevue, WA 98004
 (206) 454-5755
 Fax: (206) 454.5744

Real-Time Function Libraries
 Interwork
 Block Island Technologies
 15455 NW Greenbrier Pkwy., Ste. 2 10
 Beaverton, OR 97006
 (503) 690-718 1
 Fax: (503) 6457732

DIVVY
 Drumlin Division

Micro/sys
 3447 Ocean View Blvd.
 Glendale, CA 91208
 (818) 244.4600
 Fax: (8 18) 244-4246

Real-Time Executives
 AMX
 Kadak Products, Ltd.
 206-1 847 West Broadway Ave.
 Vancouver, BC
 Canada V6J 1Y5
 (604) 734.2796
 Fax: (604) 734-8 1 14

SuperTask!
 U S Software
 142 15 NW Science Park Dr.
 Portland, OR 97229
 (503) 64 1-8446
 Fox: (503) 644.2413

VRTX
 Microtec Research
 2350 Mission College Blvd
 Santa Clara, CA 95054
 (408) 980-1 300
 Fax: (408) 982.8266

Real-Time Operating Systems
 QNX
 QNX Software Systems, Ltd.
 175 Terrence Mathews Cres
 Kanata, ON
 Canada K2M 1 W8
 (613) 591-0931
 Fax: (613) 591.3579

Lynx
 Lynx Real Time Systems

16780 Lark Ave.
 Los Gatos, CA 95030
 (408) 354-7770
 Fax: (408) 354-7085

OS-9000
 Microware
 1900 NW 114 St.
 Des Moines, IA 50325
 (515) 224-1929
 Fax: (5 15) 224-1 352

psos
 Integrated Systems Inc.
 3260 Jay St.
 Santa Clara, CA 95054
 (408) 980-1 500
 Fax: (408) 980.0400

VxWorks
 Wind River Systems
 1010 Atlantic Ave.
 Alameda, CA 94501
 (5 10) 748-4 100
 Fax: (510) 814.2010

VENIX
 VenturCom, Inc.
 2 15 First St.
 Cambridge, MA 02 142
 (617) 661-1230
 Fax: (617) 577-1 607

IRS

422 Very Useful

423 Moderately Useful

424 Not Useful

RECHARGEABLE BATTERY 12 VOLTS 9 A/H

Here's a great deal on new 12 volt 9 amp/hour rechargeable lead acid batteries. These batteries were recently manufactured and prepped for use in a product which hasn't yet come to market. The OEM didn't want to hold them in stock, so we got them at a greatly reduced price. They are 5.9" x 2.55" x 4.42" high and have a strip of foam padding stuck to one side. They have standard 1/4" quick-connect/ solder terminals. Wt: 8 lbs.



CAT # GC-129 \$28⁵⁰ each 8 for \$208.00

PHOTO RESISTOR

Silonex # NSL 4582
 Cadmium Sulfide photoconductive cell.
 22 ohms in bright light. 2 meq ohms dark.
 Plastic encapsulated.
 0.29" x 0.26" x 0.07".
 1.5" long wire leads.



CAT # PRE-12 75¢

10 for \$6.00 • 100 for \$50.00

SUB-MINIATURE SNAP ACTION SWITCH

CHERRY DG1C-BJ07 S.P.D.T.
 Body size: 0.51" long X 0.255" high
 X 0.255" wide. Lever with ridged end
 simulates roller action. Rated 3
 amps @ 125 Vac, 2 amps @ 30 Vdc.
 PC pins on 0.2" centers.
 Large quantity available.



CAT# SMS-133

2 for \$1⁰⁰ 100 for \$45.00
 1000 for \$350.00

PHOTOVOLTAIC DETECTOR

Miniature silicon solar cell converts light impulses directly into electrical charges which can easily be amplified, using a transistor for example, to activate a control mechanism. Unlike a conventional photo diode or transistor, it generates its own power and does not require any external bias. Generates 0.4 vdc in moderate light. Silicon cell is mounted on a 0.31" x 0.23" x 0.07" thick plastic carrier and has pc leads on 0.2" centers.



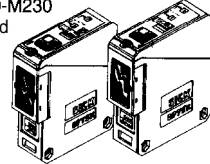
Large quantity available.
 CAT # PVD-2

75¢ 10 for \$6.50 • 100 for \$50.00

Visit Our World Wide Web Site...
<http://www.allcorp.com/>

RANGE 20 METERS (65.6 FEET) INDOOR/OUTDOOR PHOTOELECTRIC SWITCH

SICK Optex # WS/WE260-M230
 Invisible infrared beam and switch is useful for a variety of security, safety and sensing applications such as perimeter alarms, electric gate or garage door controls or conveyor line sensing. Modulated beam covers a 20 meter distance even with poor visibility. IP-66 enclosure rating makes it dust proof and splash resistant, suitable for most outdoor conditions. Light/dark switching selector. Sensitivity adjustment control. LED power and sense indicators. Two position cable entry gland (90 deg. apart) accepts 1/2" threaded conduit and a variety of wire sizes. Glass fiber-reinforced plastic housings are 3.05" x 2.5" x 1". Supply voltage: 22-240 Vac. Solid State Output: 0.5 amp (must have load to operate). Includes mounting brackets and hardware UL and CSA listed.



CAT # PES4 \$40⁰⁰ each

ALL ELECTRONICS CORPORATION

ALL ELECTRONICS CORP. • P.O. Box 567 • Van Nuys, CA 91408-0567
 FAX (818)781-2653 E-Mail - allcorp@allcorp.com

NO MINIMUM ORDER. Shipping and handling for the 48 continental U.S.A. \$5.00 per order. All orders including AK, HI, PR or Canada must pay full shipping. All orders delivered in CALIFORNIA must include local state sales tax. Quantities Limited. NO COD. Prices subject to change without notice.

ORDER TOLL FREE **1-800-826-5432**
 Visa, Mastercard, American Express or Discover accepted

CALL, WRITE, FAX or E-MAIL for a FREE 64 Page CATALOG
 Outside the U.S.A. send \$2.00 postage.

Packaging Embedded PCs

Far too often, packaging is left to the end of the design phase. However, it can make the difference between a design being a kludge and a reliable product. Russ categorizes design types, suggesting which type is best to use when.

In the past few columns, I've looked at the attributes which make embedded PCs popular, investigated the various buses and boards available, and explored some of the intricacies of driving clusters of small displays in embedded applications.

It's time now to take a look at the various options available for physically packaging embedded systems.

Generally, the end use dictates how an embedded system must be packaged. The options fall into four main categories:

- open-frame construction
- rack-mount systems
- PC embedded within the display
- portable and hand-held units

I'll look at each of these approaches in detail, checking out the specific hardware components available to the system designer in each category. Since PC/104 packaging was covered in "PC/104 Quarter" (INK 67), I'll avoid PC/104-based systems here.

Packaging often takes a back seat to the more glamorous aspects of implementing an embedded system, such as system design, CPU and interface selection, and programming.

Yet, packaging is critical to the implementation of a robust and efficient design. It requires that you take into account the overall degree of integration, shock and vibration, cooling, and powering aspects.

Packaging can make the difference between creating an unreliable kludge that never seems to get completed or work right and a reliable, serviceable product.

OPEN-FRAME CONSTRUCTION

Anyone following Circuit Cellar's HCS applications is familiar with Steve Ciarcia's propensity for screwing modules and system components to large wooden boards and hanging them on the wall.

This structure represents the extreme in "open-frame" construction! Few embedded system applications lend themselves to such a state of "open" construction.

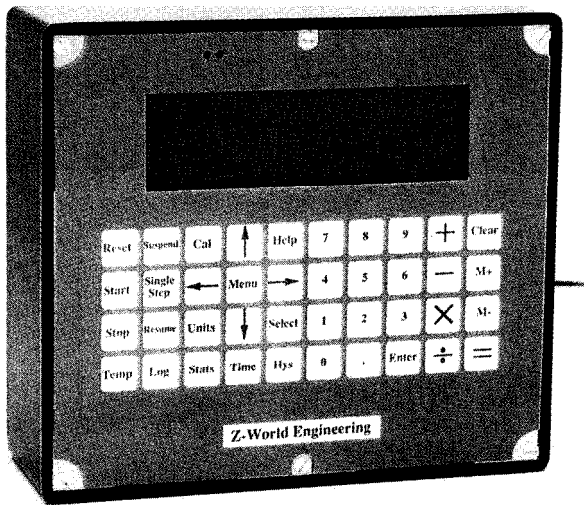
Nevertheless, the benefits of this approach are many:

- rapid and flexible configuration
- ease of servicing
- simple modification and modular expansion
- adaptability to various form factors
- robustness and good cooling in benign environments

When using this approach, you just need to select the system components and then conveniently arrange, fasten, and interconnect them. With modifications, this approach can be used in other applications.

I've mentioned before an ATM I designed for bank lobbies. In this case, the system components were certainly not bolted to a piece of wood hanging on a wall. But, the major components (CRT display, keypad, ticket printer, deposit envelope chute, gating solenoid, and collection bin) were fastened to various custom brackets fastened to the enclosure.

photo 1: **Modular LCD/key-boardpack**ayer like the one pictured here from Z-World Engineering **offer** a quick and inexpensive operator interface in some low-end embedded PC applications. Purchasing an off-the-shelf unit can save valuable engineering time and effort.



The rest of the electronics—a floppy disk drive for booting the program and saving transaction data—were bolted open-frame fashion to a large metal shelf within the ATM housing. A wiring harness with numerous disconnect points provided electrical interconnect between the major components and the electronics shelf.

In this manner, any element of the system could be easily removed for replacement or servicing. While the approach may seem unorthodox, it was perhaps one of the most cost-effective, yet serviceable, systems I've ever seen.

When using this open-frame approach, you need to give careful consideration to mounting stiffness. Warping, twisting, flexing, or bending of electronic components can wreak havoc with circuit-board reliability and lead to early failure.

Often, getting the right rigidity requires design and fabrication of custom brackets to handle each subassembly. While thicker plywood works for prototypes and home use, sheet-metal panels are normally used. They should be reinforced with cross-braces to resist warping.

When a number of circuit boards are needed to implement the embedded PC element of the design, a rack of STD, VME, or MicroPC boards can serve as one of the subsystems mounted to the electronics shelf.

In many systems, interconnection with real-world AC and DC signals is required. When discrete digital I/O is involved, connection can be achieved through Opto-22 assemblies (named after their originator, but available from many other sources).

These racks, which are most likely familiar to many of you, house a wide variety of

AC or DC input/output modules for interfacing logic levels to real-world AC and DC voltages and currents. The optoisolated interfaces provide excellent isolation from transients caused by starting motors and inductive kickback from relays for the computer and other low-voltage logic signals.

A similar panel supports signal-conditioning equipment for real-world low-level sensors such as thermocouples, optical sensors, pressure transducers, and so on. Similar in appearance to the Opto-22 carrier boards, these units accept industry-standard "5B" analog and digital I/O signal-conditioning modules.

The output of these modules then connects to A/D and D/A converters within the embedded PC. Modules are available for connecting low-level thermocouples (complete with cold junction compensation) and a host of other data-acquisition functions.

For user interaction in embedded systems, a conventional CRT monitor and PC keyboard are often unsuitable. However, small modular keypad and display subsystems, like that shown in Photo 1, may be adapted quickly to meet custom needs.

The output of these modules then connects to A/D and D/A converters within the embedded PC. Modules are available for connecting low-level thermocouples (complete with cold junction compensation) and a host of other data-acquisition functions.

For user interaction in embedded systems, a conventional CRT monitor and PC keyboard are often unsuitable. However, small modular keypad and display subsystems, like that shown in Photo 1, may be adapted quickly to meet custom needs.

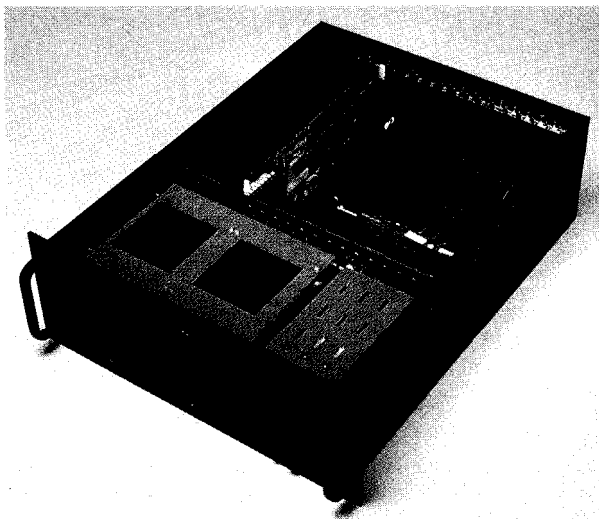


Photo 2: Interlogic Industries' **RK720** industrial rack-mount computer supports ISA cards with a **20**-slot passive backplane, room for 3.5" drives, and single or dual-redundant power supplies **running from either 48 VDC or 110 VAC**.

You might also consider the small embedded displays I discussed last time (INK 67).

RACK-MOUNT SYSTEMS AND CARD CAGES

The most classical housing for industrial and laboratory embedded systems is the conventional card cage which mounts in relay racks. The cages are best suited to situations where other large instruments are needed to complete the overall system.

The components generally are bulky and heavy, requiring a sturdy structure to hold them all together. Relay racks have been used for this purpose for as long as there have been electronic instruments.

Relay racks come in a variety of sizes and styles. **They** are usually sized to accommodate devices 19" wide, though some handle 24" units. The racks range from 1' to over 7' in height and may be bolted into bays for very large systems.

Stylish units are available which mount under a desktop work surface. With depths of 30" or more, it is often possible to mount equipment from both the front and rear, which essentially doubles capacity.

When computers first arrived, it was natural to mount them with the rest of the system components. However, to accommodate the many circuit cards that make up the computer and I/O electronics subsystem, card cages are now used.

These cages are available for all common embedded PC bus standards: STD, VME, ISA, MicroPC, and those typically employing a passive backplane. The cages usually include provision for mounting power supplies, floppy and hard drives, CD-ROMs, and other computer peripherals.

A typical STD bus card cage is featured in Photo 1 of my INK65 column. Optionally, complete indus-

trial PCs, like shown in Photo 2, use of conventional ISA and PCI cards in robust rack-mounted systems.

Getting signals in and out of the card cage are big concerns. Two approaches are normally used. Connections can be made to the backplane via transition connectors which terminate on custom panels mounted to the rear of the relay rack.

However, when it's more convenient to bring signals out the front of the rack, the ancillary access plates [see Photo 3] provide a neat, clean, simple, and reliable solution.

Powering the electronics in the card cage is also important. Separate rack-mounted power supplies can be used when the load is great, or you can use plug-in supplies that fit into the card cage itself.

The reliability of the power supply has become an important issue for embedded systems. It seems that the power supply is often the weak link keeping systems from running reliably without interruption.

Ziatech recently introduced an STD32 card cage sporting dual power supplies which keep the system running even if one supply should fail. The backplane is designed to hot swap, so the faulty supply can be replaced without powering down the system. This redundant power supply is also available for Interlogic's rack-mount unit shown in Photo 2.

Embedded systems frequently operate under extreme environmental conditions in vehicles (trucks, tractors, aircraft, and wa-

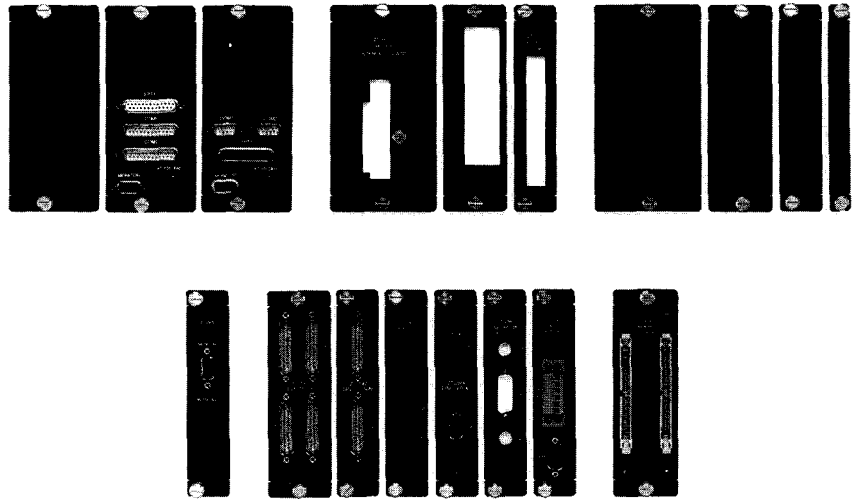


Photo 3: **Ziatech's** removable and interchangeable I/O access plates provide a convenient and robust means of bringing signals in and out of an embedded card cage PC.

tercraft) and industry. Kinetic Computer has several rugged solutions.

Their NEMA 4/12 and MIL-STD-8 10, -975, -1540, -461-rated card cage (Model RCC32 shown in Photo 4) tolerates up to 20-G shock and 12-G random vibration as well as water spray, dust, and other contaminants. This ruggedness is achieved through special backplane connectors, the lock of the card and cage, special fins which remove up to 100 W of heat from the sealed fanless enclosure, and gasketed front panel for sealed MS connectors.

EMBEDDING PCs IN DISPLAYS

In embedded PC applications where human interaction or data presentation is paramount, the display is the most obvious and important feature. With reduction in

the size of computer electronics, you can embed the entire PC within the display subsystem itself. The number and variety of such offerings are staggering.

If this approach suits your needs, you should search carefully for the most highly integrated solution possible. After all, there's not much sense in having all but 10% of your system in one neat enclosure, when you then have to go to extremes to find a home for the remainder of the components.

There's also little sense in attempting to create a package yourself. Subtle considerations like cabling, noise pickup, compatibility of flat-panel displays and touch screens, and integration of operating systems and support software can quickly destroy any savings.

At the low end, nearly every operator-control-panel vendor offers a simple, open-frame display pack, like the one shown in Photo 5. As you can see, it gets the display portion of your design over with in a hurry.

Some of these units contain nothing but a monochrome or color flat-panel display and an optional touch panel. Others include electronics which interface either a digital or an analog level with a controller in the PC assembly.

The analog flat-panel displays are particularly convenient as they directly replace color VGA (and larger) monitors with no hassle. But, these are often only partial solutions that leave a lot of electrical and mechanical design in your court.

More fully integrated solutions, like the one shown in Photo 6, also include built-in power supplies and provide numerous slots for ISA cards and for mounting disk drives.

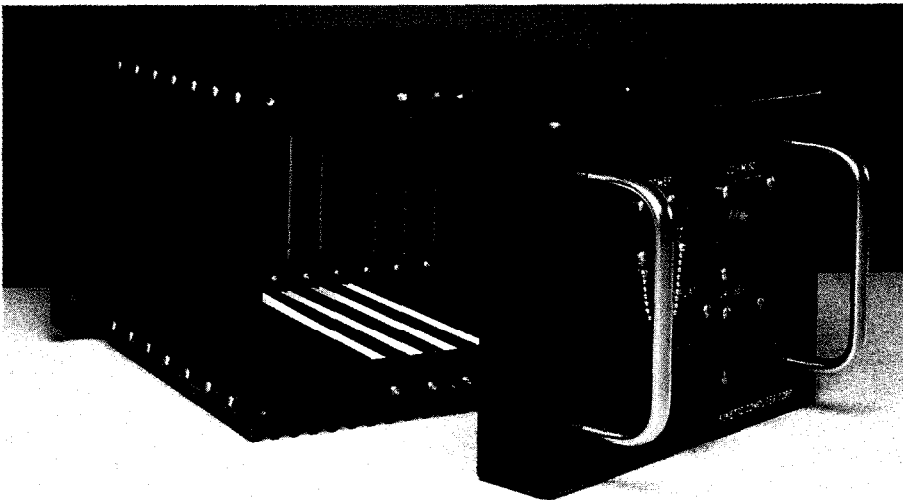


Photo 4: For embedded PC applications in extreme environments involving water spray, high shock, and vibration, Kinetic Computer's RCC-32 rugged card cage provides a convenient solution for STD-based systems while meeting **MIL** and **NEMA** standards.

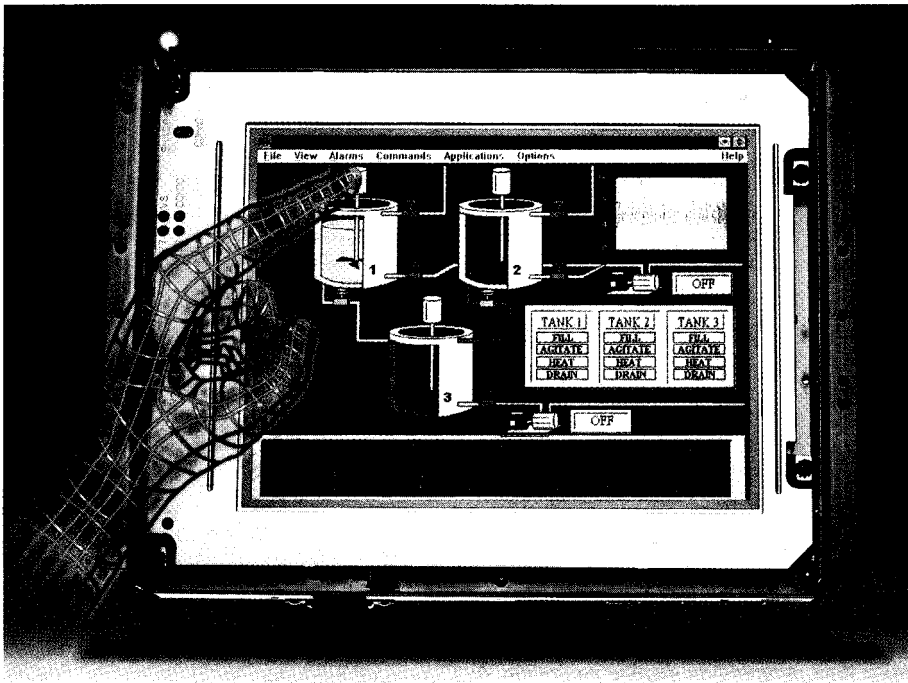


Photo 5: Simple OEM display requirements can often be met with modular packages like this one from Teknor. Manufacturers of similar units include Computer Dynamics and Interactive Display Systems.

A solution which combines the convenience of an integrated operator control panel with complete embedded PC, keyboard, and mouse, and is also rack mountable is the **VuePoint/SlimLine** offered by Interactive Display Systems. It's shown in Photo 2 of my September column (*INK 62*).

There are also two other extremes in embedded PC display systems. You can have very small LCD subsystems like that offered by **Ampro** (see Photo 7). It can marry neatly with a stack of PC/104 computer modules.

At the other end are the large, **bulky** CRT-based industrial operator consoles which have been around for years. These cumbersome units have given way to flat-panel technology. The consoles often include touch screens and/or sealed membrane keypads and function buttons, which survive wash down in extreme environments.

HAND-HELD PCs

Only a few years ago, it would not have been conceivable to carry an embedded PC in your attache case or pocket. But now, there are devices which permit just that.

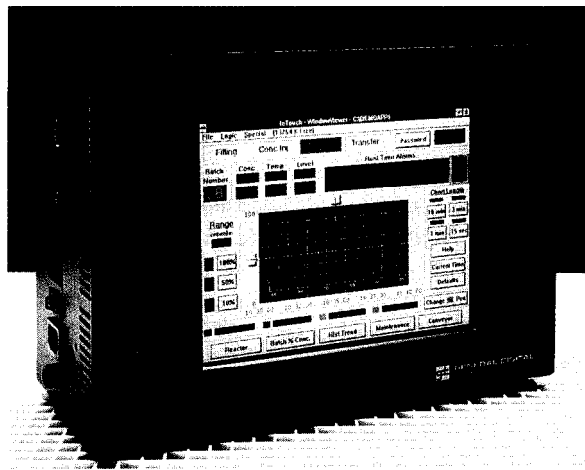


photo 6: Interactive Display Systems' deluxe full-size enclosure provides not only a flat-panel display (color or mono) and optional touch panel, but also a complete embedded PC and power supply.

Hand-held PCs are rapidly emerging with almost unlimited applications. They're no longer limited to simple nonDOS solutions built around 8051s in custom packages. You're hardly ever forced to roll your own or face high costs for portability.

While Two Technologies calls its offerings hand-held terminals, in fact the company offers a complete line from dumb terminals to complete hand-held PCs, like the one shown in Photo 8.

At about \$1000, these little computers offer solutions to many portable computing

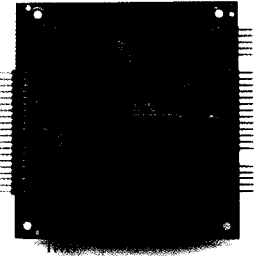
rttd[®]

Sets the Pace in Low Power, High Performance PC/104 Technologies



50 MHz
486 SXLC2
\$543

Features:
486SXLC2
50 MHz
387SX opt.
8K cache
2MB DRAM
3W typical
SSD
EEPROM
WDT
IDE & FDC
2 Serial
1 Parallel
PS/2 mouse
Keyboard
Quick Boot
Virtual devices

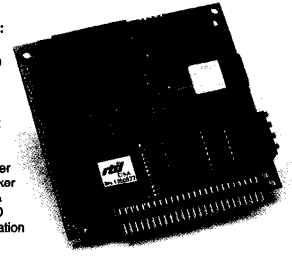


**CMi486SXLC2-1 Fully Integrated PC-AT
with Virtual Device Support**



200 kHz
12-bit DAS
\$498

Features:
200 kHz
12-bit A/D
16SE/BDI
Scan
Burst
Multiburst
1K CGT
Triggers
FIFO buffer
Data marker
12-bit D/A
16-bit D/O
+5V operation



**DM5408-2 200 kHz Analog I/O Module
with Channel-Gain Table**

Make your selection from:

9 cpuModules™

SuperXT™, 386SX, 486SXLC2, 486DX2, and 486DX4 processors. SSD, 6MB DRAM, RS-232/422/485 serial ports, parallel port, IDE & floppy controllers, Quick Boot, watchdog timer, power management, and digital control. Virtual devices include keyboard, video, floppy, and hard disk.

7 utilityModules™

SVGA CRT & LCD, Ethernet, keypad scanning, PCMCIA, intelligent GPS, IDE hard disk, and floppy.

16 dataModules®

12, 14 & 16-bit data acquisition modules with high speed sampling, channel-gain table (CGT), sample buffer, versatile triggers, scan, random burst & multiburst, DMA, 4-20 mA current loop, bit programmable digital I/O, advanced digital interrupt modes, incremental encoder interfaces, opto-isolated digital I/O & signal conditioning, opto-22 compatibility, and power-down.

&Real Time Devices USA

200 Innovation Boulevard • P.O. Box 906
State College, PA 16804-0906 USA

Tel: 1 (814) 234-8087 • Fax: 1 (814) 234-5218
FaxBack®: 1 (814) 235.1260 *BBS: 1 (814) 234-9427

RTD Europa RTD Scandinavia

Budapest, Hungary Helsinki, Finland
Tel: (36) 1 325-1 130 Tel: (358) 0 346-4538

RTD is a founder of the PC/104 Consortium and the world's leading supplier of PC/104 CPU and DAS modules.

needs. Their PC-Lite offers LCD display, many keypad options, '286 DOS PC with 1-MB RAM, solid-state disk, up to 115-kbps RS-232 interface, PCMCIA expansion, and even micro hard drives.

Kila Systems has a similar offering touted as a PC-in-a-Box with OEM pricing as low as \$500. This system offers three RS-232/485 serial ports, a printer port, and a 24-bit parallel port. They're powered from internal batteries (4-20 hours of operation) or external DC power for extended use.

Unfortunately, they're monochrome only and have restricted display resolution (192 x 128 pixels). Of course, they only show a portion of a conventional DOS screen. But, it's only a matter of time before we'll get full-color screen, hand-held PCs.

CONCLUSIONS

While the form your embedded PC takes varies tremendously depending on your application, it's best to consider the options and details of packaging early in



Photo 8: Operating from batteries or a wall pack, **hand-held PCs** like this **PC-Lite unit** from Two Technologies permit the embedded PC to travel anywhere for on-site data entry and collection, machine setup, data acquisition, and other remote computing needs.

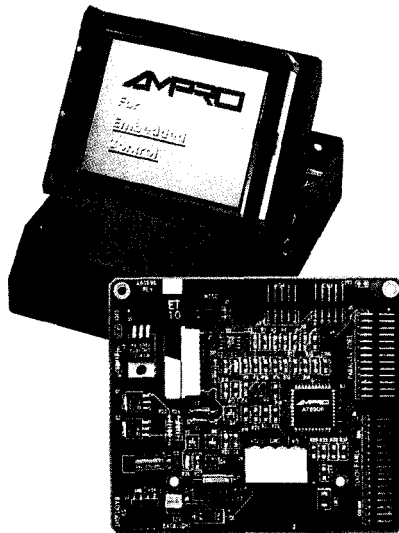


Photo 7: **Ampro's** Mini-LCD Adapter permits compact Sharp **LQ6** color **LCDs** to operate with their minimodule display controller. A miniature **PC/104** embedded system complete with operator display can be assembled around these **components**.

the development cycle. The quality of packaging affects everything that follows.

With the plethora of packaging options available—from simple backplanes to card cages, industrial boxes, complete PC-based display systems—there is seldom a good reason to reinvent the wheel. Investigate several approaches and multiple vendor offerings before deciding on this important aspect of your embedded PC design.

TYING THE BOW

It's been a pleasure this past year to introduce you to the fast growing and exciting world of embedded PCs.

However, due to work pressures, I'm forced to resign my position as columnist of Applied PC. There's much I've not yet touched on, but I'm sure others will pick up where I've left off.

I may be back from time to time to discuss specific topics. And, you'll still see my PIC projects and other INK feature articles when I can find the time to bring them to you.

Again, it's been a pleasure. **APC.EPC**

Russ Reiss holds a Ph.D. in EE/CS and has been active in electronics for over 25 years as industry consultant, designer, college professor, entrepreneur, and company president. He may be reached at russ.reiss@circellar.com or 70054.1663@compuserve.com.

CONTACTS

Analog and Digital **I/O** Panels
Opto-22
43044 Business Park Dr.
Temecula, CA 92590
(909) 695-9299
Fax: (909) 695-2712

Advantech
750 East Arques Ave.
Sunnyvale, CA 94006
(408) 245-8268
Fax: (408) 245-8268

LCD/Keyboard Modules
Z-World Engineering
1724 Picasso Ave.
Davis, CA 95616
(916) 757-3737
Fax: (916) 753-5141

STD Bus Card Cages and Accessories
Ziatech
1050 Southwood Dr.
San Luis Obispo, CA 93401
(805) 541-0488
Fax: (805) 541-5088

Kinetic Computer Corp.
270 Third St.
Cambridge, MA 02142
(617) 547-2424
Fax: (617) 547-7266

Computer Dynamics, Inc.
16530 Commerce Ct.
Middlebury Heights, OH
(216) 243-3900
Fax: (216) 243-3901

Industrial Rack-Mount Embedded PCs
Interlogic Industries
85 Marcus Dr.
Melville, NY 11747
(516) 420-8111
Fax: (516) 420-8007

Display Pack Systems
Teknor Microsystems, Inc.
616 Cure Boivin
Boisbriand, QC
Canada J7G 2A7
(514) 437-5682
Fax: (514) 437-8053

Interactive Display Systems
198 Freshwater Blvd.
Enfield, CT 06082
(860) 741-7171
Fax: (860) 741-7017

Miniature LCD Display Modules and PC/104
Ampro Computers, Inc.
990 Almanor Ave.
Sunnyvale, CA 94086
(408) 522-2100
Fax: (408) 720-1305

Hand-held PCs
Two Technologies, Inc.
419 Sargon Way
Horsham, PA 19044
(215) 441-5305
Fax: (215) 441-0432

Kilo Systems
2300C Central Ave.
Boulder, CO 80301
(303) 444-7737
Fax: (303) 786-9983

IRS

425 Very Useful
426 Moderately Useful
427 Not Useful

78 Firmware Furnace

84 From the Bench

90 Silicon Update

98 ConnecTime

FIRMWARE FURNACE

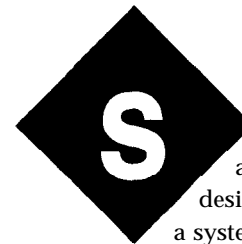
Ed Nisley

80x86 Performance Cache Craziness Redux



The ads
boast a
133-MHz
Pentium

with L1 and L2 cache. What does it all mean and how does it affect your final productivity? Ed begins taking a look at how memory cache helps (and sometimes hinders) system performance.



Some years ago, an innovative PC designer came up with a system board that wrung incredible performance from seemingly stock hardware. Nobody understood quite how it worked, but the standard benchmarks agreed that it ran rings around the competition!

This guy [I can't bring myself to call him an engineer] noted that benchmark programs use standard PC timing facilities. The usual approach to high performance packs more instructions into each second. His approach simply made each second slightly longer.. .

Closer to home, when I upgraded my test system from a 33-MHz '386SX to an 80-MHz '486DX2, the Firmware Furnace Task Switcher's protected-mode performance jumped by a factor of six. A few tests showed that the new system's cache memory accounted for essentially all of the performance increase. A better CPU, wider data path, and nearly tripled processor clock mattered hardly at all!

In this column, I'll explain why memory limits PC performance, show how caching can help, describe how a Direct Digital Synthesis loop reveals the cache in action, and present a simple circuit we'll use next month to get some detailed timing information.

If you've ever wondered how a 133-MHz CPU uses 70-ns DRAM, read on.

WHAT'S THE PROBLEM?

The Firmware Furnace Task Switcher, for those of you new to the column, is a 32-bit, protected-mode, cooperative multitasking program I built last year. While not a full-fledged operating system, it exercises and demonstrates a big chunk of the mysterious protected-mode hardware found in current PCs.

When I began writing the FFTS keyboard handler, I discovered that I had zapped the '386SX system board's keyboard hardware sometime in the last two years. Rather than fix it, I simply jacked up the box and slid another system board underneath. After twiddling a pair of delay loops that matched some peripheral gadgets to the CPU speed, the rest of the FFTS code worked perfectly.

After I adapted the code, I wondered just how much difference the new hardware actually made. Because the FFTS kernel displays the number of task switches per second on the video monitor, I used that as a simple figure of merit. Table 1 summarizes the results, with the old '386SX-33 setting the performance baseline.

The second line of that table may raise some eyebrows. The FFTS kernel actually ran 6% slower on the new board when I disabled both caches! Obviously, memory performance dominates everything else for this combination of hardware and software.

Enabling just the external cache boosted task-switching performance by more than 50%. Enabling only the internal cache gave better results, improving performance by a factor of five. Running with both caches enabled added another 20%.

You might think that every '486 system should run with both caches enabled. It turns out that not all '486 CPUs have the same internal cache size, some embedded PC boards lack external caches, and sometimes caching hurts.

But, if a six-times performance difference doesn't get your attention, flip to the next column!

WHO'S RESPONSIBLE FOR THIS?

In the beginning, the Original PC ran an 8088 CPU at 4.77 MHz. Each

CPU Type	CPU Caching		Switches per s	Speed Ratio
'386SX	none		400	1.00
'486DX2	none		377	0.94 (!)
'486DX2	external		577	1.45
'486DX2	internal		2068	5.17
'486DX2	both		2444	6.11

Table 1—The 80486 CPU's internal cache contributed *heady* all the factor-of-six performance improvement I saw when porting the protected-mode Firmware Furnace Task Switcher to the new board. In fact, the '486DX2-80 actually ran slower than the '386%33 when I disabled both caches!

instruction required several 210-ns clock cycles, which quite nicely matched the system's 250-ns DRAM access time. After all, when a simple JMP took 15 cycles, waiting a cycle or two for each memory access was no big deal.

Today, however, an 80-MHz '486-DX2 executes that same J M P in three 12.5ns CPU clock cycles. The five CPU cycles required for a 60-ns DRAM access look downright poky in comparison. When the processor stalls every time it reads or writes memory, a fast CPU clock doesn't buy very much performance.

The subject of caching would never arise if every PC used zero-wait-state memory. However, filling your PC with 8 MB of blazing RAM might pose a problem, even if your Mastercard and air conditioner were up to the challenge. Contrary to popular belief, just sticking fast RAM in a slow box won't make it go any faster.

Fortunately, most programs don't use all 8 MB at once. Your programs (usually) execute a few close-together instructions that access a few close-together variables, most of the time. Loops, for example, repeatedly execute the same instructions over and over again. You don't need 8 MB for a few dozen instructions!

A cache depends on two principles that computer science ma-

Figure 1—A typical program exhibits locality of reference because most of its memory accesses lie in a few areas. A relatively small cache can hold the programs working set, even for a system with 8 MB of RAM.

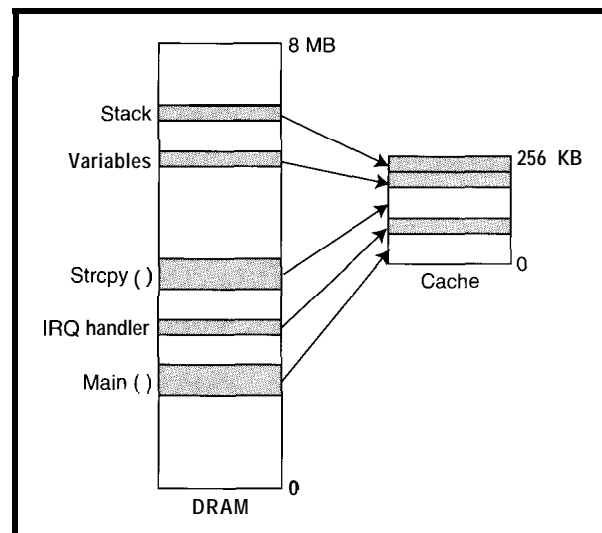
jors call *spatial* and *temporal locality*. The former means that programs typically access memory in small, sequential areas. The latter means that, once a program accesses an area of memory, it generally accesses it again soon.

Obviously, those principles cannot always be true or GUI programs would still fit on a single diskette. Comp Sci types refer to the collection of memory in use at any instant as the program's *working set*. That set changes as the program executes different subroutines, handles interrupts, and so forth. You can think of the working set as a small peephole sliding over your program's code and data.

Figure 1 shows how a relatively small cache can hold the few sections of main memory required at a given moment. The working set and thus the cache contents may be completely different every few milliseconds as the program moves on to new instructions and data.

The cache, being small, can run much faster than main memory. Accessing the cache thus imposes much less delay on each instruction, letting the CPU run faster. In the limiting case, with all instructions and data in the cache, the CPU sees zero-wait-state memory. In the general case, some accesses miss the cache, and the CPU must slow down for the corresponding external memory references.

Success has a simple metric: the cache hit ratio. Because the cache control logic cannot always predict precisely what the program will do next, the hit ratio must be less than



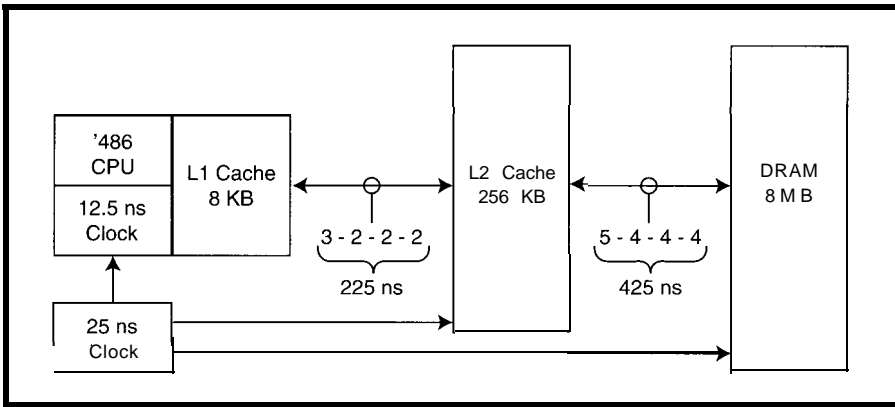


Figure 2-Accessing data from the external cache takes half as long as reading DRAM, but whether the CPU waits for 425 ns or 650 ns when data isn't in the cache depends on the system design. You hope for the former but may well get the latter!

1.00. When the ratio falls below about 0.70, designers get worried.

As the hit ratio declines, the time spent filling the cache with unnecessary data occupies memory bus cycles that delay the proper data. For some applications, disabling the cache actually improves the system's overall performance!

Collecting precise cache usage figures requires either complex CPU simulation code or expensive instrumentation. While we can't venture into that territory, we can certainly gain some insight into PC cache memory with the aid of some low-level code, a few I/O bits, and nothing more exotic than a decent oscilloscope.

First, let's find out what we're up against.

CLOCKS AND CACHES

Current PC ads show true specsmanship in action. Numbers like 66, 75, 80, 90, 100, and 133 MHz belie the simple fact that system-board designs remain stuck at about 40 MHz. The CPU doubles, triples, or quadruples the external clock to produce both its internal clock and those magic numbers. Regardless of your system, however, the PC Compatibility Barnacles and simple physics prevent a full-throttle design.

For example, my 80-MHz '486DX2 has a system clock of 40 MHz and

thus a 25-ns basic clock cycle. It uses 70-ns DRAM, but the recommended BIOS settings allow five cycles for the first memory access of a burst. That 125-ns delay provides enough time to detect an external cache miss, set up the memory access, and fetch data from the DRAM chips.

For reasons that we won't go into yet, memory accesses generally occur in bursts of four cycles. The next three accesses of a burst each require four cycles or 100 ns. On the average, then, a 16-byte access costs 26 ns per byte.

However, add the numbers and weep. Each memory access requires a total of 17 clock cycles! An instruction that normally completes in a few 12.5-ns CPU cycles must wait 425 ns if it touches DRAM. I don't recall seeing those numbers in any PC ad.

The mismatch between CPU clock rates and DRAM access times is now so dramatic that '486 and Pentium systems sport tandem caches. A primary, internal cache on the CPU chip provides zero-wait-state access, while a secondary cache on the system board reduces the effect of slow DRAM on the primary cache.

The Intel '486 series has an 8-KB internal cache, also called the L1 cache. Other vendors have different opinions of the proper L1 cache size: the IBM '486SLC2 has 16 KB while the Cyrix '486SLC has only 1 KB. Don't look for those numbers (at least the smaller ones!) in the ads either.

My system board has a 256-KB secondary cache, known as the L2 cache. You'll find L2 options ranging from zero on some embedded systems to half a megabyte or more on high-end PC boards. As with internal caches, the effect depends on your application.

The default BIOS settings for the external cache on my board allow three cycles for the first access of a burst and two cycles for each of the remaining three accesses. This works out to 225 ns total or 14 ns per byte.

The system board specs don't reveal whether the external cache copies incoming DRAM data while passing it directly to the CPU. If so, the CPU waits only 425 ns. If not, missing both caches costs 425 ns while the external cache fills, then 225 ns to load the internal cache. In other words, there's 650 ns of pure, unadulterated delay!

Figure 2 shows the layout of the two caches. The L1 cache can offer zero-wait-state access because it lives on the same chip as the CPU. The L2 cache, while dramatically slower, provides data at twice the speed of DRAM. For '486s, all data transfers occur four bytes at a time, with four transfers in each burst.

Most PC references simply advise you not to worry about caching, because you can't control it.

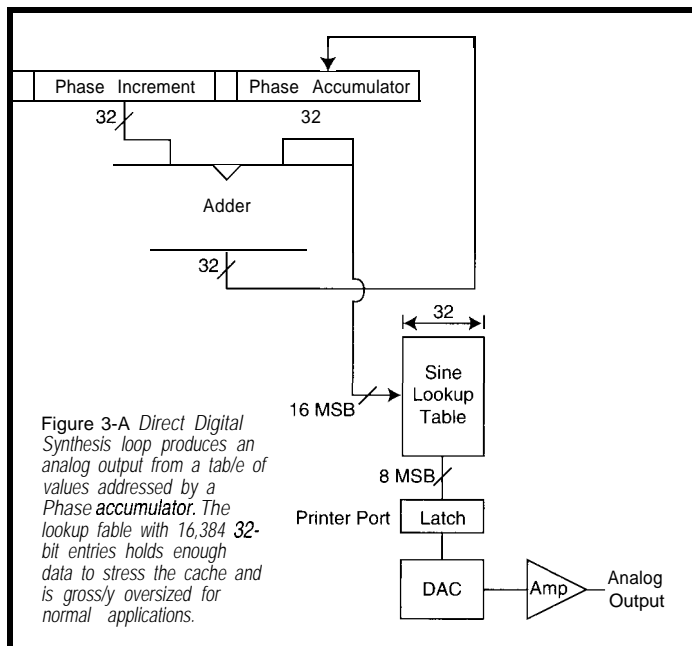


Figure 3-A Direct Digital Synthesis loop produces an analog output from a table of values addressed by a Phase accumulator. The lookup table with 16,384 32-bit entries holds enough data to stress the cache and is grossly oversized for normal applications.

Color me paranoid, but I want to know more about anything that drags an 80-MHz CPU to a crawl.

Don't you!

DIRECT DIGITAL SYNTHESIS

Although the FFTS kernel provides a dramatic introduction to the subject, it is neither convenient nor particularly adaptable for our purposes. I decided to write a small test routine that exercises the cache under carefully controlled conditions.

You have surely seen DDS (Direct Digital Synthesis) described elsewhere, probably in the context of generating good sine waves with fine frequency control. I've perverted the idea into a cache exploration tool.

Engineers generally implement DDS in hardware because the algorithm requires very little control logic. Figure 3 shows the basic idea: add an increment to a phase accumulator that addresses a table. Each pass through the table produces a single cycle of the output value, with the frequency determined by the size of the increment.

Notice that the phase increment and accumulator registers have many more bits than needed to address the output lookup table. This overkill allows phase increments much smaller than a single output step and gives the DDS very fine frequency resolution.

The lookup table's width has no

relation to its number of entries. You can have a tall, skinny table or a short, fat one. You can produce a respectable analog output from an *g*-bit-wide table and get an excellent waveform with 16 bits. The system's amplitude resolution spec determines the table width and its frequency range sets the length.

A normal DDS loop produces output values based on a fixed timing reference. Each clock tick latches a new (increment + phase) value into the phase accumulator register and sends a new table output to the DAC. The output frequency equals the clock frequency times the phase increment divided by the maximum possible phase increment.

For example, had I written the DDS loop to produce one output value on each BIOS tick, the lowest frequency would be $18.2 \text{ Hz} \times 1/2^{32} = 4.2 \text{ nHz}$. Yes, 4 nanohertz. One cycle every 2.7 days. Good for timing glacier races.

The highest frequency depends on how ugly a sine wave you can tolerate. Assuming you can stand 32 outputs per cycle, the BIOS-ticked loop would run at $18.2 \text{ Hz} \times 2^{27}/2^{32} = 586 \text{ mHz}$. Yes, half a cycle per second.

Microcontroller applications that need a fairly low output frequency can use DDS loops. Suppose you updated a 16-bit accumulator with a 100-kHz interrupt. The minimum frequency would be $100 \text{ kHz} \times 1/2^{16}$, or about

1.52 Hz. The maximum, again assuming 32 outputs per cycle, tops out at $100 \text{ kHz} \times 2^{11}/2^{16}$ or 3 kHz. The DDS loop gives you a 3-kHz sine wave that you can adjust in 1.5Hz steps.

Now you can see why real DDS loops require hardware registers with fast clocks.

Homework: figure the output frequency range and resolution for a 300-MHz clock applied to 32-bit increment and accumulator registers.

Hint: check the Analog Devices AD9950 DDS Phase Accumulator chip for more details. The state of the art moves right along, but only ECL logic can keep pace!

You must use a fixed clock to get a known output frequency regardless of the code's execution time. In my test code, however, I simply let the DDS loop run as fast as it can, updating the phase accumulator and producing output values whenever it can. Given a fixed CPU clock and no other distractions, the update rate depends on the DDS loop's execution time.

Here's the punch line: the DDS loop executes faster when it accesses data in the CPU cache. By watching the update rate on a scope, we can detect cache misses as they occur. By adjusting the phase-increment step size, we can control the amount of data passing through the cache and, thus, the cache hit ratio.

More homework: what cache hit ratio would you expect with a phase increment of 1? What about 1024?

Hint: your first guess is probably wrong, but I haven't given you quite enough information to solve the problem yet.

THE HARDWARE CORNER

Although we'll deal primarily with the loop's low-level timing, I built a simple 8-bit DAC circuit that generates an analog output voltage. A PC printer port has (barely) enough bits to drive the DAC, produce a few timing signals, and read a few DIP switches. Every PC has a printer port, however, which means we can concentrate on code rather than hardware.

Figure 4 details the minimal circuitry and port connections. Photo 1 gives you an idea of the layout on a

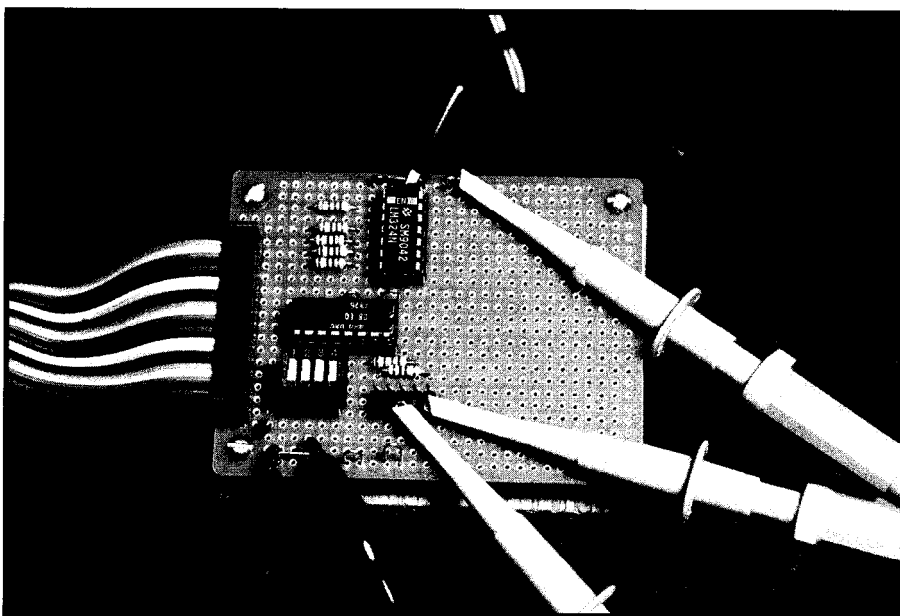


Photo 1—A custom protoboard provides convenient access to the PC parallel-port signals we use while investigating cache performance.

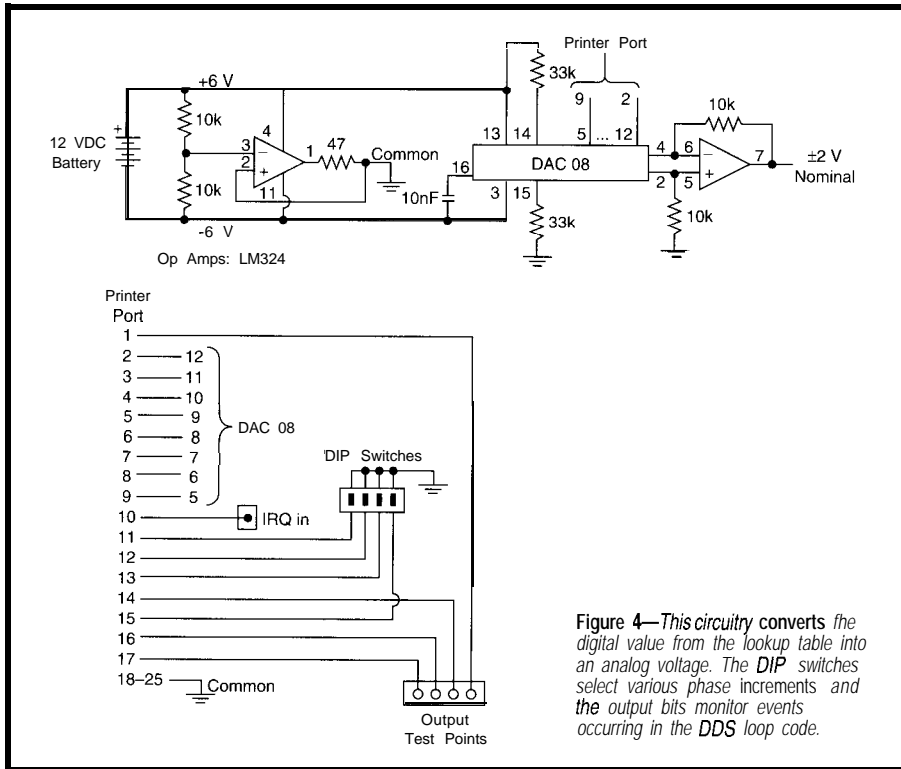


Figure 4—This circuitry converts the digital value from the lookup table into an analog voltage. The DIP switches select various phase increments and the output bits monitor events occurring in the DDS loop code.

2.5" x 3" Radio Shack perfboard. I used a ribbon cable crimped between a 2 x 13-pin socket and a DB-25F connector. For this application, the additional shielding in a round cable makes no difference.

A 12-V battery provides ± 6 -V power for the circuit, with an LM324 op-amp driving the centered common (ground) voltage. This technique works well for the single milliamp required by this circuit. I happened to have a 12-V sealed lead-acid battery on my desk and a quad op-amp in the parts box, but a ± 6 -V split supply works as well.

You must, however, connect at least one printer port ground pin to the analog common voltage. Don't short either side of the battery to the port's ground connection, lest you release the magic smoke inside the ICs.

A four-position DIP switch sets the printer port's four status input pins. My DDS code doesn't, use the port's IRQ signal on pin 10, but you may as well wire it to a terminal. I connected the port's four output pins plus the IRQ signal to a five-pin terminal strip. They're not the best scope probe test points, but they work well enough.

The digital input bits from the DIP switch appear in the printer status port at (base + 1). The output bits are in the

control port at (base + 2). Check your references for the details of which bits have what polarities in each port. The good news is, as long as you follow the schematic, everything works OK.

The BIOS stores the LPT1 printer port base address in the word at 0040:0008. The more-or-less standard value is 0378 hex, but you may find 03BC or 0278, depending on the hardware in your PC. When you write your own programs, use the BIOS address rather than hard-coding an assumption that works on just your PC.

Hardware checkout shouldn't require much effort. I used DOS Debug to set the LPT1 data port to 00, 7F, and FF and a voltmeter to verify the corresponding minimum, midrange, and maximum output voltages. The circuitry I used produces a ± 2 -V output.

Debug also sets the control bits and displays the status bit, albeit with less convenience than you'd like. The disk with *The Undocumented PC* includes IOSpy, a TSR that monitors and displays PC I/O ports. I watched the bits flip as I toggled the DIP switches, which made life much easier.


If your parts box includes a single-supply DAC and op-amp, the design may be even simpler. There are no critical timing specs, as the software

DDS loop produces a new output value roughly every 10 μ s. The output table I used presumes the DAC accepts raw binary coding.

In a real DDS application, you must pay much more attention to the analog circuitry. What you see in Figure 4 provides a quick-and-dirty look at the sine output. It is certainly not the final word on the subject!

RELEASE NOTES

I can't release the code implementing the DDS loop until I have room for the caveats. Believe me, it is a truly hostile beast. Most programs don't disable all interrupts and shut off RAM refresh, but that's what we need for precise timings.

Next month, we'll take a look at the DDS software and probe '486 cache behavior in serious detail. 

Ed Nisley (KE4ZNU), as Nisley Micro Engineering, makes small computers do amazing things. He's also a member of Circuit Cellar INK's

staff. You **may reach him** at ed.nisley@circellar.com or 74065.1363@compuserve.com.

REFERENCE

Van Gilluwe, Frank. *The Undocumented PC*. Reading, MA: Addison-Wesley, 1994. ISBN 0-201-62277-7.

SOURCE

Analog Devices sells their data manuals and app notes through their literature hotline. If you intend to do DDS right, get their **Data Converter Reference** manuals and the **System Applications Guide**.

Analog Devices
One Technology Way
P.O. Box 9106
Norwood, MA 02062-9 106
(617) 461-3392
Fax: (617) 821-4273

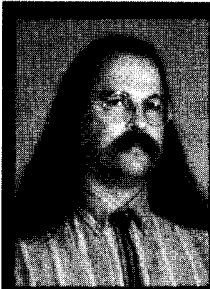
IRS

428 Very Useful
429 Moderately Useful
430 Not Useful

FROM THE BENCH

Jeff Bachiochi

LEDs Finally Fill the Rainbow



The last hurdle preventing full-color

LEDs is now being cleared. After reviewing how the eye and color LEDs work, Jeff hooks the LEDs up to a PIC-controlled circuit which manages current to produce all the colors of the rainbow.



Recently, my second son Ryan portrayed the crippled money mongrel Mr. Potter in the high school's rendition of *It's a Wonderful Life*. The original movie, starring Jimmy Stewart and Donna Reed, was filmed in black and white, as were all movies until color was introduced during the filming of *The Wizard of Oz*. Deeming it too expensive to reshoot the beginning Kansas scenes, the directors left them as black and white, while everything in Oz is in splendid technicolor. As it turned out, the split is an artistic quirk of fate.

Wheeler-dealer Ted Turner has funded the coloration of many older black-and-white films, including *It's a Wonderful Life*. Luckily, he balked at colorizing Oz.

I watched many a television show in black and white. And, although I've shared years of *Sesame Street* with my kids, I was raised on *Ding Dong School* (with *Miss Francis*).

Anyone else cultivated on black-and-white programming knows how living in full color helped one imagine in the grays. The perception of black and white is a bit different for those raised strictly on color programming.

Photography has gone through the same transformation. Early black-and-white photos were developed on glass, metal, and finally paper.

The introduction of Kodachrome has all but wiped out black-and-white photography. Still, to many professional photographers, black and white continues to be the medium of choice.

If all this seems a bit outdated, look at laptops. The relatively new flat-screen technology has brought full color to LCDs and led to the portable-computer revolution. Flat-panel, large-screen, high-definition LCD TVs are in our not-too-distant future.

The last hurdle preventing full-color LEDs is now being cleared. Even though blue LEDs were actually announced years ago, it took a hefty wallet to purchase one until recently.

One of the first available to you and me is a multicolor LED. In reality, it has four LEDs on the same substrate: one red, one yellow, and two blues.

ELECTROMAGNETIC RADIATION

The limits of the electromagnetic spectrum extend from audio to cosmic rays. See Table 1 for a view of the electromagnetic spectrum.

Of all the frequencies we are bombarded with each day, the narrow band from 430 to 750 THz is of most importance to our limited bodies. We receive the majority of our everyday reality from this otherwise insignificant slice of the universe's spectrum.

The magic of the rainbow begins with the color red at a wavelength of about 700 nm, and continues with orange (620 nm), yellow (580 nm), green (520 nm), blue (420 nm), and violet (400 nm).

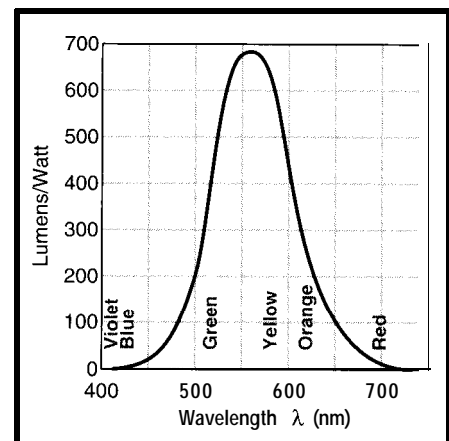


Figure 1—Our eyes are most sensitive to green and yellow light (i.e., the tip of the bell-shaped curve) as opposed to red or violet, which are located near the ends of the curve as depicted here.

The radiant energy of this light is measured in W/m^2 . However, the eye has a bell-shaped sensitivity curve which peaks at 555 nm (yellow-green).

Therefore, the actual brightness as seen by the eye depends on the frequency. Figure 1 offers a graph of photopic luminosity in lumens per meter squared (lm/m^2) versus color (wavelength).

This function is similar to a bandpass filter centered on 540 THz. It means our eyes need to receive higher radiation densities of reds and violets to perceive them at the same brightness as yellows and greens.

If we assume a point light source, we can count the number of lumens which fall on the area of a sphere surrounding it. By comparing the power used by the source to the total lumens falling on the surface of the sphere, we get an idea of the efficiency of the source.

LEDs don't generally expend their light uniformly in all directions. Therefore, luminous intensity must be measured and graphed with reference to its axis of symmetry, either vertical or horizontal. (Some LEDs don't have an axis of symmetry, but I'll leave that for another day.)

The number of lumens falling on an area of the sphere equal to the radius of the sphere squared is known as the lumens per steradian (lm/sr), or candela (Cd), which is actually about $1/42$ the sphere's area of $4\pi r^2$.

In Figure 2, the candela is graphed in relation to the center axis of a typical LED. Maximum candela is referenced as 1.0. This spatial distribution pattern depends on the lens system used by the LED.

Video cameras today are rated in lux, which indicates the sensitivity of the CCD receiver. One lux is equal to $1 lm/m^2$.

P-N JUNCTION

Like semiconductor diodes, LEDs are made up of positive-negative (p-n) junctions. The base material has an outer electron shell that is not

Frequency	Wavelength	Name
16 Hz-16 kHz		Audio Frequency
10-30 kHz	30-10 km	Very Low Frequency
30-300 kHz	10-1 km	Low Frequency
300 kHz-3 MHz	1 km-100 m	Medium Frequency
3-30 MHz	100-10 m	High Frequency
30-300 MHz	10-1 m	Very High Frequency
300 MHz-3 GHz	1 m-100 mm	Ultra High Frequency
3-30 GHz	100-10 mm	Super High Frequency
30-300 GHz	10-1 mm	Xtra High Frequency
300 GHz-3 THz	1 mm-100 μm	
3-30 THz	100-10 μm	
30-300 THz	10-1 μm	
300 THz-3 PHz	1 pm-100 nm	Visible Light
3-30 PHz	100-10 nm	X-rays
30-300 PHz	10-1 nm	
300 PHz-3 EHZ	1 nm-100 pm	
3-30 EHZ	100-10 pm	
30-300 EHZ	10-1 pm	Gamma rays
	1 pm-100 fm	
	100 fm-10 fm	Cosmic rays
	10-1 fm	

Table 1—The electromagnetic spectrum as we know it today contains some interesting names. The frequency and wavelength of visible light is highlighted.

complete on its own. Thus, outer-shell electrons can be shared between adjacent atoms (covalent bonding) and form crystalline structures.

In this form, the materials are non-conducting because all atoms have full outer shells. However, by adding an atom of an element with one less electron (the element one lower than the base element in the periodic table), a crystalline structure is created with a missing electron (or an extra hole).

Similarly, if an atom with an extra electron is added to the base material, the crystalline structure contains an extra electron.

The material with extra electrons is considered to have a negative charge (n-type) while the material with an electron deficiency has a positive charge (p-type). A p-n junction is formed when n- and p-type materials are combined. The extra electrons and holes attract one another, which creates a current flow in the forward-bias direction and prevents reverse flow.

Material	Color	Rise Time	Fall Time
GaP	red	700 ns	3000 ns
GaP	yellow-green	800 ns	700 ns
GaP	green	50 ns	400 ns
GaAsP	red	200 ns	150 ns
GaAsP	orange	300 ns	220 ns
GaAsP	yellow	300 ns	220 ns
GaAlAs	red	40 ns	60 ns

Table 2—Each color of LED exhibits different rise and fall times which depend on the LED's manufacturing process.

When a free electron recombines with a free hole, radiant energy is released equal to the difference between the two carriers prior to recombination. The wavelength of the emitted energy varies with the potential difference and the material used. If the emitted wavelength falls within that of visible light, an LED is born.

LIGHT-EMITTING MATERIALS

The most common materials used in today's LEDs are GaP, GaAsP, and GaAlAs.

Gallium phosphide (GaP) is capable of emitting red (4% efficiency), green (0.15%), or yellow-green (0.3%). The emitted wavelength is controlled by the dopants: zinc and oxygen for red and nitrogen for yellow-green.

Gallium arsenide phosphide (GaAsP) is capable of emitting red (4% efficiency), orange (0.3%), or yellow (0.12%). Again, nitrogen is used as a dopant for orange and yellow.

Gallium aluminum arsenide (GaAlAs) is capable of emitting red. It has the highest luminous efficiency, about 15%.

Not only does the material used affect the wavelength of the color emitted, it also affects the rise and fall response time (see Table 2). This variation is of concern to those interested in fiber optics. Notably, the efficiencies given above also vary based on manufacturing process.

ENTER THE BLUES

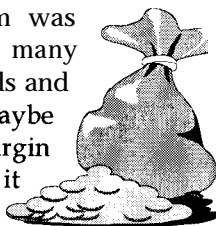
Because the gallium materials used in these LEDs don't have the potential for emitting higher wavelengths (the blues), other materials have been investigated.

Although initial results showed promise, research into gallium nitride (GaN), zinc sulfide (ZnS), and zinc selenium (ZnSe) was discontinued for practical reasons. It was found that silicon carbide (SiC) had the most favorable manufacturing characteristics, as well as the highest conversion efficiencies.

The Whole is less expensive than the sum of it's parts.....

You're shipping the first system; out the door, the customer is happy, and you can breath a sigh of relief now, right? But can you really afford to relax now?

If that system was built using many different boards and circuits, maybe your profit margin isn't what it should be.



if you Integrate!

HiTech can combine all those boards and circuits onto a single board, saving you big money. Reduce your unit cost with less inventory and fewer vendors, faster assembly time, fewer cables, and even smaller package size. Built to your specifications, an integrated board from HiTech can include analog, digital, FPGA and even custom mixed signal ICs, all for less money than your current solution.

Case: the 188SBC

One customer needed an x86 class processor with 16 channels of 12 bit A/D and 8 channels of 12 bit D/A, LCD, Keypad and Opto-rack interface, two serial ports, a printer port, real-time clock, and more. A multi-board solution would cost around \$1200 each. The 188SBC costs only \$749, and includes a power supply and a custom FPGA!

Call now and find out how we can help you reach your profit potential with integrated solutions.



HiTech Equipment Corp.
9400 Activity Road
San Diego, CA 92126
[Fax: (619) 530-1458]

Since 1983

(619) 566-1892



E-mail: info@hte.com - Ftp: <ftp://hte.com>
Web: <http://www.hte.com>

For the blue LEDs as with the others, p- and n-type materials are produced by doping the base material with tiny bits of a material that has more or fewer electrons in its outer shell. Adding aluminum to silicon carbide produces the p-type region, and nitrogen to the SiC produces the n-type material.

The switching speed of the blue LEDs is in the 800-ns range. Efficiency is lacking. Not only is the dominant wavelength beyond that of the visual spectrum, but the blue end of the visual spectrum, like the red end, is not as sensitive as the green-yellow peak. Therefore, very small millicandela (mCd) output is typically produced.

THE MULTICOLORED LED

By packaging red, green, and blue LEDs in a single package, white light (as well as all the other colors) can be emitted. Since blue LEDs are so inefficient, two blue LEDs are included with a single red and green.

The red LED has the highest output efficiency-it can produce 12 mCd. The green can produce 2 mCd, but the two blue LEDs together output only 1 mCd. Table 3 lists the currents that each LED requires to produce the complete spectrum of visual light.

Notice the high currents needed in both blue LEDs. This requirement demonstrates the low efficiencies of today's blue LEDs. In addition, while

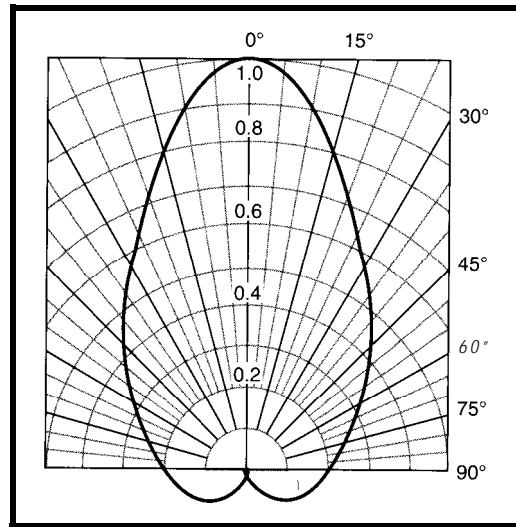


Figure 2—Depending on the LED's lens makeup, radiation is not equal in all directions.

ries-dropping resistor which limits the current, based on your VCC. It's not a big deal for single-color operation.

By placing a switching transistor in the circuit (one per color), you can add digital on/off control for each color. The two blue LEDs can be run from the same transistor, provided that each has its own series-dropping resistor.

Choose a series-dropping resistor that provides the maximum current (see Table 3) for each color. By biasing the transistors either on or off, you can satisfy many combinations simply with maximum or minimum (zero) current.

This static state doesn't offer much control without operating the transistors in their linear region. Digitally, you can acquire that kind of analog control with much simpler circuitry. If

LED	red	org	yel	grn	aqu	blu	vio	wht
red	5 mA	5 mA	5 mA	0	0	0	5 mA	5 mA
green	0	12 mA	30 mA	20 mA	20 mA	0	0	20 mA
blue1	0	0	0	0	35 mA	35 mA	35 mA	35 mA
blue2	0	0	0	0	35 mA	35 mA	35 mA	35 mA

Table 3—The different-colored LEDs require different amounts of current to produce particular colors. The current required depends on both the LED efficiency and the eye's sensitivity to the LED's color.

the red and green LEDs have a forward drop of about 2 V, blue LEDs require about 3.5 V, so they're not appropriate for 3-V systems.

I SEE THE LIGHT

To operate a multicolor LED in any one color, choose the appropriate se-

you bias the transistor with a (50/50) square wave and use a frequency high enough to eliminate any visible flicker, the LED is perceived as dimmed (operating at a lower current).

This method is used with multiplexed displays to increase peak current while keeping the average current

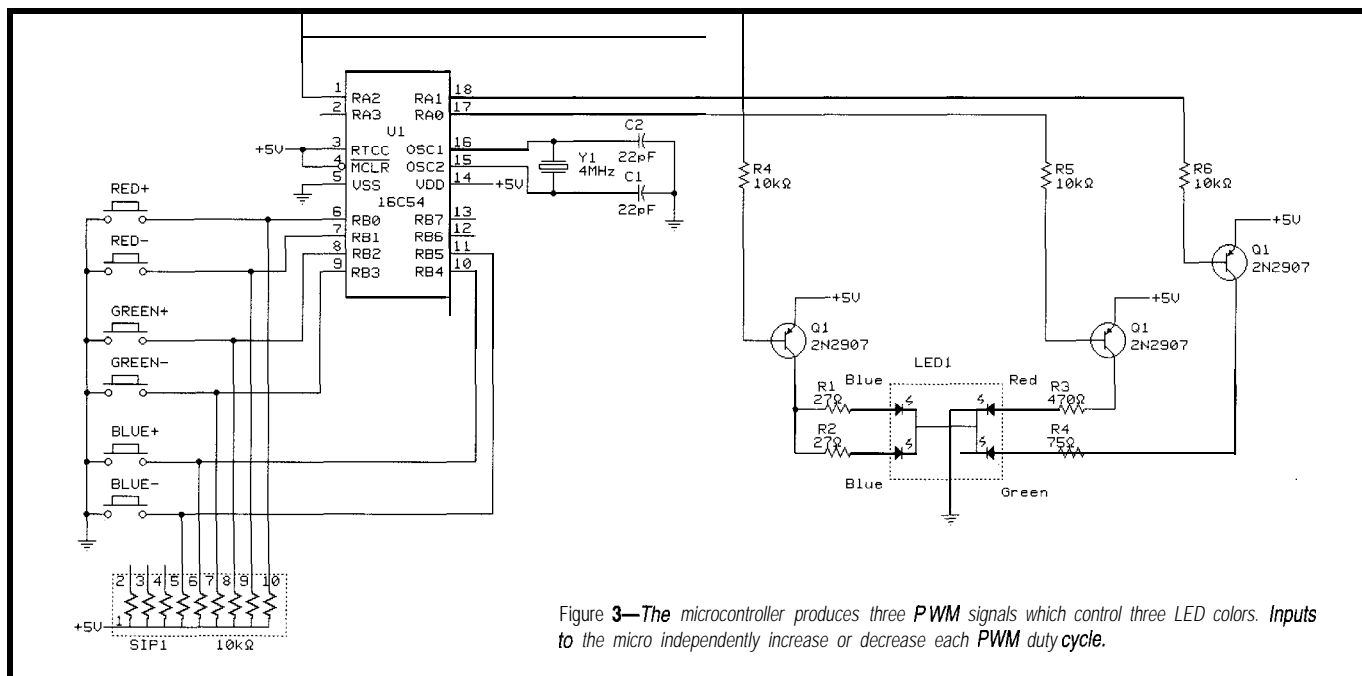


Figure 3—The microcontroller produces three PWM signals which control three LED colors. Inputs to the micro independently increase or decrease each PWM duty cycle.

down. In this application, I vary the on- and off-time of each cycle to adjust the average current through each LED. By increasing on-time, the average current and luminosity go up to 100%. By decreasing it, the average current and luminosity go down to 0%.

This type of PWM control is widely used in the control of DC motors. To create three PWM outputs, I used an inexpensive microprocessor.

Refer to the schematic in Figure 3. Here, six push buttons are connected to one of the micro's ports and configured as inputs. The remaining port is configured as outputs which bias the three LED drive transistors. Some micros can drive LEDs directly, but not to the extent necessary for the blue LEDs (35 mA each).

An alternate circuit might use four LMC555 timers. The first creates a stable oscillator, which is fed into the trigger inputs of the remaining three '555s configured as PWMs. The control inputs of each PWM come from three potentiometers. The PWM outputs provide the bias for each of the LED's transistor drivers.

There is an advantage to using a micro in the investigation of the multicolor LED. Although both circuits enable the user to raise and lower the light levels of each LED, the circuit using the micro can be easily altered through software.

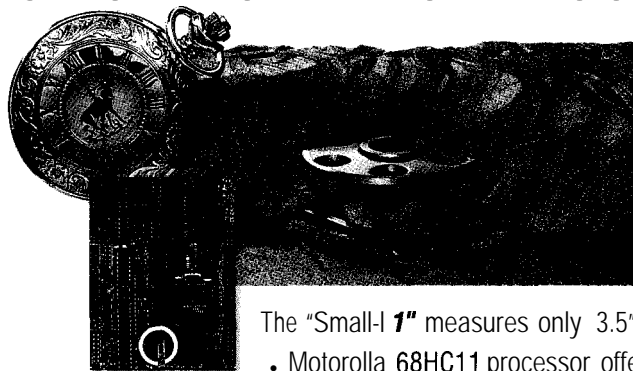
TRI-PWM CONTROL

The micro's code takes up around 100 words (12-bit instructions). This code includes generation of a master oscillator with three PWM outputs. Six push buttons (two for each color)

increase or decrease the three PWM percentages.

The real-time clock/counter (RTCC) is initialized with a prescaler to create a clock tic about every 4 us. The master PWM frequency is set by

When you don't have time to reinvent the wheel



Eliminate the time and hassle of extra design turns with the Small-I 1™ controller

The "Small-I 1" measures only 3.5" x 4.5"

- Motorola 68HC11 processor offers familiar programming and bullet-proof reliability
- 8K battery-backed RAM • Watchdog timer
- Up to 32K on board memory • Real-time clock
- RS422/485, and dual-RS232 ports
- Keypad and LCD interfaces • Expansion bus to I/O interface • Single 5 volt supply

**Micro & Control
Diagnostics™**

Micro Control & Diagnostics, LLC
300 Main St., Suite 201, Lafayette, IN 47901

Phone: (800) 429-6797 or (317) 429-6777 • FAX (317) 429-6544 • Web: <http://www.mcontrol.com>

writing a value of E7h to the RTCC and waiting for it to roll over (once every 25 tics). One output cycle equals 100 rollovers (**10** ms). Each cycle has 100 equal parts, so the PWM control is equal to 1 in 100 parts (**1%**).

Our reaction time is far slower than that of a loop time (RTCC rollover = 100 us). To prevent a button push from being sampled and action taken too quickly, buttons are only sampled once every cycle (10 ms).

A button must be sampled 8 times (**80** ms) before action is taken. This procedure slows actions to a reasonable level.

The two buttons associated with each color increment and decrement a PWM value associated with that color. The value can range from 0 to 100 percent. Each time the master PWM count rolls over, all the outputs are cleared. Each time the master PWM count is incremented, it is compared to each of the color's PWM values.

When a match occurs, the associated output is set high, turning on the LED. If the color's PWM value is low, its off-time is short because a match with the master's PWM count is made early. If a color's PWM value is high, its off-time is long.

Three colors, three independent values, and three independent PWM outputs produce about a million color combinations. Once you've been through all of them, you'll probably want to take this a step further.

How about reprogramming the micro to pick pseudorandom values for each of the three colors?

You'll be able to generate more colors than the old NBC peacock ever did. Although the LED, like that peacock, has been around for many years now, it is finally here in living color. 📺

Jeff Bachiochi (pronounced "BAH-key-AH-key") is an electrical engineer on Circuit Cellar INK's engineering staff. His background includes product design and manufacturing. He may be reached at jeff.bachiochi@circellar.com.

Bliss, John. "Theory and Characteristics of Phototransistors,"

Motorola Optoelectronics Device Data, Ap. Note AN440. 1989.

Smith, George. "LEDs and Photometry," **Siemens Optoelectronics Data Book**, Ap. Note 1. 1993.

Weyrich, Dr. Claus. "Blue-Light Emitting Silicon-Carbide Diodes—Materials, Technology, Characteristics," **Siemens Optoelectronics Data Book**, Ap. Note 31. 1993.

SOURCES

CREE CRGB-5D18 multicolor LED
Digi-Key Corp.

P.O. Box 677
Thief River Falls, MN 56701
(218) 681-6674
Fax: (218) 681-3380

PIC16C54
Microchip Technology, Inc.
2355 W. Chandler Blvd.
Chandler, AZ 85224-6199
(602) 786-7200
Fax: (602) 899-9210

Motorola Optoelectronics Device Data

Motorola Literature Distribution
P.O. Box 20912
Phoenix, AZ 85036
(602) 952-4103
Fax: (602) 952-4067

Sharp Optoelectronics Data Book: Visible Light Emitting Diodes

Sharp Electronics Corp.
Microelectronics Group
5700 NW Pacific Rim Blvd., Ste. 20
Camas, WA 98607
(206) 834-2500
Fax: (206) 834-8903

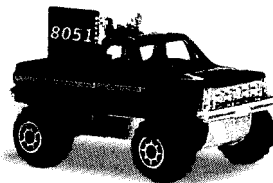
Siemens Optoelectronics Data Book

Siemens Components, Inc.
Optoelectronics Div.
19000 Homestead Rd.
Cupertino, CA 95014
(408) 257-7910
Fax: (408) 725-3439

IRS

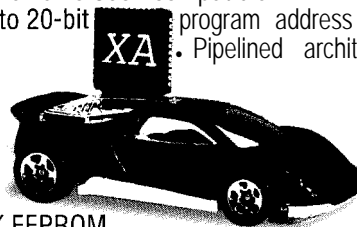
431 Very Useful
432 Moderately Useful
433 Not Useful

First in performance . . . First to market



**The Small-XA™
Controller
leaves 8051
boards in the
dust, without
time-consuming
source-code
rebuids**

- 16-bit 80C51XA operates up to 30 MHz, yet remains 8051 compatible
- Up to 20-bit program address space
- Pipelined architecture



- 64K EEPROM
- 64K battery-backed RAM
- Expansion bus to I/O interface
- Real-time clock and watchdog timer
- RS422/485, and RS232 serial ports
- Available I/O expansion modules, development software, and other accessories
- Custom-engineered solutions

**Micro & Control
Diagnostics™**

Micro Control & Diagnostics, LLC
300 Main St., Suite 201, Lafayette, IN 47901

Phone: (800) 429-6797 or (317) 429-6777 • FAX (317) 429-6544 • Web: <http://www.mcontrol.com>

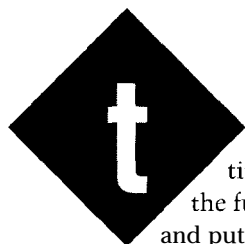
REFERENCES

Bueche, F. **Principles of Physics**.
McGraw-Hill, USA, 1965.

Fuzzy PID-Pong

SILICON UPDATE

Tom Cantrell



his month, it's time to get back on the fuzzy logic beat and put the AL220 (*INK*

67) through its paces in an actual design.

The goal is to cut through all the hype, hope, and hoopla of fuzzy and decide once and for all if it is good (i.e., worthy of consideration for mainstream designs and applications).

In "Third Thoughts on Fuzzy Logic" (*IEEE Micro*, August 1995, p.80), well-known fuzzy skeptic Robert A. Pease notes that most technical papers on fuzzy logic "are written in esoteric symbols and couched in obscure, scholarly phraseology."

Fortunately, for R.A.P. and other like-minded pragmatists in the audience, the only thing I ever write with "esoteric symbols" is a C program, and long-time readers certainly know that, when it comes to phraseology, you can count on me to eschew obfuscation.

He also writes that such papers usually "show trivial or 'toy' examples."

I have to admit the PID-Pong machine (Photo 1) has certain toy-like qualities. Notably, when I carried one into the local Shlep & Ship delivery franchise, a gaggle of kids surrounded it while the parents shrunk back. Is that the definition of a good toy or what?

While it's true the PID-Pong machine is quite fun, it also proves a surprisingly tough challenge for a controller. Indeed, since the original article ("PID-Pong Challenge," *INK 42*), incarnations of the machine have popped up at various trade shows and conferences. R.A.P. accepts the PID-

Pong machine as a worthy opponent when he observes that the gadget is "fairly nonlinear and not easily controlled."

Finally, he laments that the typical fuzzy-logic-application example fails to disclose the gory technical details (such as the type and performance of sensors) that distinguish pipedreams from purchase orders.

Without even getting to the issue of whether R.A.P. and I agree on the subject of fuzzy logic itself, I certainly concur with his complaints about dubious marketeering. I hope this article, with real hardware addressing a nontrivial problem, passes the R.A.P. snake-oil test.

PID-PONG MACHINE BASICS

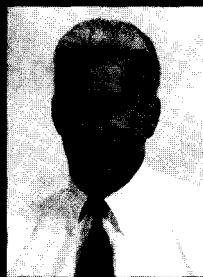
Readers of the earlier articles will note from Photo 1 that the PID-Pong machine has undergone quite a makeover. I'll be the first to admit that my previous machines, featuring tape as a primary construction material, could hardly be classified as robust or even presentable. So, this time around, I solicited the assistance of a mechanically enabled colleague (thanks go to Jim Deckert) for a much nicer version.

Having built a few machines myself and walked a number of other people through it, I've learned there are a number of traps and pitfalls for the unwary. I'm trying to persuade Jim to make the various bits and pieces of the PID-Pong machine available for those of you more interested in controlling the machine than building and debugging it.

Nevertheless, remembering R.A.P.'s admonition to open the kimono, here's everything you need to know about building your own PID-Pong machine.

As shown in Figure 1, the machine relies on the popular Polaroid TI01 ultrasonic transducer to determine the ball's position. Just like the "sonar" in old war movies, the transducer sends out a "ping" (49.4-kHz burst) and listens for the echo.

Multiplying the delay between ping and echo by the speed of sound and dividing by two yields the distance. The TI01, with $\pm 1\%$ accuracy up to 35', is widely used in autofocus cameras, collision avoidance (e.g., RV



In *INK* 67, Tom introduced Adaptive Logic's

fuzzy logic chip, the AL220. This month, he puts the chip through its paces using the PID-Pong machine. His goal: to automatically control the fan to levitate a ping-pong ball.

backup alarm), electronic tape measures, and the like.

The TI01 (see Figure 2) uses seven positions of a nine-pin connector. Notice that somewhere along the line, the TI01 pin assignment seems to have changed (this pinout is for a part number 615077).

Tip 1: Check your module's pin assignment carefully to avoid an ugly debug session.

Starting with power, the good news is the TI01 can handle anything within 4.5-6.8 V. The bad news is, when it pings, the TI01 calls for a whopping 2 A, which is not unexpected, considering it's really just a very tweety tweeteer demanding a hi-fi-like 10 W to be heard.

Tip 2: Make sure your +V is up to snuff.

Depending on the duty cycle (which shouldn't be more than about 10 pings per second to meet the module's 80-ms recycle spec), you can get away with a smaller power source using a suitable capacitor since the TI01 power drops to 100 mA when it isn't pinging (i.e., when it's listening for the echo).

The remaining five lines (INIT, ECHO, BINH, BLNK and OSC) compose the digital interface. Of these, only INIT (ping) and ECHO are mandatory in most cases.

For reference, the OSC output is a spare 49.4-kHz clock for external use, typically to drive a counter. BINH (Blanking Inhibit) causes early termi-

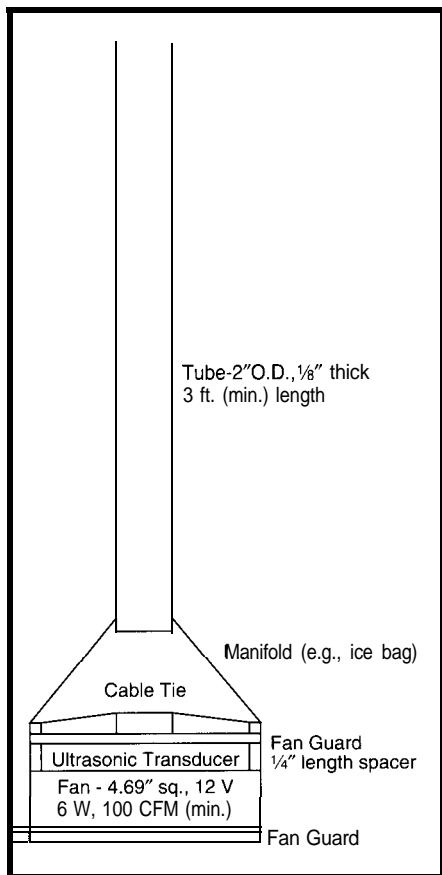


Figure 1—The P/D-Pong machine relies on an ultrasonic transducer and a 4" 12-VDC fan to test the mettle of would-be controllers.

nation of the TI01's built-in automatic blanking since the TIO 1 must suppress the input for a short time after transmission, lest it hear itself ringing.

The blanking interval dictates the minimum distance that can be measured, with about 16" being the de-

fault. Past experience indicates that BINH can be used. But in this application, it only cuts the minimum a little (to perhaps 12").

BLNK is associated with a multiple-echo feature, which clearly isn't needed or desired in the PID-Pong application.

Tip 3: Remember to ground BINH and BLNK if you aren't using them.

Most of the action centers around INIT and ECHO, and it couldn't be simpler—just drive the INIT input high and measure the elapsed time until ECHO goes high, as indicated in Figure 3. Well, actually it could be a little simpler, say along the lines of TTL compatible.

Turns out the ECHO output may need a pullup (e.g., 4.7 k Ω), depending on what it's connected to. How many hours have been spent debugging an otherwise perfect TI01 setup whose ECHO output was a few millivolts shy of a full 1? Too many, I know..

Tip 4: Don't forget the !@#\$ing pullup.

In fact, it's highly recommended you get a small TI01 test program or jig working to your satisfaction before you lash the whole machine together. Verify that distance changes are accurately detected and that there aren't any glitches or dropouts in the distance readings.

Tip 5: Ensure there's a good ground and cable between the TI01 and the controller to avoid false echoes.

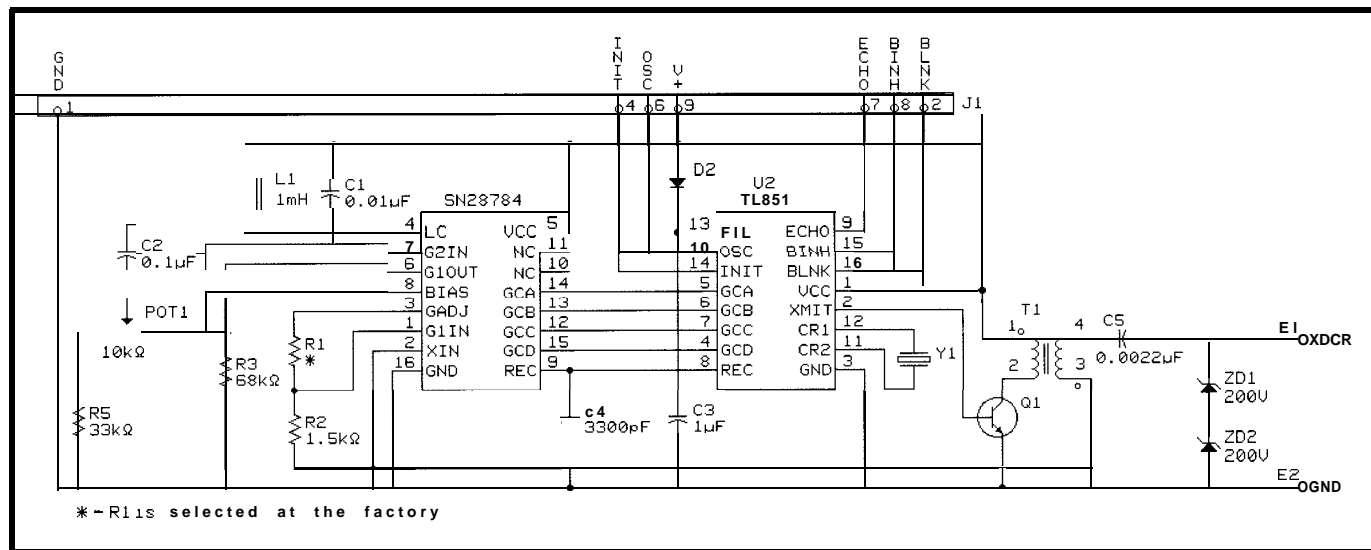


Figure 2—The TI01 handles the low-level details of accurate ($\pm 1\%$) measurement across an extended range (6"-35") including 49.4-kHz modulation, automatic gain control, and high-voltage (200V) transducer drive. Note that some modules have different pin assignments and connectors.

This is also a chance to verify your driver software. For instance, the test program I used indicated a distance reading glitch every now and then. At first glance, these blips appeared to be electrical problems like those associated with failure to properly heed Tips 4 and 5.

Fortunately, I noticed a periodicity not seen in the typical noise-related interference. It turns out that the real-time kernel on the SBC I used interrupts the test program to update its tic counter, a distraction dispensed with by disabling interrupts altogether.

Tip 6: There's nothing worse than debugging hardware (that probably actually works) with software that doesn't work, especially if you don't know it doesn't work. Vet your software thoroughly, so you don't stumble blindly if (more like, when) something goes wrong.

With a rock-solid TI01 interface, the only other critical step is positioning the sensor relative to the bottom of the tube. The dilemma, illustrated in Figure 4, is that the sensor starts to deliver false echoes if it's located too far below the tube.

On the other hand, if it's too close to the tube, the TI01 physically impedes air delivery, resulting in severe power loss and turbulence. Using your own test setup, verify the limits of TI01 positioning (typically about 1") and mount it accordingly.

Tip 7: TI01 positioning and alignment is critical.

Another aerodynamic foible explains the machine's open-frame construction. It turns out that the airflow is noticeably degraded by any obstructions around the fan.

Tip 8: Leave lots of breathing room for the fan, and why not throw in a finger guard for safety's sake?

Once you've got the machine completely assembled, go back to your previously proven test jig and verify

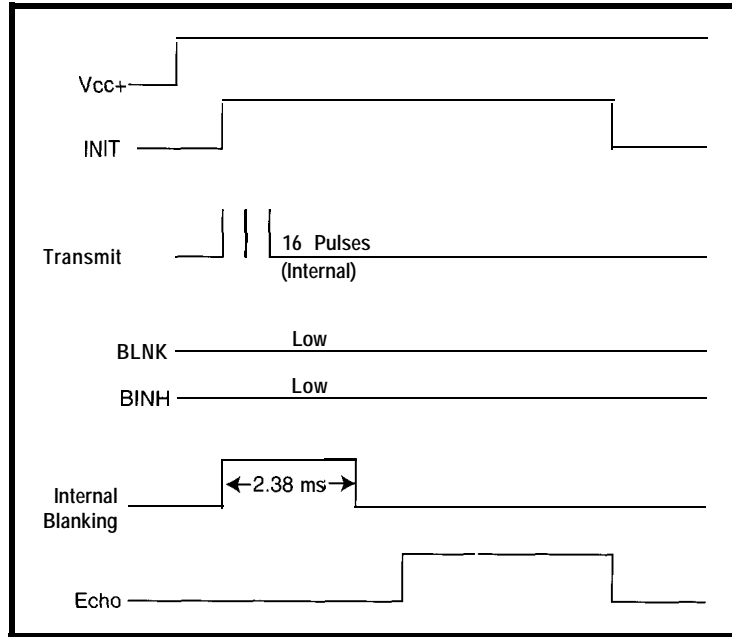


Figure 3—The simplest mode of TI01 operation exploits the built-in blanking feature to reduce the interface to two lines, INIT and ECHO.

everything still works. Hook up a variable supply to the fan and move the ball around to make sure the distance readings from the TI01 seem to correlate.

Notice how the automatic blanking makes the distance reading bottom out at about 16". It's also a good idea to place a flat object on top of the tube and then modify the test program to confirm that the distance reading never changes. Just let it run for awhile.

THE GRASS IS ALWAYS GREENER

With the PID-Pong machine up and running, it's time to turn our attention back to the AL220 and work up a hardware interface.

It doesn't take long before you're struck by a reversal of typical roles (i.e., we've got an analog processor that needs to talk to digital I/O!). I imagine a corporate version of musical chairs in which all the analog companies go digital at the same time the digital companies go analog, each believing the other guy must know something they don't.

Enlightened by this viewpoint, the first step is to reexamine assumptions (notably, a digital controller) underlying the original design decisions.

For instance, the earlier fan-control design (see Figure 5a) relied on a digital

PWM input to set the speed. Certainly, there's nothing controversial about that, and as described in the previous article, the AL220 can generate a PWM if necessary.

But, wait. It may be conventional wisdom these digital days to control a DC motor with a TTL pulse train, but didn't they once use dials? There must be an easy way for the 0-5-V AL220 output to control the 0-12-V fan.

One idea would be to try to operate the 2N-3055 power transistor used in the previous design in its linear re-

gion. These gadgets, though normally turned hard on or off, can work as an amplifier when driven with an intermediate value. This idea was filed due to concerns about resolution (i.e., the linear region may be quite small) and transistor-to-transistor variations in this typically unspecified parameter, but it hasn't yet been proven unworkable.

Another idea is to use op-amps and simply multiply the output voltage, perhaps adding an offset recognizing that the fan takes a good 5 V or so to start. As you know by now, I'm rather analog challenged and prefer to get by without fooling with a rat's nest of chips, Rs, Cs, trim pots, and so on.

Fortunately, Jim is as good with a transistor as he is with tools. He came up with the disarmingly simple solution in Figure 5b. He relies on a plain old variable voltage regulator, the LM350.

I'm not quite sure why the LM350, classically connected in a feedback configuration, works in this open-loop mode, but it does. Basically, it appears to drive the output about 1 V above the adjust input. Yes, converting 0-5 V to 1-6 V doesn't help much, but the next trick is to connect the fan's black lead to -5 V, instead of ground. Now we've got 0-5 V translated to 6-11 V, which works quite well.

A 12-V regulator input is used for convenient connection with a power supply. However, remember that from the regulator's point of view, the minimum output is about 1 V. Thus, at low-speed settings, the regulator sees a 10-V (or more) voltage drop and has to dissipate a few watts. Originally, a smaller regulator that looked like it might make it on paper was tried.

However, even with a heatsink, it got quite hot and would even go into thermal shutdown if the fan was left idling (i.e., $V_{adj} = 0$ V) for a long time. By contrast, the much beefier LM350 barely gets warm and probably doesn't even need a heatsink. Alternatively, a lower input voltage (say 9 V instead of 12 V) might resurrect the smaller-regulator option.

With success returning the fan to its analog roots, the same strategy was contemplated for the TI01. After all, what's really being measured is time, which—though I'm no Einstein—seems rather analog to me.

Knowing more about the TI01 than the AL220, I tried to come up with a

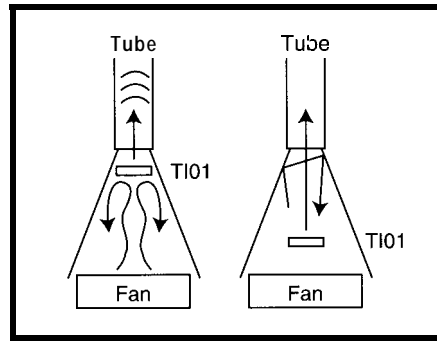


Figure 4—Transducer positioning relative to the bottom of the tube is quite critical. If it's too close, it chokes the airflow, but if it's too far, false distance readings start to occur.

way to give the AL220 exactly what it wants (i.e., 0-5 V). The classic lean-and-mean way to convert time to voltage uses an RC that starts charging on the leading edge of INIT and is sampled on receipt of echo.

However, at this point, I didn't want to impose any timing or synchronization burden on the AL220, fearing I would regret it later. What's needed is a way to decouple the TI01 and AL220 timing (i.e., provide a sample-and-hold capability).

The seemingly minor decision quickly led to a very slippery slope. It turns out S&H amps are rather arcane items, with prices to match (\$3-6 at Digi-Key). I may as well just use a counter and a DAC.

For that matter, how about a PIC processor? I could exploit its rail-to-rail outputs and a resistor network DAC. Gee! What about one of those newfangled analog PLDs? Come to think of it, why not just use another AL220 in a fuzzy multiprocessing configuration?

Talk about tail wagging the dog. I could see myself spending a lot more time and transistors just to get the TI01 and AL220 talking than in getting them to do something useful.

I finally recognized I'd been avoiding (perhaps, denying) the inevitable. Clearly, the optimal interface relies on extensive insight into the detailed operation of the AL220.

Staring at the pile of chip and tool documentation down, I knew what had to be done. Yes, it's a messy job, but somebody's got to do it.

PC Notebook Comscope™ Protocol Analyzer

- Menu Driven
- Time Stamping
- Small-Portable
- Travel Case
- Built-in Bert Tester
- Works with all PC's Via Serial Port
- Monitors & Emulates X.25, HDLC, Async & Bisync
- Split Screen Decode of X.25 & SDLC
- State Machine Programming
- Applications Interface Programmable



The Model 904 PC Notebook Comscope is the latest addition to Telebyte's line of powerful and flexible PC based protocol analyzers. The Model 904 is a high performance data line monitor, bit error rate tester and emulator in a small configuration that supports all PC's. The combination of the 904 and PC produces a powerful protocol analyzer.

Phone: 1-800-835-3298
(516) 423-3232
Fax: (516) 385-8184
e-mail: sales@telebyteusa.com
Internet: http://telebyteusa.com
270 Pulaski Road Greenlawn, New York 11740

TELEBYTE
TECHNOLOGY, INC.

REMOTE POWER CARD!

3 VERSIONS:

RESET

WATCHDOG RESETS PC IF IT HANGS FOR HARDWARE OR SOFTWARE REASONS

PHONE

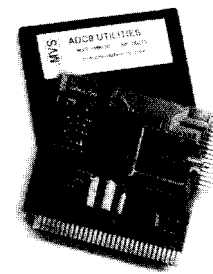
TURN ON PC WITH PHONE, SHARE VOICE / MODEM LINE, CONTROL AC APPLIANCES

TIMER

WAKEUP OR SHUTDOWN PC, LATE NITE BACKUP / MODEM, CONTROL AC APPLIANCES

95\$ 27\$ OEM

ALL MVS CARDS INCLUDE ASM, BASIC, AND C SOURCE FOR PC OR SBC



8 CHAN ADC

DATA ACQUISITION, SERVO CTL, AUDIO
8-BIT RESOLUTION 22KHZ SAMPLE RATE
SHARP CUTOFF ANTI-ALIAS FILTER
CREATE STEREO BLASTER (.VOC) FILES

95\$

2 CHAN DAC

VOICE MAIL, MUSIC, ALARMS, CTL VOLT
8-BIT RESOLUTION 44KHZ SAMPLE RATE
PLAYS MONO / STEREO BLASTER FILES
FUNCTIONS AS DIGITAL ATTENUATOR TOO

75\$

MVS

MVS BOX 850
MERRIMACK, NH
(508) 712 4507

5 YEAR LIMITED WARRANTY
PING IN USA

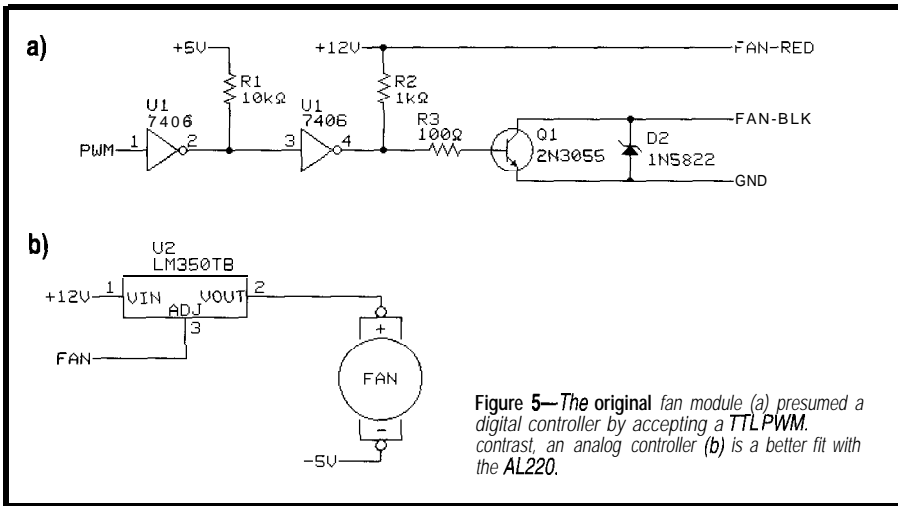


Figure 5—The original fan module (a) presumed a digital controller by accepting a TTL PWM. In contrast, an analog controller (b) is a better fit with the AL220.

And, it ain't me, I thought. Picking up the phone, I explained directly to the experts at Adaptive Logic. Could they help me finesse the interface a bit? Needless to say, I gladly accepted their offer of assistance.

Having successfully delegated all the hard work to others, the interface answers suddenly came to me. Yes, I see now that the only acceptable solution is to make the AL220 deal with

di-
 O So, just have the AL220 drive INIT, sample ECHO, evaluate the rules, and drive the fan all at once. It sounds good, except there's a small problem. The speed needed for good resolution (i.e., -50 us) also means the maximum loop delay, using the AL220 8-bit variables, is only about 12 ms. This rate is much too fast, notably show stopped by the TIO1's ~100-ms cycling spec.

TIME OUT

To start to get a feel for the AL220 and TIO1 interface, let's take a quick look at some of the basic timing and accuracy considerations.

The TIO1 specs typical absolute accuracy of $\pm 1\%$, which is about $\frac{1}{2}$ " given the length (36") of the tube. Sound takes about 0.9 ms to travel a foot, which is doubled to determine the echo delay. Thus, detecting a $\frac{1}{2}$ " change in ball position requires 75- μ s echo resolution.

Remember from **INK 67** how the AL220 runs in kind of a batch mode. All the inputs are sampled, rules evaluated, and outputs set in a 1024-clock chunk. Thus, the AL220 has roughly 50- μ s timing resolution running at top (20 MHz) speed. Quite fortuitously, it looks like the AL220 is speedy enough to resolve a $\frac{1}{2}$ " difference and fully exploit the accuracy of the TIO1.

So, just have the AL220 drive INIT, sample ECHO, evaluate the rules, and drive the fan all at once. It sounds good, except there's a small problem. The speed needed for good resolution (i.e., -50 us) also means the maximum loop delay, using the AL220 8-bit variables, is only about 12 ms. This rate is much too fast, notably show stopped by the TIO1's ~100-ms cycling spec.

The folks at Adaptive were clever enough to bring the RC delay idea off the bench for another go. In this case, the AL220 loops an output back to an input via an RC to establish a leisurely overall machine cycle time without giving up TIO1 resolution.

Putting it all together, Figure 6 and Photo 2 show the refreshingly minimalist AL220-based PID-Pong machine controller. Besides the previously discussed LM350 fan controller (with an added pot to limit maximum fan output) and cycle time RC network (R1 and C2), there's little more to the hardware than the usual clock and power-up-reset circuits.

One input is dedicated to a Manual/Auto switch. In Manual mode, the AL220 simply copies the input voltage on Setpoint to the Fan output with no regard for the actual ball position (i.e., open loop). This option is useful for calibrating and testing the machine and also trying to control the ball yourself to see how hard it is.

In Auto mode, the AL220 monitors and regulates the position of the ball based on the Setpoint input. Changing the voltage input on Setpoint exercises

*Under the hood of a stock
 Cimetrics RS-485 Protocol Stack ...*

*... plenty of software power to drive
 an embedded microprocessor network!*

- > Performance
- > Robust Implementation
- > Payback
- > Reliability

Exactly what you would expect from the leader in microcontroller networks for RS-485, BACnet™, Ethernet™ and ARCNET™ environments. Have your RS-485 network up and running smoothly in just *hours* – at a fraction of the development and support cost – with full support from the experienced team at Cimetrics. Please call, or FAX us at 617-350-7552.

Cimetrics • Boston, MA • 617-350-7550

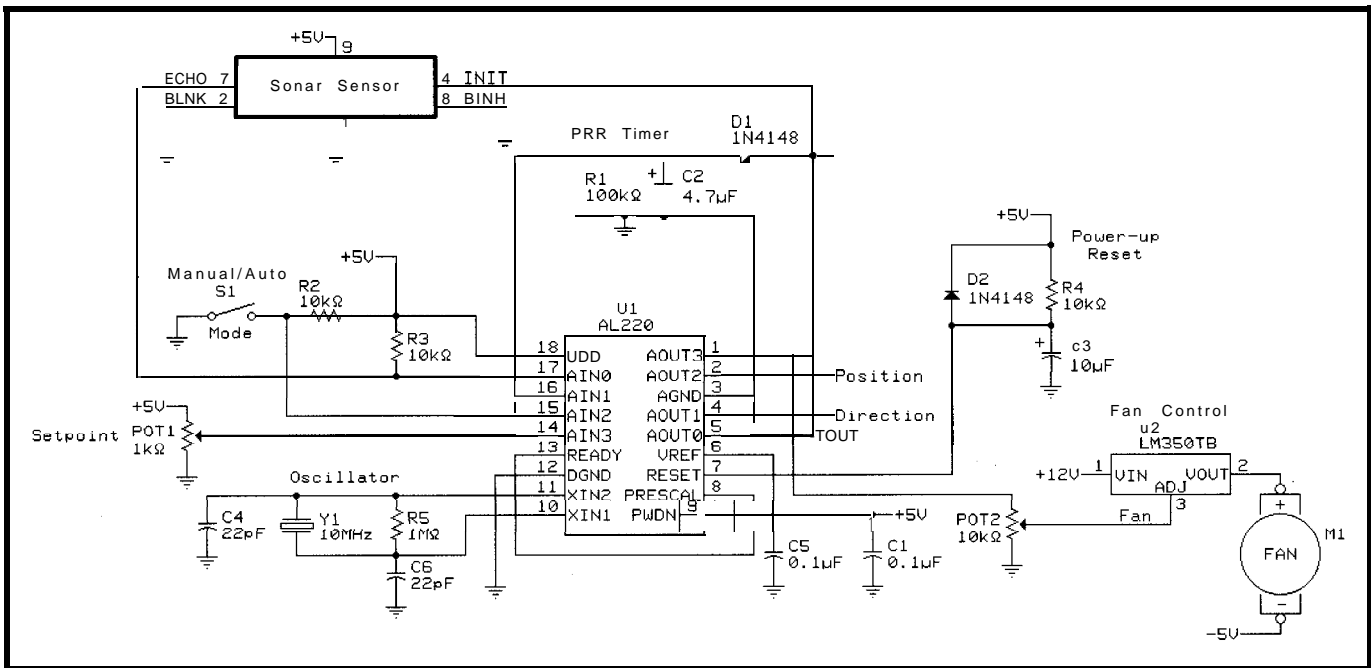


Figure 6—It's rather snug, but the AL220 proves up to the PID-Pong challenge.

the step response of the controller and machine.

Notice the Direction and Position outputs don't seem to go anywhere. This means these are being used as buried outputs (i.e., they're fed back internally). However, the signals are available on the pins providing a useful debug and instrumentation aid.

I think all must agree the design is quite compelling. Despite being coerced into talking on the TI01's terms (i.e., digital), the AL220 certainly appears to hold its own against any micro in terms of chip count and system cost.

Final judgment on whether the AL220 deserves a spot in the PID-Pong hall of fame depends on how well it plays the game.

PID-PONG PLAYBOOK

The simplest drill is just to move the ball to a position and hold it there. This move is rather easy—even a human can do it—especially with nothing said about the speed or accuracy of the movement. Needless to say, this minimal level of control must be mastered before moving on to advanced play.

To move the ball back and forth between setpoints ups the ante. It sounds easy, but a few attempts at manual control illustrate just how

tough it is. The machine exhibits severe lag (on the order of a second) and also an intriguing variety of nonlinearities, including gravity and mysterious aerodynamic influences.

However, even this challenge is manageable by humans (or a real simple controller), once you know the trick. The loophole is that nothing is said about speed and accuracy. The

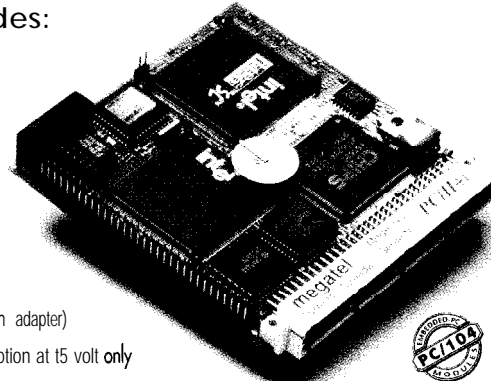
• WORLD'S SMALLEST

486TM

Embedded PC with
Hooting Point,
Ethernet & Super
VGA Only 4" x 4"

• The PC/II +i includes:

- 486dxTM CPU at 25MHz or 33MHz clock frequency
- Full 8K Cache with Floating Point
- Ethernet local Area Network
- Local Bus Super VGA Video/LCD
- Up to 2MBytes FlashTM with TFFS
- 4 or 16MBytes User DRAM
- PC/104 or ISA Bus compatible option (with adapter)
- 4" x 4" Format; 6 watts power consumption at 5 volt only



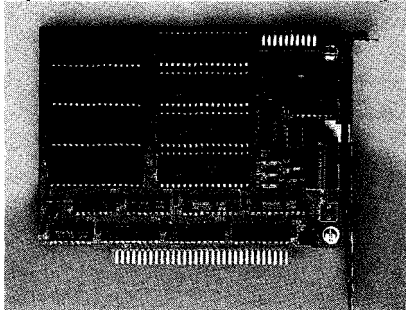
486DX and Flash are registered trademarks of Intel Corp. as are PC, AT of IBM, megatel of Megatel Computer (1986) Corp.

(416) 245-2953

megatel[®]

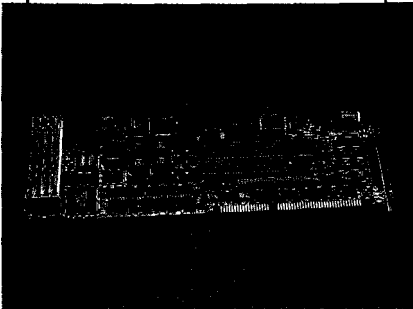
125 Wendell Ave. • Weston, Ont. • M9N 3K9 • Fax: (416) 245-6505

VMAX[®]
QUALITY PRODUCTS
RESPONSIBLE SERVICE
RELIABLE DELIVERY



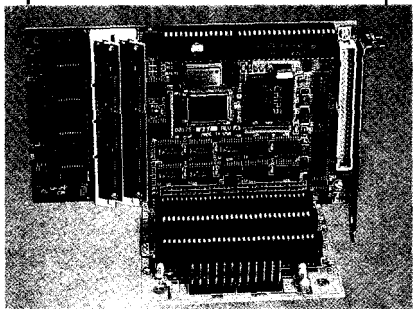
ENHANCED SOLID STATE DRIVE — \$144*

4M Total, Either Drive Bootable
 1/2 Card 2 Disk Emulator
 Flash System Software Included
 FLASH & SRAM, Customs too



486 SLAVE PC — CALL

Add up to 4 Boards to One Host PC
 Fast Data Transfer and I/O
 PC-104 Port, IDE & Floppy Control
 Independent Processors on One Bus
 No Special Compilers Needed



TURBO XT w/FLASH DISK — \$266*

To 2 FLASH Drives, 1 M Total
 DRAM to 2M
 Pgm/Erase FLASH On-Board
 CMOS Surface Mount, 4.2"x6.7"
 2 Ser/1 Par, Watchdog Timer

All Tempustech VMAX[®] products are
 PC Bus Compatible. Made in the
 U.S.A., 30 Day Money Back Guarantee
 *Qty 1, Qty breaks start at 5 pieces.

TEMPUSTECH, INC.
 TEL: (800) 634-0701
 FAX: (941) 643-4981

Fax for fast response! 295 Airport Road
 Naples, FL 33942

Photo 1—The latest and greatest version of the PID-Pong machine looks a lot nicer than the original (and doesn't use any tape).

generic strategy (whether human or electronic) is simply to govern the fan power high and low limits to a range that barely lifts or drops the ball.

The control strategy then devolves to a simple bang-bang [i.e., on/off) algorithm similar to:

```

IF POSITION <
  SETPOINT THEN
  FAN=ON
IF POSITION >
  SETPOINT THEN
  FAN=OFF

```

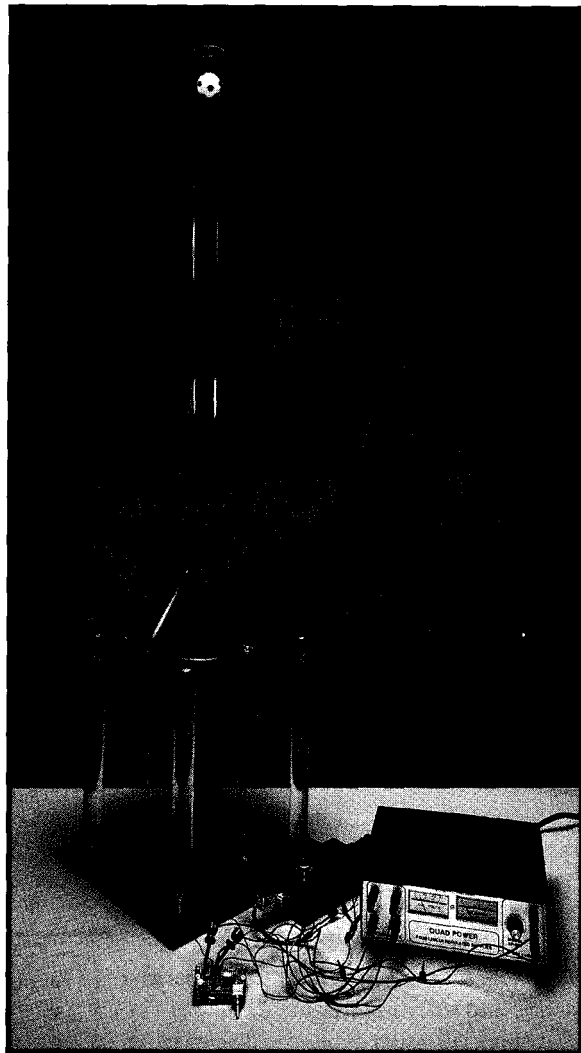
It's only by adding speed and accuracy requirements that the machine starts to challenge a controller and humble a human operator. For instance, try the following challenge:

Move the ball between **18"** and **30"** setpoints as fast as possible for **1** minute, achieving stability within $\pm 5\%$ for at least 3 seconds at each setpoint.

The score equals the number of moves completed.

My experience is that most mortals won't be able to do it at all, while an on/off controller might hit **1-2** times as it's limited by sluggish rise and fall times. By contrast, an optimal controller, able to move the ball quickly, could achieve perhaps 5-6 moves a minute.

Even this tougher test, like any benchmark, is arguably too simple and prone to manipulation and hijinks. The problem is the fixed high- and low- setpoint pattern. It's possible to optimize (i.e., cheat) by tweaking gain factors and fan limits based on knowing the setpoints a priori.



A more realistic test could substitute a random or other interesting pattern of setpoints for the fixed sequence of the previous test. The pattern length (i.e., test time) would have to be long enough to ensure statistical validity.

An interesting embellishment might explore the subject of adaptive control. For instance, the setpoint pattern could be repeated *n* times with the controller allowed to tune itself between trials. Comparing the score after *n* trials to that of the first would (hopefully) show the speedup due to learning.

But wait, it gets even better. We haven't even contemplated the interesting subject of disturbances yet. Disturbances are what make real-world control problems challenging, a reality the PID-Pong machine itself teaches. Nothing like admiring your finely (and time-consumingly) tuned

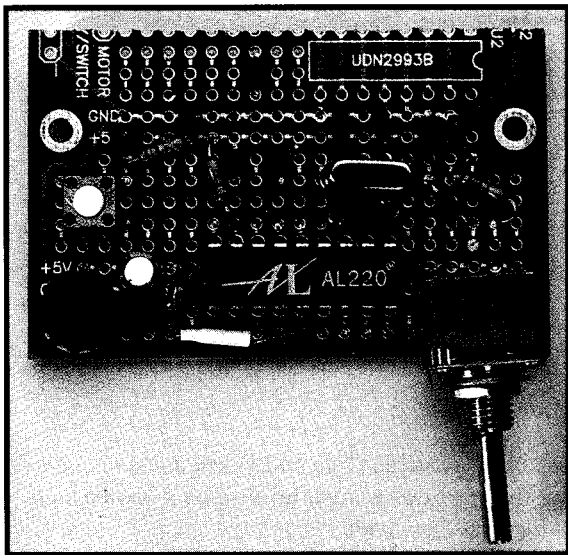


Photo 2—The AL220 controller board requires minimal external support components.

controller volleying back and forth, only to have the ball land in your lap when the air conditioning kicks on.

So, once you've got a controller you think passes the previous tests well, try introducing some disturbances. For instance, lay a pencil or ruler across the top of the tube to in-

troduce an offset. It's not uncommon to see a formerly well-tuned controller struggle or even fail to adapt to relatively minor changes.

Another torture test would be to insert noise of various sorts into the feedback or control loop. What the heck, why not goof up both loops and see how well your controller might work across a satellite link.

With all these ideas racing around, now is not the time to define any hard-and-fast rules for the PID-Pong game except one—have fun and learn something. That's just what we'll do next time when the AL220 finally takes on the PID-Pong machine.

Let the games begin! 🎮

Tom Cantrell has been working on chip, board, and systems design and

marketing in Silicon Valley for more than ten years. He may be reached by E-mail at tom.cantrell@circellar.com, by telephone at (510) 657-0264 or by fax at (510) 657-5441.

CONTACT

Adaptive Logic, Inc.
800 Charcot Ave., Ste. 112
San Jose, CA 95131
(408) 383-7200
Fax: (408) 383-7201
<http://www.adaptivelogic.com/>

SOURCE

TIO1 Ultrasonic Transducer
Micromint, Inc.
4 Park St.
Vernon, CT 06066
(860) 871-6170
Fax: (860) 872-2204

434 Very Useful
435 Moderately Useful
436 Not Useful

Does your Big-Company marketing department come up with more ideas than the engineering department can cope with? Are you a small company that can't afford a full-time engineering staff for once-in-a-while designs?

Steve Ciarcia and the Ciarcia Design Works staff may have the solution. We have a team of accomplished programmers and engineers ready to design products or solve tricky engineering problems.

Whether you need an on-line solution for a unique problem, a product for a startup venture, or just experienced consulting, the Ciarcia Design Works is ready to work with you. Just fax me your problem and we'll be in touch.

REMEMBER... A
CIARCIA DESIGN WORKS
FAX: (860) 871-8986

B.G. MICRO, INC. P.O. Box 280298
Dallas, Tx 75228
Orders Only 1-800-276-2206
Tech Support 214-271-9834
Fax 214-271-2462
Local Orders 214-271-5546

Our Internet address is BGMICRO@IX.NETCOM.COM

SUPER CONTACTOR **CABLE IT**

From IBM-3/6 Pole S.T.N.O. 115 Amp 30/50 Hp. contactor Telemechanique (Rowan Control) 2160BX760JA-6-29-47. Inductively rated to 600 VAC.

Features two stacked sets of 3-pole contacts. Contact pairs may be separated by removing connecting buss bars to provide up to 6 poles (reducing stacked pole rating by half). Or, parallel all 6 poles to make a S.P.S.T. unit for operation up to 345 Amps. One N.O. auxiliary contact included, with provision for another. Additionally, there is provision for mounting a micro switch. 24VDC coil with dual windings-6 Amp pull-in, >50 mA, holding, with pre-wired changeover switch. Coil may be changed (for different Voltage/frequency) by removing 4 screws from the bottom. These contactors are brand new. Nameplate ratings (as 3-pole contactor) are: 208/220 Volt-30 Hp., 440/550-50 Hp. Micro size measures 5-7/8"Hx4-1/4"Wx6"D (over terminals). Originally sold for \$901.00 each. You pay only... **\$44.00**

RE-CHARGE IT

That's right a 3 volt lithium coin cell rated 3 volt/50 mAh, 20.5mm mounting. Compare at \$7.85 each. Great price. While they last... **\$1.99**

Nickel Hydride-Used on many computer boards. 3.6V/60mAh. Measures 1/2"Dx3/4"L. Mounting 3/4". Steal'em... **\$2.95**

Nickel cadmium 7.2V/100mAh 1/2"Dx3-1/2"L... **79 or 3/\$2.00**

And finally gell cell. Heavy duty, 6V/33Ah. Weighs 12 pounds and measures 6-1/4"x3-1/4" and is 6-1/2" tall. New Price **\$17.50 each**

SWITCH IT

Switching Power Supply #1

A full 220 Watts of awesome computer power Completely enclosed w/fan - Manufactured by Level -UL & CSA Approved.

Input: 110/220 Vac
Output:
+5 Volts @ 23A.
-5 Volts @ 500mA.
+12 Volts @ 8.5A.
-12 Volts @ 500mA.
Measures: 8-3/8"x 5-7/8"x5-7/8" **\$17.99**

TERMS: (Unless specified elsewhere) Add \$4.00 postage, we pay balance on Orders Under 5 lbs. Orders over \$50.00 add .85 for insurance. No C.O.D. Texas Residents add 8 1/4% Tax. 90 Day Money Back Guarantee on all items. All items subject to prior sale. Prices subject to change without notice. Foreign Orders U.S. Funds Only. We cannot ship to Mexico or Puerto Rico. Canada, add \$7.50 minimum shipping and handling. Countries other than Canada, add \$15.00 minimum shipping and handling.

Call or write for a copy of our free catalog

CONNECT TIME

conducted by Ken Davidson

The Circuit Cellar BBS
300/1200/2400/9600/14.4k bps
24 hours/7 days a week
(860) 871-1988—Four incoming lines
Internet E-mail: sysop@circellar.com

In our first discussion this month, we look at a topic familiar to many, but a mystery to others: Gray codes. Why are they applied and how are they generated? Read on to find out.

In the other thread, we compare the use of shunts to Hall-effect devices for measuring high currents. It's anything but cut and dry.

Gray code

Msg#: 4172

From: Dave Ewen To: All Users

Anyone spend much time fooling with Gray code? I'm wondering what the simplest logic is for generating it. I'd like to fit it in a high-speed PAL and run it like a counter.

Msg#: 4308

From: Pellervo Kaskinen To: Dave Ewen

The following presentation is picked pretty much directly out of Hamming: Numerical Methods for Scientists and Engineers, second edition, ISBN 0-07-02887-2.

Building Gray code recursively

1-bit	0	
	1	
2-bit	00	i.e., 0 & 1st term of 1-bit
	01	0 & 2nd term
	11	1 & 2nd term
	10	1 & 1st term
3-bit	000	i.e., 0 & 1st term of 2-bit
	001	0 & 2nd term
	011	0 & 3rd term
	010	0 & 4th term
	110	1 & 4th term
	111	1 & 3rd term
	101	1 & 2nd term
	100	1 & 1st term

Hamming gives the example to 4-bit level, but I believe the process is clear enough from the description at the right of each term, with just up to 3-bit level. The 4-bit level of course contains 16 terms, eight of which are obtained by padding with a leading 0 and the new eight by padding with a 1 and taking the terms in reverse order as I indicated above.

There are other possible Gray codes, but because the rule of obtaining this particular one is so simple, it has become to be known as *the* Gray code.

I hope this covers your needs.

Msg#: 4316

From: Dave Ewen To: Pellervo Kaskinen

Thanks. Doesn't look like something that can be done in a PAL too easily, though. I received a suggestion from a friend that I split it up into nybble-wide Gray-code generators-each with its own count enable-and then be a little tricky with the enables to step them only one at a time.

Msg#: 4353

From: Rufus Smith To: Dave Ewen

For lurkers (and participants) out there who may be wondering what Gray code is and why it's useful, file this away under interesting tidbits.

The beauty and usefulness of Gray code is that only one bit changes at a time between successive values. This is extremely important in the world of position encoder devices which detect the position of a mechanism linearly or rotationally.

The significance is in the transition between values where a bit change (edge) could be read as either on or off. In straight binary code, for example, going from 3 to 4 is a single step, but all three bits are in transition: 011 to 100. This means that your encoding pickup is on three edges, each of which could pick up either 1 or 0, potentially reading any value from 000 to 111.

Binary	Changed Bits	g e d	B i t s
000	—	000	—
001	0	001	0
010	1,0	011	1
011	0	010	0
100	2,1,0	110	2
101	0	111	0
110	1,0	101	1
111	0	100	0
rollover to 000	2,1,0	000	2

Gray Code, having only one bit in transition at any edge, always gives a correct reading within one significant bit.

CONNECTIME

Incidentally, anyone who has worked with quadrature or incremental encoding has been working with 2-bit Gray code.

There are several puzzles (the hand-held ring/string/rod manipulation puzzles) that are solved through the use of Gray code.

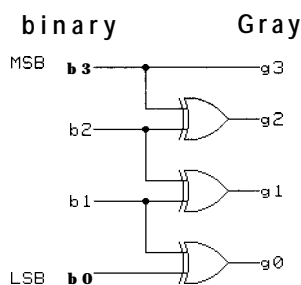
Also the infamous Towers of Hanoi puzzle is also an example of Gray code at work. The size of the disk to move is the bit that changes.

I think there is a conversion routine which involves successively XORing adjacent bits low to high or high to low, but I don't recall it.

Msg#: 4360

From: James Meyer To: Dave Ewen

Here's a binary-to-Gray converter.. .



Note that you will still have to worry about glitches on the Gray code side because the binary-side bits change more than one at a time with some counts. A D-type flip-flop latch per bit with a gate derived from the count input fixes that, though.

Do your counting in binary and convert that to Gray code. I'll bet you'll be able to fit that into a PAL or two.

Msg#: 4455

From: Dave Ewen To: James Meyer

Hey thanks! Simple XORing of adjacent binary bits seems to work fine! That's something I didn't know. Where did you run into that?

Msg#: 4467

From: James Meyer To: Dave Ewen

At the risk of exposing myself as *not . all-powerful and all-knowing, I must admit that I pull a lot of excellent stuff out of *The Art of Electronics* by Horowitz and Hill. The first edition has a catalog number of ISBN 0-521-23151-5. There's a newer second edition out now (ISBN 0-521-37095-7).

If I could only have one book, it would have to be *The Art*. You should not rest until you have found a copy.

Hall effect vs. Shunt

Msg#:14443

From: Mike Mager To: All Users

I want to measure current in an automotive application up to ± 255 A with 1-A resolution. I have tentatively chosen Hall-effect transduction over resistive-shunt methods for the following reasons:

To keep the voltage drop low at high current, a low-resistance shunt is necessary (no problem, Stuart-Warner uses 3-m Ω and 650- $\mu\Omega$ units, and possibly lower). A 1-m Ω shunt gives 1 mV/A and would require gain to drive the PIC's ADC, a higher-resolution lower-reference-voltage ADC, or something like that (true?).

A copper or aluminum shunt has a temperature coefficient of resistance of approximately 40% in 100°C, although NiCr or constantan are better.

To use a shunt, with the shunt in the ground return, the PIC's lower reference (Vss) would be connected to the low end of the shunt, and to accurately read several shunts would require make-before-break switching of the system's Vss from shunt to shunt. With the shunt in the high side, I'd need an op-amp with a positive supply above the system's battery voltage (true?). The shunt idea seems impractical here.

So, then, the Hall-effect approach..

I need to sense several points, in each of several vehicles, and a Hall-effect device would seem to allow lossless and floating sensing, needing only a small amount of gain to drive an ADC.

Does anybody have suggestions about a source of sensor units!

Does anybody have comments on building sensor units using a slit-core and analog Hall-effect devices? I saw something in an old *Popular Electronics* (I think it was, temporarily lost now) using microswitch Hall-effect devices, a slit core, and calibrated gain.

Does anybody know about a dedicated absolute-value circuit, or do I need to use an active full-wave rectifier with a comparator for sign?

Relative to this project are tow trucks with electrically driven hydraulic systems. They use about 225 A at the relief valve setting, BCI group SD batteries, 1300 CCA (400 minutes reserve), and Leece-Neville 2800JB 160-A alternators. The alternators are typically used on Cummins engines in Kenworth trucks, fire trucks, school buses with wheelchair lifts, and similar applications. I overhauled them and adapted them to the gasoline engines in the tow trucks.

(Oh, yeah. This is to be used with a PIC16C74, having an 8-bit ADC, 5-V referenced.)

Thanks!

CONNECT TIME

Msg#:14541

From: Lyndon Walker To: Mike Mager

A trip through the yield around \$10 in small quantities, would be site-value circuits. But looking in Analog Devices' a I the AD737 RMS-to-DC converter suggests that it can be used as an absolute-value output from a DC input.

Msg#:16181

Thanks for taking time to find an absolute-value circuit! I hadn't thought of the RMS converter what it does. Sadly, it doesn't have the sign output, so it's just an expensive rectifier sorts of stuff that was almost, in the Maxim books. If you're

Msg#:19082

From: Pellervo Kaskinen To: Mike Mager

There are several possibilities, some even beyond what you have mentioned so far. But, I'll start with the items you presented.

A commercial shunt is likely to be defined at a fixed, full-scale voltage level of 50 mV, whatever its current rating. This is due to the fact that any analog meters pretty much had to be standardized. Actually, in Europe a 60-mV standard emerged, while the U.S. trend setters [Weston, GE, and others) chose 50 mV.

Anyway, if you go to an electric supply house, such as GE supply, Graybar, or several others that are more localized, you can buy almost any shunt you want, with some more popular ones off the shelf. A full scale of 255 A, though, is a far call still today. More likely, you would find 200-A and 300-A units. But no problem with your 225 A, in most cases: Just use a 200-A unit. They are very robust and the 25 extra amperes won't harm them any time soon. Of course, you have to provide adequate ventilation around the shunt or the accuracy would fall out of the usual 0.25% spec.

Some companies, such as Empro, make shunts to your specifications. Instead of the 50 mV, you can spec 75 mV or 100 mV if they better fit your needs. The higher voltages naturally mean higher power dissipation. That again tends to translate into larger physical size. At the suggested 200-A level, the power is 10 W, 15 W, or 20 W for the 50-, 75-, and 100-mV units, respectively. At your 225 A, it would be about 20% more.

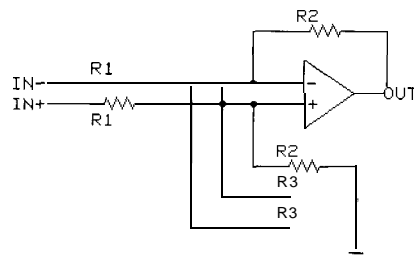
Contrary to your belief, no extra power supply above the positive rail would be necessary in several schemes, even when the shunt is on the high side. This does not mean the

extra power supply idea should not be considered. In fact, an isolation power supply from Burr-Brown would yield around \$10 in small quantities, would be

You could use an isolation amplifier. Those amplifiers have their own power isolation and could be using optocouplers. You could build your own for 30-100 bucks from components available from

linear device based on a Siemens chip (I forget the exact name something like VL6-300). but, yeah,

Another option would be the signal conditioning module in the smyncat. It's a Burr-Brown 5411 series from Analog. Another option would be the low-cost option: Use an extended common-mode differential amplifier. Burr-Brown sells their model INA117 that operates from ± 15 V and has a common-mode range up to ± 200 V, while providing a unity gain for the differential input to the single-ended output. To provide a good enough common-mode rejection, they laser trim the resistors inside, but I use plain 1% resistors in my designs with adequate results. Here is the basic circuit:



The INA117 has essentially $R2 = R1$. But if you want to get gain, you can make $R2 > R1$, with $R3$ affecting only the common-mode range.

Say, you have $R1 = 10$ k Ω , $R2 = 100$ k Ω , and $R3 = 1$ k Ω , to produce a gain of 10 and a common-mode range of at least 100 V. Maybe I need to state that the 10-k Ω input resistors might not be too happy about the 100 V, unless you use at least 1-W resistors. But, the actual resistance values can be scaled by any factor to accommodate the power dissipation requirements.

To avoid a need for trimmer potentiometers, the resistors should be as accurately matched as you can make them. The easy way would be to pick 0.1% resistors that offer a 60-dB common-mode attenuation off the box. Otherwise, get 1% resistors and find matched pairs with a 4 $\frac{1}{2}$ -digit DVM or other resistance meter.

You can also use a bridge technique, where you at first find four resistors that produce a balance no matter what order the four elements are combined. Then use two of

CONNECT TIME

these resistors as a basis for a permanent bridge to measure any other resistors. Put the other two away for any future replacement need.

Then to the Hall-effect front..

While you were concerned about the thermal stability of copper as a shunt material, the Hall-effect devices are worse! They really make quite sensitive temperature-measurement devices, with a secondary use for magnetic-field detection. :-)

The way out of that dilemma, of course, can be using two of them in a half bridge. One would be mounted so the magnetic field does not affect it, while the second one would be in the gap of the core. That corrects the bulk of the sensitivity shift by temperature. It still leaves a major nonlinearity. The way out of both the temperature and the linearity problems is to make a feedback arrangement.

A Hall-effect current-measuring device with feedback is based on the idea that the Hall-effect sensor is operated at zero flux. Any deviation from zero is amplified and fed to a secondary winding on the same core. If the primary winding is 1 turn and the secondary winding is 5000 turns, then a 250-A primary flux would be fully canceled by 50 mA at the

secondary. Then, all you need is to measure the 50 mA. A 100-G resistor would generate 5 V for your PIC ADC.

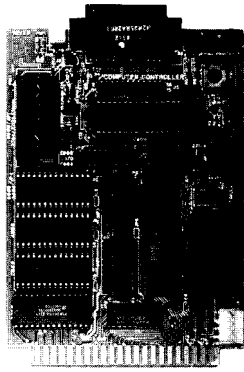
You need a power op-amp to force the 50-mA current, but it is regularly done. To my understanding, the first source for such devices was LEM. Now also Honeywell and F.W. Bell offer feedback-based current transducers.

Now, a few words about some other possibilities. You can measure DC currents with a device called a *transducer*. It is based on the DC primary shifting the two saturation points of two mumetal cores that are driven by AC excitation. Again, the DC primary is a single turn, while the secondary consists of thousands of turns on each of the two cores. The secondaries are connected in series, but in opposite directions.

As long as there is no DC, they offer a high impedance to the AC. But when the saturation points change, the AC excitation starts producing current. In fact, it is pretty straight proportional to the DC current.

I have used them in the past, but now find them cumbersome and only applicable when the currents get to the >100,000-A range. (Imagine the power dissipation of a 50-mV shunt at 100 kA!)

BCC52 BASIC-52 COMPUTER/CONTROLLER



The BCC52 controller continues to be Micromint's best selling single-board computer. Its cost-effective architecture needs only a power supply and terminal to become a complete development system or single-board solution in an end-use system. The BCC52 is programmable in BASIC-52, (a fast, full floating point interpreted BASIC), or assembly language.

The BCC52 contains five RAM/ROM sockets, an "intelligent" 2764/128 EPROM programmer, three 8-bit parallel ports, an auto-baud rate detect serial console port, a serial printer port, and much more.

PROCESSOR

- 80C528-bit CMOS processor w/BASIC-52
- Three 16-bit counter/timers
- Six interrupts
- Much more!

Input/Output

- Console RS232 - autobaud detect
- Line printer RS-232
- Three 8-bit parallel ports

- EXPANDABLE!
- Compatible with 12 BCC expansion boards

MEMORY

- 48K RAM/ROM, expandable
- Five on-board memory sockets
- Either 8K or 16K EPROM

To ORDER Call 1-800-635-3355

TEL: (860) 871-6170

FAX: (860) 872-2204

NEW LOWER PRICES!

BCC52	Controller board with BASIC-52 and 8K RAM	\$179.00	Single Qty.
BCC52I	-40°C to +85°C industrial temperature version	\$204.00	
BCC52CX	Low-power CMOS, expanded BCC52 w/32K RAM	\$199.00	

CALL FOR OEM PRICING

MICROMINT, INC. 4 Park Street, Vernon, CT 06066
In Europe: (44) 1285-658122 • In Canada: (514) 336-9426 • Distributor Inquiries Welcome!

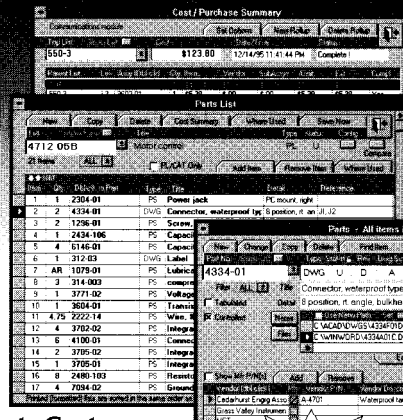
Designing Something? This Parts List Software will help you stay organized.

For Engineers,
Product Designers
and Prototypers

Easily create
and manage
multi-level
parts lists for
products in
development.

Keep track of:

- Part Specs
- Drawings
- Suppliers
- Product and Parts Costs
- Engineering Stock



V1.5

Parts Vendor\$™

for independent and departmental projects. Use alone or in a workgroup.

Version SE: \$99 + shpg Call 800-280-5176

Extended: \$299 + shpg
Requires Win 3.x or Win95, 486, 8 meg min. ram.
Trilogy DESIGN 916-273-1985
916-477-9106
P.O. Box 2270, Grass Valley, CA 95945

CONNECTIME

Sometimes you may not need very complicated DC measuring at all. If you can put a standard AC current transformer on the AC leg before the rectifier (in the alternator), you get a good representation of the DC current. Actually, if you put in a current transformer on all three AC legs to the rectifier and then put a three-phase bridge rectifier to the output of the three current transformers, you get an exact representation of the DC. It is scaled by the transformer turns ratio. Connecting the necessary (compulsory) load resistance to the DC leg of the secondary rectifier eliminates the diode drop effect of the secondary rectifier.

You can scale the output voltage by selecting the load resistance value. But you have to make sure the transformers are not saturated by the imposed load voltage. Also, many alternators use a three-phase half bridge as the main rectifier. Such an arrangement is not well suited for the current transformer method described above.

This list is by no means exhaustive, but I hope you find it thought provoking. After all, even though your description of the application needs was quite thorough, there still may be details that I do not know. So the final choices remain yours.

Msg#:19262

From: Mike Mager To: Pellervo Kaskinen

Part of my requirement was for the transduction to be low enough priced that several points on each of several tow trucks could be measured. These would be battery current, alternator current (positive only), and hoist motor current (negative only).

I would like to get $\pm 5\%$ to $\pm 10\%$ accuracy, in a practical sense, not in a laboratory sense. I'm with you on the shunts, as far as power dissipation, voltage drop, and so on, and the compromise would be between low voltage drop and low required gain. I'm concerned about noise with a high gain.

The feedback method of linearizing a Hall-effect device seems interesting! I could build them, I think. Have you any references on the practicalities?

Msg#:24059

From: Pellervo Kaskinen To: Mike Mager

I am not sure, but I might find some useful info from LEM documents. I'll take a look. On the other hand, the F.W. Bell parts are quite low in price.

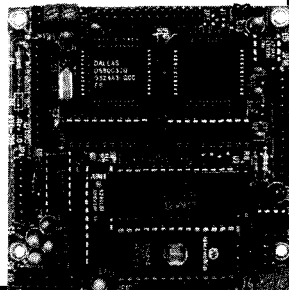
BLAZING SPEED RTC320 AND SMART TOO!

One of Micromint's hottest-selling products for the past five years has been the RTC31/52 stackable controller. It has been a leading price/performance choice among our customers. With our new RTC320 board, we have expanded the value of that relationship even more.

Occupying the same small 3.5"x3.5" RTC footprint and using S-V-only power, the RTC320 uses the new Dallas Semiconductor 80C320, which is 8031 code compatible and 3-5 times faster. At 33 MHz, the RTC320 is an 8-MIPS controller! Along with the new powerful processor, the RTC320 board accommodates up to 192 KB of memory, two serial ports (RS-232 and RS-485), 24 bits of TTL parallel I/O, and a 2-channel, 12-bit ADC. The RTC320 puts some real firepower under the abundant variety of RTC I/O expansion boards. Plugging in your favorite ICE or EPROM emulator is the easiest way to develop code. For the diehards who like to twiddle the bits directly, we have a ROM monitor specifically designed for the Dallas '320.

RTC320-1 (22 MHz)
\$239 \$179/Qty.100

(Call for pricing on 33 MHz)



CALL 1-800-635-3355 TO ORDER
MICROMINT, INC.

4 PARK STREET • VERNON, CT 06066 • (860) 871-6170 • FAX (860) 872-2204

#119

We invite you to call the Circuit Cellar BBS and exchange messages and files with other Circuit Cellar readers. It is available 24 hours a day and may be reached at (860) 871-1988. Set your modem for 8 data bits, 1 stop bit, no parity, and 300, 1200, 2400, 9600, or 14.4k bps.

ARTICLE SOFTWARE

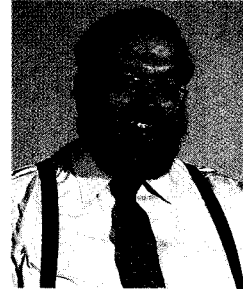
Software for the articles in this and past issues of *Circuit Cellar INK* may be downloaded from the Circuit Cellar BBS free of charge. It is also available on the Internet at <ftp://ftp.circellar.com/pub/circellar/>. Web users should point their browser at <http://www.circellar.com/>. For those with just E-mail access, send a message to info@circellar.com to find out how to request files through E-mail.

For those unable to download files, the software is also available on one 360 KB IBM PC-format disk for only \$12. To order Software on Disk, send check or money order to: Circuit Cellar INK, Software On Disk, P.O. Box 772, Vernon, CT 06066, or use your Visa or Mastercard and call (860) 8752199. Be sure to specify the issue number of each disk you order. Please add \$3 for shipping outside the U.S.

437 Very Useful 438 Moderately Useful 439 Not Useful

PRIORITY INTERRUPT

Not Just Ferengi Values



there are two ways to learn things in life: trial and error (experience) or by what someone tells you (education). Your behavior is also affected by what you've learned. If you grew up always thinking that wearing the color green means you're Irish, then as a Russian or Greek, you'd probably avoid green. Similarly, if you grew up on the impoverished side of town and never saw an upwardly mobile role model, you might resign yourself to a life with few options.

For many young people today, film and broadcast media have become primary education and information sources. Unfortunately, in my opinion, these media people view their influence and effect with about the same responsibility as those selling a box of Tide. However, because this influence is real, it is important to challenge these obvious biases.

One area where I believe film and broadcast media are biased is in their negative attitude toward business. Cloaked in dramas or comedies, businesses are often equated with dark escapades, indiscretion, and greed. Only when observed in historical perspective are they correctly viewed as having contributed to the potency and prestige of the United States' industrial power. Other than that, business is a four-letter word.

I'm not disputing that some corporations appear to treat employees as cannon fodder. I'm merely criticizing the media bias which results from their obsession with these "business detractors." Such an attitude shouldn't be applied to all situations involving a business entity.

The Ferengi value system offers definitive rules to deal with the best and worst aspects of capitalism. It seemed kind of ironic that a recent episode of *Star Trek: Deep Space Nine* would, while presenting business in a detrimental light, offer as its primary malevolence, that which perpetuates its positive existence.

In a dispute over working conditions in Quark's bar, the issue of collective action was presented as a suggested maneuver. Of course, the Ferengi employees were initially appalled at the concept. Under their value system, the control Quark was dispensing they would also seek to command if they were in his place. Undermining the process that enables Quark to dominate merely undermines their potential for ever achieving equivalent authority.

In a society which has only business to support its continued existence, we delude ourselves to think otherwise. Government doesn't make money, it only spends money. It is the positive relationship between people working for businesses that produce products and the resulting wages they spend buying products that creates money in an economy.

The realization is that, unless you are supported by government, you are sustained by business. The engineering consulting you do is a business product; the embedded controller you're hoping to market is a business product; to feed your family, the job you have is in a business.

I apologize if this sounds like preaching. Perhaps it's people like me who have started businesses who feel the greatest need to protect a system that offers such opportunity. Describing something always in negative terms is hardly incentive for others to join the cause. It's time to balance the equation with a little experience...Flame off.

steve.ciarcia@circellar.com