#165 April 2004

CIRCUIT CELLAR®

www.circuitcellar.com

THE MAGAZINE FOR COMPUTER APPLICATIONS

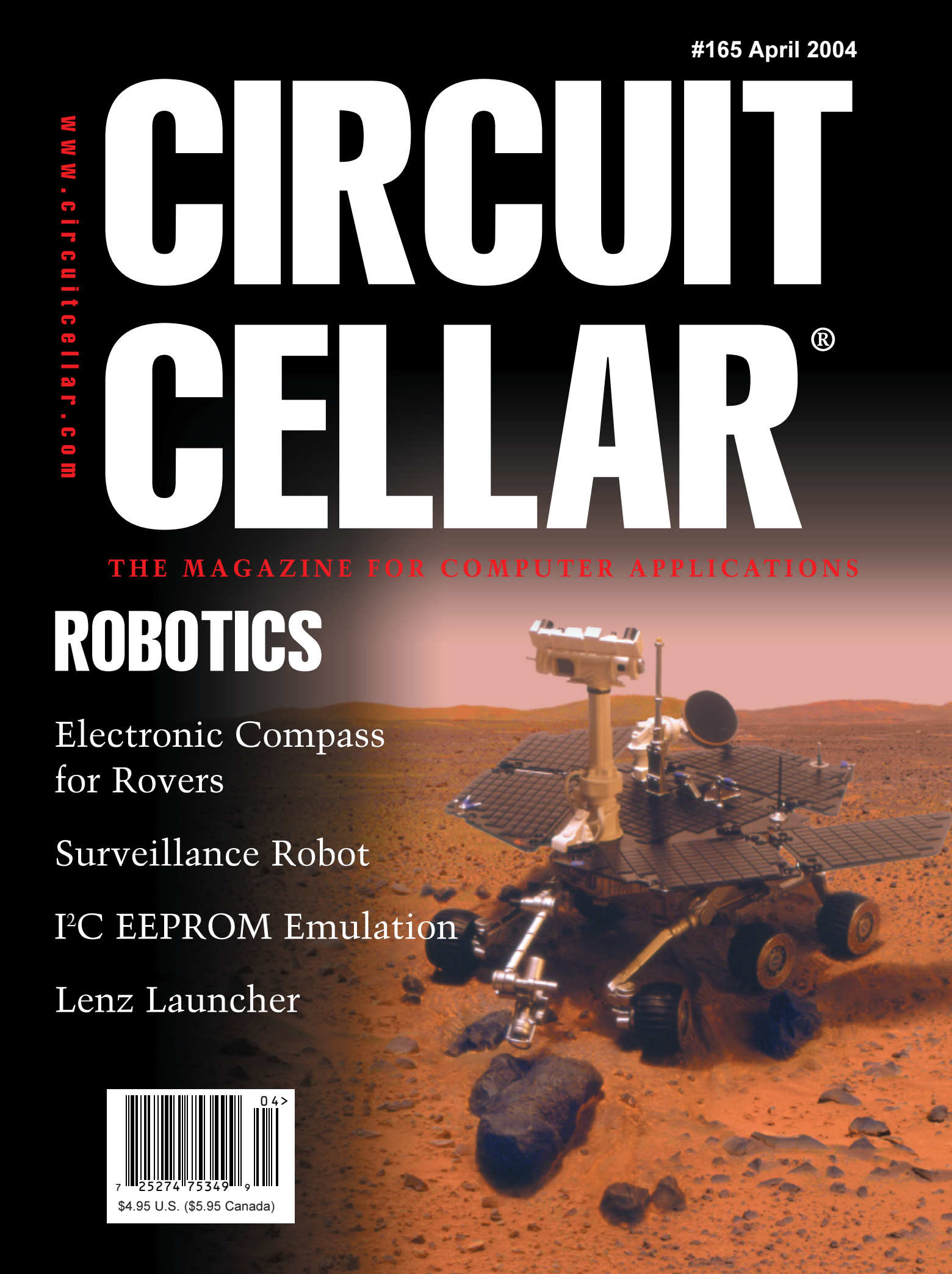
ROBOTICS

Electronic Compass for Rovers

Surveillance Robot

I²C EEPROM Emulation

Lenz Launcher



04>
7 25274 75349 9
$4.95 U.S. ($5.95 Canada)

NetBurner's Complete Solution
What's in your development kit?

NetBurner
is a single source for hardware, software, development kits, tools, technical support and custom design services that will speed development of embedded ethernet devices.
Concentrate on developing your product instead of reinventing network connectivity.

Development Kits $499

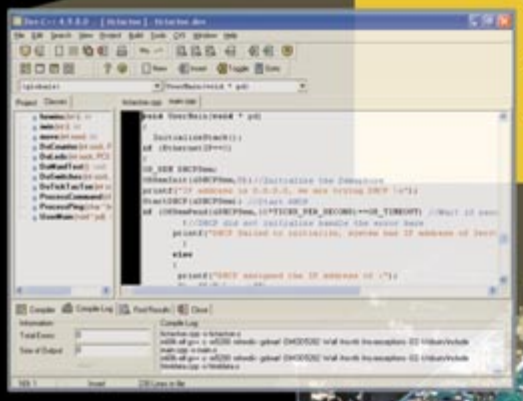
Modules from $72

The NetBurner Solution includes all the following components:

- Deployment Tools
- Real Time OS
- TCP/IP Stack
- Web Server
- C / C++ Compiler & Linker
- Integrated Development Environment
- **Your Product**
- Graphical Debugger
- Hardware Designs
- PPP
- E-Mail
- SSL*
- SNMP*
- Wireless* 802.11b
- Flash File System with Flash Card Support*

KEY BENEFITS: Tested. Integrated. Ready on day one!

- NetBurner hardware and software is fully tested and ready for your application.
- No Porting! The integration effort has already been done and all components work together perfectly.
- You can be up and running the day you recieve your kit.





1-800-695-6828 • www.netburner.com

* Optional modules



NetBurner

Networking in 1 Day!

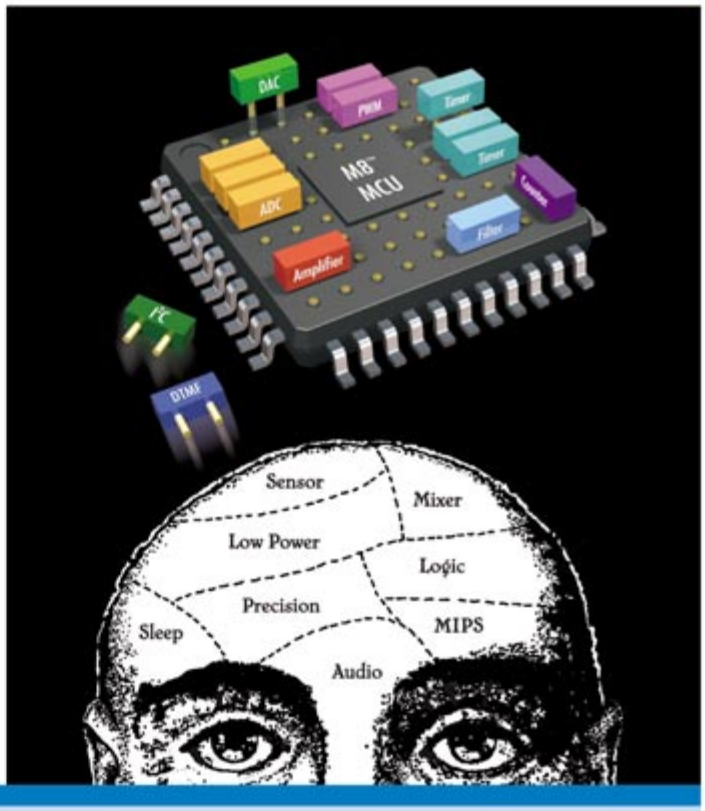
30% OFF
All PSoC Items in Cypress Online Store
www.cypress.com/ad/psoc-cea1

PSoC™ Mixed-Signal Array.
It ll change the way you think about embedded design.

Now with instrumentation-quality programmable analog.

Powerful new programmable analog and digital blocks with memory and MCU for less than $2. Winner of the EDN Innovation of the Year award, our PSoC™ Programmable System-on-Chip™ device is changing the face of embedded design.

- Replace 1,000s of fixed-function devices with a couple of keystrokes
- Dynamically reconfigure a PSoC device, changing functionality on the fly in any application
- Select from hundreds of predefined blocks in our mixed-signal library
- Already designed into 1,000s of applications; check out our online app note library



Reduce board size and BOM up to 80%

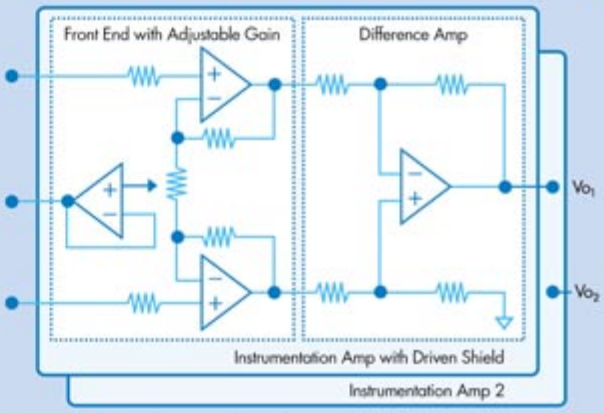


PSoC Mixed-Signal Arrays with M8 Microcontroller

	CY8C27X	CY8C24X	CY8C22X
CEA ANALOG BLOCKS	12	6	3
DIGITAL BLOCKS	8	4	4
HI REL SONOS FLASH	16K	4K	2K
SRAM	256	256	256
COST	low as $1.99	low as 99¢	low as 69¢

CYPRESS ENHANCED ANALOG™ (CEA™)

• Rail-to-rail analog • Instrumentation amps • Lower voltage offset
• Lower input leakage currents • Programmable gains • Better stability



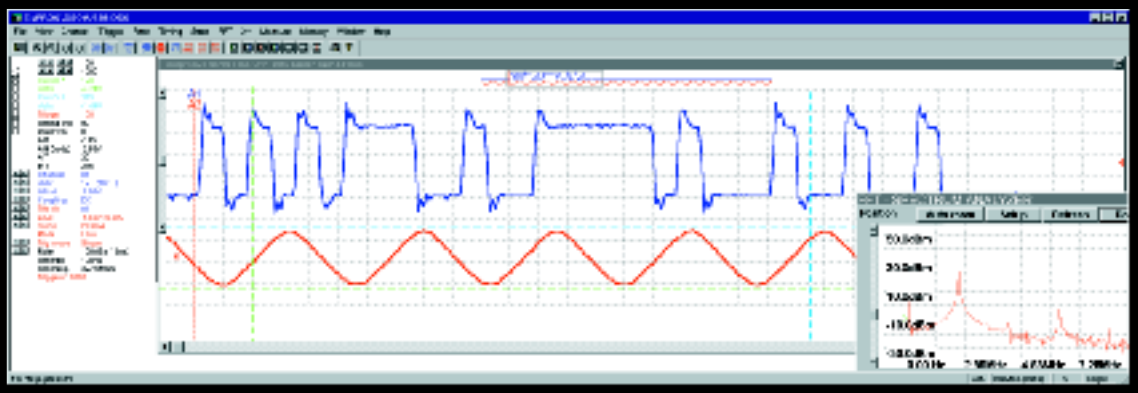
One of 1,000s of examples of programmable analog blocks.

Check out our free online training and our 4-hour applications support:
www.cypress.com/ad/psoc-cea1

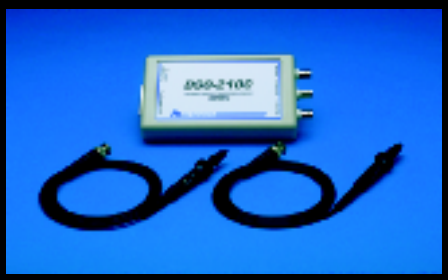
CYPRESS MICROSYSTEMS
A Subsidiary of Cypress Semiconductor

©Cypress Microsystems, Inc. 2003. PSoC, Programmable System-on-Chip, Cypress Enhanced Analog and CEA are trademarks of Cypress Microsystems, Inc. All other trademarks are the property of their respective owners.

Digital Oscilloscopes

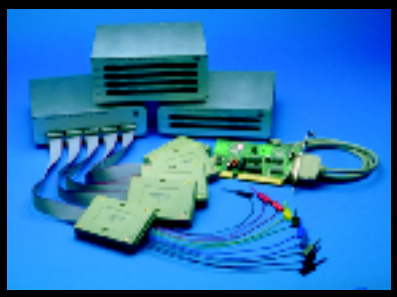


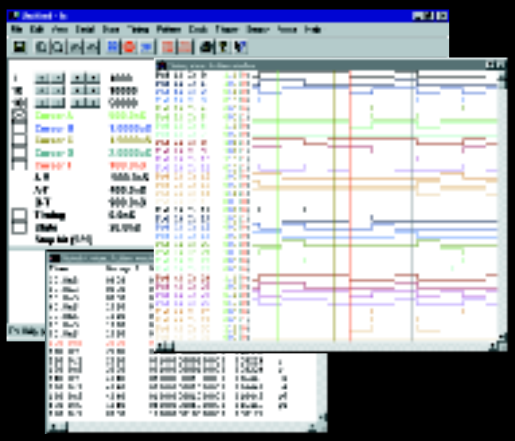
- 2 Channel Digital Oscilloscope
- **100 MSa/s** max single shot rate
- 32K samples per channel
- Advanced Triggering
- Only 9 oz and 6.3" x 3.75" x 1.25"
- Small, Lightweight, and Portable
- **Parallel Port** interface to PC
- Advanced Math options
- FFT Spectrum Analyzer options

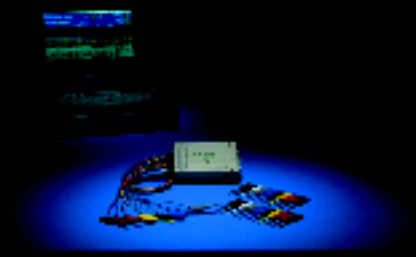


DSO-2102S $525
DSO-2102M $650
Each includes **Oscilloscope**, Probes, Interface Cable, Power Adapter, and software for Win95/98, WinNT, Win2000 and DOS.

Logic Analyzers







- 40 to 160 channels
- up to 500 MSa/s
- Variable Threshold
- 8 External Clocks
- 16 Level Triggering
- up to 512K samples/ch
- Optional Parallel Interface
- Optional 100 MSa/s Pattern Generator

LA4240-32K (200MHz, 40CH)	**$1350**
LA4280-32K (200MHz, 80CH)	**$2000**
LA4540-128K (500MHz, 40CH)	**$1900**
LA4580-128K (500MHz, 80CH)	**$2800**
LA45160-128K (500MHz, 160CH)	**$7000**

All prices include Pods and Software

- 24 Channel Logic Analyzer
- 100MSa/S max sample rate
- Variable Threshold Voltage
- Large 128k Buffer
- Small, Lightweight and Portable
- Only 4 oz and 4.75" x 2.75" x 1"
- Parallel Port Interface to PC
- Trigger Out
- Windows 95/98 Software

LA2124-128K (100MSa/s, 24CH) Clips, Wires, Interface Cable, AC Adapter and Software **$800**

www.LinkIns4.com

Link Instruments • 369 Passaic Ave • Suite 100 • Fairfield, NJ 07004 • (973) 808-8990 • Fax (973) 808-8786

Seetron Serial LCDs

...almost too easy!

LCDs have a well-deserved reputation for being a pain in the neck to interface. Their non-standard parallel busses and weird timing are enough to turn your hair gray—if you don't pull it out first. Our LCDs have an easy-to-program serial interface (up to 9600 bps) that makes them a snap to use, particularly with friendly micros like BASIC Stamps®.

• *Easy hookup and mounting.* We don't stop at simplifying display interfacing and programming. For easy, quick connections, we offer a variety of ready-made wiring harnesses. Need mounting hardware? We've got that, too.

• *In stock right now.* Hate to wait? Our products are in stock and ready to ship the day you place your order.

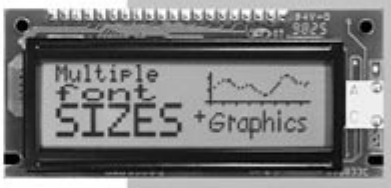
• *Online satisfaction.* Our customers love our web site—it's fast, uncluttered, and jammed with tech data on our products. The online store runs weekly sales with savings up to 40%. Checkout is a breeze with a simple one-page form.



BPI-216N
2x16 serial LCD
ultra-low power
45.00



ILM-216L
2x16 serial LCD
w/ switch inputs,
config EEPROM
49.00



SGX-120L
120x32-pixel
serial graphics LCD
99.00



TRM-425L
4x20 serial LCD
w/ keypad input,
95-screen memory
119.00

www.seetron.com

Many more models available at seetron.com

Scott Edwards Electronics, Inc.
1939 S. Frontage Rd. #F, Sierra Vista, AZ 85635
phone 520-459-4802 • fax 520-459-0623
web www.seetron.com • email sales@seetron.com

TASK MANAGER

Model Robot

Revolutionary engineering has led to extraordinary advances in robotics. As I'm writing this editorial, NASA scientists have just announced the discovery that Mars was once a wet planet. Their achievements in geological research could not have been possible without the impressive capabilities of the Opportunity and Spirit rovers. These robots were tasked—on an $820-million mission—with collecting data that NASA believed might indeed prove that liquid water existed on the red planet. The six-wheeled rovers have been able to traverse the landscape and perform critical research, yielding incredible results.

This is not to say that there haven't been complications along the way. NASA scientists have grappled with malfunctions on both rovers. In late January, there were problems with the transmission feeds from Spirit. A month later, a malfunction that caused the on-board heater to stay on threatened to drain power from Opportunity. Despite these issues, the geologist robots have continued working successfully.

Here on Earth, we're getting ready to see the latest and greatest fire fighting robots. The 2004 Trinity College Fire Fighting Home Robot Contest will be held in Hartford this month. The competition attracts everyone from curious onlookers to amateur hobbyists to highly skilled engineers. Challengers from around the world demonstrate the speed, dexterity, and efficiency of their small robots in extinguishing candle flames. We've definitely been impressed by some of the past entries. In fact, a few of the robots featured in our past issues debuted at the Trinity contest. Good luck to everyone in the competition!

The robots featured in this issue share many of the characteristics that have made the Mars rovers successful and will surely be demonstrated in the winners of the Trinity contest: efficiency, innovativeness, and accuracy. Like the scientists at NASA, when faced with obstacles, the engineers behind these projects met their design challenges with workable solutions.

In "Mini Rover 7," Joseph Miller discusses the electronic compassing scheme for his robot (p. 14). As he explains, precise navigation is a primary concern for most builders, which is why designing an effective heading system is so important. By calculating a more accurate bearing, Joseph was able to achieve better navigation capability.

Ingo Cyliax is back with an article about TeleBot, his robot surveillance unit (p. 30). The small, inexpensive robot is modeled after similar units used by municipalities to check for pipe damage in sewage systems. Ingo, too, used electronic compassing for his autonomous robot, which provides heading measurements with accuracy of ±2°.

Plus, we also have the scoop on Lego's Spybot. Jay Francis has transformed this already interesting toy into a sophisticated autonomous robot by interfacing to the original electronics. Using a Microchip PIC16F876, he was able to emulate an I²C EEPROM. Those of you who enjoyed Jeff Bachiochi's Mindstorms project in last year's Robotics issue (*Circuit Cellar*, issue 153, April 2003) are sure to find Jay's design interesting.

On a side note, I wanted to thank all of you who have already participated in our current reader survey. I invite anyone who hasn't yet to take a few minutes to tell us how we're doing. This is your opportunity to share your opinions and help shape our content. I look forward to reading your responses. You'll find a link to the survey at the top of our home page.

jennifer.huber@circuitcellar.com

CIRCUIT CELLAR®

THE MAGAZINE FOR COMPUTER APPLICATIONS

EDITORIAL DIRECTOR/FOUNDER
Steve Ciarcia

MANAGING EDITOR
Jennifer Huber

TECHNICAL EDITOR
C.J. Abate

WEST COAST EDITOR
Tom Cantrell

CONTRIBUTING EDITORS
Ingo Cyliax
Fred Eady
George Martin
George Novacek
Jeff Bachiochi

NEW PRODUCTS EDITOR
John Gorsky

PROJECT EDITORS
Steve Bedford
Ken Davidson
David Tweed

CHIEF FINANCIAL OFFICER
Jeannette Ciarcia

CUSTOMER SERVICE
Elaine Johnston

ACCOUNTANT
Jeff Yanco

ART DIRECTOR
KC Prescott

GRAPHIC DESIGNER
Mary Turek

STAFF ENGINEER
John Gorsky

QUIZ COORDINATOR
David Tweed

ADVERTISING

PUBLISHER
Dan Rodrigues E-mail: dan@circuitcellar.com

ASSOCIATE PUBLISHER/DIRECTOR OF SALES
Sean Donnelly Fax: (860) 871-0411
(860) 872-3064 E-mail: sean@circuitcellar.com
Cell phone: (860) 930-4326

ADVERTISING COORDINATOR
Valerie Luster Fax: (860) 871-0411
(860) 875-2199 E-mail: val.luster@circuitcellar.com

ADVERTISING ASSISTANT
Deborah Lavoie Fax: (860) 871-0411
(860) 875-2199 E-mail: debbie.lavoie@circuitcellar.com

Cover photograph Chris Rakoczy—Rakoczy Photography
PRINTED IN THE UNITED STATES

CONTACTING CIRCUIT CELLAR

SUBSCRIPTIONS:
INFORMATION: www.circuitcellar.com or subscribe@circuitcellar.com
To Subscribe: (800) 269-6301, www.circuitcellar.com/subscribe.htm, or subscribe@circuitcellar.com
PROBLEMS: subscribe@circuitcellar.com

GENERAL INFORMATION:
TELEPHONE: (860) 875-2199 Fax: (860) 871-0411
INTERNET: info@circuitcellar.com, editor@circuitcellar.com, or www.circuitcellar.com
EDITORIAL OFFICES: Editor, Circuit Cellar, 4 Park St., Vernon, CT 06066
NEW PRODUCTS: New Products, Circuit Cellar, 4 Park St., Vernon, CT 06066 newproducts@circuitcellar.com

AUTHOR CONTACT:
E-MAIL: Author addresses (when available) are included at the end of each article

For information on authorized reprints of articles, contact Jeannette Ciarcia (860) 875-2199 or e-mail jciarcia@circuitcellar.com.

CIRCUIT CELLAR®, THE MAGAZINE FOR COMPUTER APPLICATIONS (ISSN 1528-0608) and Circuit Cellar Online are published monthly by Circuit Cellar Incorporated, 4 Park Street, Suite 20, Vernon, CT 06066 (860) 875-2751. Periodical rates paid at Vernon, CT and additional offices. **One-year (12 issues) subscription rate USA and possessions $21.95, Canada/Mexico $31.95, all other countries $49.95. Two-year (24 issues) subscription rate USA and possessions $39.95, Canada/Mexico $55, all other countries $85.** All subscription orders payable in U.S. funds only via VISA, MasterCard, international postal money order, or check drawn on U.S. bank.

Direct subscription orders and subscription-related questions to Circuit Cellar Subscriptions, P.O. Box 5650, Hanover, NH 03755-5650 or call (800) 269-6301.

Postmaster: Send address changes to Circuit Cellar, Circulation Dept., P.O. Box 5650, Hanover, NH 03755-5650.

Circuit Cellar® 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® disclaims any responsibility for the safe and proper function of reader-assembled projects based upon or from plans, descriptions, or information published by Circuit Cellar®.

The information provided by Circuit Cellar® is for educational purposes. Circuit Cellar® makes no claims or warrants that readers have a right to build things based upon these ideas under patent or other relevant intellectual property law in their jurisdiction, or that readers have a right to construct or operate any of the devices described herein under the relevant patent or other intellectual property law of the reader's jurisdiction. The reader assumes any risk of infringement liability for constructing or operating such devices.

Entire contents copyright © 2004 by Circuit Cellar Incorporated. All rights reserved. Circuit Cellar and Circuit Cellar INK are registered trademarks of Circuit Cellar Inc. Reproduction of this publication in whole or in part without written consent from Circuit Cellar Inc. is prohibited.

Their boards come with a packing slip. Ours come with a Microsection Analysis Report

In today's competitive climate, offering the best product at a competitive price is a must to satisfy your customers. Sierra Proto Express offers the fastest, most reliable, turns at the highest quality. And we'll prove it in every shipment with our unique Evidence of Quality reports, so you know your board is right the first time. One proof of quality is our Microsection Analysis Report, as featured here, so you can see the quality inside your board. And that is just part of our comprehensive quality tests. It's in our process, not in our price. In fact, it is our commitment to total quality that enables us to run our operation cost effectively. And that comes back to you in our competitive price. Talk to a Sierra Proto Express Account Manager about our commitment to quality, our range of technology, and our 99% on-time track record of delivery. Call **1.800.763.7503** from 6 a.m. to 6 p.m. PST or email your design to **files@protoexpress.com** and receive a quote. **Mention code: PPDC00093**

14 Layer Board Microsection

Learn more about our unique Evidence of Quality report that comes with every PCB at www.protoexpress.com



For proven quality that never costs extra, put Sierra Proto Express on your team today.

QUALITY LEADER
SIERRA PROTO EXPRESS
MIL-P-55110 ISO9002
800-763-7503 www.protoexpress.com

Quality On Time

April 2004: Robotics

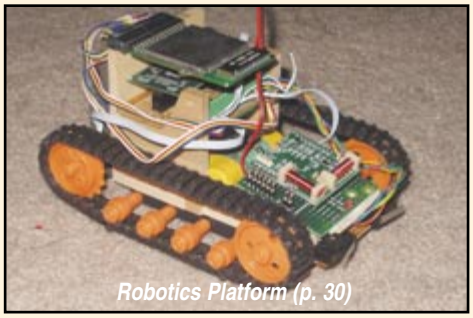
FEATURES



14 **Mini Rover 7**
Electronic Compassing for Mobile Robotics
Joseph Miller

24 **BasicCards 101 (Part 2)**
Use in a Liquid Nitrogen Monitor
Brian Millier

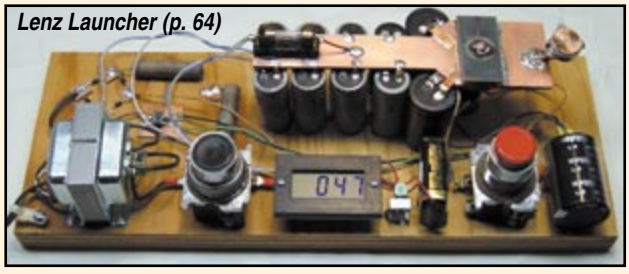
30 **Build a Small Robotics Platform**
Ingo Cyliax



38 **Low-Cost Intelligent Sensors Network**
Victor Kremin
Flash Innovation 2003 First Prize Winner

48 **Robot Upgrade**
Use a Microcontroller to Emulate an I²C EEPROM
Jay Francis

COLUMNS



58 **APPLIED PCs**
A Wireless Ethernet Solution for the People
Fred Eady

64 **ABOVE THE GROUND PLANE**
Pennies to Heaven
The Lenz Launcher
Ed Nisley

68 **FROM THE BENCH**
USB in Embedded Design (Part 1)
The Undeniable Benefits
Jeff Bachiochi

76 **SILICON UPDATE**
Powered Points
Tom Cantrell

DEPARTMENTS

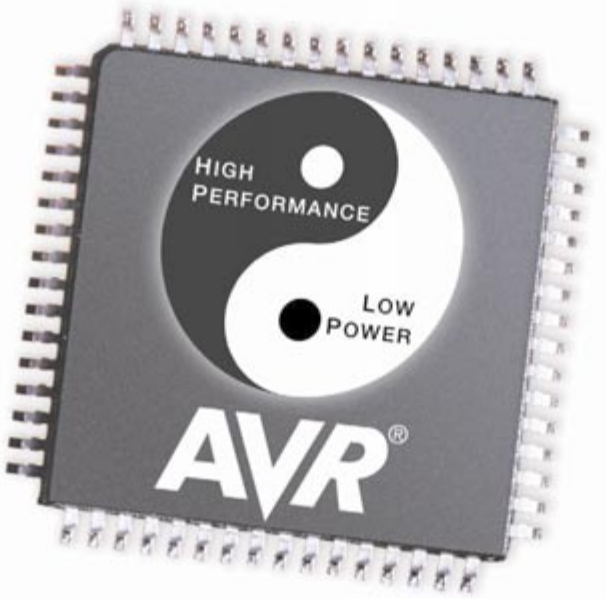
4 **TASK MANAGER**
Model Robot
Jennifer Huber

9 **NEW PRODUCT NEWS**
edited by *John Gorsky*

12 **TEST YOUR EQ**
edited by *David Tweed*

94 **INDEX OF ADVERTISERS**
May Preview

96 **PRIORITY INTERRUPT**
Beach Day
Steve Ciarcia



AVR microcontrollers
The perfect combination of performance and power consumption

Specifically designed for the latest battery operated portable applications, Atmel's new Low Power AVR microcontrollers provide six low-power operation modes. These devices work down to 1.8 volts providing maximum battery life. In low power modes such as standby, AVR turns off all internal peripherals and the core. Standby mode leaves only the external crystal running to deliver start-up times as low as 6 clock cycles.

AVR is a RISC CPU that executes instructions in a single clock cycle, with a rich CISC-like instruction set and 32 working registers. AVR microcontrollers have a very high code density and extremely fast execution speeds up to 24 MIPS. The Flash program memory and on-chip EEPROM will help you slash months off your development efforts – save thousands of dollars in project costs, and save YOUR energy as well.



AVR comes in a full selection of package and performance options, and is supported by a full compliment of integrated tools to make your development process fast, simple and successful. Atmel dedicated staff as well as certified technical consultants are available to help and support your projects.

See for yourself: Atmel's Butterfly evaluation kit demonstrates AVR's low power capabilities, and also serves as a development kit for the mega169 microcontroller.

So get your project started off right. Learn more about AVR at www.atmel.com/ad/lowpoweravr, and **register to win a free butterfly evaluation kit.**
See why AVR offers the perfect combination.

Check out AVR today at: www.atmel.com/ad/lowpoweravr

Low Power AVR Typical Power Consumption				
Operating Voltage	Power Down	Real Time Counter	Active 32 kHz	Active 1 MHz
1.8 Volts	Less than 100 nA	4.5 µA	15 µA	240 µA
3 Volts	Less than 250 nA	7 µA	26 µA	420 µA

ATMEL®
Everywhere You Are™

© 2004 Atmel Corporation. Atmel®, AVR® and the Atmel logo are registered trademarks. "Everywhere You Are™" is a service mark of Atmel Corporation.



Hobbyist?

You might be if you used a **SCHEMATIC** to explain something **ON A FIRST DATE**, or your **SOLDERING IRON** is next to the **COFFEE MAKER**. If you wired up the **DOG RUN** with a **TEMPERATURE CONTROLLED MISTING SYSTEM** that activates via **ETHERNET**... and you don't have a dog...

Want your application featured on our site?

Show off your work!

When we publish your application we will send $150.

imaginetools.com

MICROPROCESSOR STARTER KIT

C-Programmable, 8-bit Rabbit 3000® Microprocessor, application based, experimentation board, mad science...

Application Notes Included:

- RangerBot
- GPS Clock
- RS-232 Voltmeter
- DC Motor
- Metal Detector
- Many More...



imagine tools™

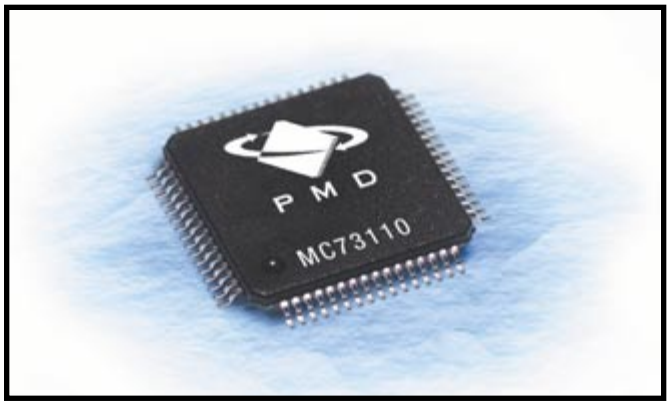
Think
Learn
Build

NEW PRODUCT NEWS

Edited by John Gorsky

ADVANCED MOTOR CONTROL IC

The **MC73110** is a new motor control IC for brushless DC motors. This intelligent single-axis controller is the first member of a new line of motor control devices—is designed to provide ultra-high performance digital current loop, velocity loop, and commutation for brushless motors.



The MC73110 operates in one of three modes: Internal Velocity Profile mode, Velocity mode with an external command signal, and Torque mode with an external command signal. It provides software-programmable velocity and current loops, six-step and sinusoidal commutation, analog or digital command input, profile generation, and six-signal symmetric PWM waveform generation.

Stand-alone operation is achieved using preprogrammed flash memory-stored parameters or through a serial port interface using microprocessor-style commands. To create a complete motion controller, all that is required is to connect the MC73110 to three half-bridge switchers, typically MOSFET or IGBT-based.

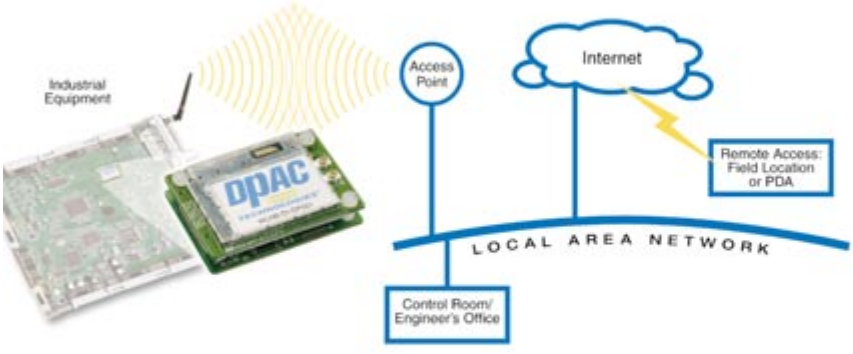
The MC73110 is ideal for building high-performance, low-cost brushless motor amplifiers. It can be used in applications such as motion control amplifiers, motor drives, medical automation, centrifuges, tape drives, machine tools, scientific instrumentation, and semiconductor equipment.

The MC73110, which is packaged in a compact 64-pin plastic quad flat pack (PQFP), operates from 3.3 V. Prices start at **$18** in OEM quantities.

Performance Motion Devices, Inc.
(781) 674-9860
www.pmdcorp.com

DPAC TECHNOLOGIES

Web-Enable Your Products with
Airborne™ Wireless Products



Technology	IEEE 802.11b DSSS, WiFi compliant
Frequency	2.4 ~ 2.4835 GHz (US/Can/Jpn/Europe 2.471 ~ 2.497 GHz (Japan)
Modulation	DQPSK, DBPSK and CCK
Channels	11 Channels - USA/Canada 13 Channels - Europe 14 Channels - Japan 4 Channels - France
Data Rate	11, 5.5, 2, 1 Mbps
MAC	CSMA/CA with ACK, RTS, CTS
RF Power	+15 dBm (typical) Approx. 32 mW
Sensitivity	-82 dBm for 11 Mbps -86 dBm for 5.5 Mbps -88 dBm for 2 Mbps -90 dBm for 1Mbps
Security	WEP standard encryption, 64/128 bits
Antenna	Supports diversity antennas, using U.FL coaxial connectors (50ohm)
Supply	3.3 VDC
Current	420mA - transmit mode (typical) 350mA - receive mode (typical) 200mA - sleep mode (typical)
Temp	Industrial -40C ~ +85C
GPIO	Up to 8 digital I/O ports and status
Serial	UART up to 230.4 Kbps IIC Master to 400KHz SPI up to 20MHz
Analog	Up to 8 channels, 10bit resolution
Connector	36pin (PN: HRS DF12-36DS-0.5V)

+ Radio with Base-band Processor
+ Built-in Web Server
+ Application Processor with Integrated RTOS
+ Built-in TCP/IP Stack
+ Configurable Serial/Digital/Analog I/O Ports
= Drop-In 802.11b Solution

For Design-In Information:
1-800-642-4477 x164
www.dpactech.com/wirelesscc_dsp.asp

NEW PRODUCT NEWS

MOTOR CONTROL PIC

The new 40-pin **PIC18F4331/4431** and 28-pin **PIC18F2331/2431** microcontrollers provide a complete solution for extremely precise, energy-efficient operation in sophisticated motor control applications through advanced analog and digital feedback, and the new three-phase complimentary PWM control system.

Three new modules specifically designed for advanced motion and power control applications are key to the PIC18Fxx31 family. The power control module has a three-phase PWM structure. The motion feedback module includes a quadrature encoder interface. The third module consists of a high-speed ADC, which operates at 200 ksps and can be synchronized with the PWM. Together, these modules eliminate the need for any external motor control components (e.g., a high-speed converter and a resonator), thereby providing a lower-cost solution and saving board space.



Other significant features include: up to eight PWM channels with 14-bit resolution, center alignment, programmable dead time, and two external fault-protection inputs; two capture/compare/PWM modules; and an internal RC oscillator with eight selectable frequencies ranging from 8 MHz to 31.25 kHz. Also included are four general-purpose timers, a clock monitor, and an enhanced USART supporting RS-485, RS-232, and the LIN protocol. Prices start at **$4.61** for the PIC18F2331 in 10,000-piece quantities.

Microchip Technology, Inc.
(888) 628-6247
www.microchip.com

FULLY PROTECTED MOSFET SWITCH

The **BSP75G** is the first fully self-protected MOSFET device in the IntelliFET low-side array-based platform integrating a configurable component array with a vertical power transistor on the same die.

The 60-V, 550-mΩ N-channel BSP75G is protected against over-temperature, over-current, over-voltage, and ESD. The MOSFET offers logic-level input control and will automatically restart on removal of any fault type. The SOT223-packaged part suits general-purpose switching duties and is rated for automotive applications.

At a preset chip temperature of approximately 175°C, the device shuts down and a 550-mJ active clamp turns the device off before it goes into avalanche breakdown. Thanks to its novel over-current limit, the device can handle an unusually high continuous current of 1.6 A. Internal bidirectional diodes provide human body ESD protection.

As well as protecting itself, the BSP75G also actively protects the load. By limiting the load current, dissipation in the load is greatly reduced, which can prevent destruction of the load during a load dump condition. Short circuits are also handled. After a permissible start-up current pulse, the device settles to a current that is fixed by the internal over current protection circuit.

The BSP75G is particularly well suited for loads with a high inrush current such as lamps and motors. It will switch all types of resistive, inductive, and capacitive loads. And it acts as a microcontroller-compatible power switch in 12- and 24-V circuits. The BSP75G costs **$0.41** each in 10,000-piece quantities.



Zetex, Inc.
(631) 360-2222
www.zetex.com

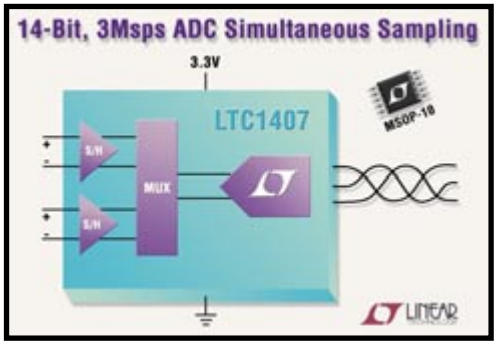
TINY TWO-CHANNEL, 14-BIT ADC

The **LTC1407A** and **LTC1407** are two new 14- and 12-bit, 3-Msps ADCs with two simultaneous sampling inputs. These high-speed serial ADCs are available in a 10-pin MSOP package less than half the size of an SO-8, allowing the design of compact high-speed data-acquisition systems. The LTC1407A device achieves 73.5-dB SINAD and guarantees 14-bit no missing codes. Its small size in combination with high speed and simultaneous sampling make the LTC1407 the ideal choice for medical applications, instrumentation, multiphase motor control, and narrowband I & Q demodulation applications.

The LTC1407A is powered from a single 3-V supply. Power dissipation is typically 12 mW. When the device is not converting, power dissipation can be reduced to 3.3 mW in NAP mode, with the on-board 2.5-V reference remaining active, and 6 µW with everything powered down in Sleep mode. The two conversion results are delivered sequentially to high-speed DSP serial ports via a three-wire SPI interface.

The ADCs are available in the commercial and industrial temperature ranges. Pricing begins at **$7** for the LTC1407A and **$4** for the LTC1407 in 1000-piece quantities.

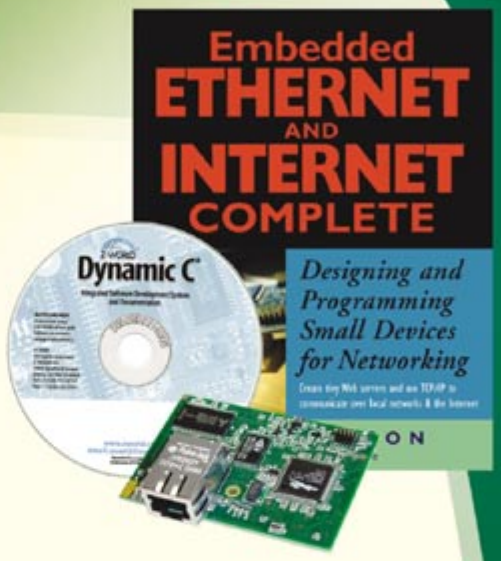
Linear Technology Corp.
(800) 454-6327
www.linear.com



Jumpstart Your Internet Designs

FREE
Book With Dev. Kit
$49.95 value

RCM3200 Kit + Jan Axelson = saved development time and cost



RCM3200 Development Kit Provides a Complete Embedded Design System:

Kit Includes RCM3200 core module, Dynamic C® software (with royalty-free TCP/IP stack with source) and complete documentation on CD-ROM, prototyping board and serial cable for programming and debugging. *Embedded Ethernet and Internet Complete* valued at $49.95 will be included at no charge with priority code order.



$72 qty. 100
$63 qty. 1000

FEATURES	RCM3200
Ethernet	10/100Base-T
Processor	Rabbit 3000® @ 44.2 MHz
Power	3.3 V operation
Memory	Over 1 MB code/data
I/O	52 digital
Serial Ports	6 (SDLC/HDLC, Async, SPI)

"My sample applications were up and running quickly with Rabbit's RCM3200 and Dynamic C programming software. Everything I needed was provided in a comprehensive, easy-to-use kit."

Jan Axelson, *author*

KIT PRICE $349 USD

Order Online!

Free *Embedded Ethernet and Internet Complete* with your RCM3200 Development Kit order at:

www.rabbitsemiconductor.com/9418
Limited Time Offer

RABBIT Semiconductor

www.rabbitsemiconductor.com
530.757.8400

NEW PRODUCT NEWS

OS EMBEDDING KITS

The new **OS Embedder** kits simplify the implementation of operating systems on a wide variety of Octagon's ruggedized SBCs. Operating systems include Linux, Windows CE.net, Windows XP Embedded, QNX, and DOS. They are available on a range of hardware platforms from 586, 133-MHz products to Intel mobile P III at 933 MHz. EBX and PC/104 form factors are available.

All drivers are developed and tested for full functionality on the hardware platform of the user's choice. All OS Embedder kits include cables for connectivity, memory, custom drivers, application examples, and the target hardware platform. OS Embedder kits reduce the time to market. Free and unlimited technical support on both the hardware and software is available.

Linux OS Embedder kits include an optimized 2.4 Linux kernel preinstalled on a CompactFlash. Users get all the benefits of the Linux performance open-source environment with no license fees.



The QNX OS Embedder kit comes with a preprogrammed evaluation image on a CompactFlash, which is a 30-day evaluation version of QNX for a host system. QNX systems offer the deterministic stability of hard real time.

Windows OS Embedder kits feature either Windows CE.net or Windows XPe. Windows CE.net kits include an evaluation version of development software from Microsoft for a host system, instructions for building a target image, and methods for transferring that image to the CompactFlash. Windows XPe evaluation software requires downloading the software from Microsoft's web site. The basic OS Embedder kit (including SBC) pricing starts at **$995** for single quantities.

Octagon Systems Corp.
www.octagonsystems.com

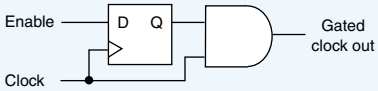
CIRCUIT CELLAR Test Your EQ

Edited by David Tweed

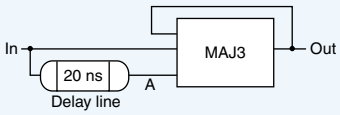
Problem 1—What are the correct units for bulk resistivity?

Problem 2—What are the correct units for resistivity in materials constructed as thin sheets?

Problem 3—The figure shows one way of generating a gated clock signal. What's wrong with this approach? How can this be fixed?



Problem 4—The following circuit is a glitch filter for a clock signal.



The MAJ3 gate is a special primitive available in some FPGA families. Its output is the same as the majority of its inputs; in other words, if any two inputs are high, the output is high. How exactly does it work, and what would be the effect of inverting the output of the delay line?

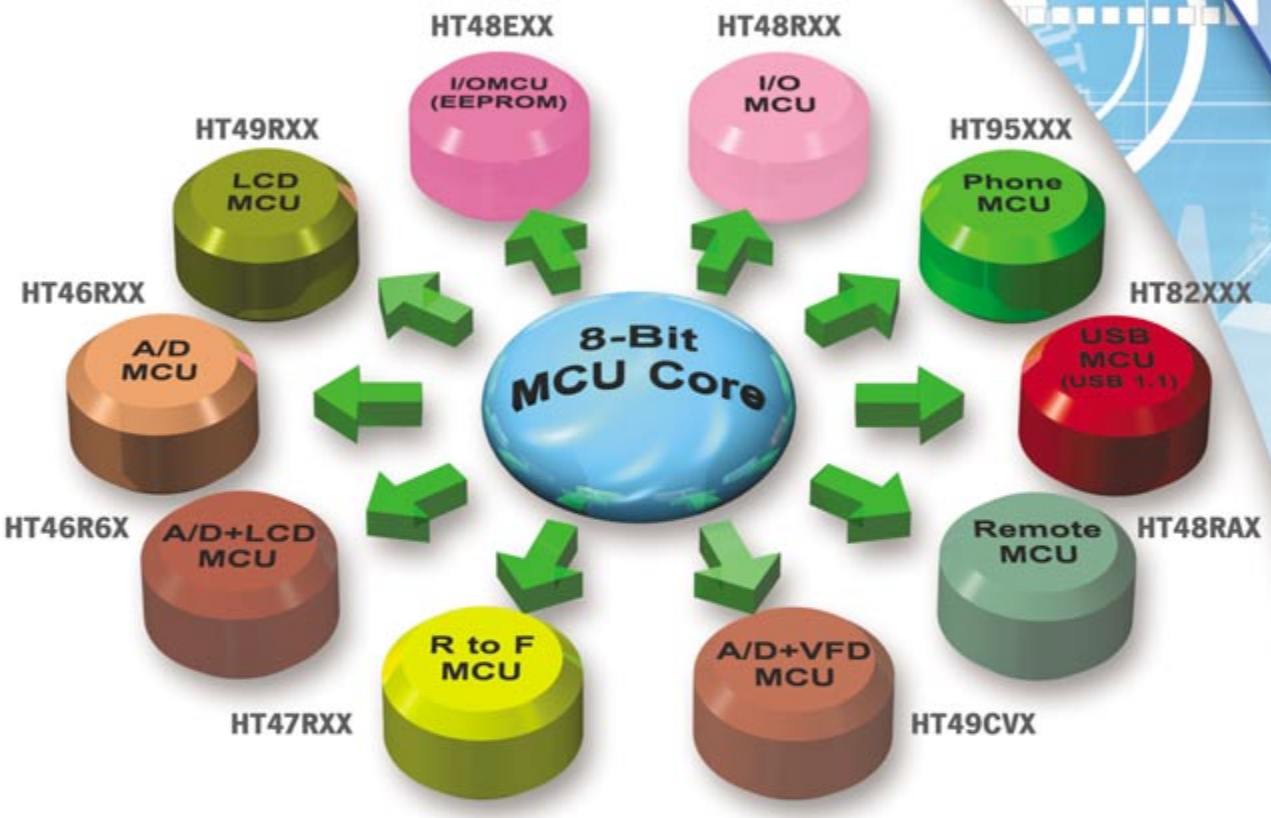
Contributed by David Tweed

What's your EQ?—The answers are posted at www.circuitcellar.com/eq.htm
You may contact the quizmasters at eq@circuitcellar.com

HOLTEK

Comprehensive device range and superior performance for all your intelligent controller solutions

The Holtek MCU Family



- Cost-effective solutions
- Industrial Specifications: -40°C~+85°C
- High noise immunity and electrostatic protection
- Comprehensive device range for various applications
- OTP and Mask device compatibility

New range of development tools and volume production type OTP writers now available !



HOLTEK SEMICONDUCTOR INC.

www.holtek.com

◎ USA Sales Office : Holmate Semiconductor, Inc.
46712 Fremont Blvd., Fremont, CA 94538 Tel: (510) 252-9880 Fax: (510) 252-9885
www.holmate.com

Mini Rover 7

Electronic Compassing for Mobile Robotics

Electronic compassing is one of the most intelligent ways to provide absolute heading information for a mobile robot. In this article, Joseph explains why the PNI V2Xe compass turned out to be the best fit for his Mini Rover 7 robot, which he modeled after the NASA/JPL Rocky 7 Mars rover.

Recent technological growth has yielded some impressive tools for tackling increasingly difficult tasks. The robotics field has been a large beneficiary of these advancements because it encompasses so many disciplines, each heavily dependent on technology. Mobile robot navigation is one particular recipient of recent technological advancements. Accurate and reliable navigation is fundamental to the success of any mobile robotic application.

Navigation is typically given one of the highest, if not the top, consideration when designing an autonomous mobile robot. You have to consider numerous hardware and software options. These days a GPS receiver is one of the first tools that you think about to fulfill your navigational needs. However, the area that many mobile robots operate in is within GPS's present 3- to 15-m accuracy range.[1] Even long-range mobile robots still have to maneuver successfully within that 3- to 15-m resolution uncertainty.

Compound these resolution limitations with satellite signal interference, and it becomes clear that a supplemental system is essential. Heading information, along with positional data from other sources, can provide interim position data as well as augment GPS data to resolve higher accuracies.

A magnetic compass has many advantages as a provider of heading information. Compassing is one of the only methods that can provide absolute heading information without external references for calibration. Today's electronic compasses easily interface with microcontrollers and come with a host of other features like low-power consumption and built-in local distortion correction, such as any other instrument. Compasses have their own uncertainties and issues. Understanding how compasses work, as well as the behavior of the environment that they measure, will better prepare you to manage their uncertainties and best apply the technology to a project like my Mini Rover 7 robot.

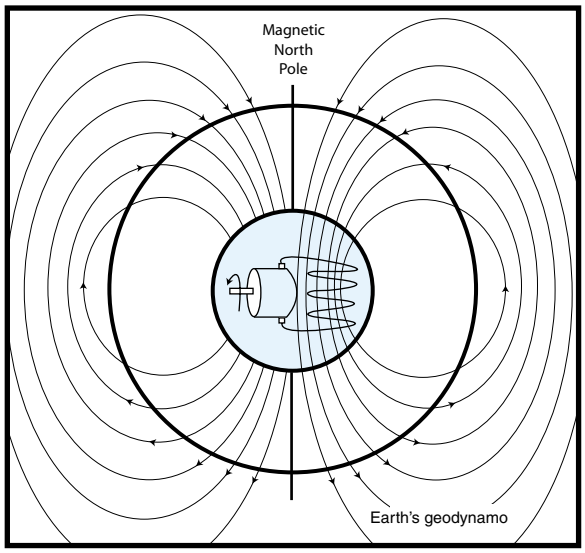


Figure 1—*As you study the Earth's geodynamo magnet field pattern, note how the field lines are not horizontal to the surface except along the equator. The rotating dynamo and coil in the center represent the Earth's magnetic field being generated by its ever-flowing iron outer core.*

BASIC ROBOT NAVIGATION

Dead reckoning (DR) is the fundamental navigation method used in mobile robotics. It is a method of mathematically tracking your present position by measuring speed and direction traveled at regular intervals, or distance and direction at any convenient interval. The latter is easier to do on most wheeled mobile robots if you use wheel encoders. This is also referred to as odometry.

The basic tools for DR in mobile robotics are a compass and an odometer. In more complex robots with advanced navigation resources, DR is still used to navigate between absolute position fixes. Position tracking by DR is used to get to and from fixed locations, help create maps of surroundings, and keep track of position when moving around unexpected obstacles.

Position errors have a tendency to accumulate over time when navigating with DR. A robot happily hobbling down a path can easily drift off the path because of the imperfect traction of its drive wheels or limbs. Furthermore, the terrain may be uneven, inconsistent, obstacle-ridden, or even move, as is the case for aquatic vehicles. A robot could simply correct its steering to stay

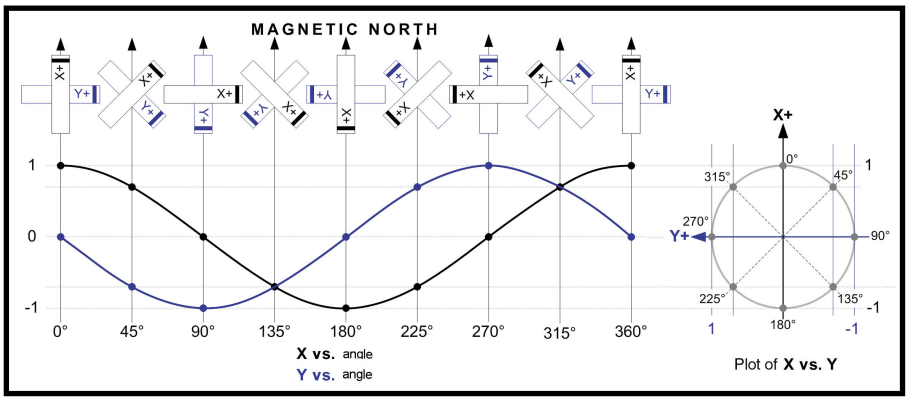


Figure 2—*I plotted the magnetometer sensor output versus the angle. You can also see the x-y plot of the magnetometer sensor output.*

on course (or on bearing), but that little off-course excursion could add a small position error. The magnitude of the position error is the integral of time, distance, and off-course angle traveled while the robot is off course. What is really required is a new bearing calculation to increase the robot's chances to reach the desired target.

HEADING DETERMINATION

For centuries, we have been taking advantage of the Earth's magnetic field to orient ourselves with respect to its magnetic poles. Both the mechanical needle compass and electronic compass can provide absolute heading information. It is hard to beat the compass for this purpose. With the exception of GPS, other systems that exist require an external heading reference as calibration.

Gyroscopes use mechanical angular momentum changes to measure angular and linear movements. Traditional flywheel gyroscopes are fast-spinning gimbaled flywheels with encoders. The encoders are situated about the pivotal axes of the gyroscope's gimbals and register angular movement of the spatially stable flywheels to its base, which is fastened to a host vessel. Modern gyroscopes use micro electromechanical systems (MEMS) and optical technologies in place of the bulky flywheels.

Gyroscopes have fast response times and are insensitive to magnetic anomalies. They are also relative angular position sensors, which require an external reference heading to initially set. A special kind of gyroscope called the gyrocompass can align itself with the Earth's rotational axis, but it tends to be a large and costly instrument.

Differential wheel encoding is another technique used to determine heading. Relative heading changes can be computed by taking the difference of distance traveled by two opposing wheels. This technique has the same traction and terrain issues associated with the aforementioned wheel encoder odometry.

A single-antenna GPS can provide heading information, but it is not instantaneous. It inherently lags the movement of the robot or vehicle because the derived heading requires previous position data. A GPS could not tell you where you are heading if you were to stop and change directions. Like compasses, GPS receivers do not require external reference heading calibration. Once moving, the GPS heading update rate is a maximum of approximately 1 Hz, although some receivers add damping, which increases this time constant even more. A dual-antenna GPS receiver can provide instantaneous heading—or yaw—information, although the recommended distance between the two antennas is 1 m. This fact, along with its large price tag, can be a limiting factor for many mobile robot applications.

A combination of techniques is the best approach. There are many ways to determine heading, each of which has its own strengths and weaknesses. None of them are infallible. For this reason, some systems use two or more methods cooperatively to increase system accuracy and reliability. The deciding factors are cost, accuracy, efficiency, features, availability, ease of use, speed, and size.

Kalman filters are typically used to integrate the data from multiple sensors to produce a more reliable and accurate heading. Kalman filtering is a statistical method that combines the dynamic model of the system with the statistical behavior of system errors. It enables navigation systems to handle periodic GPS signal interruption, odometer slippage, magnetic anomalies, and other sensor irregularities with minimal degradation of accuracy. Kalman filters also can be extremely complicated. You must fully understand the dynamic behavior of your systems and the statistical and systemic errors of your sensors in order to make proper use of Kalman filters. It might be easier and more feasible in less demanding projects to use other software-based analytical tools like averaging, weighted averaging, limiting, and majority voting to improve heading data reliability.

MAGNETIC COMPASSING

The Earth's magnetic field is created deep in its iron core by a regenerative magnetic field generator that's sometimes referred to as the geodynamo. This iron core has a liquid outer section and a solid inner section. The flow of electrical current in the turbulent liquid iron outer section creates the magnetic field.

The simplest description of the Earth's magnetic field spatial pattern is that of a dipolar field with magnetic flux emanating from the South Pole and converging at the North Pole. The Earth's magnetic field pattern is a little more complex than a simple bar magnet model. As previously mentioned, the Earth's geodynamo is constantly moving. Presently, the magnetic poles are tilted about 11° away from the geographical poles, and they are not at exactly at opposite sides of the world either. The magnetic North Pole is located in northeastern Canada, and the magnetic South Pole is located in the Antarctic Ocean south of Australia.

The geographical North Pole is also known as true north. The angular difference between the true poles and the magnetic poles at a given location is called the declination angle. Depending on your location, true north could appear to either the east or west of the magnetic North Pole.

The Earth's spherically shaped geodynamo produces a magnetic field as shown in Figure 1. Note that the field lines are not horizontal to the Earth's surface, except at the Earth's magnetic equator. Unlike the Earth's straight geographical equator, this one mean-

ders but is located in roughly the same area. At the magnetic poles, the field lines are vertical. The angular vector of the magnetic field with respect to the horizontal plane at any given location is known as the dip angle, or inclination angle. The density of the magnetic field also varies around the world. The magnetic field density is approximately two times as dense at the magnetic poles as it is at the equator.

The magnetic field vector is sometimes referred to as having two separate components, a horizontal component and a vertical component. At the magnetic equator where the magnetic field is horizontal, the field has no vertical component. At the magnetic poles, the field is purely vertical and has no horizontal component. At places where the inclination angle is 45°, the horizontal and vertical components are equal.

The U.S. Geological Service (USGS) and the Nation Oceanic and Atmosphere Administration (NOAA) maintain web sites that have global maps and on-line programs that chart declination angles, field intensities, dip angles, and much more. I will focus on the horizontal component of the magnetic field, because that is the portion that contains the heading information that I wish to measure.



MAGNETIC MEASUREMENTS

A magnetometer is an instrument that can measure the flux density of a magnetic field. It uses one of any number of types of sensors to convert magnetic flux to voltage, current, frequency, or some other electronically measurable form.

There are numerous types of magnetic field sensors: the saturable core magnetometer (or fluxgate magnetometer), the Hall effect sensor, the magnetoresistive sensor, and the magnetoinductive sensor. A two-axis magnetometer, in which the two sensors are in quadrature (orthogonal) orientation, can be used as an electronic compass to compute heading. When it is parallel with the measured field, the magnetometer sensor's output is at maximum for the given amount of magnetic flux density that is present. When the magnetometer sensor is perpendicular to the magnetic lines of flux, the sensor will output no signal. A plot of the x-sensor output versus the y-sensor output results in the heading being represented around the polar axis of the coordinate system origin (see Figure 2).

This form is preferred as a visual analysis tool for sensor and system performance analysis and troubleshooting. Notice that the y-axis is inverted from that of a typical Cartesian coordinate system. This was done so that the compass coordinates would be produced in its correct orientation. When operating with compass coordinates, it is important to remember to make the proper translations from a Cartesian coordinate system to a compass coordinate system, especially after using trigonometric functions.

At angles between parallel and antiparallel with respect to the magnetic lines of flux, the sensor's output signal, X, is a product of the applied magnetic flux density, β, and the cosine of the angle, θ, of the sensor from being parallel with the flux lines.

$$X = \beta\cos(\theta) \quad [1]$$

If a second sensor is added, and if it is

positioned at a right angle to the first sensor, its output, *Y*, will have the same function as *X*, but will be 90° out of phase. The y sensor will be in the east position, and the x sensor will be in the north position. The two sensors are said to be in quadrature with one another. The equation for output *Y* is the following:

$$Y = \beta \sin(\theta) \quad [2]$$

You now have enough data to compute heading from the output values of the x and y sensors. Use the trigonometric identity:

$$\tan(\theta) = \frac{\sin(\theta)}{\cos(\theta)} \quad [3]$$

Combining Equations 1–3 yields:

$$\theta = \text{Arctan}\left(\frac{\beta \sin[\theta]}{\beta \cos[\theta]}\right)$$

or

$$\theta = \text{Arctan}\left(\frac{Y}{X}\right) \quad [4]$$

The arctangent, or inverse tangent ($\tan^{-1}$), is inherently restricted to ±90°, which covers only two quadrants of the coordinate system over its entire input range of –∞ to ∞. This function also operates in the Cartesian coordinate system, which is rotated 90° from compass coordinates.

To convert the Cartesian θ to a compass heading coordinate, a translation must be performed. Table 1 shows the translation based on the output polarity of the sensors when the sensors are oriented as shown in Figure 2. Note that magnetometer polarities may vary depending on the manufacturer.

As demonstrated in Equations 1 and 2, the output signals of both sensors are proportional to field density (β) and its angle relative to magnetic north. The field density can be extracted at any angle of the quadrature pair by computing the geometric sum of the two sensors outputs:

$$\beta = \sqrt{X^2 + Y^2} \quad [5]$$

Equation 5 computes the horizontal component of the overall magnetic field density from the perspective of the robot's horizontal plane. By monitoring this value, you can spot magnetic anomalies and tilting, which effect heading accuracies.

SOFT/HARD IRON DISTORTION

As you now, producing a compass heading with a pair of sensors is fairly straightforward. However, there's trouble in the neighborhood. Magnetic anomalies are all around you. Other magnetic sources and alternate magnetic flux pathways are corrupting your intended signal—the Earth's magnetic field.

Magnetic fields are generated by electrical current (electromagnetism) in your circuitry and neighboring appliances. There are permanent magnets in your robot's motors, and there is magnetized metal in the robot's construction materials.

Measured sensor value X	Measured sensor value Y	Quadrant(s)	Heading Calculation
≥ 0	> 0	270–360	360 – Arctan (Y/X)
≥ 0	≤ 0	0–90	0 – Arctan (Y/X)
< 0	All	90–270	180 – Arctan (Y/X)

Table 1—*Use this table to convert the limited range, Cartesian coordinates system output of the Arctan function into full-range compass coordinates.*



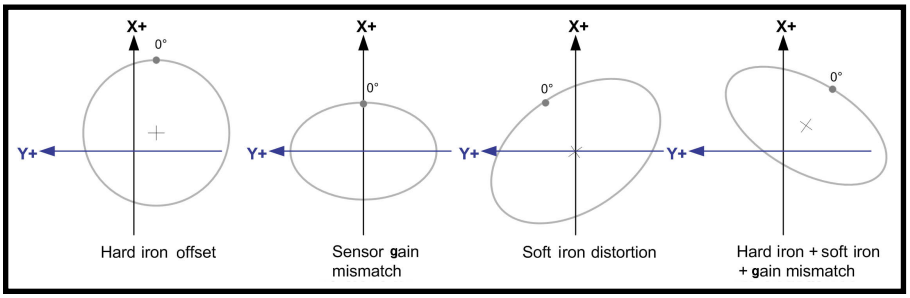


Figure 3—*The x-y plots show various magnetic distortion effects.*

Items external to the robot, such as appliances, underground pipes, and ducts may have their own fields, and can locally offset or change the direction of the Earth's field. Items that create offsetting fields are classified as hard iron. A soft iron material is anything that alters the natural path of a magnetic field's lines of flux. By "natural path" I mean its path through air. Put another way, any material with a relative permeability other than one is a soft iron material. A soft iron also can have a hard iron effect. Usually soft iron and hard iron materials are referred to by the effect they have on the surrounding magnetic field. These effects are easy to recognize when an x-y graph is plotted of the compass sensors output values as the compass is rotated through 360°. Figure 3 illustrates these effects.

All of the plots assume that the culprit material is local, which is to say that it is onboard the robot when it is rotated. These traces, or profiles, can be thought of as the magnetic signatures of the robot's local environment. Notice how the soft iron effects create an elliptical profile. Soft iron material attracts, bends, and condenses magnetic flux toward itself, creating sparse field densities on its lateral sides. At the same time, it will also create dense magnetic fields in areas that are in line with it and the source field direction, as shown in Figure 4. The distribution and quantity of soft iron material will determine the alignment and offset of the ellipse's foci.

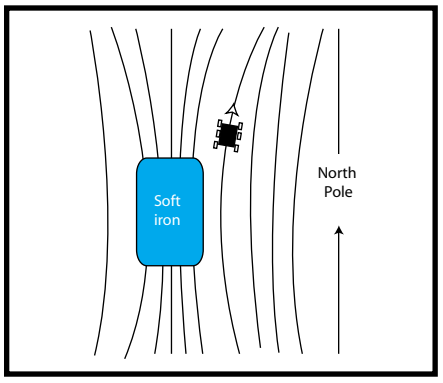


Figure 4—*Here you can see the effects of soft iron distortions in the Earth's magnetic field.*

Nearby electromagnetic fields, ferrous construction materials, and motors make robots especially tough environments for compasses. Before using an electronic compass in a mobile robot, it is a good idea to analyze its magnetic signature by creating an x-y sensor plot. From this plot, you can identify the existence of the aforementioned simple distortions and, more importantly, the more complex ones. The plots should give you an indication of the compass's chances of providing accurate readings, as well as information to help you make corrective adjustments to the robot or sensor placement.

Modern electronic compasses have little trouble mathematically correcting for simple soft iron and hard iron distortions, sensor gain mismatch, and even sensor misalignment (orthogonality), which are all shown in Figure 3. Electronic compasses calculate the correctional coefficients after completing a one-time calibration procedure that involves rotating the robot or vehicle through at least one complete rotation so that it can analyze the geometry of the x-y sensor data.

When testing your robot and creating plots for your analysis, the patterns to watch for are excessive hard iron offsets and multimode soft iron distortions, which look like lumpy circles or ellipses. Excessive hard iron offsets can bring sensors into their nonlinear regions. This looks like a flat spot on the x-y circle plots, which are perpendicular to the axis of the saturated sensor. It is also a good idea to create plots for the robot when it is rotating both clockwise and counterclockwise, and then compare the two to see if the motor currents affect sensor readings.

Figure 5 shows x-y sensor plots (magnetic signature plots) running on my Mini Rover 7 robot. The red trace shows that the robot has some hard iron offsets but no soft iron distortions. I rotated the robot in both directions under its own power and saw nothing significantly different, although it would be hard to see any rotational offsets with these plots. There are ways to test this.

The slightly elliptical blue trace is the result of placing a 9-V battery next to the compass. I placed a small magnet in the robot when I generated the light blue trace. The dark blue trace is the result of placing a PC chassis next to the robot when it performed a rotation. The small amplitude is to be expected because the external soft iron material attracts the magnetic flux away from the robot, therefore creating an area with a sparse field, much like the one in Figure 4. You should not calibrate the compass in your robot under this condition because it does not represent a normal condition. Any electronic compass with hard iron and soft iron compensation should have no problem producing accurate heading data using any of the vehicle magnetic signatures shown in Figure 5.

As you have seen, with a little effort a compass can adapt itself to your robot's unique internal magnetic signature by using its built-in calibration algorithm. However, when operating after calibration, your robot will most likely come across external hard iron

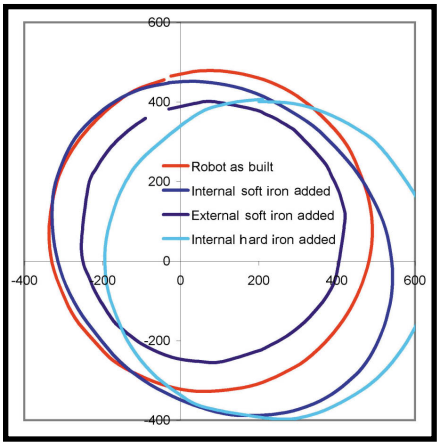


Figure 5—*Here are several magnetic signature plots of the Mini Rover 7 with additional effects.*

ATMEL®



Limitless Possibilities

AVR® 2004 DESIGN CONTEST

When you choose an Atmel AVR microcontroller, not only will you be a winner for choosing one of the fastest, smartest, and most readily supported microcontrollers, your design could also win you a share of

$15,000 in prizes.



Deadline June 30, 2004

www.circuitcellar.com/AVR2004

and soft iron objects, which will alter the Earth's magnetic field direction.

Figure 4 shows how a soft iron object can alter the heading of a robot as it attempts to keep steady (north in this case) when it moves past the object. Notice that the field density is sparse in this area. As long as the sensors are relatively close (so that they experience the same field density ratios), the compass abilities are not significantly diminished. This is because the heading calculation for a quadrature sensor pair uses the arctangent of the ratio of the two sensors (see Equation 2). The overall field density, or magnitude of the field density, can be monitored by calculating the geometric sum of the two sensors (see Equation 4), which can alert you to such anomalies. This information can be used to weigh your options.

You may decide to temporarily trust differential wheel encoder outputs for tracking heading while the overall measured field density falls outside established thresholds. A Kalman filter or your own statistical algorithm can use this information with other sensor data to make improved estimates of actual heading. It is important to use the compass's calibrated x-y sensor data to compute field density because it already has been compensated for your robot's personal magnetic signature. Modern electronic compasses like the PNI V2Xe can report the overall field density in proportion to the field that it experienced at every heading angle during a calibration cycle so that local field distortions do not interfere with the assessment of external magnetic fields.

EFFECTS OF TILT

Tilting an electronic compass can create heading errors. When tilted, the sensors no longer receive the magnetic field in the proportions measured during calibration. Equations 1 and 2 characterize the sensor output value with respect to the magnetic field's angle to the sensor's orientation in the horizontal plane. Tilting the sensor would expose the sensor to portions of the Earth's vertical field component and reduce its exposure to the horizontal field component, which could either be an increase or decrease in field density depending on local inclination angles and direction of tilt.



Photo 1—*No, it isn't the Spirit rover you've seen on the news; it's my Mini Rover 7 robot.*

Equation 6 gives the relationship of heading error versus tilt angle when a compass is tilted in the north-south rotational axis (also known as pitch):

$$\theta_{ERR} = \text{Arctan}(\sin\alpha \tan\varphi) \quad [6]$$

where θ_{ERR} is the heading error caused by tilt, α is the pitch angle compass, and φ is the inclination angle of the Earth's magnetic field.

To use an example, if your compass were located in San Francisco, which has a magnetic field inclination angle of 61°, you could expect a heading error of up to 1.8° for every degree of pitch for the first 10° of tilt. Tilt in the east-west direction (roll) or any compound angle of the two tilt axes creates similar errors, although the worst-case errors could still be characterized by Equation 6 by simply substituting the pitch angle, α, with a tilt angle in any direction. Another accuracy-degrading factor would be the lack of hard iron and soft iron distortion correction in the pitch and roll axes. The two most common ways to make a compass insensitive to tilt is to mount a two-axis compass on a gimbal and to use a three-axis, tilt-compensated compass.

COMPASS FEATURES

A good compass should have high dynamic range sensors. The need to compensate large hard iron offsets is not uncommon, and can account for 75% to 90% of the sensor's operating range. A good compass will have an operating range that is at least four times the Earth's magnetic field density.

A compass also should be able to resolve the Earth's horizontal magnetic field component to 1:115 for 1° resolution and 1:1146 for 0.1°. These numbers don't account for soft iron distortions that effect sensor gains and hard iron offsets, which would increase these requirements.

A good compass should be temperature-compensated. Magnetic sensors have temperature dependencies like most sensors. Fortunately, the sensor's common temperature effects drop out of the heading calculations because the arctangent function's input value is a ratio of the sensor pair (y/x). Sensor nonlinear temperature effects, external hard iron offsets, and soft iron distortions do not share this temperature cancellation characteristic. Of course, a compass that has hard iron and soft iron compensation is essential. The necessity of tilt compensation depends on the robot's requirements and your budget.

V2Xe COMPASS MODULE

I used the V2Xe compass in the Mini Rover 7 robot. This is a 1″ square electronic compass module that uses an SPI interface as a means of communication. It consumes less than 3 mW of power, and has an output resolution of 0.01° with a heading accuracy of 2°. Its field measurement range is about 20 times that of the Earth's field, which means that it can operate with extremely large hard iron offsets that are common in robotic applications.

The V2Xe compass can be calibrated using one of two methods. Distortion-compensation coefficients along with declination settings are stored in non-volatile memory. The V2Xe can provide raw sensor data and compensated field magnitude. It has an adjustable digital low-pass filter for heading.

A continuous calibration is the simplest calibration method to perform on the V2Xe. Send the calibration start command to the V2Xe to begin

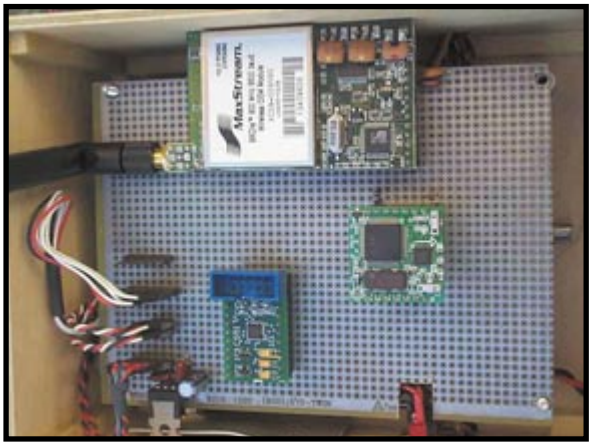


Photo 2—*I thought you'd like a look inside the Mini Rover 7's enclosure.*

the calibration process, rotate the robot in one or two complete circles, and then send a calibration stop command to end the calibration process. After completing the calibration, you can retrieve the heading data from the V2Xe as necessary.

V2Xe IN THE MINI ROVER 7

The Mini Rover 7 depicted in Photo 1 is a working scale model of the NASA/JPL Rocky 7 Mars rover. It incorporates six drive motors, and it has a rocker-bogey suspension as well as a zero turn radius steering system that's capable of rotating the rover in place. The robot contains a Maxstream 900-MHz wireless modem for receiving commands and transmitting data to and from a terminal program on a host PC, which has a matching wireless modem.

The Mini Rover 7 also sports a C8051F330 microcontroller that's mounted on a CStamp, which is a BASIC Stamp II socket-compatible module. It contains two regulators and a JTAG programming/debugger port connector. I used the free integrated development environment and evaluation C compiler supplied by Silicon Laboratories along with its EC2 serial-to-JTAG adapter. The wiring diagram for the robot is shown in Figure 6.

Photo 2 shows the placement of the compass in the Mini Rover 7. What is not clear from this photo is that I raised the compass to distance it from the motors and current-carrying wires in an effort to reduce distortion. Additionally, the wires are twisted in complimentary pairs to alternate the magnetic field direction and to minimize the distance of the radiated field generated by current flow; this effectively reduces offsets to the measured field.

The end result is illustrated by the red trace in Figure 5, which shows that the robot's magnetic signature has only a small hard iron offset. The compass can easily compensate for this simple offset mathematically after a calibration is performed.

To explore the compass's performance in various environments, I wrote a simple program to perform basic moves and transmit the robot's heading, as well as the measured field magnitude and raw sensor values. You may download the code from the *Circuit Cellar* ftp site.

The wireless modem that links the Mini Rover 7 to a remote terminal enables you to use the keyboard on the remote terminal to send the Mini Rover 7 commands for maneuvering and to select the type of data you want the robot to report. For instance, sending a "c" to the robot tells it to begin calibrating of the V2Xe compass. An "s" stops the calibration process and any movements. Sending an "h" tells it to include only heading data in its periodic data transmissions.

You can command the robot to rotate in a circle ("r" or "l"). It also can send raw sensor data ("b") in its datastream, which I was able to cut from a terminal program receive buffer window and paste into a spreadsheet program to create the x-y magnetic signature plots shown in Figure 5.

Another good test is to command the robot to move forward in a straight line. You can then observe the heading and magnitude values for irregularities. For this test, I commanded the Mini Rover 7 to travel down a hall in my house and into the living room. Along the way, the robot passed the kitchen. As it passed the backside of the refrigerator, the heading reported by the compass drifted.

1-800-492-2320

PRODUCT HIGHLIGHTS

Products & services that bring fast, simple wireless to OEMs.



Flexible Mesh RF Network

New MeshRF™ protocol supports rapid deployment, scalable architecture and expandable range. This advanced system is embedded into AeroComm's compact, long-range 900MHz transceivers. Call **AeroComm** for details, (800) 492-2320.

- *Peer-to-peer—**no need for a master.***
- *Self-healing—**no connection failures.***
- *Auto-routing—**instantly redirects data.***
- *Secure links—**FHSS, data encryption.***



RF Links for Serial Devices

ConnexLink™ stand-alone transceivers can be set up in minutes to cut the cables between RS232/485/422 devices. Their flexibility, durability and low price suit both industrial/commercial applications. Contact **AeroComm,** (800) 492-2320.

- *Global acceptance: **2.4GHz, 900MHz***
- *Point-to-point or point-to-multipoint.*
- ***Small & portable** for mobile settings.*
- *Software enables **custom configuration.***

WWW.AEROCOMM.COM

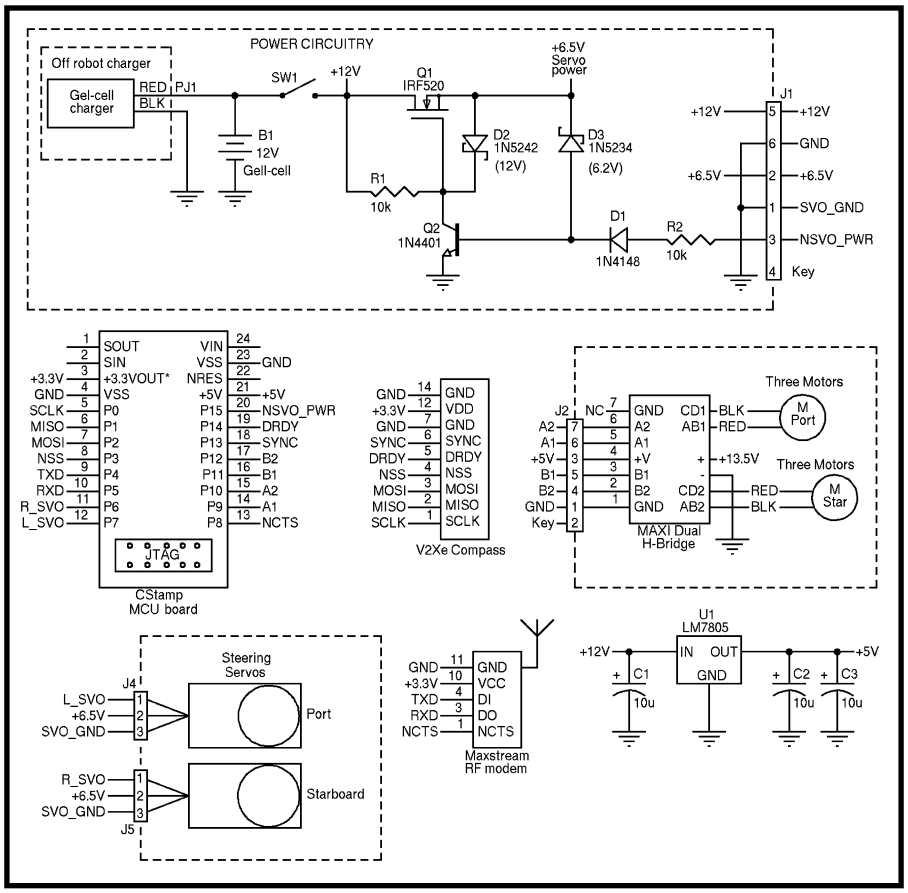


Figure 6—*Because I like to make my robots as expandable and serviceable as possible, most subsystems are modularized and plug into the main board using connectors.*

The magnitude was simultaneously changing, therefore signifying a magnetic field anomaly.

Figure 7 is a graph of the distance the robot traveled versus the heading and magnitude. Notice that the field magnitude is a good indicator of magnetic anomalies, which can be used to determine the quality of the compass heading information.

I performed an outdoor test that proved more successful than the indoor test. The outdoor environment didn't reveal any of the magnetic disturbances that I had observed indoors because I didn't encounter manmade iron objects.

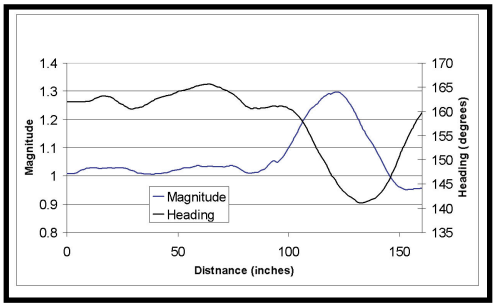


Figure 7—*Here you have the robot's distance from the origin versus heading and magnitude during a straight-line test with a soft iron distortion effect near the 125″ mark.*

THE NEXT MOVE

The program listing posted on the *Circuit Cellar* ftp site also contains a PID-based steering control system that keeps the robot on a steady course—if it doesn't encounter magnetic anomalies. The next step for this project is to add two wheel encoders to create a differential odometer. I plan to experiment with this data along with compass heading and magnitude information in a weighted averaging algorithm to increase heading accuracies.

Joseph Miller is a Senior Electrical Engineer at PNI Corporation, where he designs consumer and OEM sensor- and network-based products. He earned an A.S.E.E. degree from the Heald Institute of Technology in San Francisco, and has been an electrical engineer for approximately 20 years. Joseph enjoys competing and demonstrating his robots. You may contact him at compass@sonomatron.com.

PROJECT FILES

To download the code, go to ftp.circuitcellar.com/pub/Circuit_Cellar/2004/165.

REFERENCE

[1] Garmin Ltd., www.garmin.com/aboutGPS/.

RESOURCES

H.R. Everett, *Sensors for Mobile Robotics: Theory and Applications*, A.K. Peters, Wellesley, MA, 1995.

R.B. Langley, "The Magnetic Compass and GPS," *GPS World*, September 1, 2003.

E. Kaplan ed., *Understanding GPS: Principles and Applications*, Artech House, Norwood, MA, 1996.

M. Miller et al., *The Personal Robot Navigator*, A.K. Peters, Wellesley, MA, 1998.

PNI Corp., "Multipoint Calibration Primer," 1001766 R01, 2003.

U.S. Geological Service (USGS), geomag.usgs.gov.

SOURCES

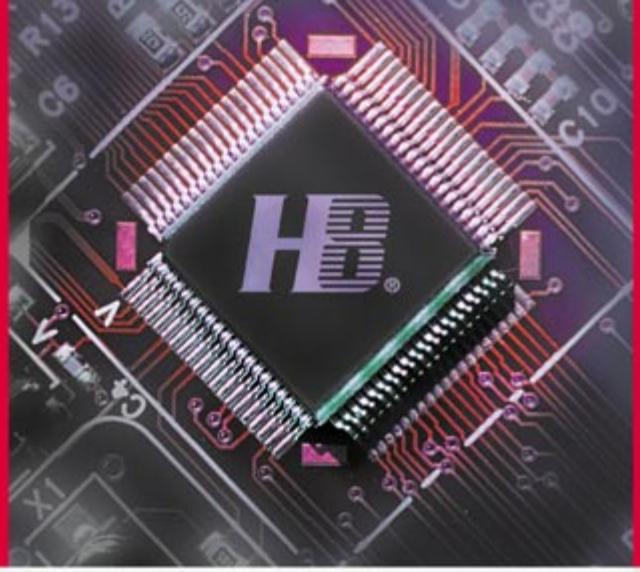
CStamp Module
Ken Gott
cstamp51@earthlink.net

Maxstream Wireless modems
Maxstream
(866) 765-9885
www.maxstream.net

MAXI Dual H-Bridge two-motor driver kit
Mondo-tronics, Inc.
(800) 374-5764
www.robotstore.com

V2Xe Compass module
PNI Corp.
(707) 566-2260
www.pnicorp.com

C8051F330 Microcontroller
Silicon Laboratories
(877) 444-3032
www.silabs.com



Everywhere you imagine. RENESAS

MCU Solutions Matched To Your Design Imagination

H8 Advantage

- **Single architecture spanning 8-, 16- and 32-bit MCUs**
- **Wide range of Flash memory options from 8KB to 1MB**
- **Widest selection of on-chip peripherals in the industry**
- **Very low power consumption, suitable for battery-operated applications**
- **Very low EMI and EMS**
- **Integrated safety mechanisms: POR, LVD and WDT**

Fully Upward Compatible

H8SX
87 instructions
11 address modes
- Data bus width extended to 32-bits
- Enhanced memory operation support
- Enhanced address modes
- Enhanced bit operation instruction

H8S/2600
69 instructions
8 address modes
- Four MAC instructions added

H8S/2100 • 2200 • 2300
65 instructions
8 address modes
- Multiple Load/store instruction

H8/300H
H8 Tiny
64 instructions
8 address modes
- 8/16/32 bit arithmetic and logic instructions
- Enhanced push and pop instruction
- Signed/unsigned MULT/DIV instruction
- Signed/unsigned extension instruction
- Enhanced trap exception handling

H8/300L • 300L SLP
57 instructions
8 address modes
- 8/16-bit arithmetic and logic instructions
- Unsigned MULT/DIV instructions
- Powerful bit manipulation instructions

Low-cost Renesas Toolchain

- **Fully integrated suite of hardware and software tools**
- **Highly-optimized C/C++ compiler – FREE (64K version)**
- **Low-cost toolkit (target board, on-chip debug emulator and compiler) starting at $248**
- **Easy online access to H/W and S/W tools – FREE**

http://america.renesas.com

Device	Board	Serial Emulator (E7)	Compiler C/C++
HD64F38024H	US38024-BAG1 $49*	HS0007TCU01H $199*	FREE (Download from Web (64K version)
HD64F3672FY	US3664N-BAG1 $49*		
HD64F2327BVFBL	Kit includes board, JTAG emulator, compiler. US2329-KIT1 $1,529.73*		

* Suggested volume independent resale price in US dollars from Renesas authorized North American Distributors only.

Get your product to market faster using Renesas Interactive!

Renesas INTERACTIVE

Evaluate, test, and learn about Renesas products in an interactive online environment.

www.renesasinteractive.com

Popular Devices

Device	Bits	Package	Freq.	Vcc (V)	Flash	RAM	Timer	10-bit A/D	Serial	Features	Price*
HD64F38024H	8-bit	80-pin TQFP	16MHz	2.7 - 3.6	32KB	2KB	9 ch	8 ch	1 ch	LCD, Sub-clock	$4.68
HD64F3672FY	16-bit	48-pin TQFP	16MHz	3.0 to 5.5	16KB	2KB	6 ch	4 ch	1 ch		$2.75
HD64F2327BVFBL	16-bit	128-pin QFP	25MHz	3.0 - 3.6	128KB	8KB	24 ch	8 ch	3 ch	D/A, DMAC	$5.99

* Suggested volume independent resale price in US dollars from Renesas authorized North American Distributors only.

BasicCards 101 (Part 2)

Use in a Liquid Nitrogen Monitor

Now that you're familiar with ZeitControl's BasicCard, it's time to take a closer look at the process of programming one and incorporating it in a design. Brian shows you how he integrated BasicCard technology in the design of a liquid nitrogen generator.

Last month, I introduced you to ZeitControl's BasicCard, which is a smartcard that is programmed in BASIC. In Part 2, I'd like to further explore the programming of these cards and discuss a project.

I built a device that allows you to use BasicCards like debit cards to control the dispensing of liquid from a liquid nitrogen generator/storage tank. The idea is to issue each liquid nitrogen user here at Dalhousie University with a BasicCard. Each card is personalized by entering the user's name, account number, and a zero balance. Card personalization is done on a PC using a Windows-based application. This user-friendly application interfaces to the BasicCard using the CyberMouse reader that comes with the development kit.

To access some liquid nitrogen (LN_2), insert your BasicCard into the custom controller connected to the LN_2 generator. The controller checks to ensure that the card is properly personalized for this use, and displays your name, account number, and the number of liters of LN_2 previously consumed during the current billing period. You then enter the amount of LN_2 desired on the controller's keypad. After updating the BasicCard EEPROM variable, which stores the accumulated LN_2 usage, the controller activates a relay that opens a valve and lets the liquid nitrogen flow for a long enough period to dispense the necessary amount of LN_2.

Periodically, the BasicCards are collected and another PC application is run that zeroes out the accumulated LN_2 total. It transfers that figure to a table that is used for the actual billing process.

In the past, an honor system was used in which users entered their LN_2 usage on a sign-out sheet posted next to the generator. We lost some LN_2 billings here at the university because of dishonesty and low-ball estimates of the actual amounts taken. Also, there was a significant amount of clerical time spent tallying all of the entries on the daily sign-up sheets.



Photo 1—*Check out the monitor before it's mounted in its cabinet. Note the BasicCard sticking out of the card socket on the right-hand side.*

CIRCUIT DETAILS

Because the prototype would be the only unit I would need, I wanted to use a commercial development board and just add the few custom parts needed for the design. I chose the Futurlec 8535 development PCB because it's inexpensive, contains headers for commonly used peripherals, and has a large enough prototype area to mount the few extra components I needed to complete the design.

Figure 1 is a diagram of the circuit. Some of the circuitry on the Futerlec board that is not required for this project is not shown in the diagram. For example, the in-circuit serial programming port for the 8535's flash program memory is not shown. There is one small change that I had to make to the Futurlec board itself. The board comes fitted with an 8-MHz MCU clock crystal. I changed that to 3.579 MHz so that I could use a common clock for both the MCU and BasicCard, which has a 5-MHz maximum clock rate and communicates at 9600 bps when clocked at 3.579 MHz.

I mounted an Amphenol C702 10M0008 2834 smartcard socket in the prototype area of the PCB. This socket contains an NC switch that opens up when a card is inserted, making it easy for the MCU to know when to establish communications with the card.

Although I connected this line to port D3 of the '8535 (INT1), I don't use this pin as an interrupt input, instead I poll the pin to see when a card is either inserted or removed. There are a few different phases in the program, and the removal of a card must be handled differently in each case, so using an interrupt to signal card-in-socket status was not ideal for this project.

The BasicCard interfaces to the '8535 using only two port lines. The *RESET input to the BasicCard is connected to port D2, and the I/O line connects to port D4. As I mentioned earlier, the BasicCard gets its clock signal directly from the '8535's XTAL1 pin, which is its oscillator buffer output. Although you have to be careful tapping off a clock signal from an MCU's oscillator output, the BasicCard presents, in this case, such a small load that the oscillator is unaffected by its presence.

The user interface is the standard keypad/LCD. I had a Grayhill series 88 keypad on hand, which contains the numbers in a telephone-like arrangement, as well as several extra keys labeled Enter, Clear, etc. that provide all the necessary input functions. Eight lines are needed to interface this Matrix keypad, and I used port A for that purpose. Doing so meant I could not make use of the eight-channel ADC contained in the '8535, but I didn't need an ADC for this project. The Futurlec board has a 10-pin header connected to this port, which made it easy to connect the keypad using a ribbon cable.

I didn't need a large LCD, because there isn't a lot of information to display. Therefore, I used a 2 × 16 display, and connected it to the '8535 in 4-bit mode using six lines of port C. The Futurlec board also provides a 14-pin header for the LCD, with all the signal and power wiring taken care of, as well as POT1 for contrast adjustment.

Apart from wiring a few lines to the smartcard socket, all the circuitry I needed to add was the driver and relay to activate the dispensing valve. This was a solenoid valve that required 120 VAC to operate, so I decided to use a relay to keep any inductive kickback from its coil away from the MCU circuitry. Any relay with a 12-V coil and contacts able to switch 1-A AC will do for K1. If the coil needs more than a few hundred milliamps to operate, you'll need a larger transistor than a 2N3904 for Q1. Diode D1 is placed across the relay coil to protect Q1 from the spike that occurs when the relay turns off.

The 5-V power supply regulator is contained on the Futurlec PCB, but the power transformer, bridge, and filter capacitor are mounted externally. Photo 1 shows the development board with the extra components needed for this project.

The development board comes with a MAX232 for RS-232 communications purposes. Although I did not use this feature, it would be easy to wire the controller to a remote PC, for example, if you want to log all the transactions in real time to a computer file.

Atmel's AT90S8535 is a versatile MCU. In some respects it is overkill for this application. With the program written in compiled BASIC, much of its 8-KB flash program memory is unused. It contains 512 bytes of EEPROM, which is unused apart from a few bytes used to store some valve time calibration parameters. The eight-channel ADC, UART, and SPI functions weren't needed either. However, the assembled Futurlec development board, including the '8535 MCU and programmer cable, cost less than $30, so I thought it made a lot of sense to do things this way.

INTERFACE TO THE AVR MCU

Before I ordered the BasicCard development kit, I wanted to be sure I could use the cards with my MCU of choice, the AVR series from Atmel. The BasicCard development kit contains software to program the cards themselves, as well as a few different programs to develop PC applications. However, the kit doesn't provide any

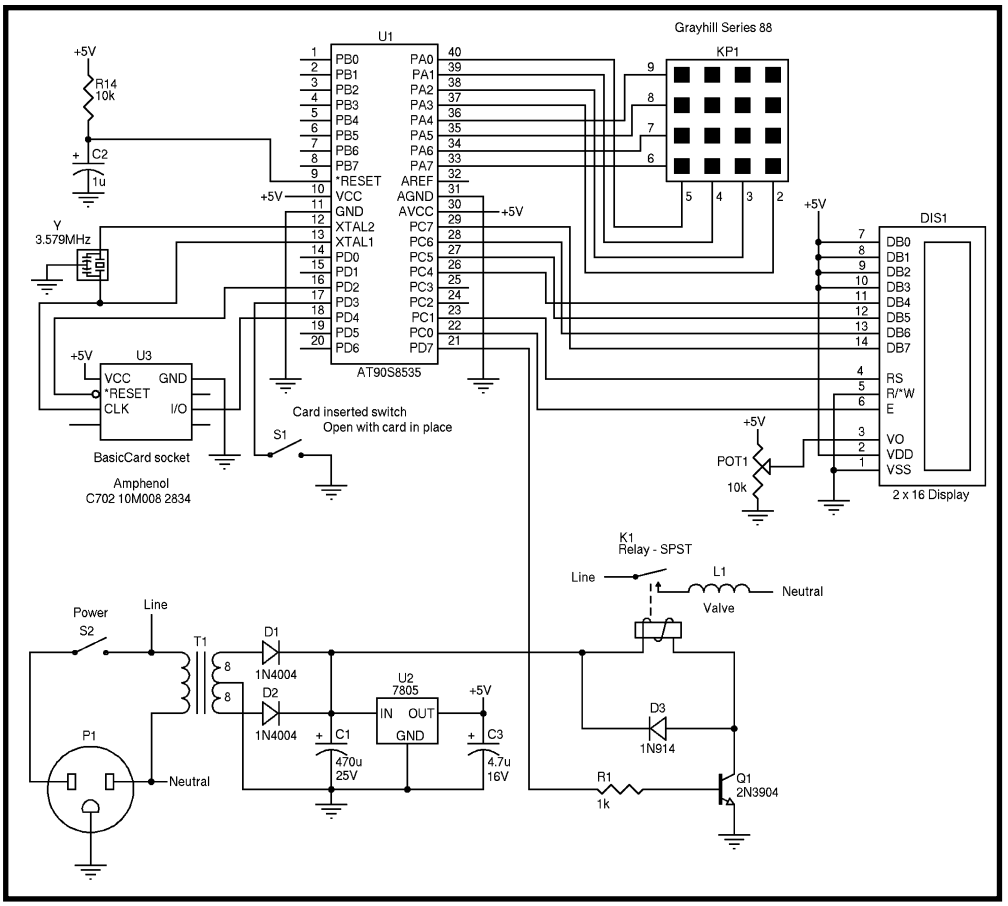


Figure 1—*The liquid nitrogen dispensing monitor was built on a Futurlec 8535 development board. Apart from the card socket, keypad/LCD, and relay driver, most of the circuitry already exists on the development board.*



Command Name	Description	Used by
PersonalizeCard	Initial card with name, account number, and balance	PC
IncreaseAmount	Increase balance by amount dispensed	AVR
GetCardData	Read name, account number, and balance	AVR and PC
CancelLastTransaction	Self-explanatory	AVR
ReadLastTransaction	Needed by previous command	AVR
PRDisplay	Display information on key fob balance reader	Balance reader

Table 1—*These six commands are implemented in the BasicCard for this application. As you can see, some commands are used by both the LN_2 dispensing controller and the PC application that personalizes and reads the cards.*

support for interfacing them to MCUs, apart from providing a technical reference document describing the ISO/IEC 7816-3 standard that the cards use for communication. From what I could see, it would have been quite time-consuming to personally write a suitable driver to allow AVR MCUs to talk to BasicCards. As luck would have it, the developer of the BASCOM AVR compiler (the language I use exclusively for all of my AVR-based projects) had a beta version of a driver library available. This was one of those good news/bad news situations. I was able to get the library free from the developer, but I was, at that time, the first and only beta tester! I ended up figuring out on my own some of the patches needed to make it work.

To interface a BasicCard to an AVR MCU using the BASCOM compiler, you must run BASCOM version 1.11.6.8 or newer. You must also load the BasicCard library (available from MCS) into BASCOM's LIB folder.

Next, within your BASCOM program, you must do the following four things. First, use the `CONFIG BCCARD` command to tell the driver which port pins you have connected your BasicCard socket to. This is simple because you only have to tell it which port you are using and to which pins the *RESET and I/O lines are connected. This is clearly described in the documentation that comes with the library.

Second, you must declare each of your BasicCard commands to the BASCOM program using the `BCDEF` command. Because you have already written the BASIC code that runs in the BasicCard, you will have already defined the commands that the BasicCard recognizes. These command names, followed by their parameter lists, are used with the `BCDEF` command.

Next, instruct the compiler to use the proper data rate. Use the $BAUD = 9600 and the $Crystal = 3579000 directives. Finally, issue a `BCRESET` command to initialize the BasicCard as soon as you see that a card has been inserted into the card socket.

After this preparation/initialization, all you have to do to access a particular command in the BasicCard is use the `BCCALL` procedure. This is similar to any BASIC procedure, except that it includes some BasicCard-specific parameters. The following is the syntax for this command:

```
BCCALL name( nad , cla, ins, p1,
  p2 [param1 , paramn])
```

where `name` is the name of the procedure in the BasicCard to call. It must be defined first with `BCDEF`. The name used with `BCDEF` and `BCCALL` does not need to be the same as the procedure in the BasicCard, because the CLA and INS bytes actually tell the BasicCard which command to execute. However, it makes sense to use the same names for consistency.

`nad` is the node address byte. BasicCards respond to all node address values; zero is used here as a default. `cla` is the class byte. It is the first of a 2-byte CLA-INS command. It must match the value you used for the command in the BasicCard program itself.

`ins` is the instruction byte. It is the second of the 2-byte CLA-INS command. The same consideration applies as for the CLA byte. `p1` is parameter 1 of the CLA–INS header. (Use a zero for your purposes.) `p2` is parameter 2 of CLA-INS header. (Again, use a zero for your purposes.) `param1` through `paramn` are the parameters you want to pass to the BasicCard (as required by the command).

The BasicCard operating system

employs a comprehensive error-reporting scheme. This is necessary in any device used for commerce, particularly in a device that you can pull out of its socket in the middle of program execution! Describing the error handling is beyond the scope of this article, but I will mention that the BCCARD library supports it to some extent. In essence, if the BasicCard returns an error, the library will set the BASIC variable ERR and two status bytes—SW1 and SW2—will contain error codes. Many of these codes are predefined in the BasicCard protocol, but you can also set the value of these variables yourself within your BasicCard program should your code encounter an error condition.

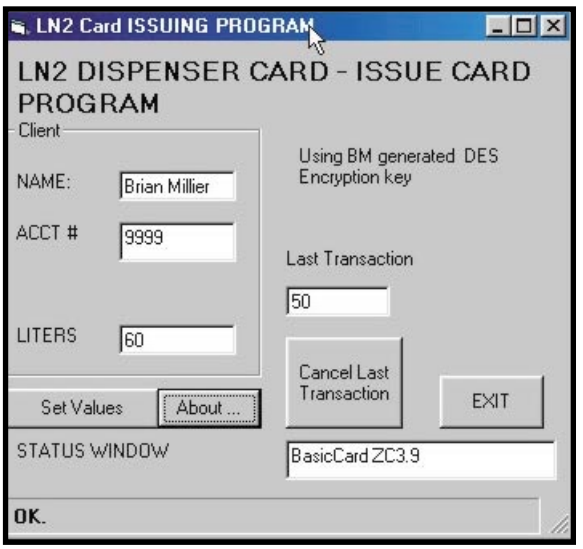


Photo 2—*This is a screen shot of the PC application that initially personalizes the BasicCard and reads out its balance periodically.*

In most designs, the card-in-socket switch will notify the program if the card is removed during use; but, if an actual data transaction is in progress when this happens, the BasicCard driver within BASCOM may hang the program until a time-out interval passes. Depending on a number of variables, this might take up to a minute, so beware of this during debugging.

Before trying your own Basic-Card program, I recommend that you load the sample Bccard.bas program into your AVR MCU, program a BasicCard with the Calc demo program provided with the development kit, and try that combination. If everything is satisfactory, you will see the results of the communication between the two devices. You can then enter some numbers into your AVR MCU, and let the BasicCard act as a calculator.

CARD FIRMWARE

Table 1 shows a list of the commands that I've defined in the BasicCard for this application. The last column in this table indicates where the command is used: The LN_2 Dispenser is labeled "AVR." The PC application personalizes the card and later reads the accumulated total usage (labeled "PC"). The balance reader is the little key fob device that comes with the development kit.

The PersonalizeCard command, which the PC application uses to initialize the card with user information and zero out the balance, is the only command that requires encryption in order to work. This guards against the possibility of fraudulent card production. ZeitControl's Visual Basic API, which is part of the development kit software, includes support for encryption. This command is only used by the PC program that personalizes the cards, and that is written in Visual Basic. Although not a concern in this application, the fact that the BASCOM AVR BasicCard library does not currently support encryption could be a disadvantage in other applications.

For this application I used the BasicCard ZC3.9 enhanced card, because it was included with the development kit. However, the lesser models of the card would work as well because this is a simple application. In the first part of this series I gave you some hints on how to program the BasicCards by supplementing the instructions in the user manual.

AVR FIRMWARE

Next, I'll describe the program that runs on the '8535, which controls the LN_2 dispenser. After some initialization of the ports, the program basically enters a loop. The program prompts you to insert the card and then waits for this to happen. The ReadCardData command is then issued to the BasicCard, and the user information contained in the card is displayed on the LCD. You are then prompted to enter the amount of LN_2 needed. This amount is first added to the BasicCard's balance variable, and then the program waits until you indicate you're ready for the dispensing to start. It then activates the relay for the amount of time necessary to dispense the necessary LN_2.

This is where calibration is needed. When dispensing LN_2, it takes a little while for the transfer tube to become chilled enough to pass the liquid. Thereafter, the liquid flows steadily, with the amount dispensed being proportional to the time the valve remains open. After this amount of time, the program waits until the card is removed, and then goes back to the start of the loop. Calibration involves determining the initial chill time and the time/liter parameters empirically, and storing them as calibration parameters.

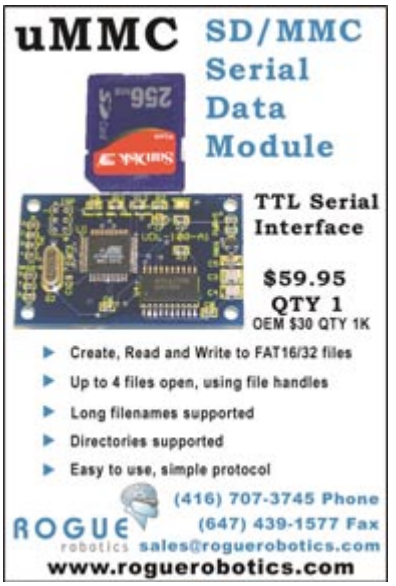
It's important that this calibration can be done in a way that's both convenient for the operator of the generator and secure from tampering. To that end, I designed the program so that when it sees a card with an account number equal to zero (the operator account), it goes into a calibration routine and prompts the operator for an initial (transfer tube chilling) time, as well as the number of seconds required to transfer 1 liter of LN_2. These parameters are subsequently saved to EEPROM in the '8535 and used in all future liquid transfers, unless the calibration is changed.

If you remove the card before entering the desired number of liters to dispense, the LCD indicates that the card has been removed prematurely and returns to the start of the loop.

CARD ISSUER PROGRAM

Photo 2 shows a screen capture of the Visual Basic Card Issuer application. This program is run on a secure PC because it is used to personalize the BasicCard with the username and account number, and to set the initial







balance variable to zero. Obviously, anyone with access to this program can insert their card and wipe out its balance easily.

After start-up, the program waits until a card is inserted. It checks that the card has the proper program in it for this application. It does this by reading the `applicationID` variable contained in the card's EEPROM. If it matches, the program displays the user information and the current balance. If it's a new card that's been programmed with the LN_2 application but not yet personalized, it will show blank fields and allow you to enter the applicable name/account information and also a balance amount (typically zero for a new account).

There is also a provision to cancel the last transaction. The amount of the last transaction is shown, and there is a button to subtract this amount from the accumulated total. The purpose of this function is to allow the operator of the machine to cancel a transaction if the machine fails to deliver LN_2 as requested (i.e., the tank is empty or a valve sticks).

I also included a box that displays the type of card that has been inserted into the reader. This card-specific information is returned by all smartcards as part of the answer-to-reset (ATR) routine, which is invoked at card insertion.

GIVE IT A TRY

I must admit that I was intrigued by the fact that these tiny BasicCards contain more resources (except for I/O) than the typical MCUs I routinely use. Furthermore, all of this capability is sandwiched into the small area under the gold contact pad. The rest of the card is just plastic filler to make the card easy to handle.

I am definitely in the compiler camp, and generally avoid MCUs like BASIC Stamps, which run as interpreters. However, for this purpose, I have to admit that the BasicCard interpreter design is quite useful.

I suspect that many readers, like myself, are not quite up to the task of designing their own device drivers for high-level PC languages like Visual Basic running on Windows. It was a different story in the past, when you could easily design an ISA card, plunk it into the PC, and access it using direct in and out instructions. Now, it's important that the device manufacturer supplies APIs that allow these compilers to easily interface to the device. ZeitControl certainly has taken care of this aspect, and it provides an extremely inexpensive development kit. For BASCOM AVR users, the necessary BasicCard drivers exist as well. So, there's really no reason not to give this a try if you have an application in need of a secure smartcard.

Brian Millier is an instrumentation engineer in the Chemistry Department at Dalhousie University in Halifax, Canada. He also runs Computer Interface Consultants. You may reach him at brian.millier@dal.ca.

PROJECT FILES

To download the code, go to ftp.circuitcellar.com/pub/Circuit_Cellar/2004/165.

SOURCES

C702 10M008 2834 Acceptor
Amphenol-Tuchel
(734) 451-6400
www.amphenol-tuchel.com

AT90S8535 Microcontroller, AVR assembler and simulator
Atmel Corp.
(714) 282-8080
www.atmel.com

90S8535 Development board
Futurlec
www.futurlec.com

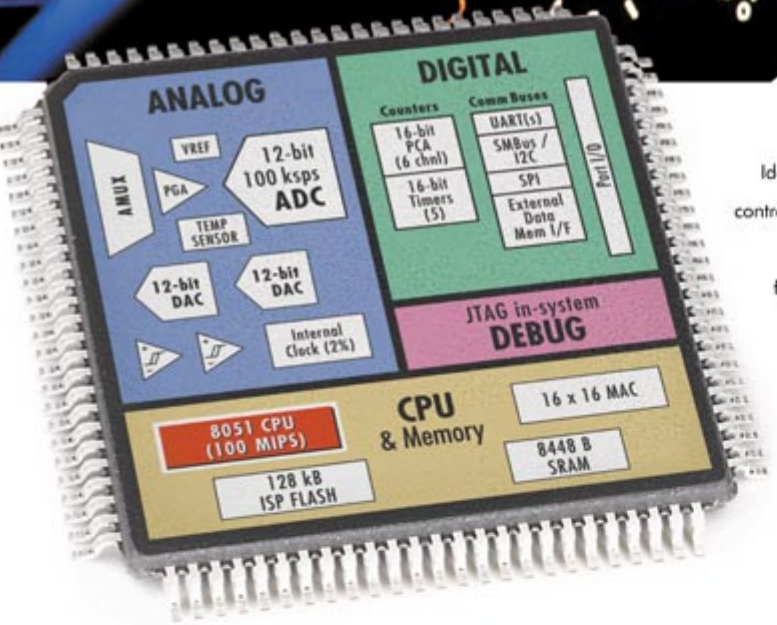
Series 88 keypad
Grayhill, Inc.
(408) 354-1040
www.grayhill.com

BASCOM AVR Compiler/programmer, BasicCard driver library
MCS Electronics (Holland)
+31 75 6148799
www.mcselec.com

BasicCard
ZeitControl Cardsystems
+49 0 571-50522-0
www.zeitcontrol.de

Precision Mixed-Signal MCUs with a supercharged 8051. Small. Fast. High-performance analog.





Ideal for data acquisition and process control applications, the Precision Mixed-Signal family of advanced MCUs features a supercharged 8051 CPU, high Flash densities, UART, SMBus and SPI serial interfaces, internal oscillator and numerous digital I/O options. On-chip analog circuits include multi-channel ADCs, DACs, voltage reference and a temperature sensor.

C8051F0xx - C8051F1xx	
MIPS	20–100
Flash	16–128 kB
ADC	12-bit, 8-ch, 100 kbps



Test drive Silicon Laboratories' MCU family by attending a hands-on workshop in your area.
To register call 1-800-677-7716, or sign up at: www.insight.na.memec.com/silabs-mcu.

WIRELESS | WIRELINE | OPTICAL | **MICROCONTROLLERS**

SILICON LABORATORIES

©2004 Silicon Laboratories, Inc. Silicon Laboratories and the Silicon Laboratories logo are trademarks of Silicon Laboratories.

www.silabs.com

Build a Small Robotics Platform

Inspired by the robots used by public works departments to investigate pipe damage in local sewage systems, Ingo set out to build his own robotic surveillance unit. Although small, this RCM3610-based robot boasts bump sensors and an odometer. Whether you need an inexpensive surveillance solution or an upgradeable base module, this robotics platform is the first step in the right direction.

If you have read my previous *Circuit Cellar* articles, you know that I'm interested in robotics. Having done some work for a couple companies that design sewer robots, I have always been intrigued by them. You may have seen sewer robot vans in your town; most municipalities own or contract them to survey sewage systems. These little robots are equipped with video cameras to examine pipes for damage. Some even have mechanisms to patch broken ones. Well, this type of system and my general interest in robots have inspired me to build a small surveillance robot.

In this article I'll describe my TeleBot, which is a small treaded robot I've been working on. It's based on the Tamyia bulldozer kit, which I purchased at a local electronics outlet (Fry's), but is available via mail-order sources on the 'Net. Photo 1 shows the complete robot with all of its widgets installed.

BASIC COMPONENTS

The Tamyia includes a chassis, DC motors, a transmission, and treads that can be used in different configurations. The transmission and motors also can be bought separately if you want to build your own chassis. The original kit was designed to be remote controlled via a battery and switch box that comes with it. And it includes a movable shovel, which you may or may not include in your robotics project.

The idea for this project is to build a useful robotics platform with relatively low-cost components that is extensible (i.e., the base configuration—chassis, motors, and odometer—is cheap to assemble, as robots go, and can be used to interface with a variety of controllers). Of course, you can expand it by adding sensors and actuators, like I did.

I used a Rabbit-based RCM3610 module for the controller. The module contains 256 KB of flash memory, 128 KB of RAM, and various I/O (serial, SPI, PWM, etc.). Most importantly, it has a 0.1" dual row header for mounting it on a circuit board, which I did. The main board, which is cut to fit in the chassis, contains extra circuitry like an H-Bridge to drive the motors and screw terminals to interface to the rest of the robot.

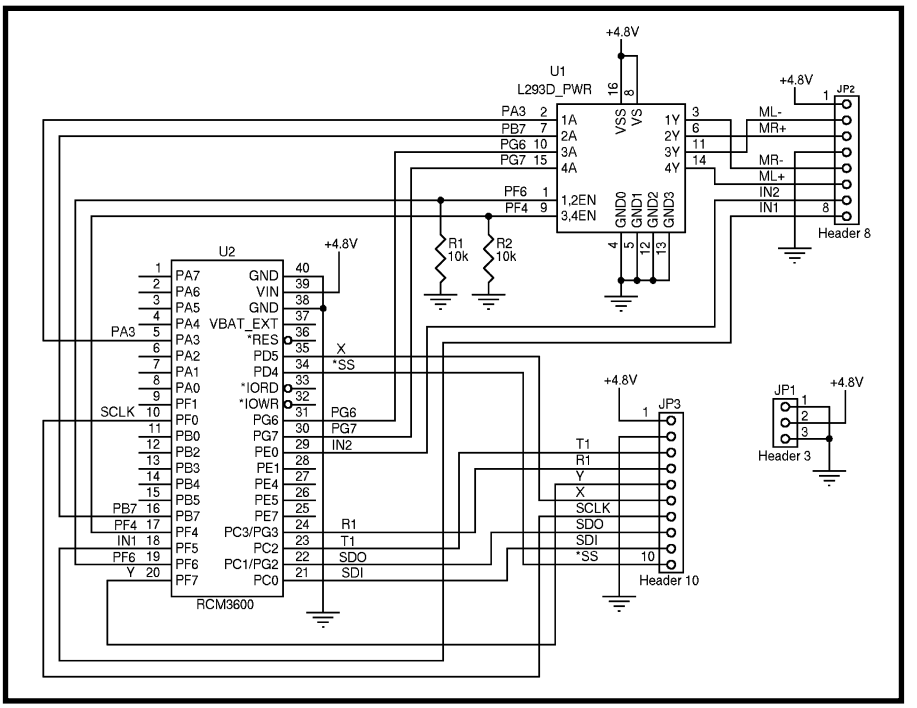


Figure 1—*The RCM3610 plugs into the main logic board. The L293D H-Bridge is used for motor control. Sensors are interfaced using miniature screw terminals (0.1"), which are really handy. Many of the I/O signals used are the same as on the starter board, which comes with the Imagine tools kit that contains the core module and C compiler for this project.*

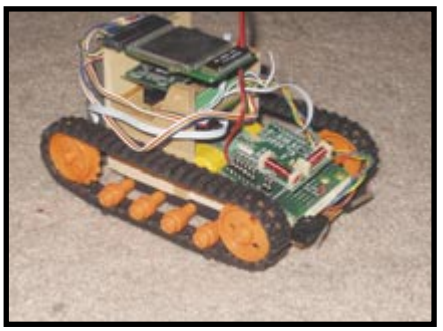


Photo 1—*The completed TeleBot bristles with sensors and wires.*

Besides the controller, the robot has a couple of bump sensors, an odometer made from a discarded mouse, a 915RF modem, and a digital compass to guide its way. For power I used a 4.8-V NiCd battery pack, like those used for RC models.

NOW FOR THE DETAILS

The chassis in the kit is made of wood, and it glues together easily. Wood may seem kind of quaint as a construction material for a robot, but it's almost perfect: you can cut and drill it easily; it's light; and most types of glue adhere to it easily. The pieces in the kit are precut and predrilled, so you don't have to do any of that.

The transmission, which has 3-V DC motors mounted on the rear of the chassis, and is neatly out of the way. So the rest of the chassis is available for the battery pack as well as all the electronics to make it go. The 3-VDC motors are perfect for driving from a 4.8-V power supply with a variable-speed H-Bridge.

I used an L293D dual H-Bridge (actually, it's a quad half H-Bridge) that's capable of driving motors at up to 36 V at approximately 600 mA—enough for what I need. H-Bridges, for those of you who have not used them, are circuit configurations to switch and reverse the polarity to a load like a motor to make it run forward and backward. There are two transistors (high-side and low-side) for each terminal of the load, four in total. The high-side transistor switches the load terminal to the power supply high side (4.8 V in this case), and the low-side transistor switches it to ground.

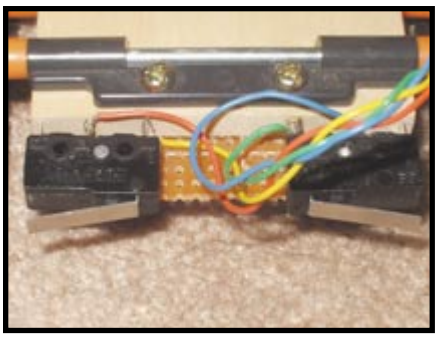
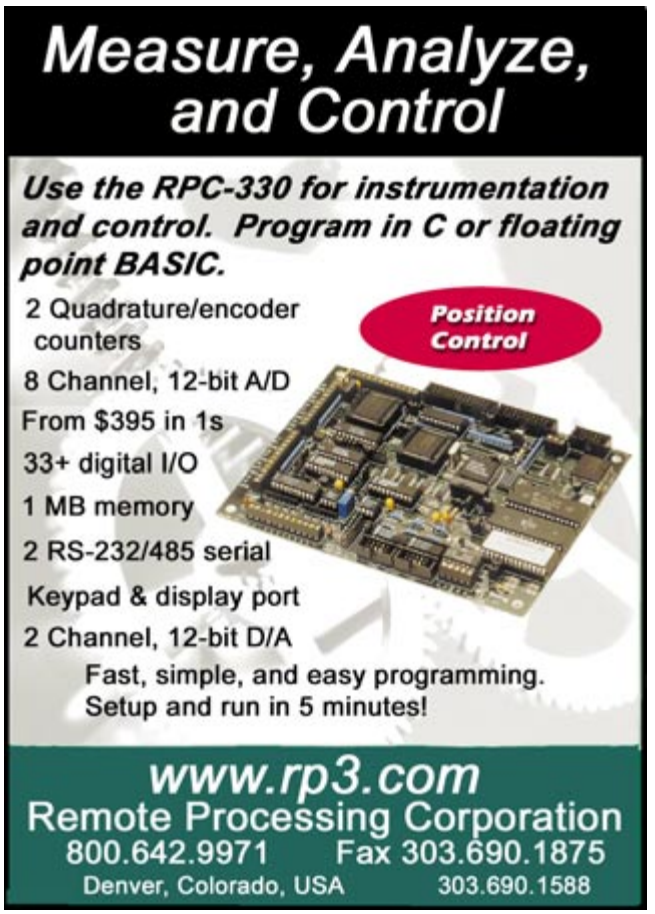


Photo 2—*I glued a simple bump sensor interface to the front of the robot.*

In order to make the motor run in one direction, one terminal is switched to power and the other to ground through the transistors. Reversing is easy by reversing the polarity. The chip I used contains four two-transistor pairs plus logic to make sure that each two-transistor set is only in one of three allowed configurations (off, high, and low).

The chip also contains thermal protection circuitry, which shuts down the power if the chip overheats. Finally, protection diodes are also provided on the outputs, which are essen-





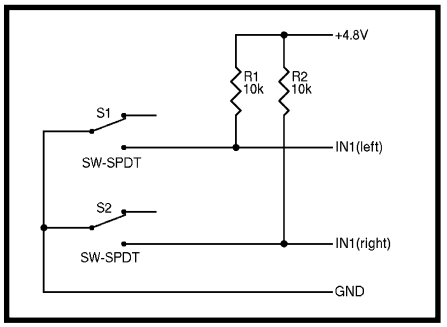


Figure 2—*The bump sensors are simple. Two micro switches are wired to two inputs.*

tial for inductive loads like motors and solenoids. The L293D, although it's fairly old technology and not as efficient as some of the newer H-Bridges available, is convenient to use and relatively inexpensive. It's also available in a DIP package, which made it perfect for this project because I wanted to mount it on the same 0.1" pad-per-hole prototype board as the CPU module.

You can drive both motors using one L293D. There are four control signals to drive each transistor pair and two enables that are shared with two of the pairs. The control signals are used to signal the direction in which you want the motor to go. The two enables (one for each motor) are wired to two PWM outputs on the CPU module. The PWM signal allows you to adjust the power to the motors in extremely fine resolution with minimal CPU overhead.

The Rabbit 3000 on the CPU module has four on-chip PWM timers that can run independently of each other. A 10-bit PWM register lets you adjust the width of the pulse from zero to 100% in 1024 steps for each channel. I commonly prescale and adjust the repetition rate of the PWM channels. Although the PWM hardware on the Rabbit is easy to use, there are library calls to make it even easier. In particular, there is `pwm_init(freq)`, which allows you to set the prescale register to the repetition frequency you want to use. The `pwm_set(chan,count)` routine lets you set the PWM value (0 to 1023) for a particular PWM channel (0 to 3).

To make it even easier, the library that comes with the development kit for the starter kit for the RCM3610 CPU module also has a routine to deal with the H-Bridge in addition to the PWM. There are two simple routines: `HBmotorinit()` initializes the I/O and PWM, and `HBmotorSet(chan,speed)` lets you control the speed and direction of two motors using a floating-point setting between –1 and 1. All I had to do was wire up the L293D like it is on the starter board to use this API (see Figure 1).

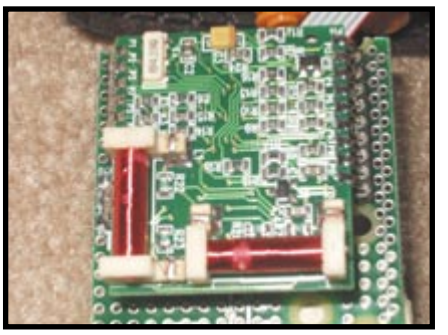


Photo 3—*Take a look at the complete electronic compass module.*

SENSORS

After I had the motor controls working sufficiently, I added sensors to the robot. The first sensor is a bump sensor. I used two micro switches that I glued on a small PCB and then glued to the wooden chassis in the front of the robot (see Photo 2). Check out Figure 2 to see how they are wired up.

Basically, each switch input is pulled to a logic high. When the switch is activated, the input is connected to ground and shows up as a logic zero to the processor. I used two separate switches so that I can sense the side to which the obstruction is located. This aids in deciding what to do.

With the motor control and bump sensor, I was now able to write an autonomous control program for the robot. Listing 1 shows a simple program. It starts up waiting for you to activate it by pushing and releasing the two bump sensors. The robot then lurches into action, going forward until it senses an obstruction. Then it stops, backs away from the obstruction, and continues moving forward. To make the bump sensors more effective, you can add rails that extend vertically and horizontally to give it a large area to cover.

Even with this simple algorithm, the robot can find its way around. It's



surprisingly effective and duplicates some of the behaviors of an ant looking for food and a way to get around obstructions. You can modify the program to play around with other strategies. Half the fun is trying different things and then being surprised at the results. The other half, in my opinion, is building it in the first place. If nothing else, a bumping robot is a fun toy to introduce to a cat, especially if you tie a string to the robot and have it drag the string around a room.

FOR GOOD MEASURE

I wanted to use some kind of encoder to measure the speed of the motors and distance traveled. Mechanical mice are a great source for encoders. Optical encoders in a mouse typically use an LED and optical sensor that detect when the beam is obstructed with a slotted wheel. Two wheels, x and y, are driven by the ball of the mouse. Because the wheel has a fixed number of slots, a counter can be used to measure the rotation of the wheel, and thus the distance traveled.



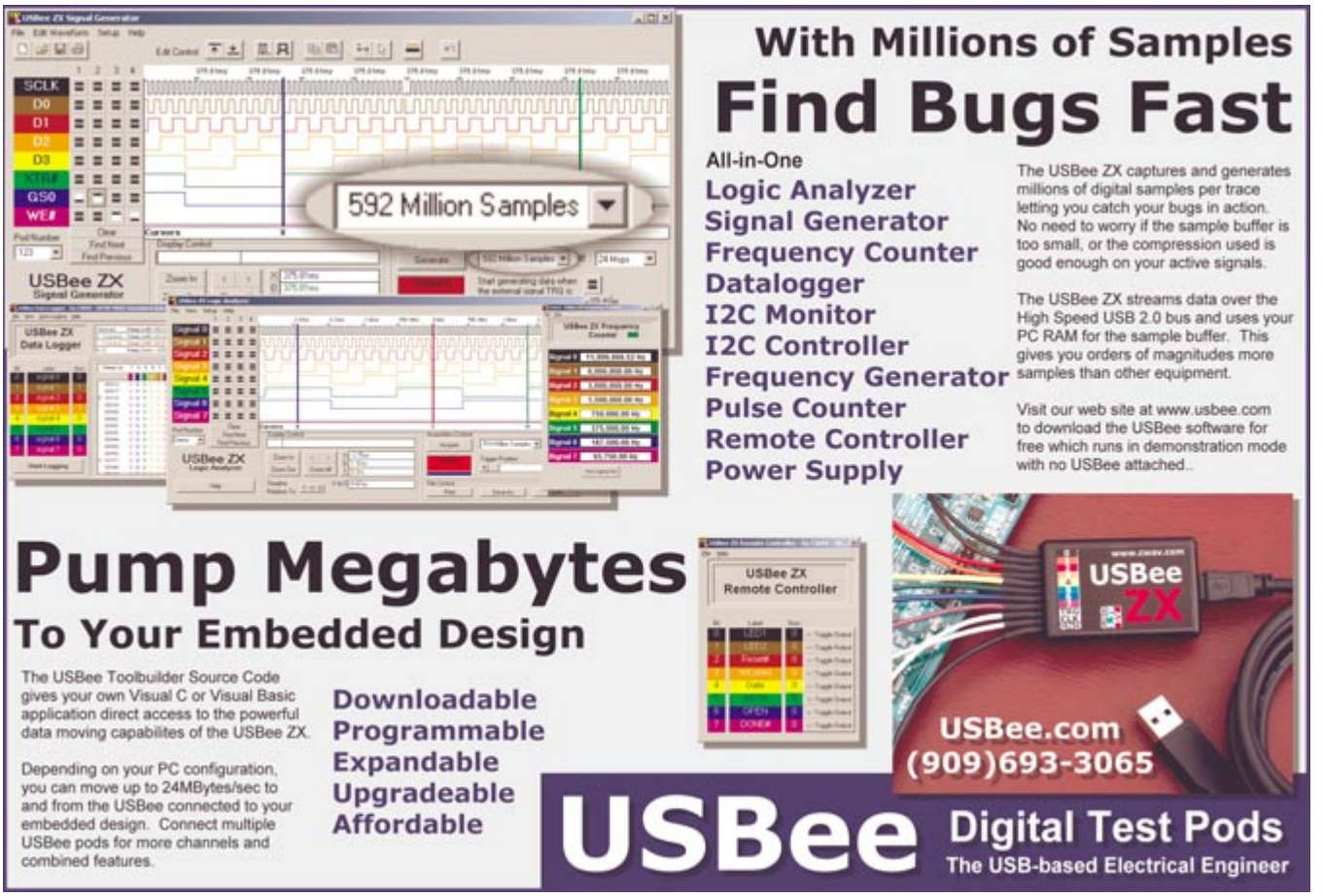
Photo 4—*The RF modem is complete with a transceiver and antenna.*

The optical encoders in mice actually have two sensors per wheel that are offset. The optimal offset is about one quarter of the distance between adjacent slots; therefore, these are called quadrature encoders. Depending on the direction, one pulse is generated in front of the other. It can measure the direction of travel as well. Coupled to a bidirectional counter, a position count can be maintained.

The Rabbit processor actually has dual hardware quadrature decoder registers, which work with a quadrature detector. However, I used only one of the sensors. I keep track of the direction traveled with software. A more sophisticated control algorithm could use quadrature information to build a highly accurate positioning system.

While trying to adapt a surplus mouse encoder, I broke the encoder wheel that came with the mouse. Making a new encoder wheel seemed like a lot of trouble, so I opted to build a simple encoder using a photo-reflective detector. A photo-reflective detector incorporates an LED and a phototransistor in one package; it is designed to work best when some reflective material (target) is placed in view—a short distance in front of the detector.

To measure the rotation of the wheel, I made a small reflective disk using some silver ribbon material left over from the holidays. I glued it on a small disk of cardboard and painted some dark lines using a magic marker on the disk and attached it to the wheel. The detector turns on when it sees the reflective material; it turns off when the opaque marker interrupts the beam. There is a detector



www.sensorsexpo.com

sensors expo & conference

The Leading Industry Event for Sensing Technologies

Conference: June 7 - 10, 2004 • Exhibits: June 8 - 10, 2004

Cobo Conference & Exhibition Center, Detroit, MI



Register Today
Use Code: 301M
www.sensorsexpo.com
or call
800-331-5706

The Next Wave of Sensing Innovation

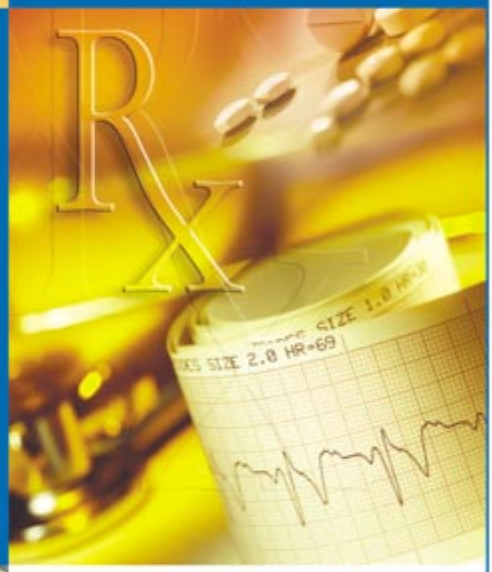
Wireless Networking

Spotlight Program
Monday, June 7



Medical/Life Sciences

Spotlight Program
Monday, June 7



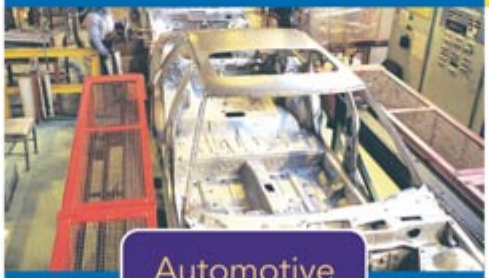
KEYNOTES



Donald L. Runkle
Vice Chairman
Delphi Corporation
Enterprise Technologies



Kermit Hoffman
President
GE Infrastructure Sensing, Americas



Automotive

Spotlight Program
Monday, June 7

NEW

INNOVATION THEATER
Hear new product introductions and other informative presentations in a quick, to-the-point format.

TECHNOTOURS
Inaugural Wireless TechnoTours
By: Dr. James P. Smith

Sensors Expo & Conference will focus on sensor related technologies for the following industries:

- Automotive
- Medical/Life Sciences
- Industrial Automation
- Process Control
- Telecommunications
- Manufacturing
- Consumer Electronics
- Information Systems

To Exhibit, call 603-924-5400

Official Publication: sensors
Produced by: ADVANSTAR TECHNOLOGY GROUP
Gold Sponsor: GE
Corporate Sponsor: Melexis Microelectronic Integrated Systems

Listing 1—*Use this code to make the robot go bump.*

```
#use "hobbyist.lib"
main()
{
int i;
float spd;
char state;
byte x;
int count;
  HBmotorInit();            // Initialize motor subsystem
  HBmotorSet(1,0.0);
  HBmotorSet(2,0.0);
  spd = 0.8;                // Speed
  state = 'f';              // Initial state is forward
  count = 0;                // Down count
  printf("Push both bump sensors\n");
  while(1){
    if((HBpinRead(HB_SW1)== 0)&& (HBpinRead(HB_SW2) == 0)) break;
  }
  printf("Release both bump sensors\n");
  while(1){
    if((HBpinRead(HB_SW1)== 1)&& (HBpinRead(HB_SW2) == 1)) break;
  }
  printf("Running...\n");
  while(1){
        // Countdown timer
    costate {
      waitfor(DelayMs(100));
      if(count) count-;
    }

// Run/bump state machine

costate {
       waitfor(DelayMs(10));
       switch(state){
         // Run forward until bump
         // Set timer
         // Back up away
         case 'f':
           HBmotorSet(1,spd);
           HBmotorSet(2,-spd);
           if(!HBpinRead(HB_SW2)){
             state = 'l';
             count = 20;
           }else if(!HBpinRead(HB_SW1)){
             state = 'r';
             count = 20;
           }
         break;
         // Back up left until timer
         case 'l':
           HBmotorSet(1,0);
           HBmotorSet(2,spd);
           if(count == 0) state = 'f';
         break;
         // Back up right until timer
         case 'r':
           HBmotorSet(1,-spd);
           HBmotorSet(2,0);
           if(count == 0) state = 'f';
         break;
         case 's':
           HBmotorSet(1,0);
           HBmotorSet(2,0);
         break;
         default: state = 's'; break;
       }
    }
  }
}
```

and encoder wheel on both drive wheels. The detector outputs, which are configured in an open-collector configuration with a 270-Ω pull-up, are wired to a couple of digital inputs (see Figure 3).

Reading the detector is pretty simple (see Listing 2). A routine wakes up every 0.1 s, samples the inputs, and compares it to the last stored state of the inputs. When a change occurs, a counter is incremented or decremented depending on whether the motor is running forward or backward. The wheel is approximately 1″ in diameter, so the circumference is 3.14 × 1, or 3.14″. There are eight pulses per revolution, which allows you to measure in steps of approximately 0.4″. Adding more lines to the encoder wheel will give more resolution.

Although using this encoder for an odometer is fine, you could also use each encoder input and measure the difference in distance traveled on each wheel to get a sense of direction when turning. This is cumbersome and error prone with the low-resolution encoders I have built. A better way would be to use a compass to gain an absolute heading.

COMPASS NAVIGATION

I chose a Precision Navigation (PNI) Vector 2x electronic compass for this project. It is a two-axis compass capable of measuring heading with an accuracy of ±2°. Because it is only a two-axis compass, it can only give accurate readings on a level surface. A three-axis compass (also available from PNI) costs more, but can give accurate readings at various attitudes. It also can be used as an inclinometer. However, the price tag for a three-axis compass was not an option for this project.

The Vector 2x in Photo 3 is a small, self-contained module that has several operating modes. It uses SPI to communicate with a host and can be set up to operate in Master mode. SPI uses three wires plus a select signal to operate. All data is clocked synchronously to a master clock (SCLK) in 8-bit chunks. There are two serial datastreams. One is the data from the master to slave devices (SDO); the other is the data from the selected slave device to the

master (SDI). When a device is not selected, the SDI line is tristated to allow other slaves to communicate.

Having a peripheral device operate in Master mode is a bit unusual. Typically, the processor is the master and the compass the slave. The Vector 2x certainly can be configured to do this. Refer to my article, "Robot Navigation Scheme," for more information (*Circuit Cellar* 81). Because the Rabbit is a 3.3-V device and the Vector 2x compass is a 5-V device with input threshold levels that would not work with the output signals of the Rabbit, I put the Vector 2x in Master mode. In this particular mode, it can generate 5-V TTL-level signals, which the 5 V-tolerant I/O pins can deal with without having to use level converters on the SCLK and select lines.

There is a polling signal to the compass. I can generate this signal easily with an open-drain I/O signal on an open drain-capable I/O port. There is already a pull-up resistor on the Vector 2x's *P/C line, so no additional components are required. Figure 4 shows the wiring diagram for the compass.

To take a reading, the software asserts the open-drain poll signal (*P/C). The compass then computes a reading and, after finishing, clocks out a 16-bit word indicating a heading. Options on the compass allow the format of the output to be either in Binary mode or BCD mode. Also, a special RAW mode allows you to read both the x and y magnetometers directly for more advanced applications. I chose Binary mode, and the software simply reads 2 bytes via its SPI port and assembles them into a integer heading reading.

Using the compass in this mode is easy; however, it only works well if the processor can actually read SPI data as a slave device. The Rabbit has hardware serial ports capable of synchronous operating in SPI slave and master mode at high clock speeds with minimal software overhead. Listing 3 shows the code to read out the compass.

OK, let's see. We have a platform, motors, motor controller, bump switches, an odometer, and a compass. But wait, there's more.

Listing 2—*Reading the odometers is simple. The delay helps debounce the detector inputs, which is mostly necessary because of my messy penmanship on the encoder.*

```
// Read odometer -> left,right
    costate {
    waitfor(IntervalMs(10));
       rs = HBpinRead(HB_PF7);
       ls = HBpinRead(HB_PD5);
       if( rs != ors && rs == 0)
              right += (state == 'f')?1:-1;
       if( ls != ols && ls == 0)
              left += (state == 'f')?1:-1;
       ols = ls;
       ors = rs;
 }
```

Listing 3—*I wish all of the sensors were this easy to interface. Simply poll the sensor and read the result.*

```
// Read compass -> heading
costate {
      waitfor(IntervalMs(100));
      HBpinLow(HB_PD4);
      waitfor(DelayMs(10));
      HBpinHigh(HB_PD4);
      SPIRead(buf,2);
      heading = (buf[0]*256) + buf[1];
}
```

Listing 4—*Simply interpret commands from the wireless modem and display sensors.*

```
// Display
costate {
   waitfor(IntervalMs(1000));
   if(count) count—;
   sprintf(buf,"%c %3d %5d %5d\r", state, heading,left,right);
   wfd cof_serCputs(buf);
}  // Costate
   // Command processor
costate {
   wfd c = cof_serCgetc();
   switch(c){
      case '0':
      case '1':
      case '2':
      case '3':
      case '4':
      case '5':
      case '6':
      case '7':
      case '8':
      case '9':
          numreg = (numreg*10) +  (c - '0');
          break;
      case '\n':
      case '\r':
          course = numreg%360;
          numreg = 0;
          break;
      case 'f':
      case 's':
      case 'r':
      case 'l':
          state = c;
          break;
      case '>':
          if(spd < 1.0) spd += 0.05;
          break;
      case '<':
          if(spd > 0.0) spd -= 0.05;
          break;
      default:
          break;
   } // Switch
}    // Costate
```

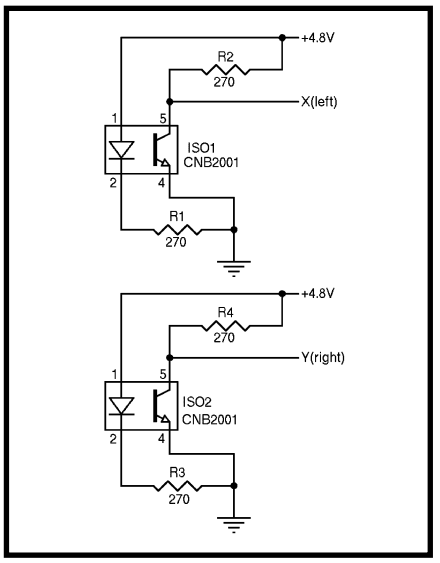


Figure 3—*The detector photo-reflective sensor is wired as open collector, and the signals go to two standard digital inputs.*

ROBOT COMMUNICATION

Because the robot is self-contained, I wanted to be able to communicate with it as well. I have a free async serial port. Actually, I potentially have about six serial ports. One is used for programming and one for SPI. Another is reserved for serial flash memory on a more advanced Ethernet-enabled version of the core module, with the same pinout. So I have approximately three serial ports free that I can use for async serial applications. I use port C in a simple two-wire serial mode with the modem.

I chose Needham's SureLink 915A-FM RF modem (see Photo 4). It's a sophisticated 915-MHz modem that also can be configured for several operating modes. It can run from a 4.8-V power supply and deal with 3.3-V signaling just fine.

The wiring is simple. Most of the pins on the modem module are used to adjust the channel number and modes. I chose to use Serial mode, which gives the application a virtual serial link between two modems. (One is configured to be the master, and the other is a slave.) The interface to the Rabbit uses a transmit-and-receive pin and runs at 57600 bps. Figure 5 shows the wiring diagram for the RF modem.

The robot talks to another modem in an optional QuickLink 232A board. This board interfaces the modem module to an RS-232 link that can be plugged directly into a PC (or laptop). It has a 9-V battery clip and can be powered remotely. I can now use my laptop to remotely monitor and control my robot. Watch out, cat!

Earlier, I showed you a simple BumpBot application for the robot. Now let's look at more sophisticated software that uses the wireless serial link to send commands to the robot and retrieve information.

The software's main job is to deal with the sensors and keep the robot on a predefined heading and speed, which is commanded remotely from a laptop. When the robot runs into something, it stops and waits for the next command. Commands are simple: tell the robot its heading and speed. The software spits out current heading, speed, and bump sensor status. Listing 4 shows the high-level control for the robot. You may download the code from the *Circuit Cellar* ftp site.

MOVING ON

There you have it—a nice platform for robotics experiments. It has the basic sensors for autonomous operations. You could add an autonomous command mode to your control software. For example, you could add the bump-and-avoid algorithm, which lets the robot navigate by itself until it reaches a certain location. This relieves you from having to monitor the robot.

You could also add a grid search pattern submode that allows the robot to search an area in a grid pattern and collect data from a new sensor, like a metal detector or wireless video camera. Or, perhaps you could incorporate an IR thermal sensor to look for intruders.

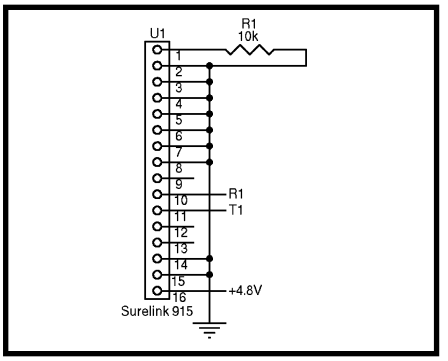


Figure 5—*The Surelink modem needs to be configured in this way to put it in Serial Link mode. TX/RX are async serial at CMOS levels.*

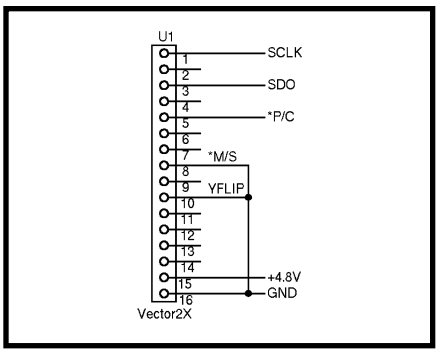


Figure 4—*The Vector 2x compass uses an SPI serial interface. The poll signal (*P/C) is driven via an open-drain output from the Rabbit to make it 5 V logic-compatible.*

Ingo Cyliax received a B.S.C.E.E. from Purdue University. Ingo has worked for universities and small companies, and he has been consulting on his own and writing for many years. He currently works as an application engineer at Z-World, Inc. Ingo's interests include photography, astronomy, aeronautics, embedded systems, wireless networking, and hardware design. You may reach him at cyliax@ezcomm.com.

PROJECT FILES

To download the code, go to ftp.circuitcellar.com/pub/Circuit_Cellar/2004/165.

SOURCES

SureLink 915A-FM RF modem
Needham's Electronics
(916) 924-8037
www.needhams.com

RCM3600 RabbitCore
Rabbit Semiconductor
(530) 757-8400
www.rabbitsemiconductor.com

RCM3600 Microprocessor starter kit
Imagine Tools
(530) 757-0911
www.imaginetools.com/MSK

L293D Dual H-Bridge
STMicroelectronics
www.st.com

Bulldozer kit
Tamiya America, Inc.
www.tamiyausa.com



Low-Cost Intelligent Sensors Network

Victor's 68HC908QY4-based intelligent sensors network is an inexpensive solution for applications that require low-speed, two-wire bidirectional data communication. A simple time-triggered protocol ensures a predictable data delivery time. The quasi-harmonic constant frequency network signals are sheltered from electromagnetic compatibility problems.

Modern homes, offices, and industrial facilities contain multiple networks for data communication, equipment control, systems state monitoring, and diagnostics. A special class of networks is intended for low-speed data communication. These networks are used for home/office automation and remote technological process monitoring. For instance, they can control alarm systems, remote light systems, and other applications that require low-speed, bidirectional data communication with minimal equipment and installation costs.

Although various network implementations (e.g., Profibus, Modbus, CAN, LIN, X10, Linet, and Ethernet-based systems) were developed to serve these needs, they all are characterized by at least one of several drawbacks: dedicated, relatively expensive modem/line interface chips must be used, which increases the network node price; the network protocols can be relatively complicated, thus demanding more expensive microcontrollers to serve them; the node power must be provided by separate wires, which increases installation and service costs; the network protocols cannot guarantee predicted data delivery time, and network performance is traffic dependent; the host must process all data because the nodes cannot perform direct information exchange; and, lastly, the network can radiate the electromagnetic noise in a wide frequency range, which, in turn, can raise electromagnetic compatibility problems or require more expensive shielded cables.

The proposed network depicted in Photo 1 is free from the aforementioned shortcomings and is characterized by several advantages. The simple time-triggered protocol assures predictable data delivery time. Only two wires are used for bidirectional data communication and providing the node power. The quasi-harmonic, constant frequency network signals are free from electromagnetic compatibility and certification problems. And finally, note that the node and host do not use proprietary or dedicated modem/line interface chips. This permits you to create an intelligent networked sensor (e.g., fire alarm, motion detector, temperature, position, proximity sensors, and valve controller) for the price of a conventional, non-networked device. Table 1 illustrates the main network characteristics.

OPERATION PRINCIPLES

The proposed network consists of the programmable single host controller and multiple nodes connected to the host with a single twisted wire pair. The host controller generates an amplitude-modulated constant frequency carrier signal. The top half wave is intended to provide the power energy to nodes. The lower half wave is used for bidirectional data communication.

Figure 1 illustrates the voltage and current signals in the proposed network. Each carrier period is used to transmit one bit, and the varying amplitude of the negative half wave is used for information coding. Low amplitude is used to code logic 1. High amplitude is used to code logic 0. If several nodes try to transmit different

Characteristic	Specifics
Carrier frequency	5 to 20 kHz, depending on the application
Data transmission method	Each carrier period is used to transmit/receive 1 bit
Node data length	4, 8, 12, and 16 bits, and broadcast messaging with 8-bit data and 8-bit type mask
Maximum nodes in network (Note: Nodes with various data lengths can coexist in the network at the same time.)	250 for 4-bit nodes, 125 for 8-bit nodes, 80 for 12-bit nodes, 60 for 16-bit nodes, and 60 for broadcast nodes
Network supply voltage and current	15-VDC, 4-A maximum, 25-V option

Table 1—*The network specifications can be easily adapted to your demands. For instance, the supply voltage and current can be increased, and the communication speed can be adjusted according to the application. The maximum quantity of network nodes can be easily increased as well with minor modification of the sources.*

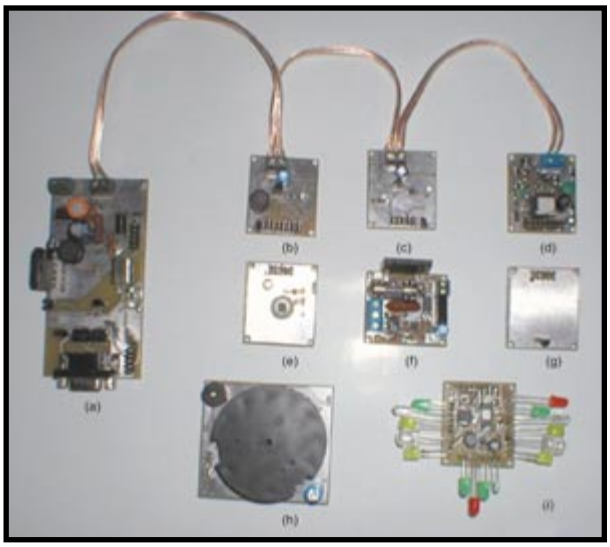


Photo 1—*Here's the whole system:* ***a****—host controller;* ***b****—general slave with a 16-pin Nitron CPU;* ***c****—slave with an eight-pin CPU;* ***d****—slave with a 16-pin CPU and switching regulator;* ***e****—infrared motion detector expansion board (EB);* ***f****—light controller EB;* ***g****—temperature sensor EB;* ***h****—smoke detector EB;* ***i****—LED controller EB.*

logic levels at same time, the resulting level will be logic 1 (wired OR).

The carrier frequency can be selected to be above audio frequencies but far lower than radio frequency bands, which guarantees low electromagnetic radiation and eliminates possible interference with other systems. The ratio between carrier signal negative half-wave amplitudes for different logic states was selected as a compromise between information detection reliability and network voltage signal spectrum width. It was set equal to five in this design, which allows you to get second and forth harmonic levels of –13 and –20 dB relative to the carrier signal level during 010101 pattern encoding.

The network uses collision-free time triggered protocol (TTP). Each node has its own timeslot to receive or transmit data. Figure 2 illustrates the protocol. In the current implementation, point-to-point communication (PPC) and Broadcast Messaging modes are supported. In PPC mode, each node can communicate with another node or host. Each node has one PPC in endpoint and one out endpoint. In addition to PPC mode, each node can support Broadcast mode and has one in-broadcast and one out-broadcast endpoint. In Broadcast mode, the node can receive and transmit messages to and from any number of nodes.

PPC mode is useful for building a distributed remote data collection system (e.g., a temperature/humidity sensor network for a grain elevator), in which the host periodically queries each node to get the node's information. Broadcast mode is intended for applications where commands can be received from numerous sources, such as a building light control system that involves different lamps that can be turned on and off from anywhere. Some applications can effectively employ both the PPC and broadcast communications at the same time. For instance, in the light control system, PPC is useful for querying the lamps' state for burnouts.

All of the nodes are grouped according to node data length. The current network protocol implementation supports nodes with data lengths of 4, 8, 12, and 16 bits. All broadcast messages have a fixed length of 8 bits together with an 8-bit mask to distinguish various kinds of broadcast messages. The 4-bit nodes are intended for simple control functions such as button state reading and valve and load control. Light control and security systems are good examples of this. The nodes with 8- or 16-bit data lengths can be temperature, humidity, or pressure sensors.

The host controller selects a node group, and then sends the preamble and the unique node group selector byte. After the selector byte is sent, the node data follows sequentially. The data transmission is divided into 4-bit (nibble) frames. Two adjacent frames are separated by a zero separator bit. This way no more than four logic ones can be transmitted consecutively. Nodes use this property to trap the preamble. The endpoint-related frame numbers are assigned to each node within its group. The nodes use these frame numbers to determine when a node must start and stop to transmit or to receive the data.

After all of the nodes within one group are processed, the host selects another group. It sends the same preamble but another group selector byte, and transmits the frame separator bits. All of the node groups are chosen one after another by the host, which, of

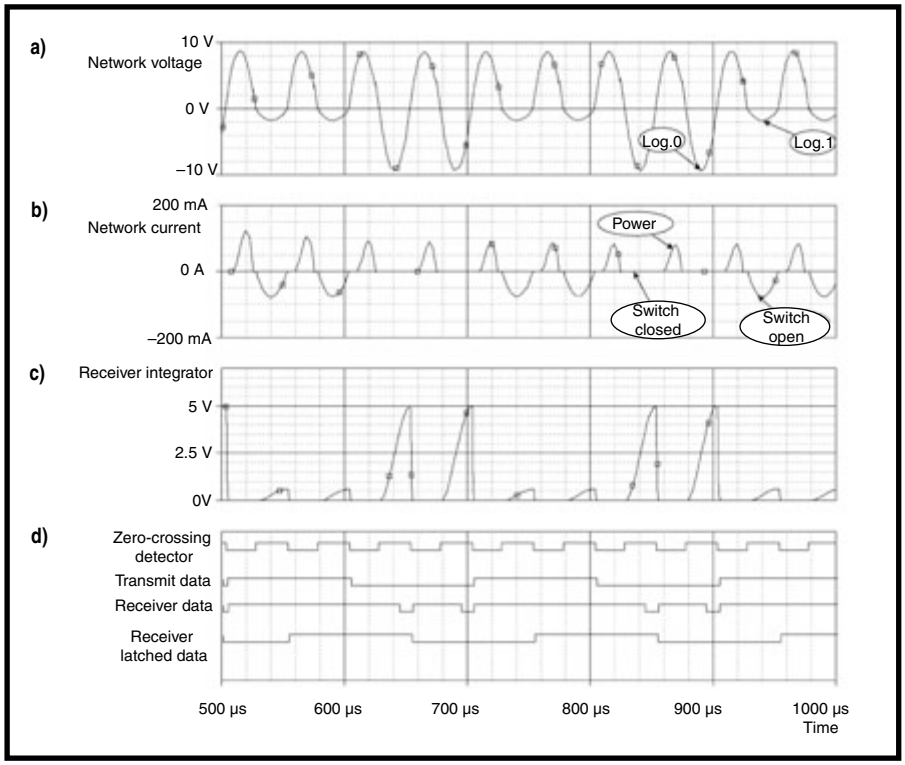


Figure 1—*The selected network signals include the twisted pair voltage (****a****), current (****b****), and receiver internal signals (****c*** *and* ***d****).*

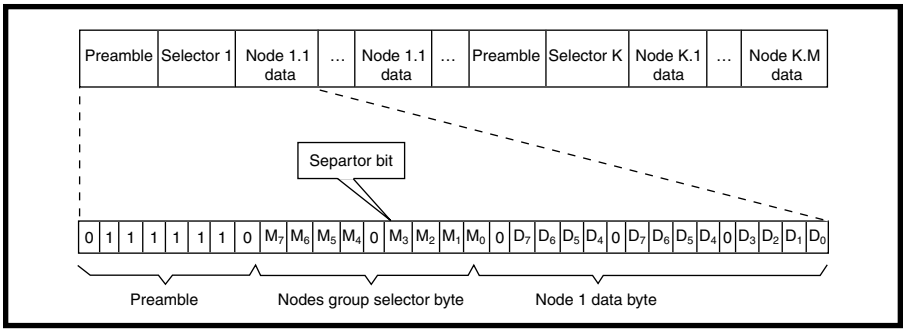


Figure 2—*The network separates the nodes into several groups according to the node data length. The groups are separated by the node group selector bytes. The unique preamble marks the start of the data exchange cycle, and selector bits are used to distinguish the preamble from nodes data and check network errors.*

course, can receive and transmit any messages to any node group.

The carrier frequency and total number of network nodes determines the maximum information delivery latency. You can estimate it with the following formula:

$$T_d = \frac{1}{F_c}\left(\sum_{i=1}^{5}\left[17 \times H(N_i) + B_i \times N_i\right] + 37\right),$$

$$H(x) = \begin{cases} 0, & x = 0 \\ 1, & x \neq 0 \end{cases}$$

$$B_i = \{5, 10, 15, 20, 20\},$$

$$N_i = \{N_{4b}, N_{8b}, N_{12b}, N_{16b}, N_{br}\}$$

where F_c is the carrier frequency. N_{4b}, N_{8b}, and so on are number nodes with 4-, 8-, 12-, and 16-bit data lengths and broadcast nodes respectively. The current implementation uses the carrier frequencies 5, 10, and 20 kHz according to the application's requirements (e.g., node power consumption and requisite speed).

HOST CONTROLLER

The host controller is shown in the block diagram in Figure 3 and the schematic in Figure 4.

The host controller consists of the galvanically isolated RS-232 interface, the software-implemented UART and packets processor, the software frame engine for network protocol support, and the control subsystem. It also includes a carrier generator, band-pass filter (BPF), power amplifier, impedance-matching network, carrier signal zero crossing detector, data receiver and transmitter, network current sense with rectifier/integrator, and an ADC for measuring network current consumption.

A carrier signal is generated by the CPU U5 internal timer, and is filtered by a band-pass filter, which is implemented on U2A. The filter selects the first harmonic in the timer output signal waveform. Note that the host controller can provide a network soft start by gradually increasing the timer's duty cycle to eliminate current surges. The low-cost audio amplifier, U1, scales the output signal to the ±12 V. For large networks, the higher output signal levels can be obtained by adjusting the gain of U1 and increasing the amplifier supply voltage without any trouble.

The matching network L1R2D1R3 provides different impedance for upper

Fighting against your PCB-Design Software?
Here's something that will spare your time and your budget!
Boards designed under EAGLE are found in patient monitoring equipment, chip cards, electric razors, hearing aids, automobiles and industrial controllers. They are as small as a thumbnail or as large as a PC motherboard. They are developed in one-man businesses or in large industrial companies. EAGLE is being used in many of the top companies. The crucial reason for selecting EAGLE is not usually the very favorable price, but rather the ease of use. On top of that comes the outstanding level of support, which at CadSoft is always free of charge, and is available without restriction to every customer. These are the real cost killers!
New Version
EAGLE 4.1
Schematic Capture • Board Layout
Autorouter
for Windows®
and Linux
Version 4.1 Highlights
Powerful library management: e.g. move devices between libraries, base library for packages, generate package variants from other libraries.
Dynamic ratsnest during routing process.
Copy function in schematic.
Rotate components in 0.1-degree steps.
Blind & buried vias and pads with off-center drill.
User-defined background color.
Miter function for (rounded) tracks.
Smash for groups.
Measure distances between arbitrary points.
Choose alternative raster on-the-fly with Alt-key.
Windows is a registered trademark of Microsoft Corporation
Linux is a registered trademark of Linus Torvalds
FREE
EAGLE 4.1 Light is Freeware!
You can use EAGLE Light for testing and for non-commercial applications without charge. The Freeware Version is restricted to boards up to half Eurocard format, with a maximum of two signal layers and one schematic sheet. All other features correspond to those of the Professional Version. Download it from our Internet Site or order our free CD.
If you decide in favor of the Commercial Light Version, you also get the reference manual and a license for commercial applications. The Standard Version is suitable for boards in Eurocard format with up to 4 signal layers (max. 99 schematic sheets). The Professional Version has no such limitations.
Prices
Light
Standard
Professional
Layout — 199$ — 399$
Layout + Schematic — 398$ — 798$
Layout + Autorouter — 398$ — 798$
Layout + Schematic + Autorouter — 495$ — 597$ — 1197$
Pay the difference for Upgrades
http://www.CadSoftUSA.com
800-858-8355
CadSoft Computer, Inc., 801 S. Federal Highway, Delray Beach, FL 33483
Hotline (561) 274-8355, Fax (561) 274-8218, E-Mail : info@cadsoftusa.com

Protect Critical Embedded Web Systems



Rapidly develop secure web interfaces for your embedded systems

As more embedded applications feature access via the Internet, secure transmission of data is growing increasingly critical. The Secure Embedded Web AppKit provides optimized software development tools to easily build secure web interfaces for new and existing applications. Built around the RCM3700 RabbitCore™ powered by the Rabbit 3000® microprocessor, the Secure Embedded Web AppKit offers Secure Socket Layer (SSL), which uses a 128-bit lock and key solution to encrypt data transfers.

The Secure Embedded Web AppKit includes:

- **RCM3700 RabbitCore**
- **Dynamic C Development Software**: Industry proven for embedded applications
- **Secure Socket Layer (SSL)**: 128-bit encryption for HTTPS
- **RabbitWeb™**: Rapid web development extensions for embedded devices
- **File Allocation Table (FAT) File System**: Ready to run Flash based file System

RCM3700
2.95" x 1.20" x 0.88"



From **$32** qty. 1000

FEATURES	RCM3700
Ethernet	10Base-T
Processor	Rabbit 3000® @ 22.1 MHz
Power	3.3 V operation
Serial Flash	1 MB
I/O	33 digital
Serial Ports	4 (IrDA, HDLC, asynch, SPI)

KIT PRICE $599 USD
Reg. $699

Order Online!

Order your Secure Embedded Web Application Kit online and receive an instant $100 discount.

Limited Time Offer

www.rabbitsemiconductor.com/9184

RABBIT Semiconductor

www.rabbitsemiconductor.com
530.757.8400

and lower half waves, and it assures amplifier stability with large capacitive loads. To measure network current consumption, the sense resistor, R8, is used together with voltage-to-current converter, U2B, Q1. The converter output current, which is proportional to the network positive half waves current, is filtered by C6R13. The voltage across C6 is measured by the internal CPU U5 ADC.

The network is powered by a single 15-V power supply. Accordingly, the 5- and –15-V voltages are produced by U4 and U3. The –15-V power consumption does not exceed several watts, so the low-power, low-cost, switching-mode inverter (U3) is satisfactory enough.

The galvanically isolated UART consists of U6 through U8. Because the processor does not contain a hardware UART, a separate processor U6 is used for serial interfacing with the PC and I/O

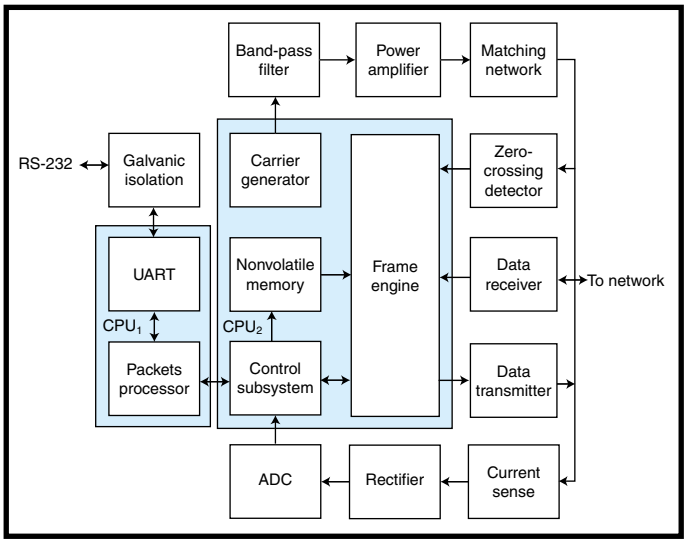


Figure 3—*The host controller consists of two microcontrollers: one CPU for UART data packet processing and one to serve the network communication protocol. Some modules are implemented with microcontroller hardware peripherals, others are implemented completely with firmware.*

packet processing. The stable UART clock signal is generated by the U5 crystal-stabilized oscillator. The current implementation uses 9600 bps to communicate with the PC or another controller.

The zero-crossing detector (Q3) creates the periodic interrupts for the central processor (U5). The interrupt requests correspond to the carrier signal rising edges. Interrupt latency up to half the carrier signal period has no influence on host operation.

The data transmitter is implemented on Q4 through Q5, and the data receiver is formed using Q2. The data receiver is based on a resettable integrator. C20 integrates the network voltage during a negative half wave, which allows increasing noise immunity for high-frequency signals. When a zero-crossing detector interrupt occurs, the C20 voltage level is read by CPU U5, and C20 is discharged by dynamically changing the port bit direction to output with a log 1 level.

Figures 1c and 1d illustrate the data transmitter and receiver operation. The received data is delayed one carrier period relative to transmitted infor-





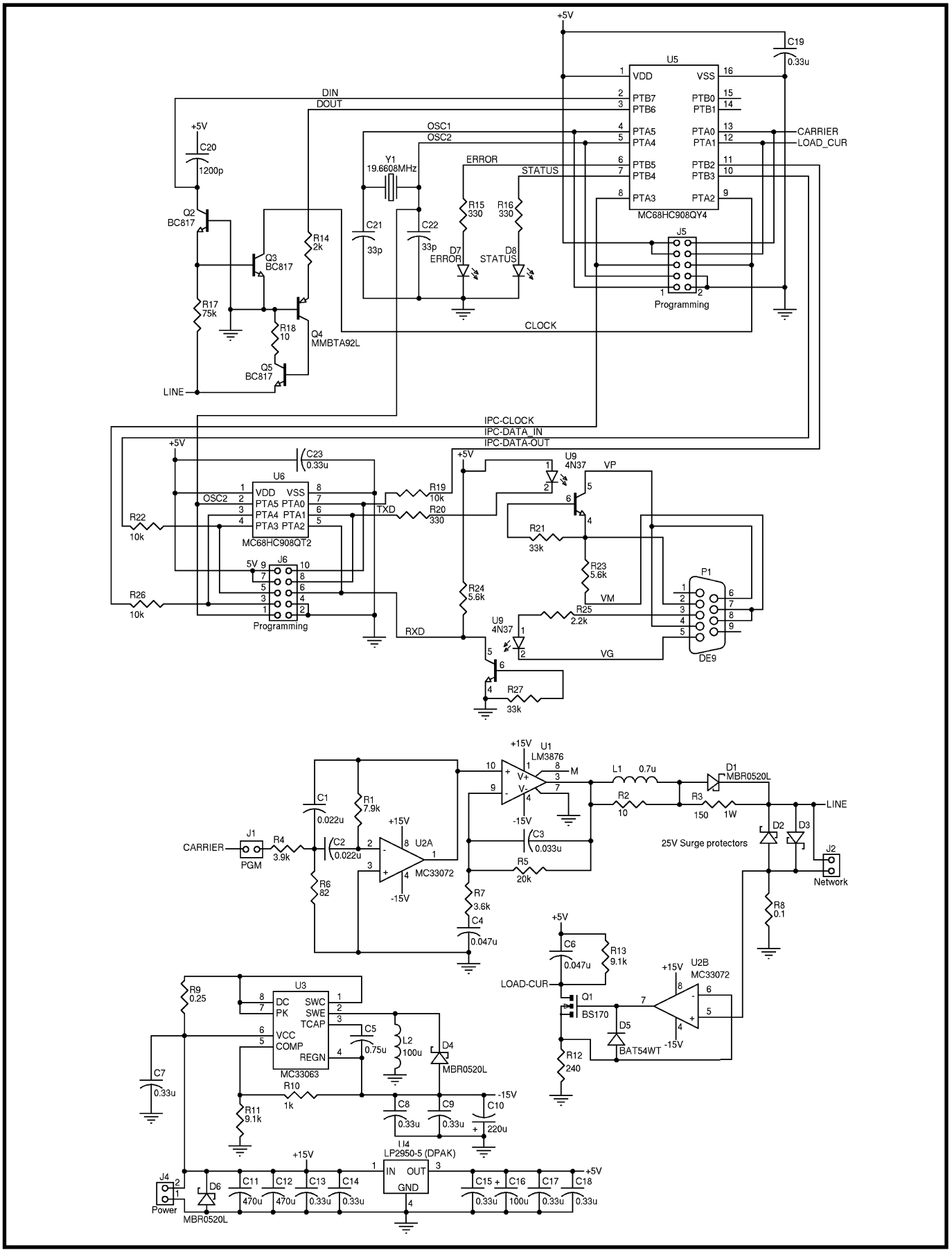


Figure 4—*R22, R26, and R19 protect the controller pins during possible firmware faults. U5 and U6 have their own programming/debug sockets.*

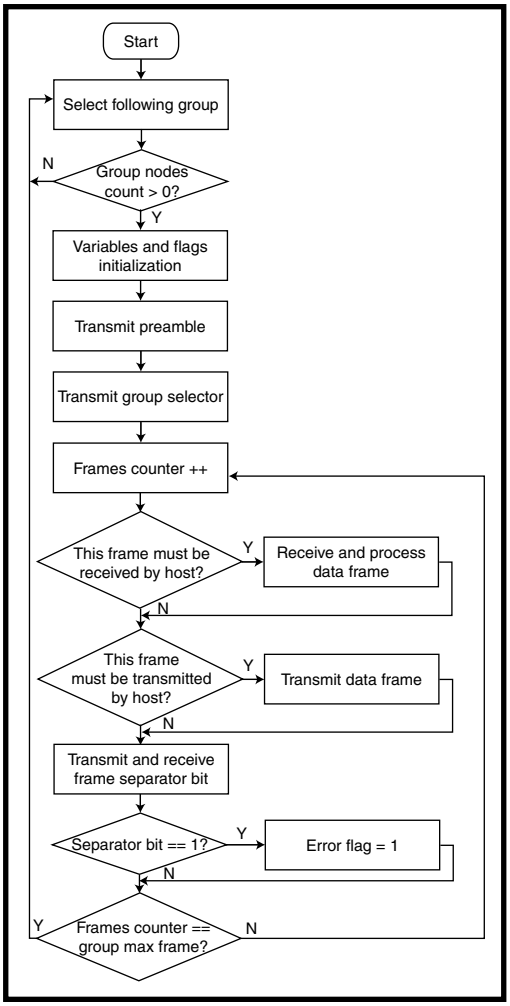


Figure 5—*The host firmware sends the preamble and nodes group selector byte for each group (if present in network), and it captures and transmits the separate frames to and from individual nodes. It also checks the network's health by zero separator bit monitoring.*

mation. This property can be used for possible network fault detection by the host or nodes.

The host firmware consists of two parts: the packet and central processor firmware. The external PC can interact with the host by sending and receiving the ACSII text messages. The packet processor receives, decodes, and retransmits these messages to the host's central processor. The interprocessor communication is implemented using the master-slave approach. The packet processor acts as master relative to the central processor. Note that the three-wire synchronous interface is used for interprocessor communication. The host's central processor receives and transmits the data based on interrupt-driven communication using the state machine.

The host central processor firmware serves several functions. It provides the node control and frame generation, supports capturing node data, and transmits messages to every node in any group. Figure 5 provides the central processor node communication and control algorithm. This algorithm was implemented in the IRQ interrupt routine using a dedicated state machine. The firmware supports network health monitoring by checking separator bits (each separator bit must be zero) and measuring the network current consumption level, therefore detecting the overload situations. If network errors are detected, they can be reported to the PC, or the host controller can try to restart the network by cycling the nodes' power.

NODE CONTROLLER

The general node controllers were developed using the eight- and 16-pin Nitron CPUs. They use Linear/Switch mode power supply options to satisfy the various application requirements. Application-specific daughter cards can be connected to these controllers to implement end applications.

The node controller consists of a carrier zero-crossing detector, a data receiver and transmitter, a voltage regulator, and a software frame engine for communication protocol support. It also includes application-specific firmware and hardware in addition to nonvolatile memory for configuration storing. The CPU timer, ADC, and most of the I/O ports are at an end application's disposal. A schematic diagram of the node controller with a 16-pin CPU and switching regulator is depicted in Figure 6. Schematic diagrams of other node types are posted on the *Circuit Cellar* ftp site.

The node zero-crossing detector, data receiver, and transmitter operation are similar to those of the host controller. The network R1D1D2C3 acts as a simple but effective hardware protector from processor hang-ups. Without this network, one failed node that constantly transmits log 1 can disturb the network operation. D3, D4, and L1 smooth the rectifier current pulses, providing the optimal load for the host power op-amp and reducing the EMC energy losses. It is a minimal power factor corrector.

The application-specific hardware is connected via expansion connector J1, which can be used for CPU in-circuit programming at the same time. The LED has end application-specific functionality, and confirms the remote node configuration or indicates the error situations. Jumper J3 is used for entering Configuration mode.

The node firmware consists of communication protocol support routines and application-specific firmware. The communication protocol is implemented by the IRQ interrupt routine. The simplified algorithm is illustrated in Figure 7. The firmware supports the PPC, broadcast in and out operations, and remote node Configuration mode. The built-in error detection suspends node operations until the next preamble is received when erroneous data are encountered.

To process the various broadcast messages effectively, a simple associative memory subsystem was implemented in the node firmware to store different broadcast messages only. Because neighboring broadcast messages are separated by at least 20 carrier periods, the search algorithm was implemented recursively to balance the interrupt latency. The associative memory allowed for the simplification of the end-application firmware and the reduction of the broadcast message buffer size.

NETWORK CONFIGURATION

The network configuration process consists of several stages. First, the required node functionality (e.g., setting the node group and choosing the appropriate endpoints and mask values) must be selected during end-application firmware development at compile time by appropriately setting conditional compilation variables and the corresponding constants.

Before network installation, the valid endpoint frame numbers must be assigned to each node. A dedicated PC-based application was developed for this purpose. To configure the node remotely, the node must be connected to the network as usual, and the configuration jumper should be set to allow for configuration. The PC software sends the special command sequence to the host,

which redirects them to the node. The node receives the configuration commands, checks the command validity, and stores the decoded configuration settings in the built-in nonvolatile memory using the processor's self-programming option. The blinking node LED confirms the process's success.

Finally, when valid frame numbers are assigned to each node in a group, the host controller must know how many nodes exist in each group to reduce the maximum node access time. Setting and storing a maximum frame number value for each node group in the host central processor nonvolatile memory can do it. Note that you can use the PC-based software for network monitoring, capturing node data, transmitting data to any nodes, and sending various commands to the host.

The PC configuration software scripting option simplifies the configuration process. The top-level application-specific software can be written according to the end use. For example, database connection, data analysis, and various charting possibilities can be added to the control software.

Please note that the PC (or another controller) is only required for network configuration or data exchange with external world.

NETWORKED APPLICATIONS

Several projects have been developed using this network. The list of nodes includes a remote temperature sensor, an infrared motion detector for security systems, a smoke detector with self-testing possibilities, a lighting control system for buildings, and an LED information panel for factory/elevator automation. The end application schematic diagrams and program source code are located on the *Circuit Cellar* ftp site.

Let's take a look at a networked smoke sensor. One of most popular methods to detect smoke is the smoke light scattering effect. The detector contains a chamber with an LED and photodiode. When smoke enters the sensor, the scattered light hits the photodiode and generates a photocurrent, which is processed by the amplifier. You may download the sensor daughterboard schematic from the *Circuit Cellar* ftp site.

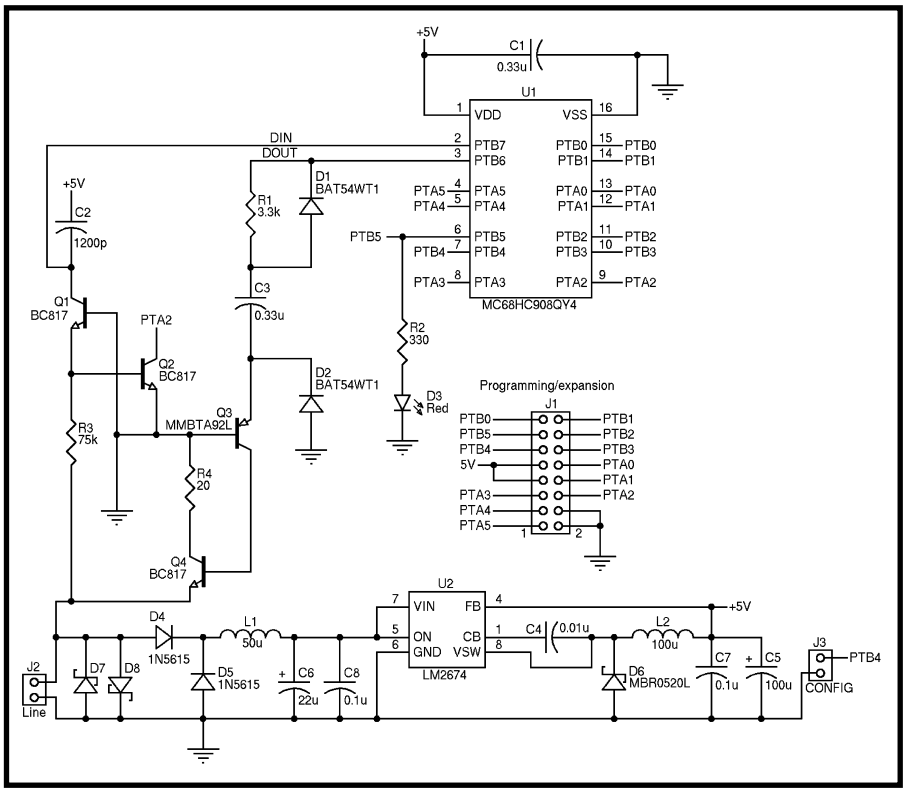


Figure 6—*The general node controller is simple. It has a switching power regulator with minimal power factor corrector, a simple network data receiver, a transmitter, and a zero-crossing detector.*

The sensor contains the LED driver (Q1), photodiode signal amplifier (U2), and horn (LS1) with driver Q2. Both piezoelectric and electromagnetic horns are supported. L1 provides relatively high voltage (up to 40 V) when the piezoelectric horn is used. This inductance is not needed for the electromagnetic buzzer. Note how the different network C4R7 limits the maximum LED on time and guards the LED (and other) nodes from possible CPU hang-ups when the processor constantly turns on the LED. The amplifier signal is sampled by the processor ADC and subsequent signal processing is done in the firmware.

The smoke detection routine is called approximately eight times per second. Dividing the timer module overflow interrupt frequency in the firmware provides the timebase. The `Samples_Counter` variable is incremented each routine call; it determines the internal sensor state. The first eight routine calls are used to estimate the background level using an averaging FIR low-pass filter. The ninth call is used to measure the scattered light level by turning on the LED for 1 ms. Smoke is detected when the measured photodiode signal is greater than the background level on $THLD_2$ threshold $THLD_3$ times consecutively. When smoke is detected, the buzzer generates the loud uninterrupted signal. The internal CPU timer module generates this signal.

The sensor has self-testing capabilities. The measured photodiode signal must be larger than the background level on the THLD1 threshold when no smoke is in the chamber by reason of the chamber light reflections. In this way, LED/photodiode contamination can be detected, indicating the need for service.

The sensor incorporates the one out PPC endpoint and bidirectional broadcast endpoint. The PPC endpoint is used to transmit the sensor state to the host. The bidirectional broadcast endpoint is used to transmit and receive the information to and from other sensors. When one sensor detects smoke, it sends this event to the other sensors via a broadcast messaging mechanism. Sensors receive these messages and turn on the buzzer. Note that a continuous signal is used to indicate local smoke and an intermittent signal indi-

cates remote smoke detection.

This example clearly illustrates the advantages of the networked solution. Each sensor can talk to others about a fire, and the host identifies which sensor caught it. This saves time for rescues.

AN ALTERNATIVE

The current implementation of the sensor network does not include extensive error detection. You can add a simple CRC mechanism to check data validity. Additionally, the majority rule that critical data should be received twice can be added to increase network operation reliability in harsh environments.

This sensor network project was designed for the Motorola Flash Innovation 2003 Design Contest. However, the methods I used worked so well that I decided to rework the network using reconfigurable mixed-signal PSoC microcontrollers from Cypress. Because of the PSoC's unique built-in analog peripherals, I was able to reduce the number of external components. This decreased the host/node price, which is important for cost-sensitive applications. The improved PSoC-based network application note should be ready by the summer. Refer to the Cypress web site for details.

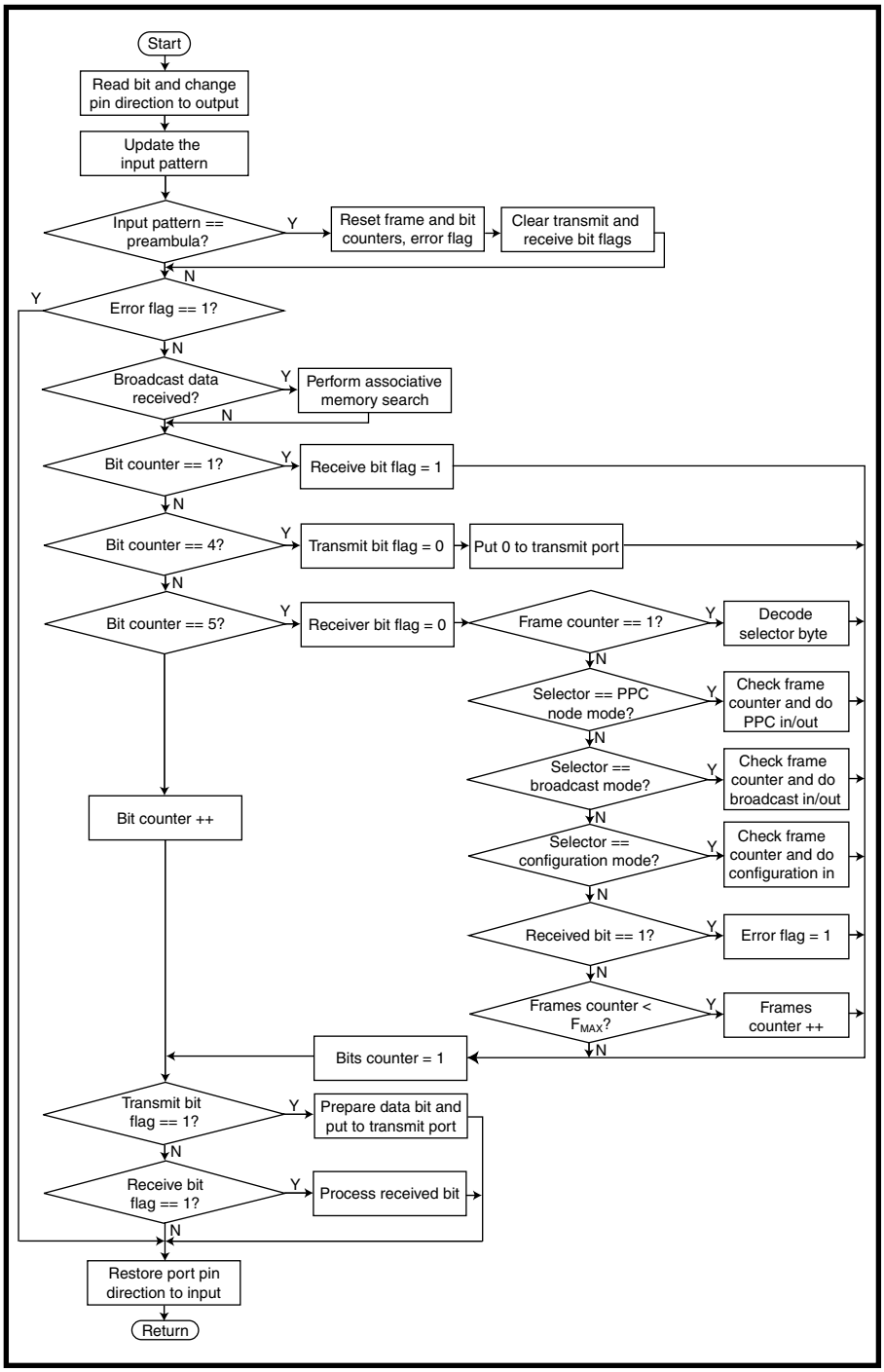


Figure 7—*The node communication protocol is implemented in the IRQ ISR. The node traps the preamble and selector bytes, counts the frame numbers, makes the bit-by-bit data reception or transmission, supports the remote node configuration, and checks the error conditions.*

Victor Kremin earned a diploma in Radio Physics from Ivan Franko National Lviv University and a Ph.D. in Computer Aided Design Systems. Currently, he's an associate professor at the National University "Lvivska Polytechnika" in the Ukraine. Victor's interests include the full cycle of embedded systems design including processors, operation systems, and target applications. You may reach him at vkremin@polynet.lviv.ua.

PROJECT FILES

To download the code and additional schematics, go to ftp.circuitcellar.com/pub/Circuit_Cellar/2004/165.

RESOURCES

Linet network, www.linet-network.com.

Motorola, Inc., "MC68HC908QY4," rev 0.1, 2002.

X10 Protocol information, meltingpot.fortunecity.com/lightsey/52/x10.html.

SOURCES

MC68HC908QY4 Microcontroller
Motorola, Inc.
(847) 576-5000
www.motorola.com

MC33072 op-amp
ON Semiconductor
(602) 244-6600
www.onsemi.com

LM3876 Power amplifier
National Semiconductor
(800) 272-9959
www.national.com

ZiLOG double deal

Twice the Speed with ZiLOG's Newest Microcontrollers

Z8 Encore!® Flash Microcontrollers

Z8F04 and Z8F08

Compare:

FEATURE	ZiLOG Z8F04	MOTOROLA MC68HC908JK3	ZiLOG Z8F08	MOTOROLA MC68HC908JB8
Actual CPU Speed	20MHz	8MHz	20MHz	8MHz
High-Speed Flash	4KB	4KB	8KB	8KB
SRAM	1K Bytes	128 Bytes	1K Bytes	256 Bytes
A/D Converter	10-bit	8-bit	10-bit	8-bit
SPI & I^2C	Both	None	Both	No I^2C
UARTs	One (9-bit)	One (8-bit)	One (9-bit)	One (8-bit)
16-bit Timers	Two-Multi-mode Independent	One-Multi-channel Synchronous	Two- Multi-mode Independent	Two- Multi-channel Synchronous

This comparison is valid as of 7/1/ 2003

ZiLOG broadens its high-speed Z8 Encore!® Flash microcontroller family from 64KB to 4KB with the 4KB Z8F04 and 8KB Z8F08. The newest members of theZ8 Encore!® 8-bit Flash microcontroller family feature the enhanced version of ZiLOG's renowned Z8® architecture.

With more powerful instructions, register-to-register architecture, and flexible memory mapping, you'll get products to market faster than with any other microcontroller. To learn more about ZiLOG's complete Z8 Encore!® and eZ80Acclaim!™ lines visit us on the web

www.zilog.com



Z8 Encore!®

Start developing today for as low as

$39.95

(Includes full-version ANSI C-Compiler)

ZiLOG, Z8 Encore! Z8, and eZ80Acclaim! are trademarks or registered trademarks of Zilog Inc. in the United States and in other countries. Other logos, product names and service names mentioned herein may be trademarks of the companies with which they are associated.

ZiLOG



eZ80 Acclaim!™ Flash Microcontrollers

Who Needs 16-bit When You've Got the ZiLOG eZ80Acclaim!™

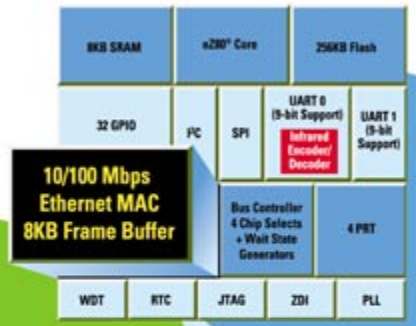
This chip rips. It's the world's most powerful 8-bit MCU. Period. Imagine a 50 MHz single-cycle core with:

- Instruction execution using a pipelined architecture
- An 8/16/24-bit register set
- A 24-bit address bus with 16MB of linear address space
- 256KB of on-chip-reprogrammable flash memory
- 8KB of SRAM
- Ethernet Media Access Controller
- A comprehensive TCP/IP protocol stack

What you wind up with is a brawny chip that performs all the day-to-day functions of an 8-bit with the throughput of a 16-bit at a lower system component cost. And it does it without even breaking a sweat. Imagine the possibilities!

Pump up your performance with the world's most powerful 8-bit MCU! Take the eZ80Acclaim!™ Flash Microcontroller for a test drive today and save $700.

eZ80F91 Block Diagram



eZ80 Acclaim!™ Flash Microcontrollers

Complete Development Kits

$99.95

Now for a limited time!

Register Now For ZiLOG's $20,000 Flash "Keis" Cash Design Contest

www.zilog.com/acclaimcontest

Robot Upgrade

Use a Microcontroller to Emulate an I²C EEPROM

It's amazing what you can do with a PIC and a toy robot. Jay presents an interesting project. He came up with a method of interfacing to the electronics of a Spybot robot by using a PIC16F876 to emulate an I²C EEPROM. The result is an enhanced robotics system that's the perfect foundation for more complex designs.

Like many robotics enthusiasts, I've converted my fair share of toys into autonomous creatures. When I evaluate a potential victim, I'm usually after the motors, gears, and other mechanical items. Because of their prices, toys can be a good alternative to building robot mechanics from scratch. The electronic brain of a toy, if any, often consists of a single device covered in a blob of epoxy and, if you're lucky, some salvageable discrete components. A few snips from the wire cutters, a little desoldering, and the toy is rendered brainless. In this article, I present an unusual toy that has such a nice brain it would be a shame to pull it out. Instead, I'll show you a method of interfacing to the existing electronics by using a microcontroller to emulate an I²C EEPROM.

SPYBOTICS

Spybots are the latest in a series of Lego robotic construction toys. There are four different Spybot models, each of which includes the same controller but varies in its locomotion. I used the tread-based Snaptrax S45 Spybot for this project (see Photo 1a).

Spybot controllers contain an infrared communication system, two motors, multiple LEDs, a piezo speaker, a bump sensor, and a light sensor (see Photo 1b). Spybots have no built-in expansion capability for adding additional sensors or actuators.

The Spybot's processor is found under a blob of epoxy, and its firmware is stored in ROM. Lego has not documented the type of processor it uses; and even if it had, you couldn't simply replace the firmware with one of your own liking. Fortunately, the Spybot's firmware is flexible enough to do some interesting things. Lego supports Spybotics programming in a software development kit that contains tools and related documentation.

The Spybot's processor executes user code as a virtual machine. It emulates a simple microprocessor and executes unique byte codes (referred to as "LASM"), storing and executing them from an I²C EEPROM. You can program directly in LASM. Refer to the documents in the Resources section of this article for more information on the virtual machine and byte codes.

There are two options to program in a higher-level language: MindScript and NQC. The former is a language developed by Lego for use in its robot products. The latter, which is short for "not quite C," is a programming language similar in syntax to C. Both languages, by way of their respective compilers, generate the byte codes to be interpreted by the virtual machine.

In addition to running the virtual machine, the processor performs background tasks such as battery monitoring and communications processing. The Spybot ROM also contains the software for Lego's mission engine architecture. In the standard Spybot software, each Spybot can be programmed to perform a mission based on various parameters.

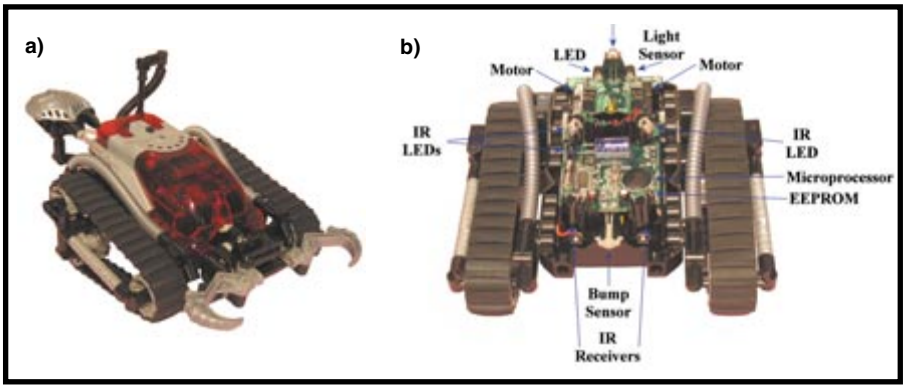


Photo 1a—*This is a Lego Snaptrax S45 Spybot with the electronics visible beneath the clear red canopy. All Spybots share the same integrated controller but use different mechanisms for locomotion.* ***b—****Unlike the Lego Mindstorms, Spybots are not expandable. They contain standard robot features such as two motors and simple sensors. Also included is a complex infrared communications system for Spybot-to-Spybot messaging.*

The missions are games with specific rules and goals. The ROM engine contains functions and control software for operating a Spybot in Mission mode. It is up to you to decide whether or not to use the ROM engine. If you do, some of the Spybot's resources are taken over by the engine and you lose a level of flexibility.

I²C EEPROM Address	PIC Memory	PIC Address
0x0000–0x007F	Data EEPROM	0x00–0x7F
0x0080–0x008F	RAM	io_buffer[]
0x0090–0x00FF	Not mapped	
0x0100–0x0FFF	Program flash memory	0x1100–0x1FFF

Table 1—*Emulated EEPROM accesses are mapped in three different memory regions of the PIC. Variable space is mapped to the PIC's data EEPROM. User-defined I/O space is mapped to an array in RAM. Code space is mapped to the PIC's program flash memory.*

OPENING THE SYSTEM

With all that the Spybot has to offer, there are plenty of things you can do without even cracking open the case. Before taking one apart, I encourage you to try a couple of missions—particularly those that require more than one Spybot—and attempt writing some MindScript or NQC code.

Any number of approaches may be taken to expand the Spybot. Reading through the documentation, I discovered that user code has access to the lowest 256 bytes of the I²C EEPROM. Lego reserved the first 128 bytes of this space for internal variable use, with the remaining 128 bytes being user defined. I chose to replace the I²C EEPROM with a processor containing I²C slave hardware and enough memory to store the Spybot's code. The processor emulates the I²C EEPROM's memory while blocking out a small region of the user-defined 128 bytes for I/O space.

I²C EEPROM OPERATION

The I²C EEPROM, an STMicroelectronics M24C32-R, contains 4 KB of nonvolatile memory accessible through an I²C bus. The M24C32-R is an I²C slave that allows an I²C master to perform read and write operations. (If you're unfamiliar with the I²C standard, refer to the Philips Semiconductor I²C web site.)

Writes are broken into two categories: byte write and page write.[1] I'm going to treat them the same because a byte write is identical to a page write when only 1 byte is written.

A write cycle begins by loading the address register. This is done by sending a start condition, followed by the device ID with the read/write bit set low (write), and then 2 bytes of address (see Figure 1a). The EEPROM acknowledges each address byte and loads the address register. Following the address bytes, up to 32 data bytes may be written. However, these bytes must all reside in the same row of the EEPROM. In other words, bits 12 through 5 of the address must not change. After the last byte, the master generates a stop condition and the EEPROM begins the internal write cycle. During the write cycle, the EEPROM ignores all I²C bus requests. A master may poll the EEPROM by sending a start condition followed by the device ID. When the EEPROM acknowledges, the write cycle is complete and the master may perform the next operation. If no acknowledgement is received, the master must wait and poll again.

Read cycles are also divided into different categories, but you can boil these down to random address and current address cycles, reading from one or more addresses (see Figure 1b). For a random address read, the master loads the address register by writing the two address bytes as it did at the start of a write cycle. A new start condition is generated, followed by the device ID with the read/write bit set high (read). The EEPROM reads from the current address (pointed to by the address regis-

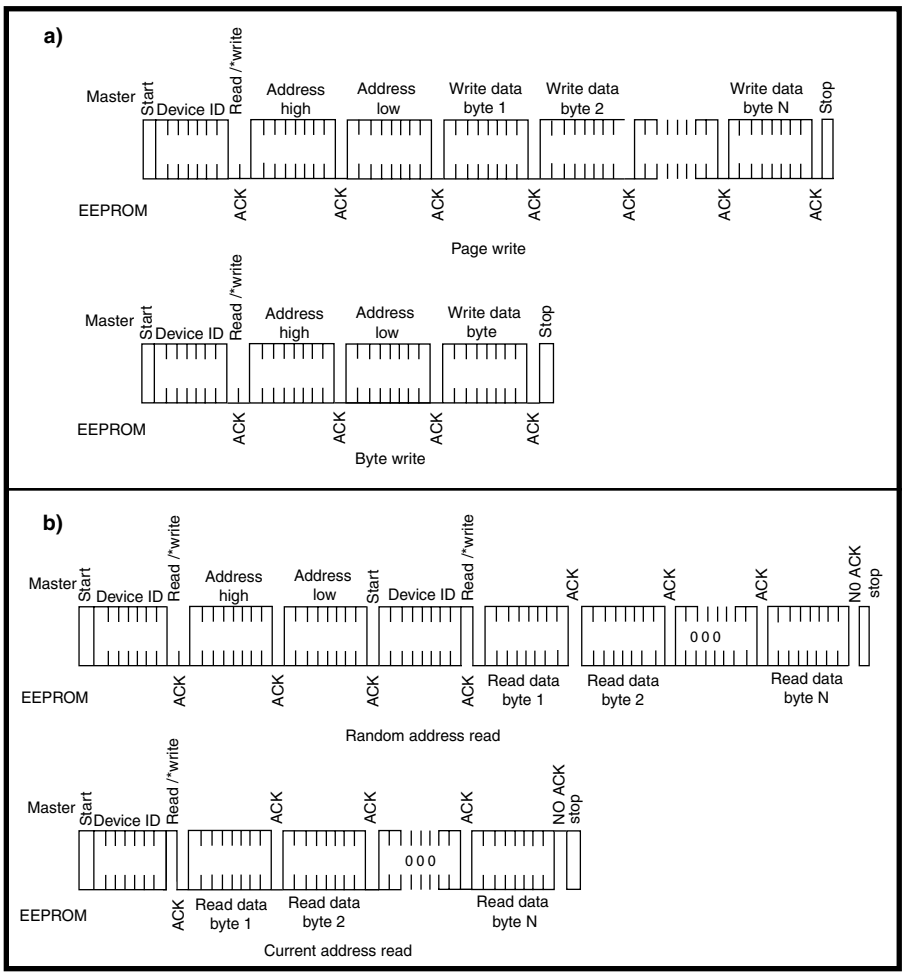


Figure 1a—*Write cycles always start by writing to the address register. Up to 32 bytes may be written at a time in a page write operation. A byte write is identical to a page write where only 1 byte is written.* ***b—****A random address read cycle loads the address register just like the start of a write cycle. Reads from memory then take place starting at that address. Current address read cycles use the existing contents of the address register.*

ter) and sends the result to the master. The address register is then incremented. If the master wants to continue reading, it generates an ACK, which causes the EEPROM to transmit another byte. When the master is done, it must not generate an ACK. This signals the EEPROM to release the bus, allowing the master to finish with a stop condition.

A current address read cycle is identical to the random address read, except that no address bytes are written to the EEPROM. The existing contents of the address register are used to determine the read address. The read portion of both the random access and current access cycles are the same.

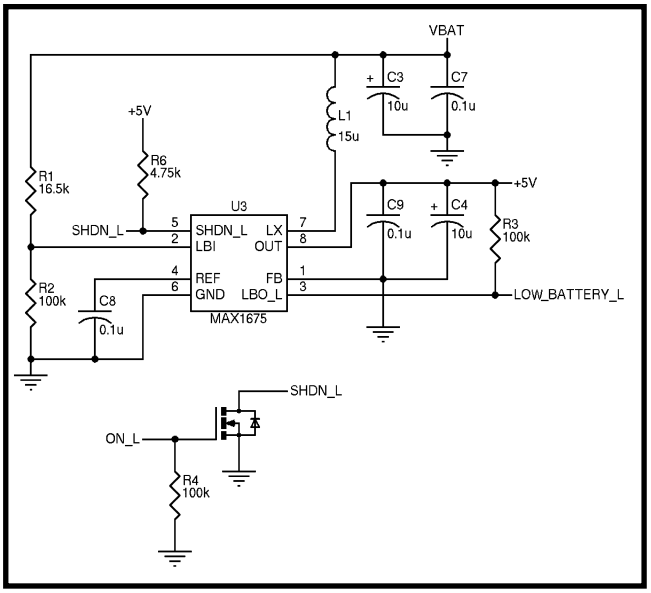


Figure 2—*To operate the PIC at the required clock speed and voltage range, a three cell-to-5 V boost regulator is used. The Maxim MAX1675 steps up the input battery voltage to a regulated 5 V.*

EEPROM EMULATION

You need to start with the basics, which means taking a close look at the M24C32-R datasheet. Examining the "Features Summary" table, the top-level requirements can be extracted. The EEPROM contains 4 KB of storage, with an endurance of more than 1 million erase/write cycles. The I²C interface needs to operate up to 400 kHz with an operating voltage between 1.8 and 5.5 V.

I chose the Microchip PIC16F876. I've used the CCS C compiler and the PIC's I²C slave hardware in previous projects. A PIC16F876 contains 8K × 14 bits of program flash memory and 256 bytes of data EEPROM. For generic EEPROM emulation, you would use half of the program flash memory for emulation and half for PIC code. In the case of a Spybot, you know that 256 bytes of the memory are used for variable storage and the rest for program space. Because the PIC's data EEPROM has a higher erase/write endurance than the program flash memory, mapping the variable space into the PIC's data EEPROM makes sense. I mapped the upper 128 bytes of the variable region as the I/O space.

For simplicity, I chose to map 1 byte of emulated EEPROM to one word (14 bits) of PIC program flash memory (see Table 1). If program flash memory





space becomes an issue, some form of compression could be done. The flash memory is reprogrammed every time the Spybot user code changes. The PIC16F876 flash memory endurance has a minimum of 1000 erase/write cycles, with the EEPROM at 100,000 erase/write cycles. With 1000 changes in the realm of possibilities for the lifespan of my Spybot, I decided to look at the PIC16F876A with its improved endurance of 10,000 erase/write cycles.

In obtaining the extended endurance, some complexity was added to the programming sequence. I prototyped with the PIC16F876 and then switched to the PIC16F876A. If you use the "A" version parts, make sure the date code is 0242xxx or later. (Refer to the PIC datasheet to see where to find date code markings.) Earlier "A" version parts may not operate reliably above 4 MHz.[2]

In order to support a 400-kHz I²C interface, the PIC needs to operate at higher than 10 MHz. The low-voltage PICs have a maximum clock rate of 4 MHz at 2 V. At this point, I began to question if my processor was the correct one, so I looked into a number of other processors from different vendors. After poring over datasheets and a bit of head scratching, I determined that they all had had similar issues. None stood out as an obvious better choice.

To run at higher than 10 MHz and down to 1.8 V, I decided to use a three cell-to-5 V boost regulator circuit. According to the datasheet, a PIC16F876A operating at 5.5 V and 20 MHz—with all I/O pins tristated and pulled up—will draw a maximum of 15 mA. I ran my first prototypes without a regulator by providing a regulated 5 V to the Spybot. I observed that 15 mA was a reasonable estimate for the processor alone. To provide for expansion, I decided to look at regulators in the 100- to 200-mA range. By keeping the current requirements reasonable, the regulator circuit can be small. I chose the Maxim MAX1675 step-up DC-DC converter for its small size and simplicity (see Figure 2).

Besides power and ground, the only connections required between the PIC and the Spybot are clock and data for the I²C bus. I connected to the positive battery voltage just after the Spybot's fuse and my ground to the location where the negative battery terminal attaches to the Spybot's PCB. Connecting to the Spybot I²C bus was a bit trickier. The I²C EEPROM is the eight-pin surface mount device between the epoxy blob and the left infrared receiver. I chose to remove the M24C32-R completely and solder wires to the proper pad locations.

My prototype also includes connections to the two Spybot infrared receivers and a minimal number of I/O ports (see Figure 3). I use a Fairchild NC7SZ157 TinyLogic multiplexer to select the right or left infrared receiver. To program and debug the PIC, I brought the in-circuit debugger (ICD) pins to a connector.

Aesthetically, I wanted to maintain the look of the Spybot. This meant the circuit board with the boost regulator, PIC, multiplexer, and connectors had to fit inside the plastic canopy. Placing a number of connec-



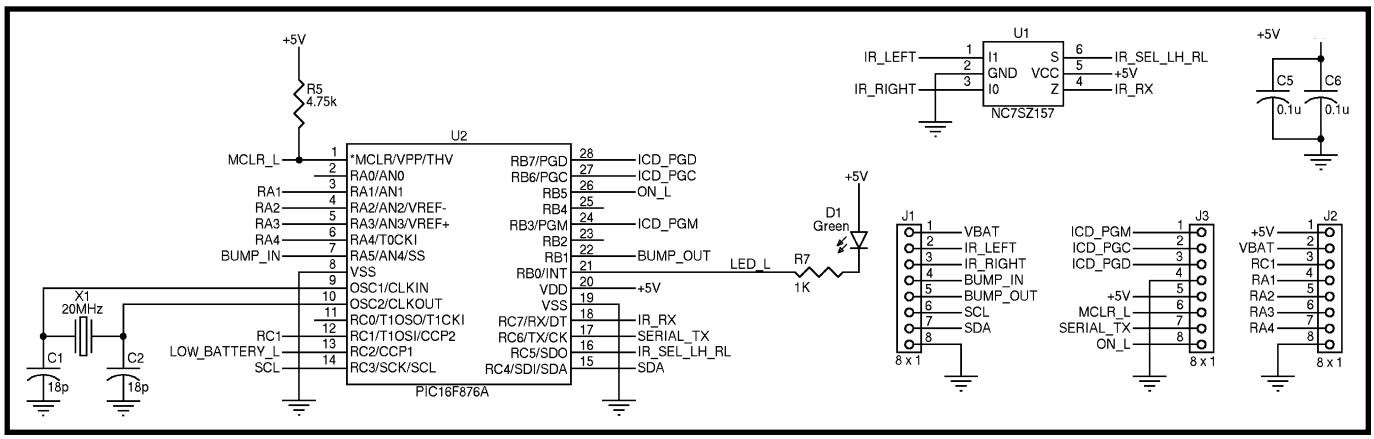


Figure 3—*A bare-bones PIC configuration includes just the basics. I selected expansion pins to include connections to the internal A/D converter and timing hardware. The multiplexer allows the PIC to decode received serial data from either of the two infrared receiver modules.*

tors on the board wasn't going to fit, so I opted to use a connector for the ICD, and the rest of the connections were soldered in place. The pins on the ICD connector needed trimming to fit (see Photo 2).

EEPROM EMULATION SOFTWARE

I used the PIC's master synchronous serial port (MSSP) hardware to implement an I²C slave. Although the CCS C compiler provides functions that simplify the MSSP configuration and basic operations, I explicitly coded these functions to show the details of what is being done. For interrupts, the compiler generates code that saves the processor state, calls the appropriate interrupt handler, clears the interrupt, and restores the processor state. If you port the code to a different compiler, make sure the interrupts are handled properly.

The configuration of the MSSP is done in the `i2c_init()` function (see Listing 1). The status register (SSPSTAT) is cleared, which also sets the I²C slew rate control for 400-kHz operation. The device ID is loaded into the address register SSPADD. The configuration register (SSPCON) enables the MSSP, and configures it for I²C Slave mode using 7-bit addressing, with the interrupts occurring for byte transfers.

The MSSP monitors the I²C bus looking for a start condition. When a start occurs, it clocks the next byte (device ID and read/write bit) into the SSPSR register. (This is an internal register, accessible only by the MSSP hardware.) Bits 7 through 1 of the SSPSR are compared to bits 7 through 1 of the device ID stored in SSPADD. If a match occurs, the MSSP will acknowledge the master, place the contents of SSPSR into the SSPBUF, and generate an interrupt.

Listing 1—*`i2c_init()` configures the MSSP as an I²C slave. The entire emulation of the I²C EEPROM is performed in the interrupt function `ssp_interrupt()`. The error-handling and read operations are shown here. Overflow and write collision errors must be cleared by software. For read operations, 1 byte is always preread from memory based on the current value of the address register. This provides for a fast data turnaround, and it limits the amount of time the master is kept waiting.*

```
void i2c_init(void) {
   SSPSTAT = 0x00;
   SSPCON2 = 0x00;
   SSPADD = I2C_DEVICE_ID;
   SSPCON = 0b00110110;
}
#INT_SSP
void ssp_interupt()
{
   byte i;
   byte d;
   int16 temp_addr;
   if (SSPCON_SSPOV || SSPCON_WCOL) {
      incoming = SSPBUF;
      SSPCON_SSPOV = 0;
      SSPCON_WCOL = 0;
      writeState = IDLE;
   }
   else if (!SSPSTAT_BF && SSPSTAT_RHWL) {
      SSPBUF = pre_read;
      SSPCON_CKP = 1;
      address++;
      if (address == 0x2000) {
         address = 0x0000;
      }
      else if (address == 0x0100) {
         address = 0x1100;
      }
      if (address & 0xff00) {
         pre_read = read_program_eeprom(address);
      }
      else if (address & 0x0080) {
         pre_read = io_buffer[((byte)(address & 0x000f))];
      }
      else {
         pre_read = read_eeprom((byte)(address & 0x00ff));
      }
      writeState = IDLE;
   }
   else ...
```

What happens next depends on the read/write bit. For write operations (read/write bit set low), the MSSP sets the buffer full flag (SSPSTAT_BF). Software must read the data out of SSPBUF, which automatically clears SSPSTAT_BF. Meanwhile, the MSSP awaits an additional byte from the master. While waiting, if a stop or start condition is detected, the MSSP logic resets and goes back to look for a device ID match. Assuming the MSSP logic is not reset, each additional byte received is shifted into the SSPSR. As long as SSTSTAT_BF is clear, the MSSP will acknowledge the byte, move it into the SSPBUF, set SSPSTAT_BF, and generate an interrupt. At this point, the software must again read the byte from SSPBUF.

The byte reception process repeats until the MSSP logic is reset. If the SSPSTAT_BF is set at any time when the MSSP attempts to move a byte from the SSPSR, it means the software hasn't read SSPBUF in time. An overflow condition will occur (SSPCON_SSPOV is set), and the MSSP will not acknowledge the byte. It is up to the software to clear overflow conditions by reading SSPBUF (clearing SSPSTAT_BF) and manually clearing SSPCON_SSPOV.

For read operations (read/write bit set high), the MSSP does not set SSPSTAT_BF, and it holds the I²C clock line low, forcing the master to wait for data. The MSSP expects software to write data into SSPBUF and release the clock line by setting SSPCON_CKP. A write collision error (SSPCON_WCOL) could occur if software writes to SSPBUF while it is full. (SSPSTAT_BF and the master are still sending the byte out on the I²C bus.) SSPCON_WCOL must be cleared by software if it occurs.

If the master acknowledges the transfer, the MSSP assumes it wants another byte, clears SSPSTAT_BF, sets the I²C clock line low, and generates an interrupt, causing the process to repeat again. When the master does not generate an acknowledgement, the MSSP releases the I²C clock, which allows the master to generate a stop condition, resets its logic, and goes back to look for a device ID match.

The entire emulation of the I²C EEPROM is performed in the interrupt function ssp_interrupt(). This function is executed whenever a byte is received from the I²C bus or a byte needs to be transmitted to the I²C bus.

The first thing to do is handle any errors (see Listing 1). Check for an overflow (SSPCON_SSPOV) or a write collision (SSPCON_WCOL). If either occurs, clear out SSPBUF, reset the error flags, and force your write logic to wait for a new transfer (aborting any operation in progress).

Determining whether a read or write operation is taking place is as simple as checking to see if SSPSTAT_BF is cleared (read operation) or set (write operation). Processing a read operation is handled next in the code, but it makes more sense to discuss write operations first (see Listing 2).

In write operations, you need to keep track of which byte is currently being processed. This is accomplished with the writeState variable, which has the following possible states: IDLE, DEVICE_ID, ADDRESS_HIGH, and ADDRESS_LOW. The states are labeled to indicate which byte has been processed in the past tense.

Listing 2—*The writeState variable keeps track of which byte has been received. After the two address bytes are loaded, data bytes are processed up to a stop condition. Timing is critical between detection of the stop condition and disabling of the MSSP.*

```
case IDLE :
    incoming = SSPBUF;
    if (!SSPSTAT_DHAL) {
        writeState = DEVICE_ID;
    }
    break;

case DEVICE_ID :
    incoming = SSPBUF;
    address = (((int16)incoming) << 8) & 0x0f00;
    count = 0;
    writeState = ADDRESS_HIGH;
    break;

case ADDRESS_HIGH :
    incoming = SSPBUF;
    address |= ((int16)incoming) & 0x00ff;
    if (address & 0xff00) {
        address |= 0x1000;
        pre_read = read_program_eeprom(address);
    }
    else if (address & 0x0080) {
        pre_read = io_buffer[((byte)(address & 0x000f))];
    }
    else {
        pre_read = read_eeprom((byte)(address & 0x00ff));
    }
    writeState = ADDRESS_LOW;
    break;

case ADDRESS_LOW :
    while (SSPSTAT_BF) {
        data[count] = SSPBUF;
        PIR1_SSPIF = 0;
        count++;
        while ((!PIR1_SSPIF | !SSPSTAT_BF) & !SSPSTAT_P);
  }
  SSPCON_SSPEN = 0;

  // Write to program flash memory, io_buffer[], or data EEPROM.
  // See full source for details.

  address = address + count;
  count = 0;
  SSPCON_SSPEN = 1;
  break;
```

writeState is set to IDLE at initialization, when you determine the current byte to be processed is a device ID (by examining SSPSTAT_DHAL), when a read operation is performed, or if an error occurs. While in the IDLE state, read the incoming byte and check to see if it is a device ID (SSPSTAT_DHAL cleared). If this is the case, set writeState to DEVICE_ID and wait for the next byte.

The two address bytes always follow a device ID in a write operation. As the two bytes are processed, map them into the address register, and the write byte counter is cleared. Recall that the writing of two address bytes precedes certain read operations. When the address is complete (in the ADDRESS_HIGH state), take the time to preread data from the appropriate location (program flash memory, data EEPROM, or the I/O buffer). This speeds up the response time for read operations, and has minimal overall effect.

Actual bytes to be written are processed in the ADDRESS_LOW state. (Remember, the states are past tense, so ADDRESS_LOW occurs after you have processed the low address byte.) This is a complex state. After it is entered, it will not exit until a stop condition occurs. A tight polling loop waits for each byte or a stop condition. As each byte is received, it is placed in a data buffer, and the write byte counter is incremented. When the stop condition is detected, clearing SSPCON_SSPEN immediately disables the MSSP. It is necessary to do this quickly because the master may start polling to see if the write operation is complete. If the MSSP has the chance, it will automatically acknowledge the master's poll request, and you will lose the opportunity to save the data. I found this to be the most time-critical section of code in the entire project.

The amount of code required to actually write the data to the program flash memory depends on which PIC is used. The PIC16F876 allows single-word writes to program flash memory, where the PIC16F876A operates four words at a time. Each byte in the data buffer is written to the proper destination (program flash memory, data EEPROM, or the I/O buffer). In the simplest case, bytes are written one at a time, therefore taking advantage of the CCS compiler functions. In the most complex case, when writing to program flash memory, you need to perform alignment operations to handle the four words at a time and are unable to completely rely on the compiler functions.

When a read operation takes place, you receive an interrupt with the SSPSTAT_BF clear (see Listing 1). The preread data is immediately output to the master by loading SSPBUF and setting SSPCON_CKP to release the clock. After incrementing the address register, the next data byte is preread and ready for another read operation.

ENHANCED OPERATION

Adding capabilities to the Spybot is done in the main() program loop. After initializing variables and configuring the peripherals, main() enters an infinite loop. There, you can read and write the contents of the io_buffer[] to perform specific functions.

Listing 3 is an example of monitoring a bit to turn on or off a heartbeat

Listing 3—*An infinite loop in the main() code monitors the io_buffer[] array for commands. This is a simple example to turn on and off the heartbeat LED. A GP2D12 is connected to an analog input and is read using the CCS read_adc() function. (The proper analog port was previously selected.)*

```
while (TRUE) {
   // Heartbeat LED control

   if (io_buffer[CONTROL_REG] & CONTROL_LED_ON) {
      output_low(HEARTBEAT_LED);
   }
   else {
      output_high(HEARTBEAT_LED);
   }
   // GP2D12 ranging

   if (io_buffer[CONTROL_REG] & CONTROL_GP2D12_SAMPLE) {
      io_buffer[GP2D12_RANGE_REG] = read_adc();
   }

}
```

Listing 4—*Accessing the PIC's I/O buffer is done on the Spybot using the MindScript eeprom[] command.*

```
program spypic_heartbeat
{
    #include<Spybot.h>
    const CONTROL_REG = 0x80
    const HEARTBEAT_LED_ON = 0x01
    const HEARTBEAT_LED_OFF = 0xfe
    main
    {
        local reg
        local x
        forever {
            reg = eeprom[CONTROL_REG]
            x = reg
            x &= HEARTBEAT_LED_ON
            if x = HEARTBEAT_LED_ON {
               reg &= HEARTBEAT_LED_OFF
            }
            else {
               reg |= HEARTBEAT_LED_ON
            }
            eeprom[CONTROL_REG] = reg
            wait 100
        }
    }
}
```

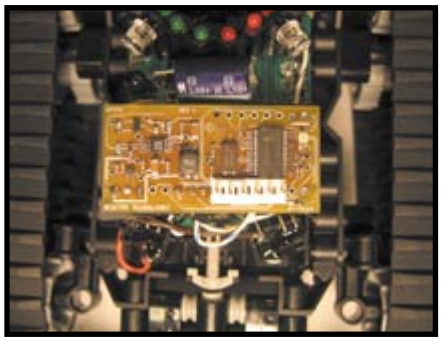


Photo 2—*The circuit board contains a PIC microprocessor, a three cell-to-5 V boost regulator, and a mulitplexer connected to the Spybot infrared receivers. In order to fit under the canopy, all connections are hard-wired in place. The white connector brings out pins for an in circuit debugger. I tried to keep all signals on the top with copper pours for power and ground the back of the circuit board.*

ACCESS FROM MINDSCRIPT

Reading and writing the I/O buffer is done in MindScript using the `eeprom[]` command. Listing 4 is a simple example of how to toggle the heartbeat LED using MindScript. Using an XOR operation would have been more efficient; unfortunately, LASM (and therefore MindScript) doesn't have one! A much more complete MindScript example that includes GP2D12 and SRF04 Sonar access is posted on the *Circuit Cellar* ftp site.

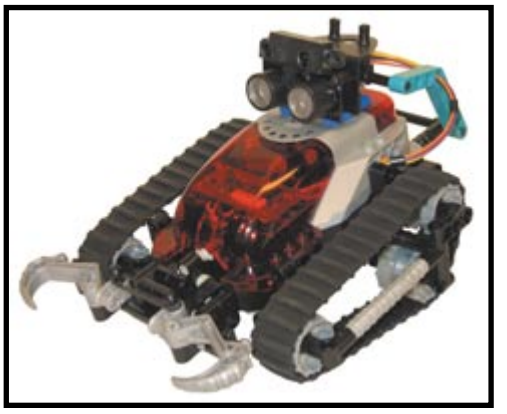


Photo 3—*The enhanced Spybot has a GP2D12 infrared range finder and an SRF04 ultrasonic ranger. The circuit board's underside is covered with electrical tape and fits snugly under the canopy.*

WHAT'S NEXT

I'm fairly happy with the results of this project, but there is always room for improvement. The boost regulator is a bit noisier than I would like. This is most likely the result of my initial component selection. A little more time spent could have brought improvements. My original plan was to use a regulator with a shutdown option because the Spybot doesn't have a global power off. The shutdown option of the MAX1675 isn't a true shutdown. In shutdown, the internal switching MOSFET conducts the battery voltage through its body diode. As long as I have batteries in the Spybot, the PIC is running.

I hope this project not only sheds light on the capabilities of the Spybot, but also provides a reasonable reference for a PIC-based I²C slave. Most of all, I LED, as well as reading the analog output of a Sharp GP2D12 infrared range finder. Adding even more sensors, such as a Devantech SRF04 Sonar, is easy (see Photo 3). You may download the full PIC source code from the *Circuit Cellar* ftp site; it includes configuration and control of the GP2D12 and SRF04 Sonar.

x86 Embedded Processor Module & Single-Board-Computer

Mity-SOC-1
386 Embedded system module
1 RS-232, 1 RS-232/485, 16 GPIO
IDE, FDD, RTC, Parallel, Watchdog
2MB RAM, 2.56" X 1.77" (Optional 4MB)
$65.00 ea. Single unit
$50.75 ea. Qty. 100 pricing

Mity-Mite
386 Embedded system module
1 RS-232, 1-RS-232/485, 16 GPIO,
Ethernet, IDE, RTC, Watchdog, K/B
4MB RAM, 3.14" X 1.96"
$100.00 ea. Single unit
$78.00 ea. Qty. 100 pricing

Vortex86-6082
Vortex86 Embedded system module
1 RS-232, 1-RS-232/485, Parallel, USB
Ethernet, IDE, VGA, K/B, Mouse, RTC,
Watchdog, 128MB RAM, 3.94" X 2.60"
$262.00 ea. Single unit
$182.00 ea. Qty. 100 pricing

Vortex86-6071
PC/104 Vortex86 Embedded SBC
3 RS-232, 1-RS-232/485, Parallel, USB, IDE
Audio, Ethernet, CRT/LCD, RTC, Watchdog
K/B, Mouse, 128MB RAM, 3.77" X 3.54"
$308.00 ea. Single unit
$214.00 ea. Qty. 100 pricing

ICOP-6027VE
3.5" 386 Embedded SBC
CRT/LCD, 1 RS-232, 1 RS-232/485, K/B
Parallel, DiskOnChip, IDE, FDD RTC,
Watchdog, 4MB RAM, 4.01" X 5.67"
$222.00 ea. Single unit
$154.00 ea. Qty. 100 pricing

ICOP-6050
PC/104 386 Embedded SBC
1 RS-232, 1-RS-232/485, 1 parallel, K/B
DiskOnChip, IDE, FDD, RTC, Watchdog
4MB RAM, 3.77" X 3.54"
$142.00 ea. Single unit
$99.00 ea. Qty. 100 pricing

Vortex86-6075
Vortex86 Embedded SBC
1 RS-232, Parallel, 3 USB Ethernet, IDE,
VGA, K/B, Mouse, RTC, Video-in, TV-out
Audio, Watchdog,128MB RAM, 4.37" X 5.24"
$266.00 ea. Single unit
$185.00 ea. Qty. 100 pricing

Vortex86-6047
3.5" Vortex86 Embedded SBC
3 RS-232, 1-RS-232/485, Parallel, USB, IDE
Audio, Ethernet, CRT/LCD, RTC, Watchdog
K/B, Mouse, 128MB RAM, 4.01" X 5.67"
$326.00 ea. Single unit
$226.00 ea. Qty. 100 pricing

Supported OS & Development Environment

DOS
Using C/C++, DOS application can be develop to run on all of our processor modules. DSock, a TCP/IP library for DOS, is provided to develop application with Internet connectivity. Sample implementation for BOOTP/DHCP, FTP, SMTP, HTTP, TELNET & TALK are available for download from our Web site. www.dmp.com.tw/dsock

Embedded Linux
Linux application can run on all of our processor modules. We provide X-Linux, an embedded Linux kernel based on the current popular distribution. X-Linux is a head-less kernel approx. 3MB in size. It includes Linux Kernel 2.4.18, SysLinux Loader, BusyBox Shell, FTP4ALL, udhcp Client, WIN HTTP, glibc & Web based administration.

Windows CE .NET
Vortex86 BSP for Windows CE .NET has been certified by Microsoft Windows CE .NET BSP certification program. Windows CE .NET applications can be develop using Embedded Visual C++, Visual Basic .NET & Visual Studio .NET with Compact .NET Framework library. Please check our Web site for Windows CE .NET SDK information.

ICOP
intelligent control on processor

ICOP Technology Inc.
Tel: (626) 444-6666 Email: info@icoptech.com
URL: www.icoptech.com

Microsoft® Windows® Embedded Partner
ICOP is a Gold-Level-Partner of Microsoft Windows Embedded Partner program. Gold WEP has been created by Microsoft to identify companies with demonstrated expertise to support Windows Embedded Technologies.

Contact us for custom design & OEM/ODM services.
Product and service names mentioned herein are the trademarks of their respective owners. Not responsible for error. Prices are subject to change without prior notice.

hope I've inspired you not to shy away from those nasty blobs of epoxy!

Jay Francis is the sole proprietor of Reactive Technologies, where he provides freelance hardware design services. As a hardware engineer for more than 12 years, his designs have spanned networking equipment, document scanners, and mobile robots. In his free time, Jay enjoys spending time with his family and racing bicycles. You may e-mail him at JayFrancis@aol.com.

PROJECT FILES

To download the code and, go to ftp.circuitcellar.com/pub/Circuit_Cellar/2004/165.

REFERENCES

[1] STMicroelectronics, "M24C64, M24C32: 64Kbit and 32Kbit Serial I²C Bus EEPROM," 2003.

[2] Microchip Technology, Inc., "PIC16F87XA Rev. B0 Silicon Errata Sheet," DS80128F, 2003.

RESOURCES

I²C information, www.semiconductors.philips.com/buses/i2c/.

Lego Co., "LEGO P-brick Script Code Language," Draft, 2000.

———"LEGO Spybotics ROM documentation," 2002.

———"Spybotics Firmware Command Overview," 2002.

Lego Users group network, www.lugnet.com.

Microchip Technology, Inc., "PIC16F87X Data Sheet," DS30292C, 2001.

———"PIC16F87XA Data Sheet," DS39582A, 2001.

———"PICmicro Mid-Range MCU Family Reference Manual," DS33023A, 1997.

SOURCES

Devantech SRF04 ultrasonic range finder, Sharp GP2D12 distance measurement sensor
Acroname, Inc. (distributor)
(720) 564-0373
www.acroname.com

PIC C Compiler
CCS, Inc.
(262) 797-0455
www.ccsinfo.com

NC7SZ157 TinyLogic multiplexer
Fairchild Semiconductor Corp.
(207) 775-8100
www.fairchildsemi.com

Software development kit (version 2.5), Spybotics robot
Lego Co.
www.lego.com

PIC16F876 and PIC16F876A Microcontrollers and ICD
Microchip Technology, Inc.
(480) 786-7200
www.microchip.com

P0770.153T 15-µH inductor
Pulse Engineering, Inc.
Mouser Electronics (distributor)
(817) 804-3888
www.mouser.com



PROTEUS

The Complete Electronics Design System



Schematic & PCB Layout

- Powerful & flexible schematic capture.
- Auto-component placement and rip-up/retry PCB routing.
- Polygonal gridless ground planes.
- Libraries of over 8000 schematic and 1000 PCB parts.
- Bill of materials, DRC reports and much more.

Mixed Mode SPICE Circuit Simulation

- Berkeley SPICE3F5 simulator with custom extensions for true mixed mode and interactive simulation.
- 6 virtual instruments and 14 graph based analysis types.
- 6000 models including TTL, CMOS and PLD digital parts.
- Fully compatible with manufacturers' SPICE models.

New Features in Version 6

- Drag and drop toolbars.
- Visual PCB packaging tool.
- Improved route editing.
- Point and click DRC report.
- Multiple design rules (per net).
- Multiple undo/redo.

Call Now for Upgrade Pricing

* PIC BASIC Plus from Crownhill Associates

Proteus VSM - Co-simulation and debugging for popular Micro-controllers

- Supports PIC, AVR, 8051, HC11 and ARM micro-controllers.
- Co-simulate target firmware with your hardware design.
- Includes interactive peripheral models for LED and LCD displays, switches, keypads, virtual terminal and much, much more.
- Provides source level debugging for popular compilers and assemblers from Crownhill, IAR, Keil, and others.

STARTING AS LOW AS

$199

With Manual on CD

Labcenter Electronics

Free DEMO CD **www.labcenter-electronics.com**

Contact - R4 Systems at 905 898-0665 or info@r4systems.com



by Fred Eady

A Wireless Ethernet Solution for the

Finding the right wireless solutions for your embedded projects can be difficult, especially if you're on a tight budget and can't get the right parts and documentation. That's why Fred set out to find an accessible solution for those of you searching for a sure-fire way to bring wireless Ethernet capability to your designs. DPAC's wireless 802.11b evaluation and development kit turned out to be the perfect fit.

No doubt about it, I'm a nerd. I was contemplating the structure of this month's column and it occurred to me that I am much like the Star Trek characters that were "looking for things to make them go." I won't classify any of you as nerds, but I attempt to write each column with the idea that you guys are looking for things that make you go as well.

This month's column had a rocky start. As you know, you need to closely examine a device's associated technical documentation to gain a working understanding of it. When beginning a project, the logical starting point is to collect all the technical information that you can from wherever you can.

So, I contacted an extremely large company (whose name I won't divulge here) about getting a technical document that explained how to write a driver for its PRISM wireless Ethernet chipset. After a few e-mails and telephone conversations, I was asked to assemble a request letter because the document I was asking for was considered "sensitive material." The company's representative needed assurance that I would not compromise any of the chipset's secret stuff by writing about it in *Circuit Cellar*. I figured that that was fair, and drafted a letter stating my intent and willingness to preserve the security of the company's intellectual capital. I was summarily kicked to the curb. Denied.

My intention was to show the *Circuit Cellar* audience how to integrate standard off-the-shelf wireless Ethernet cards that contain the PRISM chipset with simple embedded devices. Because I could not gain any insight into the PRISM technology, I could not construct the required drivers. I trashed the idea.

Sometimes there is victory in defeat. I'm an advocate for the little guy, and I was determined to find a clearer path to a working wireless Ethernet solution despite being rejected by the corporate technology holder. After some rather exhaustive research, I found a local company that was famous (on the Internet at least) for its Linux wireless Ethernet prowess. I contacted the company to talk about its Linux wireless Ethernet solution because I figured it would have some handy tips and tricks to share.



Photo 1—*Everything Wi-Fi is packed into this tiny two-story module. A 36-pin connector, which you can't see in the photo, interfaces the Airborne WLN to your circuitry. Note the diversity antenna connectors on the right-hand side of the top board of the module.*

As it turned out, the company's hands were tied by corporate nondisclosure as well. About the only thing it could recommend was that I study its Linux code and do some porting. The guys at the local Linux shop were as good to me as possible, but picking apart and porting what appeared to be some pretty extensive public domain Linux driver code was not in my future. My enthusiasm for this article began to wane because my search for a relatively easy wireless Ethernet solution seemed to be grinding to a halt.

But I kept at it. As it turns out, I ended up finding exactly what I was looking for. To make it sweeter, the wireless Ethernet technology provider's device you're about to read about comes in an evaluation kit—and you don't need some corporate guy's permission to get it!

WI-FI IN A MATCHBOX

Originally, I figured that the PCMCIA-based laptop wireless Ethernet cards would be a really slick way to add wireless Ethernet connectivity to an embedded device. In theory, I was correct; in reality, I was skipping around in wonderland.

There are drawbacks to implementing a wireless Ethernet solution using an off-the-shelf laptop card. For instance, you have to deal with the physics and logic of the card's PCM-

CIA interface. You must also obtain basic information about how the chipset works (probably an impossible task unless Bill Gates is your cousin). Then, you have to write the driver after you have the correct information. If that's not enough to scare you off, you're also tied to the physical size of the laptop card as your minimum form factor.

When you're all finished with those chores, you'll need to set aside some time to write your application code and interface it successfully with that driver code written earlier. I already know from experience that some of you gals and guys in *Circuit Cellar* readerland have the savvy to pull something like this off. However, there's an easier way to get rid of the wires.

DPAC Technologies offers aspiring wireless LAN gurus a wireless 802.11b evaluation and development kit that features its Airborne wireless LAN node (WLN) module. The analog and digital I/O-capable Airborne module that is mounted on my Airborne WLN module evaluation board is one of a family of 802.11b modules offered by DPAC.

The Airborne wireless module, which easily fits in a standard matchbox, contains all the necessary electronics to implement a radio, baseband processor, and application processor. The module's application processor is responsible for handling the analog and digital I/O. It also provides web server and TCP/IP services. This means you can forego praying to the demons of RF development, and you don't need that corporate guy's secret handbook and a stack of third-party communications software to successfully complete your wireless 802.11 project.

The Airborne WLN module eliminates the hard stuff associated with developing a wireless 802.11 solution. It drops seamlessly into any gadget you want to wirelessly enable. Because the Airborne module adheres to the IEEE 802.11b standard, it can talk to any other similarly configured 802.11b-enabled device or access point. As for data, you've got to put something in to get something out, and the Airborne module is equipped with a serial interface and some digital and analog I/O ports. There's also a high-speed SPI interface available. With all of these RF and computing resources, the Airborne module could be the sole active component in some applications.

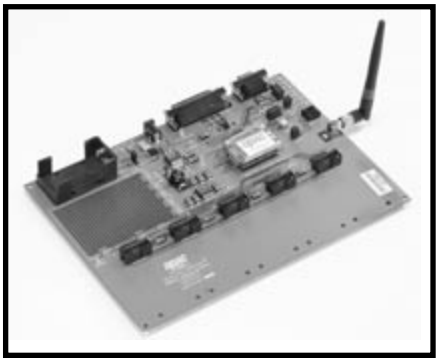


Photo 2—*There's really nothing remarkable about the Airborne WLN evaluation board; it has all the goodies you would find on any other evaluation board. The evaluation board's real purpose is to allow you to completely evaluate the Airborne WLN as a mobile unit. That's what the 9-V battery holder is for.*

Take a look at Photo 1 to see what instant Wi-Fi looks like. Now that you know what you're looking for and what to do with it when you find it, let's dig through the Airborne evaluation kit's box and see what's inside.

LOOK IN DPAC(AGE)

The star of this traveling wireless Ethernet show is the Airborne WLN module evaluation board, which is shown in Photo 2. Besides the standard DE9-to-DE9 serial cables, antenna, and power brick, the Airborne WLN evaluation board comes with a wireless access point/router. All you have to supply is a couple of PCs. One (remote) computer attaches directly to the wireless access point/router. The other (host) PC is owned by the Airborne WLN. I found that you could actually do some limited testing using a single PC.

The companion CD-ROM contains briefs describing the evaluation kit and the Airborne module, application notes, the evaluation kit user guide, a quick start guide, and the Airborne data book. There's also an evaluation utility that is essentially a control panel that gives the evaluator a hilltop view and control of all of the Airborne module's features.

DROP-IN WI-FI

Let's see just how quickly I can get the Airborne WLN on the air. As you can see in Photo 2, the Airborne evaluation board is dedicated to supplying the correct power supply voltages to the Airborne WLN and providing connecting points for every interface the module offers.

Although the Airborne WLN is capable of being set up via its 'Net interface, I started by connecting the Airborne evaluation board to my laptop's serial port and applying power. I had some lofty short-term goals in mind. So, instead of connecting the wireless access point/router that came with the evaluation board directly to a PC, I stuck it on the WAN interface of my high-speed Internet cable modem.

A Netgear MR814 wireless access point/router came with my evaluation board. After it's connected to an active WAN interface, the wireless access point/router automatically retrieves all of the necessary information from the ISP to put itself on the Internet. To keep things simple, I opted to use all of the router's local LAN default values.

After making sure the router was indeed talking to others on the Internet, I accessed its internal HTML settings page by entering "http://192.168.0.1" in the address window of an Internet Explorer window. There I noted the current WAN IP address that was being used by the wireless access point. While I was in this mode, I also visited the Wireless Settings area and assigned EDTP as the service set identifier.

The next step involved getting the Airborne WLN online and harvesting its LAN IP address, which should be given via DHCP by the Netgear wireless access point. One of the ways to communicate with the Airborne WLN is to use the command line interface (CLI) and a terminal program. So, I fired up Tera Term Pro on the laptop and steered the laptop's serial port settings to match the Airborne WLN's serial port settings. Actually, very little steering was needed because Tera Term Pro defaults to 9600 bps with 8 data bits, 1 stop bit, and no parity on COM1. All I had to do was make sure I was receiving full carriage return/line feed (CRLF) sequences instead of whacking off the LF.

The Airborne WLN logon command followed by a user name and password gets the node's attention. The Airborne WLN's defaults were used here as well ("dpac" for both user name and "oem" for the password). After I got the OK from the Airborne WLN, I instructed it to scan for access point SSIDs. There's only one wireless network in the Florida room, and had already I set its SSID to EDTP. So, I should have only received one scan result, EDTP, which I did.

After the successful scan, I ordered the Airborne WLN module to associate the WLN with the Florida room's wireless LAN. I then stored the new SSID information in the WLN's flash memory.

The WLN evaluation board is outfitted with multiple LED indicators that alert you to things like a successful Airborne WLN power-on self-test (POST), RS-232 activity, and RF link activity. After performing the SSID capture-and-store operation and doing an Airborne WLN evaluation board reset (press the Reset button), I noticed that the RF ACT LED was lit solidly. It had been blinking before. The blinking RF ACT LED indicates that the access point has found the WLN module. Therefore, this visual indicator eliminates having to write code to alert the user that things are working well in RF LAN land.

At this point I could query the Airborne WLN for its IP address. Again, I signed on through Tera Term Pro and logged onto the WLN module. To obtain the WLN IP address, I issued the CLI command. The Airborne WLN returned 192.168.0.6. Here we go! Returning to the wireless access point's settings web page, I opened the Port Forwarding window and selected Telnet from the Service Name drop-down menu. I entered "192.168.0.6" in the wireless access point's Server IP Address field. After clicking the Add button, I saw that the Telnet service defaulted to port 23, which meant I didn't have to edit the Telnet service entry to add a Telnet port number. I now had all of the IP address information necessary to finish setting up the wireless access point for the first over-the-Internet connectivity test.

Let's get our bearings. Right now, the Florida room's wireless Ethernet network consists of a laptop running Tera Term Pro that is attached serially (RS-232) to the Airborne WLN evaluation board (this is the Host per the DPAC EVB Guide), a wireless access point attached to the Internet via a cable modem with a standard Telnet portal installed on the wireless access point for access to the Airborne WLN, and a PC connected to a wired port of the wireless access point. As you have obviously ascertained from my wireless access point Telnet-enabling procedure, in addition to its local RS-232 serial port, the Airborne WLN also can be accessed via Telnet. I could have used the Telnet capabilities of Tera Term Pro to locally access the Airborne WLN, but I figured coming into the Airborne WLN via the Internet would be a better test.

I opened a Win2K Telnet window using the wired PC (the remote com-

SAELIG BRINGS YOU EASY-TO-USE SOLUTIONS!

USB in one IC!

FT232BM Single chip USB-232 solution with all Windows /Mac/Linux drivers. No programming/USB knowledge required! Easiest answer to update RS232 devices! Only $2.60 (10k)

USB<>RS485

USB-COMi (non-isolated) or **USB-485i** (isolated) offer self-powered USB to RS485 conversion with baud rates 184bps - 3Mbps. PC thinks it's talking to a COMport! Only $48/$170!

USB Logic Analyzer

ANT8 Matchbox-sized USB-powered logic analyzer. Sample 8 channels at up to 500 million samples-per-sec. View traces on PC. Print or save for later review! (16-ch. ver. too!) $199!

USB PC Scope

DS2200C 200 kS/s 12-bit 2-ch USB scope adapter for PC. Looks like a "Digital Scope" on your PC screen! Take anywhere. Great for laptops, and servicing. Remarkable price: Only $189!

USB Bus Analyzer

Tracker 110 USB protocol analyzer displays USB packets sent, decodes descriptors, errors, and measures USB s/w performance. Easy software - learn about USB now. Only $899!

USB-Serial Adapter

Serial port for laptops without one! **US232B** instantly updates older RS232 products to USB! PC thinks it's using COMport but uses the USB connection. Only $39 (1) or $$29 (100)

CANbus Cards

CANbus - Starter Packs and cards for almost any board format & OS. CAN/Ethernet bridges, industrial automation solutions from Janz AG as used by top companies all over the world.

Industrial PCs

ATX Ruggedized Industrial PCs for any budget or application. Easy mtnce, economy, reliability. AMD Athlon XP1700, shock-mounted 40GB harddrive. 100% Burn-in. CE EMC. $899^

Data Terminal

ELX customizable serial data collection terminals for HMI, machine monitor, control, etc. RS232/485 CE-approved. Backlit 4 x 20 ch. 4 isol. i/p, 32bit ctr, two relay o/p. Two non-vol. i/p. IP65 available.

I/O thru Ethernet

BITlink - easily construct control systems communicating through Intranet/Internet. **BIT2000** for process control, building monitoring, data logging, alarm systems and other industrial uses. $399 ^

RS232<>422/485

K2 9p-9p self-pwrd RS-422/485
K3 9p-9p isolated RS-422/485
K3-232 9p-9p isolated RS232
K232-ISOL 25p -25p RS232
KD485-STD DINrail - isolated
KD485-PROG programmable!

I2C for PCs

PCI93LV: industry-standard I2C card for PCs. **WINI2C/PCI** software gives windows-interface to develop and debug I2C bus systems. **UCA93LV** is new USB version. ***NOW!*** - transparently monitor at 400kHz! Only $499!

Dataloggers

VL100 2" x 3" battery-powered analog & digital loggers store events, volts, current, pressures for weeks. Download to PC and review graphic results or Excel spreadsheet. Only $200!

Dataloggers

TDS2020F makes a custom CF card A/D datalogger or controller - *quickly!* High-level software completes projects in hours. Store **GPS/CANbus/A or D** data to card. Only $199 (50 qty)

PC Scope Adapters

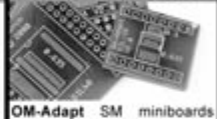
ADC-212/100 turns your PC into a great high-speed scope. Sampling to 100MS/s at 12-bit res'n. FREE s/w turns PC into powerful 2-ch scope, DVM & spectrum analyzer. **$95 - $1145**

BASIC modules

Tigers - tiny multitasking BASIC modules for quick projects. >100,000 instr's/s! **iCOM200** ready-made controller with LCD and keypad. **Touch240** controller - with touchpad and LCD display.

Crystals / Oscillators

Euroquartz - Europe's largest mfr of quartz crystals, osc's, filters and frequency products. Custom filters, high rel. rad tol. osc's. Novel **EQ-HM** spread-spectrum osc's reduce EMI problems

SM PCB Adapters

OM-Adapt SM miniboards have two footprints on either side. Now use your ultra fine pitch SMD components with more useful 0.1" inline spaced holes. One-to-one pinouts. $6.99^

*ALSO: SCALABLE LED DISPLAY PANELS. TEMP-HUMIDITY MONITORS. THERMOCOUPLE P.C. ADAPTERS. ENVIRONMENT MONITORING SYSTEMS. EDUCATIONAL SCIENCE PROJECTS. GRAPHICS SOFTARE. AutoCAD PROGRAMMING COURSE. USB-PIC BOARDS. FLASH PROGRAMMERS - IF YOU DON'T SEE WHAT YOU NEED MAYBE WE CAN FIND IT FOR YOU? - ASK FOR **SALES**!*

Saelig Co. Inc.
585-425-3753 • fax: -3835
www.saelig.com • info@saelig.com

***Saelig Co.** brings to USA unique, easy-to-use control and instrumentation products from overseas. Customers include: Intel, Philips, NEC, Kodak, Nokia, US Military, Microsoft, Dell, Xerox, Universities, T.I., Harris, Sony, J&J, Thomson, Sandisk, General Dynamics, H-P/Compaq, etc.*

puter per the DPAC evaluation board guide) and issued the Telnet open command followed by the WAN IP address of the wireless access point. Hmm, no error messages appeared. In fact, nothing appeared. So, I instinctively issued an Airborne WLN logon command with the proper arguments from the blank Win2K Telnet screen. A second or so later, I received an OK from the Airborne WLN.

Tera Term Pro is also equipped to perform TCP/IP duties. So, I tried the same operation from the wired PC using Tera Term Pro. Everything worked as advertised. Tera Term Pro allowed me to turn on local echo so I could at least see what I was typing. I was so excited that I called my friend Mark and had him sign on to the Airborne WLN from his machine shop. Instant Internet Wi-Fi indeed!

I had written 0 bytes of code and had successfully contacted my Airborne WLN evaluation board via Telnet from places afar. There was still one more way to contact my Airborne WLN—via a web browser. To get there from here, I had to revisit the wireless access point settings HTML page and open a portal for HTTP. After that, I simply entered "http://" followed by the wireless access point's current WAN IP address. I was prompted for the Airborne WLN username and password. After entering them, I was greeted by the oem.html page, which is the Airborne WLN default home page that holds the manufacturer's settings.

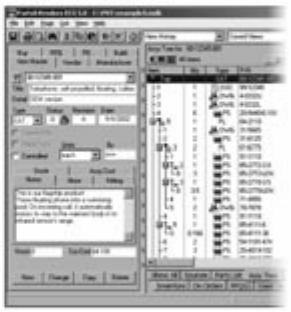
APPLY THE AIRBORNE WLN

Although I've just proven that it can be done, your itty-bitty wireless embedded project probably won't have a PC acting as the serial host for an Airborne WLN. So, let's get real and replace that PC with what you would probably find in a product that needs wireless capability—a microcontroller host.

It seems like everything is accessed these days via a web browser, which is fine if your application requires it. However, sometimes a web browser may be overkill, or there's already a perfectly good hard-coded interface application in place. Regardless of which way you go, you'll need to understand just what the Airborne WLN does when it comes to handling data.

Let's assume your product is a simple microcontroller with an RS-232 serial interface as shown in Figure 1. I don't really care what the product does at this point, but I do know that you want to wirelessly enable it by dropping in an Airborne WLN. So, couple an Airborne WLN to the microcontroller's RS-232 interface. The hardware work is done. Your effort now must turn to getting data into and out of the microcontroller.

Because the RS-232 is the only communications interface your little product has, you can assume that the application program that runs on the microcontroller accepts commands via the RS-232 port and returns data or status to the requestor through the same interface. After 802.11 enabling, the simplest and most direct method of communicating with the microcontroller is via a Telnet session. Here's

Parts List Software
for Engineers and Designers
■ Easily create and manage multi-level parts lists for products in development...and after.
■ Track sources for items with multiple price breaks.
■ Calculate product costs at any quantity.
■ Launch CAD, viewer or browser from any item.
■ Automatically generate RFQs or POs.
New Version 5.0
■ New Report Layout Editor customizes reports/labels.
■ New Connection to QuickBooks 2002/2003 Pro simplifies accounting (US version only).
■ New Multi-currency for foreign suppliers eases exchange rate calculations.

Parts & Vendors™
For Windows 98/NT/Me/2K/XP 3 Editions, starting at $99 + s/h
Visit www.trilogydesign.com and download our FREE DEMO.
Trilogy DESIGN™
Or, Call 800-280-5176
530-273-1985 Fax 530-477-9106
P.O. Box 2270, Grass Valley, CA 95945

TINI® Webserver
Internet Appliance Engine
SOM-400EM

• DS80C400 Processor
• IPv6 Fast Ethernet
• 3 Serial Ports & SPI
• SODIMM Bus Expansion
• Uses a bash shell subset
• 1 MB Flash & 2 MB RAM
• CAN 2.0B & 1-Wire® Ports
• Programmable in Java™ or C
• Telnet, FTP, and HTTP Servers
• Typical Power Consumption of 1 Watt
• Real Time Clock & Nonvolatile Memory
• Robust, Free Java™ Development Tools
• Small, 144 pin SoDIMM form factor (2.66" x 1.5")
• Carrier/Socket Board & Optional Power Supply Available
The SOM-400EM is a System on a Module, based on the Dallas® Tiny InterNet Interface (TINI®) Processor. This 8051 code-compatible processor makes it extremely easy to create a smart Network / Internet capable device. The SOM comes complete with Embedded Java OS and File System. Write sophisticated network applications in days instead of months. Single unit pricing starts at $105. For additional information visit, http://www.emacinc.com/som/.
Since 1985 OVER 19 YEARS OF SINGLE BOARD SOLUTIONS
EMAC, inc.
EQUIPMENT MONITOR AND CONTROL
Phone: (618) 529-4525 • Fax: (618) 457-0110 • Web: www.emacinc.com

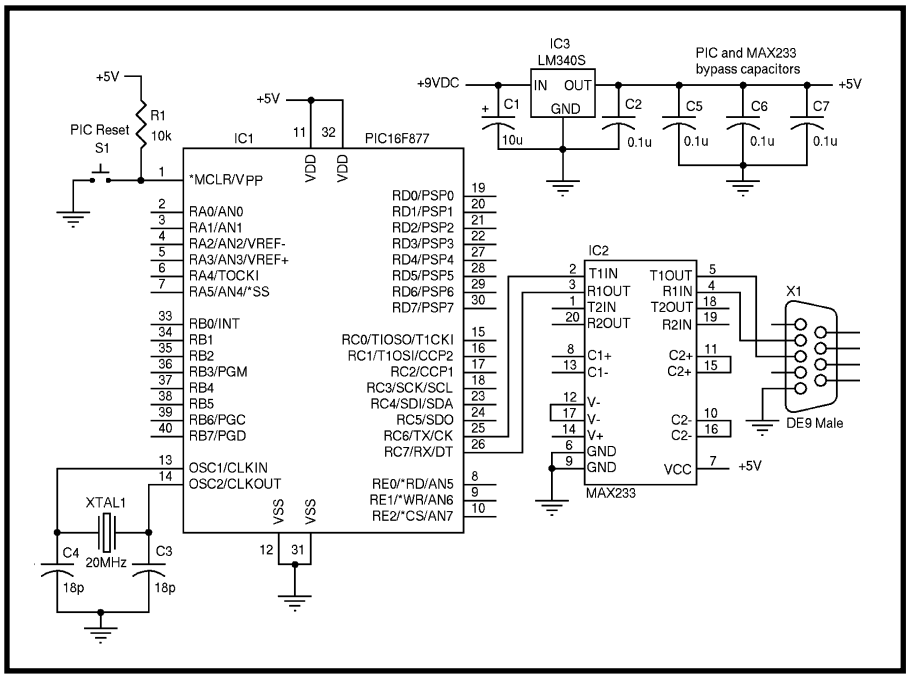


Figure 1—*A microcontroller with a serial port is all that it takes to talk to the Airborne WLN. This is a simplified example because your product's existing microcontroller probably would be rather busy on the I/O side of things. You may have something other than a PIC in there as well.*

the good part. When the Airborne WLN is involved in the communications chain, you don't have to write any microcontroller Telnet code.

The Airborne WLN has the ability to establish a Telnet session with the remote host and tunnel the binary data from the remote host through to the microcontroller via the Airborne WLN's RS-232 port, which is directly connected to the microcontroller's RS-232 port. Assuming the microcontroller controls the Airborne WLN serial port, the microcontroller can selectively allow the remote host to control and query the Airborne WLN resources or use the incoming data from the remote host for its own purposes.

Here's how that works. At power-up, the microcontroller logs on to the Airborne WLN just as one would through a Tera Term Pro terminal session. If the microcontroller application doesn't need to communicate and wants to allow the remote host to log on to the Airborne WLN and issue CLI commands, the microcontroller simply looks for the OK from the Airborne WLN after login and sits. If the microcontroller needs to transfer data to and from the remote host, the microcontroller application put the WLN module in Listen mode. Listen mode allows the remote host to issue a CLI commands to open a data tunnel.

Data Tunneling mode pushes all of the binary data coming into the Airborne WLN from the remote host out through its serial port to the microcontroller unchanged. Also, binary data from the microcontroller serial interface is sent directly to the Airborne WLN serial port and on to the remote host unchanged. In effect, invoking Listen mode causes the microcontroller to relinquish total control of the Airborne WLN serial port, which allows other LAN hosts (remote hosts) to establish connections to the WLN's serial port, while Data Tunneling mode forces the tunnel open exclusively for the remote host that establishes the first connection. The microcontroller can regain total ownership of the serial port by executing an escape command on its end. The remote host can relinquish the exclusive tunnel by entering an escape command from its end.

The downside to using the data tunnel is that the serial channel is totally consumed with the first successful remote host's tunnel activity. If you have more than one remote host that must communicate with the microcontroller, you can simply enter Listen mode at the microcontroller end, and issue individual bidirectional data transfer CLI commands from the remote host. These commands are composed of fields that include the number of bytes to be returned, a timeout value, and a payload area.

One of the WLN data transfer CLI commands expects the payload to be coded as ASCII hex digit pairs (30 for 0, 31 for 1, etc.) and requires a termination character stream. A companion CLI data transfer command simply sends raw hex values. The advantage to using the data transfer CLI commands instead of data tunneling is that the Airborne WLN serial port quickly tunnels the data to and from the microcontroller and returns to Listen mode, which allows multiple remote hosts to access the Airborne WLN and the microcontroller.

CHUCK THE WIRE

Data tunneling and direct access to the Airborne WLN's A/D converter and digital I/O can make life easier for those of you out there who need to quickly adapt an existing product for 802.11 wireless operation. If you're new to Wi-Fi technology, the Airborne WLN evaluation board is a great learning tool. The Airborne documentation is decent, and with the tools included with the evaluation board, you can get your hands on wireless technology in about 15 min.

The Airborne WLN technology is physical proof that it doesn't have to be complicated to be wireless.

Fred Eady has more than 20 years of experience as a systems engineer. He has worked with computers and communication systems large and small, simple and complex. His forte is embedded-systems design and communications. Fred may be reached at fred@edtp.com.

SOURCE

Airborne 802.11b wireless LAN node module evaluation and development kit
DPAC Technologies
(714) 898-0007
www.dpactech.com/wirelesscc_dsp.asp

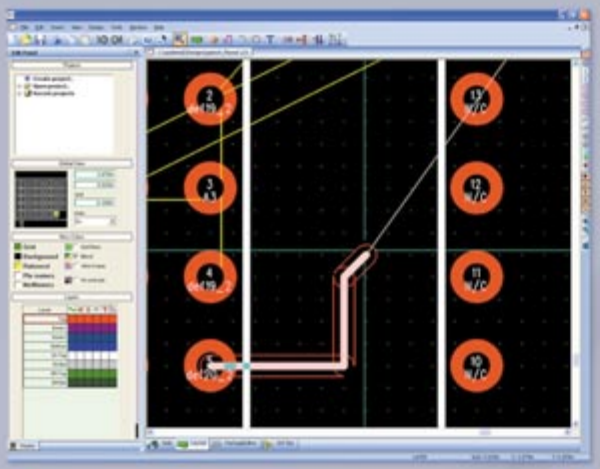
FREE
Download the new version at www.pcb123.com

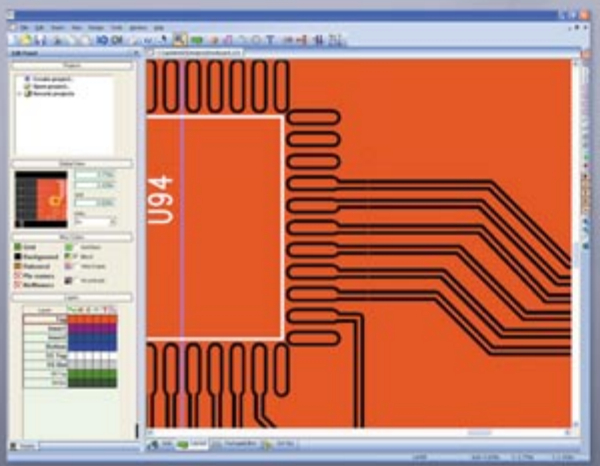
PCB123®

VERSION 2.0

PCB123 Provides the Complete Circuit Board Solution, Design Through Order

PCB123 is a FREE layout and schematic design software tool that quotes your board while you design!





New / Enhanced Features:

- → **Import Net lists from multiple CAD vendors**
- → 145,000+ parts library
- → 2, 4, and 6 layer support
- → Fast, accurate DRC
- → Copper pour with net connectivity
- → Order prototype & production quantities from within the software!

"Thanks for the help!"

Just got the first boards...perfect!!

What an easy process! Zero to finished boards in 2 weeks, including learning curve.

I was dreading ordering a PADs or OrCad package for $5K and then spending weeks learning it. Your approach really is a good option for the R&D engineer.

Bruce Sargeant
Device Lab, Irvine, CA

PCB123
13626 S Freeman Rd. Mulino, OR 97042 USA
Phone: (800) 228-8198 x236 / Fax: (503) 829-6657

Download the new version @ www.pcb123.com



by Ed Nisley

Pennies to Heaven

The Lenz Launcher

"High voltage, high current, and flying objects." Need Ed write more? Nope. He's already got your attention. Read on to learn how Ed built his Lenz Launcher, which can, as he so gently puts it, "punch conductive disks into the ceiling with a loud bang and bright sparks."

Warning: The circuitry described here has enough voltage and stored energy to kill you. Even a trivial mistake will tumble your gyros and paste your hide to the floor. You should treat this column as read-only material if you're not accustomed to working with potentially lethal equipment!

Electrical engineers learn, quite early in their education, how to model the components used in their circuits. Those models, by their very nature, describe only the most important characteristics of each component and suppress the details. Student engineers eventually learn that getting the details right will make or break a project, but you can't start out with all the complexity turned on at once.

The models actually work pretty well for common circuits: passive RLC components are nearly ideal, transistors have few shortcomings, and even interconnecting wires match up with those zero-resistance, zero-inductance lines on the schematic. Troubles arise when students mistake the models for reality and apply them in circuits where the most important characteristics may not be the most obvious ones.

Shortly after the lecture stating that an ideal voltage source provides a constant voltage regardless of the current through its terminals and that an ideal wire conducts any current with zero voltage drop, some wise guy in the back row asks what happens when you connect an ideal conductor across an ideal voltage source. Well, let's find out. We'll also stretch some models beyond their breaking points and take a trip along memory lane for you double Es.

INDUCED CURRENT

Faraday's Law tells us that the emf (voltage, for all practical purposes) induced around any closed path is proportional to the rate of change of the magnetic flux within that path:

$$\text{emf} = -\frac{d\Phi_B}{dt}$$

The negative sign in the equation defines Lenz's Law: the induced voltage causes a current that generates a magnetic field that opposes the original change in the applied field.

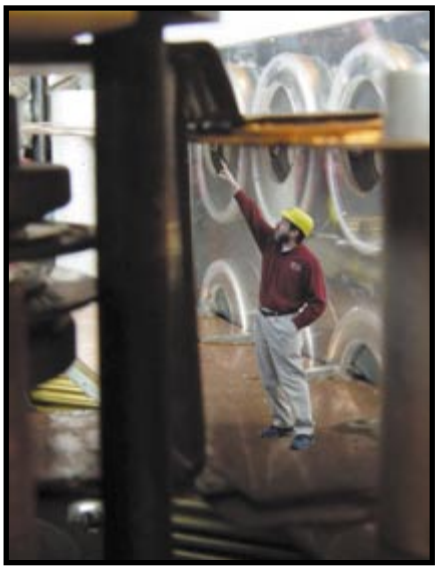


Photo 1—*Two laminated buses conduct energy from the photoflash capacitor bank to the switching transistors. The angled conductor and the vertical strap near the column join the three transistor collector terminals.*

Suppose you generate a magnetic field using a solenoid coil with the north pole pointing up as the current increases. Lenz's Law states that the induced current in a ring above the coil generates a magnetic field with its north pole pointing downward. The faster the solenoid's magnetic field increases, the more current it induces in the ring and the stronger the opposing magnetic field will be.

When you align two magnets with their north poles pointed at each other, they leap apart. In the case of a solenoid and ring, the ring flies into the air. In fact, any conductive object will work, so I cut a disk from 0.020″ copper flashing. It flies just fine!

The projectile must receive enough energy to accelerate it upward against gravity. Because it moves away from the launcher extremely rapidly, the energy transfer is essentially an impulse: energy delivered in a short time equals high power.

The ring or disk acts as the one-turn secondary winding of a rather loosely coupled transformer. By conservation of energy, the output power of a transformer equals the input power minus winding and core losses. Power is the product of voltage and current, so high power requires both high voltage and high current.

High voltage, high current, and fly-



Photo 2—*Three parallel transistors switch the kiloampere output current controlled by base current from the slightly smaller transistor below me. The beefy compression mounts contain water-cooling channels.*

ing objects. Doesn't this sound like a lot of fun?

THE BIG RIG

My friend Eks did some work with high-power pulsed supplies a few years ago, so I figured he was the go-to guy for this operation. When I described what I was thinking of doing, he said, "Come right over, Eddie, I'll show you what you should do!" I call him "Eks" because he's *my* go-to guy, not yours. Just in case you were wondering, he's also the only person who can get away with calling me "Eddie." We go back a long way.

After we caught up with each other's stories and he'd shown me around his basement shop (with a new-to-him, three-phase, full-CNC Bridgeport mill), we entered his power lab.

"You want a mighty big jolt in a hurry, so photoflash capacitors are the ticket. Ordinary electrolytics have too much resistance and inductance," Eks said.

Photo 1 shows the capacitor bank in his pulsed power supply. The conductors are all wide, flat bus bars, which have minimal inductance and resistance.

"You can't use wires, Eddie, because they have too much inductance and the magnetic field kicks 'em all over the place," Eks said. "Get yourself some single-sided circuit board, glue two sheets back-to-back to cancel their inductance, and it'll work great. Cheap, too."

The towering stack of purple transistors in Photo 2 switches the current from the capacitor bank to the output terminals. The three transistors are actually in parallel, with input and output connections through flat bus bar stock compressed between them. This supply dissipates enough power to require chilled-water cooling, but he told me, "If all you're doing is shooting pennies, your average power will be so low you don't care."

Why transistors instead of SCRs?

"Because this way you can turn 'em off. This sucker can punch a manhole cover into low Earth orbit. Sometimes you don't need that much oomph. Got it?"

When I found out how much those transistors cost, I got something else, too. As you'll see, my solution isn't nearly as nice, but it's cheap.

Eks also told me, in no uncertain terms, that a solenoid wouldn't work.

"Wind the coil with copper sheet and make it look like an electric stove coil. You're good with numbers; you can figure out why. Now, go off and build something!"

And with that, he escorted me out the door. In passing, his dog Kerberos attacked my shoe, having mistaken it for a small mammal.

STEP FUNCTION

Photo 3 is a workable Lenz launcher, albeit with a power supply that's not nearly as impressive as the one in Eks's power lab. The schematic in Figure 1 shows the irresistible force of a photoflash capacitor bank connected to the immovable object of a low-inductance coil. All the parts come from my junk box, so you should be able to duplicate it fairly easily from yours, if you dare.



Photo 3—*The Lenz launcher connects the irresistible force of a high-voltage capacitor bank to the immovable object of a low-inductance coil through a homebrew switch. The result can punch conductive disks into the ceiling with a loud bang and bright sparks. What's not to like?*

There are no safety covers or interlocks on this gadget, which isn't the right way to go. You should put a plastic cover over everything except the plastic buttons and the launching coil. You have been warned!

I extracted the photoflash capacitors from surplus camera flash units. Their 330-VDC rating sets the upper limit for the launcher's working voltage.

Some diligent rummaging produced a 24-VAC, 1.25-A wall wart and a doorbell transformer with a 240-VAC primary and 24-VAC secondary. Although the back-to-back output should be about

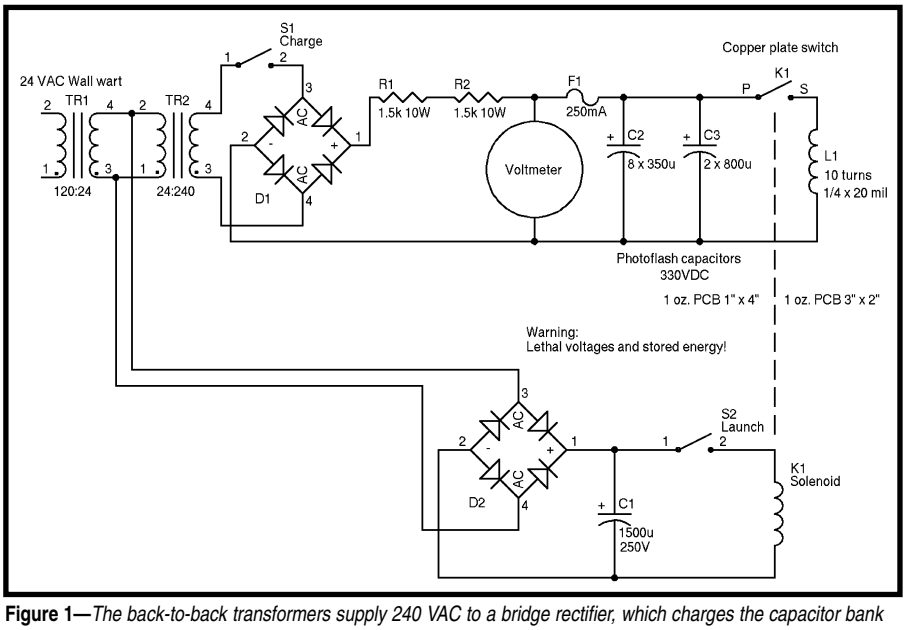


Figure 1—*The back-to-back transformers supply 240 VAC to a bridge rectifier, which charges the capacitor bank through the series resistors and fuse. S1 charges the capacitors, and S2 activates a solenoid that connects the capacitors to the Lenz launcher coil on the far right. The voltmeter must have at least a 300-VDC range. Remember, the fuse protects the circuitry, not you!*



Photo 4—*The Lenz launcher coil has 10 turns of 0.25″ × 0.020″ copper foil insulated with Mylar tape and stabilized with epoxy. I filed the top more or less flat for better coupling to the projectile. An insulating layer of tape will cover the coil's top to prevent shorts.*

330 V_p pulsating DC, my units actually produce 240 V_p under load. The two 1.5-kΩ, 10-W resistors limit the maximum charging (and short-circuit!) current to the doorbell transformer's 80-mA rating.

The energy stored in a capacitor is determined with the following equation:

$$\text{Energy} = \frac{1}{2}CV^2$$

The 10 photoflash capacitors add up to 4400 µF that hold 22 J of energy at 100 VDC. Because energy depends on the square of the voltage, they hold 5 J at 50 V and 90 J at 200 V. Photoflash capacitors have an extremely low equivalent series resistance (ESR), down in the tens of milliohms. Ten in parallel should have a fairly small ESR.

Photo 4 shows the launching coil, which is 0.75″ in diameter . I used 0.020″ copper flashing cut into a strip 0.25″ wide and approximately 1′ long. The terminals are #10 AWG copper wire soldered to a complete wrap of the flashing for a low-resistance joint. A layer of Mylar tape between the turns provides insulation, and epoxy makes the coil a monolithic lump. I removed the tape that insulates the coil from the projectile so you can see the winding.

The inductance of an n-turn pancake coil is, approximately, the following:

$$L(\mu H) = \frac{R_i^2 n^2}{8R_i + 11(R_0 - R_i)}$$

where R_o and R_i are the outer and inner radii of the pancake in inches. The calculated inductance, approximately 0.25 µH, falls well below the level I can measure with any accuracy. A field-solving simulator could calculate a better answer, but that's well above the level of this column's budget!

A coil generates a magnetic field in proportion to the current passing through it, so the rate of change of the field is proportional to the rate of change of the current. The ideal waveform would be a step function: applying an extremely high current in zero time. Ideally, you should apply an infinite current in zero time, which is what that wise guy in the back row was advocating.

The equation for the current through an inductor should look familiar:

$$i_L(t) = \frac{1}{L}\int_{t=0}^{t} V_L dt$$

At $t = 0$, the current is zero regardless of the applied voltage. A constant voltage across an inductor will produce infinite current after an infinite time, although in practice something will burn out first.

Photo 5 shows that the switch applies voltage to the coil in approximately 50 ns, which is well below my digital oscilloscope's single-shot rise time. I tried to measure the current using the differential voltage across one of the coil's solder joints, but the induced current in any closed loop obliterates low-level measurements.

I built the switch using a 12-V car door lock solenoid powered from 24 V for increased speed. The solenoid pulls the smaller top plate just to the right of the capacitors in Photo 3 firmly against the circuit board that holds the capacitors and coil. I soldered copper flashing sheets to the circuit boards for better durability, but the current often welds them together while shooting sparks out the sides. I keep a small screwdriver handy to pry the plates apart.

Eks suggested that four crossed copper bars might work better. I soldered #10 AWG wires to the plates and discovered that concentrated kiloampere current makes for phenomenal sparks and serious spot welding. Another friend suggested mounting the plates so that they rub together while closing, which might tend to keep them from welding by diffusing the current over a larger area. Again, feel free to improve the switch. Just don't expect an ordinary relay to work for long.

SOLDER MATTERS

I used lead-free Sn/Ag (96% tin, 4% silver) solder for all the joints, even though it melts at approximately 475°F (well above eutectic tin-lead solder's 361°F) and is harder to solder, even with a 230-W gun. It has two redeeming features for this application: it has lower resistivity than Sn/Pb solder, and it doesn't release lead when a switching arc vaporizes it.

Copper's resistivity is 1.7 µΩ-cm, tin's is 11 µΩ-cm, and lead's is 22 µΩ-cm, so solder is an order of magnitude worse than copper. Any solder joints must be as thin and wide as possible, but you'll probably still find that the joints determine the overall resistance of the circuit.

For example, 1-oz. copper foil on a PCB is 0.0014″ thick. The 1″ × 4″ rectangle holding the eight 350-µF capacitors is:

$$1.9\ \text{m}\Omega = \frac{1.7 \times 10^{-6}}{2.54} \times \frac{4}{1 \times 0.0014}$$

The total resistance of both copper planes is about 5 mΩ. In contrast, the silver-tin solder joint wrapped around the #10 AWG lead at one end of the launcher coil contributes

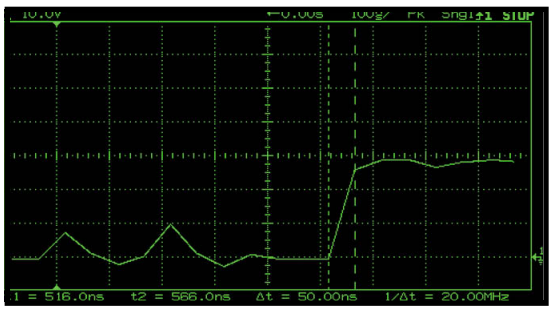


Photo 5—*The voltage across the coil resistance and inductance rises in 50 ns.*

the following:

$$5\ \text{m}\Omega = \frac{11 \times 10^{-6}}{2.54} \times \frac{0.1 \times \pi}{0.25 \times 0.001}$$

I flattened the wire for a better fit to the board, but the resistance at that joint is as follows:

$$1.5\ \text{m}\Omega = \frac{11 \times 10^{-6}}{2.54} \times \frac{0.1}{0.3 \times 0.001}$$

Similarly, the DC resistance of the coil winding works out to 1.6 mΩ, which is much smaller than the other resistances in the circuit. Very roughly, the total circuit resistance might be 20 mΩ for an L/R time constant of 15 µs. That number doesn't match up with anything I could measure, which suggests that it is completely wrong rather than just a little bit off.

Even though the capacitors carried 50 V, Photo 5 shows that only approximately 25 V appears across the coil: the switch and solder joints blot up half the total voltage. Dropping 25 V across 20-some-odd milliohms works out to 1 kA—more or less what Eks expected.

When you're working with kiloampere currents, even trivial items like solder joints and capacitor ESR become critically important. That is the sort of thing you might not learn in EE101.

MECHANICS

My copper disk projectile weighs about 3 g and hits the ceiling with the energy from a 150-V charge. Given these numbers, you can find the initial kinetic energy and the launch velocity. Raising a 3-g disk 2 m against Earth-normal gravity requires following:

$$\left(3\ \text{gf} \times \frac{0.0098\ \text{n}}{\text{gf}}\right) \times 2\ \text{m} = 0.06\ \text{n} \times \text{m}$$

Remember that a gram measures mass, and you need gram-force, which is the force exerted by 1 gm in a 1-G gravity field. The nice thing about metric units is that they actually make sense.

If half the energy goes into wind resistance—an assertion made without a shred of evidence—the launch requires 0.1 newton-meter. Given the mass and the kinetic energy at launch time, the initial velocity is as follows:

$$v = \sqrt{\frac{2 \times KE}{m}}$$

or, in this case,

$$8\ \text{m/s} = \sqrt{\frac{2 \times 0.1\ \text{n} \times \text{m}}{0.003\ \text{kg}}}$$

That number feels about right: the disk takes off at a pretty good clip!

To get an idea of how inefficient this launch technology is, however, recall that 1 newton-meter equals 1 J. The photoflash capacitors store 90 J at 150 V, so about 0.1% of the total energy goes into moving the projectile!

The remainder of the energy goes into vaporized switch contacts and heated conductors. Spot-welding the joints, rather than soldering them, might dramatically improve the launcher's overall performance.

The projectile's conductivity also matters. When Eks and I ran some informal tests, a 1964 dime worked fine, but a present-day dime barely moved, demonstrating the reduced conductivity of post-1965 "silver" coins. Pennies are much less conductive than pure copper, and quarters are hopeless.

Anyway, go forth, calculate, and measure these things for yourself. See what numbers you get, and remember to keep one hand in your pocket at all times!

CONTACT RELEASE

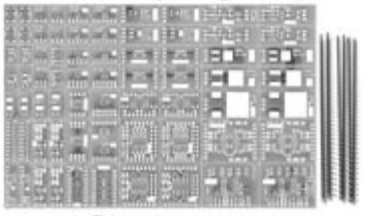
Although this Lenz launcher is a fanciful application, the effect has practical uses. Start with U.S. patent 3,730,039 at www.uspto.gov/patft/index.html and follow the references. Eks has a handful of patents on the subject, too, but they're under his real name.

As far as Photos 1 and 2 go, well, what do you think? The camera never lies, but computers often do!

Ed Nisley is an E.E., P.E., and author in Poughkeepsie, NY. You may contact him at ed.nisley@ieee.org. Put "Circuit Cellar" in the message's subject line to clear the spam filters.

Motor Speed Controls

• Use low-cost DC PM brush motors
• Proportional PWM forward/reverse
• VANTEC *CDFR* unit controls 2 motors
• Command with 8 bit parallel port
• Command 2 Units = 4 motors/port
• 5-30 V, 1-30 A motor, 60 A start surge
• Adjustable current limiting
• Electro-Dynamic brake
• Optional clutch release
• Battlebots & Robotwar wins
• www.vantec.com for complete info
VANTEC (800) 882-6832

SMT Prototyping Adapters

Snap-Apart™ PCB's with .100" pin strips
Dozens of assorted adapters on each PCB
A variety of PCB's - patterns on both sides
Eliminate the cost and delay of PCB layout / fab!
SOIC PLCC SSOP QFP DPAK
TSSOP SOT23 MSOP QSOP
SC90 D2PAK SOT89 SOT143
TSOP SC70 SC88 SC326
and many more.

Bellin Dynamic Systems, Inc.
www.beldynsys.com (714) 630-8024
- Rapid Development Solutions for the Technical Professional -

PIC-SERVO Motion Control
Low-Cost Motion Controllers for Brush, Brushless and Stepper Motors

▪ On-board power amplifiers
▪ RS232 / RS485 Interface - Up to 32 axes per port
▪ Linear and circular interpolation
▪ Software support for Windows, embedded controllers, and other platforms
▪ Excellent on-line customer support
▪ Used in Electronics, Biotechnology, Material Handling, CNC and Robotics applications around the world
Our *PIC-SERVO*, *PIC-STEP*, and *PIC-I/O* chip sets are also available for use in custom designs
J R KERR AUTOMATION ENGINEERING
www.jrkerr.com



by Jeff Bachiochi

USB in Embedded Design (Part 1)

The Undeniable Benefits

How much do you really know about USB? This column is a great place to start educating yourself about the basics. And it couldn't have come at a better time. Serial and parallel ports may soon be things of the past.

They tell you: "Prepare; the end is near. For those who art not prepared shall not enter the kingdom of the PC."

Where would your project be without access to today's GUI power base? Like it or not, PCs rule. For many a designer, they are *numero uno*, an important piece of the puzzle, and I'm not talking about a PC as a development platform.

OK, so maybe your device doesn't need that free user GUI connection that's already in every home. However, I'd find it hard to believe that you've never designed a gizmo that didn't use a serial or parallel port. Most of you have.

And now they're telling you, "You won't find those on new PC designs."

You've already seen the appearance of thin USB ports on the back, front, and side of towers and laptops. Soon the legacy serial and parallel ports will vanish completely from your computers. Whether you believe this prediction or not, it would be wise to prepare for the possibility. But don't let the promised extinction scare you, there are other reasons for you to embrace USB with open ports—eh, arms.

WISH LIST

From a user's point of view, I want to be able to connect all of my peripherals via the same cable, which means using the same connector on all products. I want to be able to plug stuff in at any time (hot pluggable) and then have the computer recognize it and automatically load whatever driver is necessary without restarting the machine. I would also like to eliminate the tangle of power supply dongles needed to externally power every peripheral.

As a designer, I don't want to have to make choices about the physical interface, serial or parallel, DB25 or DE9, and male or female. I'd be happier not having to support different data formats. Having to deal with extra level-shifting circuitry only adds to the cost.

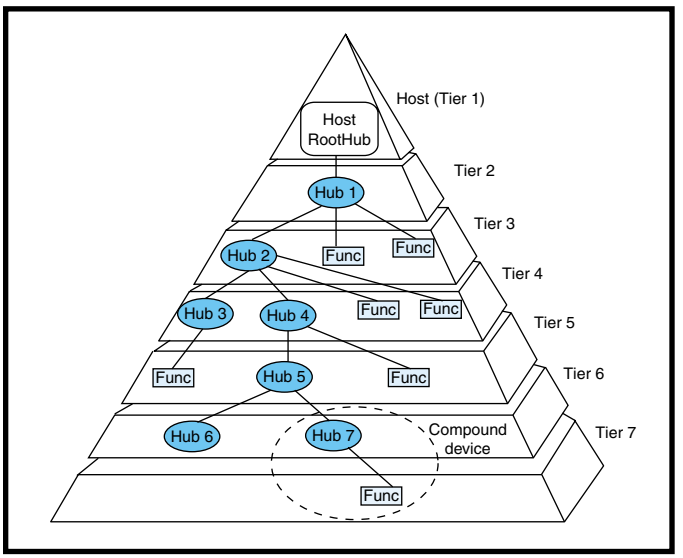


Figure 1—*The USB connection topography is a one-to-one connection between the host and a device. The hub is a specialized device acting as a repeater to enable multiple one-to-one connections through a tiered system.*

USB IMPLEMENTERS FORUM

To truly come up with a standard that is embraced by everyone requires that many different parties have input in the specification process. Presently, seven corporations are assigned the copyright to the USB standard: Compaq, Hewlett-Packard, Intel, Lucent, Microsoft, NEC, and Philips. A visit to www.usb.org will show you the level of commitment to the USB push.

Windows 95 (SR2) supported the 1996 release of USB 1.0, although it was not until Windows 98 (SE) that the hardware and software issues were under control. A faster USB 2.0 was introduced in 2000, thanks partly to a head-to-head competition with Firewire. The first USB on-the-go specifications (device as a host) were released a year later.

USB

The USB allows data exchange between a host and peripherals (devices). All USB devices share the bandwidth through a host-scheduled, token-based protocol, which allows each device to be attached, configured, used, and detached while the system remains in operation. This requires three basic ingredients: a host, a device, and an interconnect.

The host, or controller, must detect the attachment and removal of devices (enumeration and configuration),

manage the flow of control and data packets (isochronous and asynchronous), collect status (device and bus), and provide power management to the devices. A device provides special capabilities to the system and may contain multiple pipes (communications channels) through which the host may communicate with the device.

Each device must contain a control pipe connected to endpoint zero, a common access mechanism for accessing information about the device (like a table of contents). Endpoint zero contains standard information like vendor ID, device class, and power management capabilities, as well as device configuration, interface, and endpoint descriptors. Endpoint zero also contains class and vendor-specific information. Although the USB specification makes a distinction between a device and a hub, you can think of the hub as just another specialized device.

Let's look at the way USB connections are made to understand why the hub is considered so important. Figure 1 shows a seven-tier network of USB connections made possible by the hub. Between tier 1 (a host) and tier 2, you have room for one USB device. After a connection is made between the host and a device, there is no room for other connections. This allows only a single device to be connected to a host. Not too useful, eh?

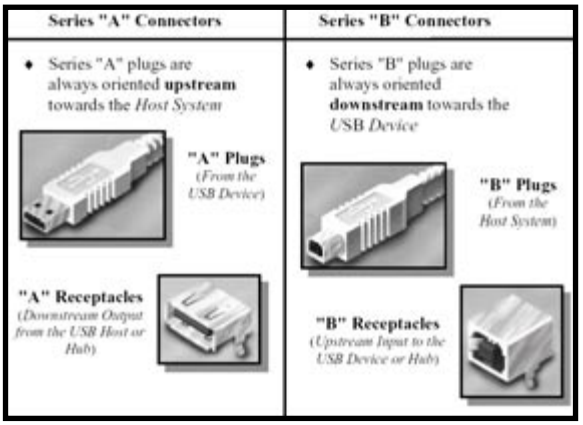


Photo 1—*Every USB cable is wired the same. Never again will you need to search for a cable with the right connectors.*

Like all devices, a hub has one connection upstream toward the host. Yet a hub offers many connections downstream to other devices. (Most PCs have a built-in hub, offering you more than one USB connection.) Every device connected to a hub (or the host) operates from a lower tier. The USB specs allow devices to operate down to tier 7 (hubs to tier 6).

The USB hub has three jobs: the hub controller, the hub repeater, and the transaction translator (see Figure 2). The hub controller controls the upstream (hub to host) communication path and can configure and monitor its downstream (hub to device) ports. The hub repeater's job is to control communications between the upstream port and the downstream ports, including hardware support for reset and suspend/resume signaling. The transaction translator handles communication speed differences between the upstream and downstream paths. (This allows each path to take full advantage of its highest operating speed.)

Each device (including hubs) is given its own address during enumeration. The host uses this address as a way of talking to one device at a time. A host broadcasts packets, which carry a device address to the entire bus. The packet burrows through every pyramid tier thanks to the hub's ability to retransmit the packet to every downstream device (including translation between bus segments of different speeds). Eventually, all devices get to hear the packet. However, only the addressed device will respond. This means one return packet (at most) is generated and works its way back up through any tiers to the host.

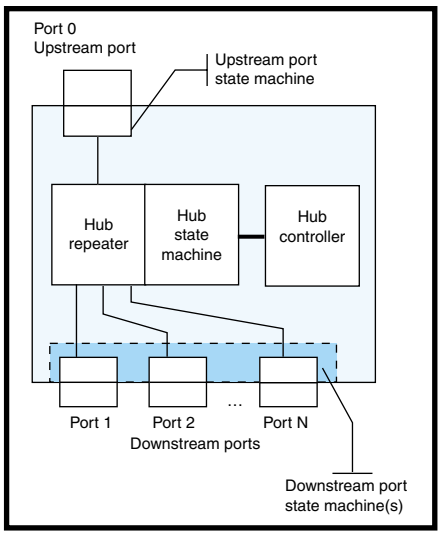


Figure 2—*The hub is a special device acting as a repeater, controller, and transaction translator. It establishes communication with downstream devices at the highest possible speed. It can receive packets at one speed and send them at a different speed to maximize bandwidth.*

INTERFACE

With the introduction of USB 2.0, the bus has three possible communication speeds: low speed (1.5 Mbps), full speed (12 Mbps), and high speed (480 Mbps). Although the specifications may differ on the transceivers and cable used for the implementation of each, this is all invisible to you. You may not even know which one each device is using. Hint: Although a device may be capable of high-speed communication, unless all of the upstream hubs support high-speed communication, the interface will throttle back to the highest supported speed. So it makes sense to know the capabilities of each device and hub and to connect them in a way that maximizes bandwidth.

The physical interface is a four-wire connection for all. Separate upstream (host) and downstream (peripheral) connectors are used. These keyed connectors ensure the cabling is correct and the signal connections are made in sequence (power first). Photo 1 shows both plugs and receptacles for upstream and downstream use. The signal connections are shown in Table 1. Data is sent in serial fashion using differential pair drivers and receivers (D+ and D–). The cable specifications require that the data pair be twisted to obtain the full benefit of using differential signals to cancel common mode interference.

Contact number	Signal name	Typical wiring assignment
1	VBUS	red
2	D–	White
3	D+	Green
4	GND	Black
Shell	Shield	Drain wire

Table 1—*Connecting USB devices requires a cable having different upstream and downstream connectors. The USB system is designed to be hot-pluggable.*

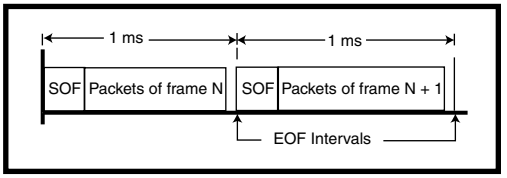


Figure 3—*The host creates 1-ms transfer frames, which can carry packets to any USB device on the bus. Packets contain ID information that allows the packet to be directed down a pipe (logical connection) to a single device's endpoint. Data packets can be formatted or non-formatted depending on the type of transfer used.*

But how is the maximum speed for each path determined if all of the connectors and connections are the same? A device on a lower tier is responsible for declaring its maximum speed to the host (or hub) it is connected to. In USB 1.0, a pull-up resistor tied to D– raises this signal slightly off of an idle state (with reference to D+), indicating a low-speed connection, while a pull-up tied to the D+ indicates a device capable of a full-speed connection.

High speed was added to USB 2.0. To stay compatible with the previous USB 1.0 specification, a device must first be declared full-speed and only then will an upstream high-speed transceiver begin a high-speed test sequence. The downstream device must interpret the test and respond indicating to the upstream port that it is a high-speed device. The full-speed pull-up is then disabled (from D+) on the device and special high-speed transceivers are enabled on the hub. Any further communication within this segment of the bus takes place at high speed.

DATA

To reduce the number of signals needed for USB, a clock is encoded along with the data into an NRZI scheme using bit stuffing, which is the automatic addition of one 0 bit after six consecutive 1 bits. This is required to force a change-of-state in the NRZI encoded data allowing a receiver to remain in sync with the data. The receiver automatically discards this bit.

USB communication can be divided into two categories—configuration and data exchange. The former is covered in the next section.

Data exchange is handled using 1-ms frames (see Figure 3). (Note that high-speed frames are further divided into 8- to 125-µs microframes.) During each frame, the host may send transaction packets to any or all devices on the bus. Each frame may be filled with packets up to the 1-ms frame time. A host transaction may be one of four types: control, bulk, interrupt, and isochronous. Control transfers are normally used for configuration at the time of attachment, but can be used for other device-specific purposes. Bulk transfers are used for bursty traffic that isn't time critical, as in a file transfer. Interrupt transfers are used for passing reliable data in real time. Isochronous data transfers stream time scheduled packets and can be bandwidth hogs. The use of timed frames allows the host to guarantee time on the bus for certain transactions.

Depending on the type, each transaction consists of two or three phases: token, data, and, for all but isochronous types, handshaking. All packets begin with a packet-ID (PID) identifying the type and phase, and may contain other information such as the USB device address and endpoint number. The token phase packet includes information on how to treat the following data as setup, data in, data out, or status. The data phase transfers data to or from the device based on information from the token phase. A handshaking phase may indicate the status of a transaction using an ACK, NAK, STALL, or NYET code.

A transaction connection between the host and device is called a pipe. Each pipe reflects the characteristics

Offset	Field	Size	Value	Description
0	bLength	1	Number	Size of this descriptor in bytes
1	bDescriptorType	1	Constant	Descriptor type
2	bcdUSB	2	BCD	USB specification. Release number in binary-coded decimal (i.e., 2.10 is 210H).This field identifies the release of the USB specification with which the device and its descriptors are compliant.
4	BdeviceClass	1	Class	Class code (assigned by the USB-IF). If this field is reset to zero, each interface within a configuration specifies its own class information, and the various interfaces operate independently. If this field is set to a value between one and FEH, the device supports different class specifications on different interfaces, and the interfaces may not operate independently. This value identifies the class definition used for the aggregate interfaces. If this field is set to FFH, the device class is vendor-specific.
5	BdeviceSubClass	1	SubClass	Subclass code (assigned by the USB-IF). These codes are qualified by the value of the bDeviceClass field. If the bDeviceClass field is reset to zero, this field also must be reset to zero. If the bDeviceClass field is not set to FFH, all values are reserved for assignment by the USB-IF.
6	bDeviceProtocol	1	Protocol	Protocol code (assigned by the USB-IF). These codes are qualified by the value of the bDeviceClass and the bDeviceSubClass fields. If a device supports class-specific protocols on a device basis as opposed to an interface basis, this code identifies the protocols that the device uses as defined by the specification of the device class. If this field is reset to zero, the device does not use class-specific protocols on a device basis. However, it may use class-specific protocols on an interface basis. If this field is set to FFH, the device uses a vendor-specific protocol on a device basis.
7	bMaxPacketSize0	1	Number	Maximum packet size for endpoint zero (only 8, 16, 32, or 64 are valid)
8	idVendor	2	ID	Vendor ID (assigned by the USB-IF)
10	idProduct	2	ID	Product ID (assigned by the manufacturer)
12	bcdDevice	2	BCD	Device release number in binary-coded decimal
14	iManufacturer	1	Index	Index of string descriptor describing manufacturer
15	iProduct	1	Index	Index of string descriptor describing product
16	iSerialNumber	1	Index	Index of string descriptor describing the device's serial number
17	bNumConfigurations	1	Number	Number of possible configurations

Table 2—*This is the format of the device descriptor table. As the first piece of information that a host requests, it contains a basic description of the device and how the host can best communicate with it.*

JAMECO® ELECTRONICS

1355 Shoreway Road, Belmont, CA 94002-4100

Call for your FREE catalog

1.800.831.4242



Open Frame Power Supplies

Jameco offers a wide selection of open-frame power supplies suitable for production, prototype and design environments. Be sure to check out our selection of Artesyn, Mean Well and dozens of other brands with universal input, PFC and other features you need on our web site today!



184866MC

ARTESYN

Manufacturer	Mfr. Cross Reference No.	Output Rating (Voltage@Current)	Jameco Part No.	Price
Skynet	SNP-0502-B	+5V@0-2A	184866MC	$12.95
Artesyn	NLP25-7608	+5V@0.2-2A; +12V@0.1-0.8A; -12V@0-0.1A	218501MC	$37.95
Mean Well	PS-65-24	+24V@0-2.7A	148646MC	29.95
Artesyn	NLP150L-96Q5366	+3.3V@0.5+10A; +5.1V@1.5-2.0A; +12V@0-2A;+12V@0-0.65A	219029MC	116.85
Digital Power	USS250-115	+15V@0-16.5A	204564MC	39.95

Closed Frame Power Supplies

MW MEAN WELL

Jameco offers hundreds of closed frame supplies from the names you trust, all priced to help you save! Call for a FREE catalog and learn more about our selection featuring low-leakage medical supplies, power factor correction, universal input and much more.



123246MC

Mfr.	Mfr. Cross Reference No.	Output Rating (Volts@Current)	Input (VAC@Hz)	Jameco Part No.	Price
Mean Well	S-25-5	+5V@0-5A	85-264@47-63	123246MC	$39.95
Mean Well	S-60-24	+24V@0-2.5A	85-264@47-63	123351MC	53.95
Mean Well	S-150-24	+24V@0-6.5A	88-132/176-264@47-63	123449MC	69.95
Mean Well	SP-300-24	+24V@0-12.5A	88-264@47-63	137402MC	181.95

Wall Adapters

We offer a complete selection of AC-to-AC and AC-to-DC wall adapters for general purpose and production applications. Most are UL approved and available with various power plugs. Be sure to visit our web site and learn more.



Mfr. Cross Reference No.	Output Rating (Volts@Current)	Size (+0%-20%) H" x W" x D"	Agency Approvals	Jameco Part No.	Price
DC1205F12	+12@500mA	2.5 x 1.9 x 1.6	UL/CSA	102496MC	$4.95
DCR1205F12	+12@500mA	3.2 x 2.2 x 1.9	—	162996MC	9.95
DC1205F5	+12@500mA	2.5 x 2.1 x 1.7	UL/CSA	102277MC	5.19

Table-Top Transformers

Outfit your product with a professional table-top transformer. Jameco offers a wide variety of configurations with regulated and unregulated outputs. Don t forget to ask for a FREE catalog and learn about our selection of domestic and international input power cords.



155213MC

Mfr. Cross Reference No.	Power (W)	Output Rating (Volts@Current)	Input (VAC@Hz)	Size (+0%-20%) H" x W" x D"	Jameco Part No.	Price
SPU50-6	60	+24@2.5A	100-240@47-63	5.6 x 2.9 x 1.6	161605MC	$44.95
P40A-3P2JU	40	+12@3.3A	100-240@50-60	5.5 x 2.3 x 1.5	155213MC	35.95
SPU50-3	60	+12@5.0A	100-240@47-63	5.6 x 2.9 x 1.5	155230MC	49.95
KWM12F-P2MU	18	+12@1.5A	100-240@47-63	4.0 x 1.9 x 1.5	216531MC	26.95
P40A6P2J	40	+24@1.66A	100-240@50-60	4.1 x 2.6 x 1.4	181884MC	35.95

Low-Noise Fans

Jameco offers hundreds of low-noise fans at outstanding prices. These agency-approved low-profile fans feature ball or sleeve bearings so you can be assured your system will be running cool for years to come. Get more details by asking for a FREE catalog today.



102955MC



We also stock CPU fans. Visit our web site for a complete listing.

Mfr.	Mfr. Cross Reference No.	Voltage (VDC)	Air Flow (CFM)	Noise (dBA)	Size H" x W" x D"	Jameco Part No.	Price
Sunon	KD1208PTB1-6	12	42.5	33.5	3.15 x 3.15 x 1.00	102955MC	$6.49
Sunon	KD1204PFB2-8	12	6.3	30	1.60 x 1.60 x 0.40	161699MC	7.95
Sunon	KD1212PMB1-6A	12	108	42	4.68 x 4.68 x 1.50	94625MC	11.95
Sunon	KD0504PFB2-8	5	5.5	23	1.60 x 1.60 x 0.40	196373MC	8.95

DC-to-DC Converters

Jameco now offers a wide selection of Isolated and Point-of-Load DC-to-DC converters suitable for distributed power architectures. Avoid the hassles of "large company" distribution channels and get immediate delivery from Jameco today!



ARTESYN

219150MC

Mfr.	Mfr. Cross Reference No.	Power (W)	Output Rating (Volts@Current)	Input Range	Jameco Part No.	Price
Mean Well	SCW03A-05	3.0	+5V@600mA	9-18V	213268MC	$15.49
Mean Well	SD-25A-24	26.4	+24@1100mA	9-18V	175812MC	40.95
Mean Well	SKE15A-05	15.0	+5V@3000mA	9-18V	155715MC	34.95
Artesyn	SIL06C-05SADJ-V	20.0	+0.9-3.3V@6	4.5-5.5V	219150MC	10.25
Astec	AA9090A	21.0	+5.1V@3750mA; +12.6V@100mA; -26V@40mA	0-20V	109276MC	2.79

Place your order today and mention VIP CC6

1.800.831.4242 or www.Jameco.com/CC6

Value-Priced Products In Stock For Same Day Shipping

Dual 10/100 Ethernet

TS-5400 $269



- PC/104
- Two 10/100 Ethernet
- 24 Digital I/O
- Compact Flash
- DiskOnChip
- 2 Serial Ports

- AMD Elan 133 MHz 586 with 32 MB RAM (64 MB optional)
- PC compatible: BIOS, DOS, RTC, I/O & memory map
- TS-Linux 3.0 is open source GNU/Linux - Glibc 2.2.5
- Low power +5v @ 800mA
- 4.1" x 5.4"

Shown with optional flash module, Compact Flash and RS-485.

PCMCIA

- 802.11b Network
- Cellular Modem

TS-5600 $279



- PC/104
- PCMCIA slot
- 10/100 Ethernet
- 2 Serial Ports
- Compact Flash
- 24 Digital I/O
- 12-bit 6 chan. A/D
- 12-bit 2 chan. DAC

Shown with optional flash modules, A/D. DAC, RS-485 and Orinoco 802.11b WiFi

- AMD Elan 133 MHz 586 with 32 MB RAM (64 MB optional)
- PC compatible: BIOS, DOS, RTC, I/O & memory map
- TS-Linux 3.0 is open source GNU/Linux - Glibc 2.2.5
- Low power +5v @ 830mA
- 4.3" x 5.6"

Linux 2.4

www.embeddedx86.com

(480)-837-5200
16610 E. Laser Drive #10
Fountain Hills, AZ, 85268 USA

Technologic SYSTEMS

Low Cost Ethernet

TS-3300

$199



- PC/104
- Compact Flash
- DiskOnChip
- 10 base T Ethernet
- 38 Digital I/O
- 2 Serial Ports
- 12-bit 8 chan.A/D

- 33 MHz Intel 386EX with 8 MB RAM (16,32 MB optional)
- DOS-ROM or TS-Linux pre-installed
- PC compatible: BIOS, DOS, RTC, I/O & memory map
- 1 MB flash drive
- Power: +5v @ 700mA
- Dimensions 4.0" x 5.4"

Shown with optional flash modules, A/D and RS-485

Low Cost PC

TS-3400

$159



- PC/104
- Low EMI
- DiskOnChip
- 37 Digital I/O
- 2 Serial Ports

- 33 MHz Intel 386EX with 8 MB RAM (16,32 MB optional)
- DOS-ROM or TS-Linux pre-installed
- PC compatible: BIOS, DOS, RTC, I/O & memory map
- 1 MB flash drive
- Power: +5v @ 500
- Dimensions 4.0" x 3.8"

Shown with optional flash modules

TS-9700 analog board

TS-9700 $59

- PC/104 peripheral board
- 8 channel 12-bit ADC
- Optional 4 channel 12-bit DAC
- Precision 0.2% reference
- 0-2.5v, 0-10v and 0-20mA input ranges
- 0-5v output range
- Optional temperature sensor



Technologic SYSTEMS

- PC/104 Computer Boards
- Enclosures
- PC/104 Peripherals
- Custom Designs
- Industrial Temp



PC/104 Peripherals

Enclosures $85

Metal enclosure for Single Board Computers provides regulated +5 VDC from 12-38 VDC input power. Connections available at back panel. Suitable for office or factory.

TS-SER4 $89

Four PC compatible COM ports using the 16550 UART, full handshaking on two ports, RS-485 option on two ports, jumper configurable for COM1-24 and IRQ's 3-15.

TS-9600 $99

IDE interface supports standard 3.5" IDE hard drives, mounting holes for directly mounting a 2.5" laptop hard drive. Standard 40 pin IDE and laptop 2mm headers.

TS-Modem $129

33.6K baud, jumper configurable as COM1-6, supports IRQs 3-15, V.32 V.42 compliant, standard AT commands, caller ID, 56K baud optional.

TS-Linux v3.0

- Introducing TS-Linux version 3.0 Our latest distribution is built from open-source featuring a 2.4.18 kernel, Apache web server, FTP client/server, bash, etc. With a footprint of 10 MB or less this is a great place to start your application.
- The TS development environment compiles application code using the libraries and headers installed in TS-Linux. Applications developed on your desktop computer copy directly.
- Our User Mode Linux patched kernel runs TS-Linux 3.0 kernel and file system as an application on your desktop computer during development.

of its associated device endpoint (i.e., bulk in using endpoint one). The default endpoint zero must always exist on a powered device to provide initial device configuration, status, and control information during the enumeration process. This discovery process allows the host to establish other pipes to the device (when necessary) in support of its function.

ENUMERATION

Most of the magic in USB happens during enumeration. Enumeration is the process by which the host assesses device communication speed, assigns the device an address, and inquires what a device is and what it requires. Because the USB bus is made up of small one-to-one connections, the host can pick and choose with whom it is going to communicate.

When a new device is plugged in, the host (or hub) senses its connection by monitoring the D+ and D– lines. If plugged into the host, it will be detected immediately. If plugged into a hub, the hub will tell the host of a detection the next time the host asks for the hub's status. The host (or hub) can now determine the speed of the new device and make contact using the default address of zero. The first piece of information requested from the device is the device descriptor table. The device responds with a list similar to that of Table 2. (Note that most tables are similar. The bLength field indicates how large the table is. The bDescriptorType indicates the format of the rest of the table.)

The host retrieves information on how large a packet it can handle and assigns the new device its own unique address (leaving the default address zero free for the next newly attached device). The host can now communicate with the device through its new address, using packets of an acceptable size to retrieve the device descriptor and learn about the device's abilities. Based on the device descriptor, the host can request information on each configuration (mode of use) for the device (most devices have a single configuration). By requesting the first 9 bytes of the configuration descriptor, a host can determine how many interfaces (sets of endpoints) the device requires. With this new information, the host rereads the configuration descriptor this time in full, which includes the interface and endpoint descriptors.

Based on the host's acquired knowledge of the device, Windows looks for and installs a driver to match. The driver asks the host to set the device to one of its configurations. The device enters its configured state with its interfaces enabled, signifying that the enumeration has succeeded.

As you can imagine, it only takes a small error in any of the descriptor tables to foil the enumeration process. Windows plays a device connect or device disconnect sound when a USB device is connected or disconnected (failed enumeration plays as a disconnect) so you can tell what is going on. Other than that, don't expect much diagnostic information about the connection or lack of it.

When a device is disconnected from a hub (or host), the hub's status indicates this. The host removes the device and any downstream devices (if this is a hub with connected devices) from its polling pool.

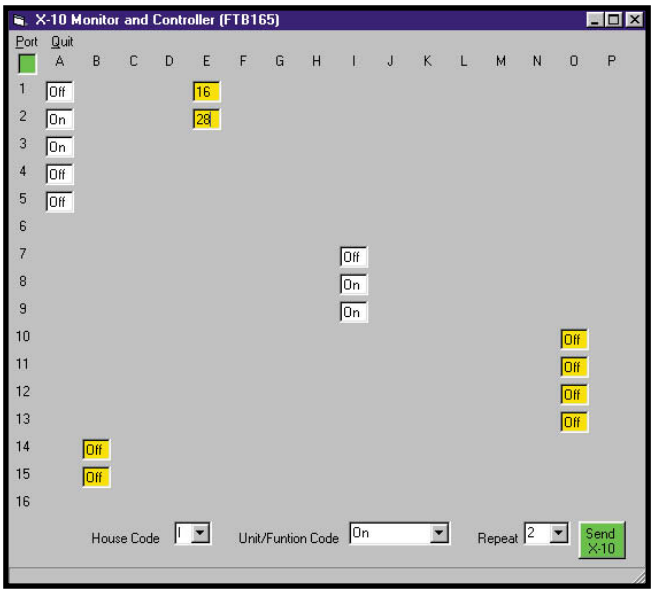


Photo 2—*The X10 monitor screen can monitor every X10 address, but it shows only those modules that have received X10 commands. Visible modules can be designated as lamp or appliance modules and therefore properly reflect on, off, and dimness levels. Drop-down boxes at the bottom allow X10 commands to be set and then sent using the "SEND X-10" button.*

COMPLEXITY VS. EASE OF USE

The PC industry has certainly concerned itself with the user. The USB effort will significantly reduce peripheral user support. Rigorous testing before obtaining USB certification ensures the compliance of any device that wants to display the official compatibility logo.

Supporting USB doesn't make the designer's job any easier. In fact, it creates another specialty area, where the average designer must put in long hours to make sense of all the bits necessary to implement a USB interface.

NOW WHAT?

Got a project that you would like to convert to USB? This seems like a good place to start trying to understand just how USB works. I've got a serial dongle that I designed a few years ago to take serial commands and convert them to X10 signals using the TW-523 as a power line interface (see Figure 4). The TW-523 is an X10 product that gives you isolated access to the utility grid. The TW-523 has two isolated control lines out (zero crossing and 120-kHz gate) and one isolated control line in (120-kHz gate).

To send X10 data, you must use the zero-crossing input as a reference point to create three potential 1-ms gates (one at each 60° point within 60 Hz of each half cycle starting with the zero crossing). X10 data consists of three gate signals (for logic 1) or the absence of three gate signals (for a logic 0) during a 60-Hz half cycle. Except for data in the start code, the second half cycle must contain inverted data of the first half cycle. The X10 commands are sent as start code, house code, and function code. (Note that the purpose of the three gates is to enable 120-kHz bursts within the TW-523 that are applied to the power line at the zero crossing

points for three-phase power, where the other phases are 60° and 120° out of phase.)

To receive X10 data, look for envelopes at zero crossings and decode the data into X10 commands. The serial dongle accepts X10 commands as 3 bytes from a PC application (house code + 128, function code, and repeat code) and translates the commands into X10 power line gating signals. If an X10 command is heard on the power line, 2 bytes are sent to the PC (house code + 128 and function code).

Commands sent to the TW-523 are also sent back to the PC as an operational check. At the time I was investigating the expanded power of microEngineering Labs's PicBasic PRO, which is a basic compiler for Microchip microprocessors.

New X10 commands had been added to the PicBasic's command set, and I wanted to experiment with them. The total basic program text is approximately two pages. Execution uses the `XIN` command to look at half of a power line cycle for a start code. If one is found, it decodes the data into a house code and function code. The program sends it out of the serial port. If a start code is not found, the program waits for approximately 5 ms (less than half a line cycle) before looking for a start code again.

The wait loop allows an interrupt to be serviced if necessary (also written in BASIC). A character received via the hardware serial port produces an interrupt. The interrupt routine grabs three characters (house code, function code, and repeat code) and checks to see if they are legal (house code must have most significant bit set), sets a data good flag, and exits the interrupt routine. Back in the main loop, the data good flag enables an `XOUT` command to be sent. All code for this module is written in BASIC and compiled into assembly code by the compiler.

The application on the PC in Photo 2 is written in Visual Basic; it monitors the serial port for received X10 commands. This application also allows you to send an X10 command to turn on and off an appliance module or control (e.g., on, dim, bright, off) a lamp module. The application monitor screen shows the status of all 255 potential modules (A–Z, 1–16). Individual modules are shown only after an X10 command has been sent or received to or from a module. You can indicate the module's use as a lamp or appliance. This allows the dim and bright levels to be displayed for lamp modules. User-modified descriptions of each module pop up when the mouse hovers over the said indicator.

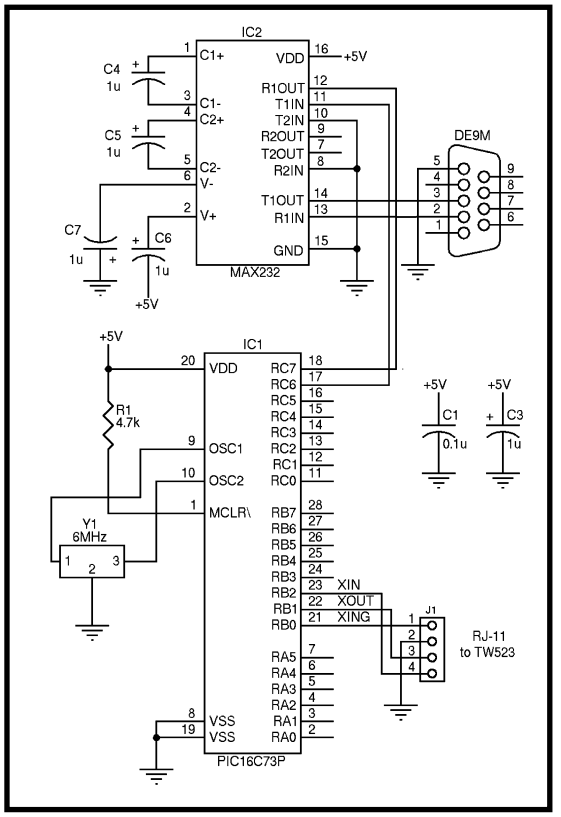


Figure 4—*This simple schematic allows a microprocessor to communicate serially with a PC application and to send and receive X10 commands using a TW-523 isolated power line interface.*

HID

In most situations you must write a PC device driver for a particular class USB device. I don't know about you, but that's much more than I wish to bite off at one time. Fortunately, there is the human interface device (HID) class. This was one of the first USB classes to exist, and it has an existing Windows driver. Although the HID class was designed as a human interface, it can be used for anything as long as the data fits with the class specifications.

Data is exchanged by the host-requesting and device-returning reports via control or interrupt transfers. A report can have multiple transactions of 8, 64, and 1024 bytes for low-, full-, and high-speed devices. Maximum transfers are therefore 800, 64 KB, and 24 MB per second.

Next month we will take a look at using the HID class for this project. Although the present interface is serial, we are not doing any bulk file transferring, so throughput is not an issue. Even when using a low-speed USB device, the specs for this project easily fit within the HID class specifications.

Jeff Bachiochi (pronounced BAH-key-AH-key) has been writing for Circuit Cellar *since 1988. His background includes product design and manufacturing. He may be reached at jeff.bachiochi@circuitcellar.com.*

PROJECT FILES

To download the code, go to ftp.circuitcellar.com/pub/Circuit_Cellar/2004/165.

RESOURCES

J. Axelson, *USB Complete: Everything You Need to Develop Custom USB Peripherals*, 2nd ed., Lakeview Research, Madison, WI, 2001.

USB Implementers Forum, www.usb.org.

SOURCES

PIC16F873A Microcontroller
Microchip Technology, Inc.
(480) 792-7200
www.microchip.com

PicBasic Pro compiler
MicroEngineering Labs
(719) 520-5323
www.melabs.com

HIDmaker
Trace Systems, Inc.
(301) 262-0300
www.tracesystemsinc.com

TW-523 X10 Two-way interface
www.X10.com



Powered Points

If you're thinking about designing a device that plugs into Ethernet, Tom suggests that you learn as much as possible about 802.3af. This column is your power-over-Ethernet (PoE) primer. Read on to see why Tom is so optimistic about PoE technology and the chips that will drive it.

Wireless is all the rage, and am I glad. With phones, 802.11, toys, keyboards, mice, and all manner of embedded gadgets cutting the cord, the historic inexorable growth of the rat's nest is finally being held at bay and, dare I hope, reversed. Lost in the hoopla is the fact that fewer wires can be as helpful as wireless. After all, until such time as there are no wires, replacing two of them with one is a worthy concept too.

That brings us to this month's topic, Power over Ethernet, or PoE for short. If there's one kind of wire that's been growing over the past decade or so in lockstep with the rise of PCs and the Internet, it's the ubiquitous Cat 5 cable, connecting 10, 100, and now 1000 Mbps Ethernets.

The bad news is that nearly every Ethernet cable brings along an unwelcome second cousin, namely the power cable for the Ethernet-connected gadget, as often as not connected to an outlet-hogging wall wart power supply. "Say hello to my little friend," indeed.

SPARE CHANGE

Actually, the PoE story isn't so new. The plotline hinges on the fact that Ethernet only uses four of the eight connections in the ubiquitous RJ-45. It's no surprise that creative homebrewers were the first to seize the opportunity to send power along with the data (see Photo 1). Just the thing for sticking a webcam on the roof or positioning a Wi-Fi access point for best coverage without having to worry about whether or not it's near an AC outlet. There are helpful step-by-step guides posted on the 'Net by these PoE pioneers (e.g., www.nycwireless.net/poe/), and there's even a web page dedicated to the subject at www.poweroverethernet.com.

Basically, homebrewing your own PoE hack boils down to a few simple tasks. The first is to verify that the spare wires are actually unused—something that's usually the case, but it's better to be safe than sorry. This caveat is especially relevant in a commercial installation, which might feature structured wiring that uses the spare wires.

Next, calculate the voltage required depending on the length of the cable and how finicky the power consumer at the receiving end is. It's a simple matter of Ohm's law and the resistance of the wire. The Ethernet spec calls for a maximum of 9.8 Ω per 100 m, or approximately 3 Ω per 100′. So, for example, to get 9 V at 0.5 A to the end of a 200′ cable, you need to inject 12 V to account for the 3-V voltage drop. E = IR (3 V = 6 Ω × 0.5 A).

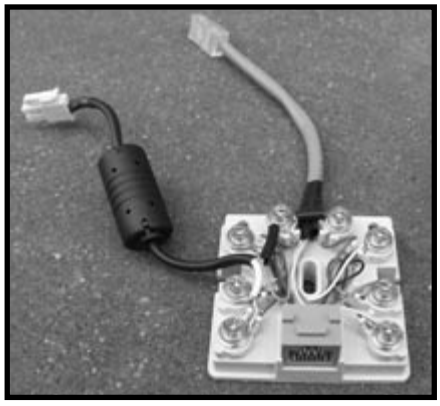


Photo 1—*For a homebrew installation, consider wiring your own PoE solution. (Photo courtesy of www.bawug.org/howto/hacks/outdoor_intel2011/pics/source/07_poe_surface_jack_exposed.jpg.)*

Keep in mind that it's the actual power consumption that matters, not what's on the label. My experience is that wall wart-powered gadgets rarely consume the entire maximum rated output of the power supply. To eliminate all the uncertainty (i.e., cable resistance and device current consumption), I suggest using a variable supply to test your actual configuration. Plug in the cable you intend to use and turn on the device to establish the actual load. Then, tweak the supply until the voltage measured at the device terminals is just right.

Furthermore, although a gadget may come with a 9-V wall wart, if it's got a linear regulator inside, it might work just fine at 7 V depending on the regulator's dropout voltage spec. For that matter, the typical 5-V regulator can tolerate a higher voltage too, but just be careful that the total power dissipation (combined current consumption and voltage drop across the regulator) doesn't make things too hot for comfort.

Consider the National LM78L05, a popular example of the 5-V linear breed. According to the spec, this puppy can deliver 5 V at 100 mA from an input voltage anywhere between 6.7 V and a whopping 35 V. However, the total power dissipation is limited by the regulator's maximum junction temperature (125°C) and package thermal resistance spec (230°C per 1 W) to

little more than 0.5 W (i.e., 125/230). That means if you actually feed an LM78L05 35 V (i.e., 30-V drop), it can only service a measly 15- to 20-mA load, which is far less than its 100-mA spec. And, even at that, you're left juggling the 125°C hot potato and contemplating thermal management, heat sinks, fans, ventilated enclosures, and all the rest.

Class	Usage	R_{CL} (Ω)	Maximum power used by PD (W)
0	Default	10 k	0.44 to 12.95
1	Optional	732	0.44 to 3.84
2	Optional	392	3.84 to 6.49
3	Optional	255	6.49 to 12.95
4	Not allowed	178	Reserved*

**Class 4 is reserved for future use.*

Table 1—*802.3af powered devices (PDs) aren't required to classify their power needs. (The default is maximum power.) But doing so enables smarter power management in mulitport power-sourcing equipment (PSE).*

The same power dissipation concern applies to the wire itself. Do not bite off more than your Cat 5 can chew. After all, shipping a lot of current over a skinny wire is the way a light bulb works. As the how-to web page notes, that's OK if you want to wrap the Ethernet cable around your water pipes to keep them from freezing, but it's more likely just a fire hazard.

ET PHONE HOME PAGE

If it were just one-off homebrew installations, this PoE story would be over. However, what has happened in the last few years is that big rack-and-stack communications infrastructure camels like Cisco and 3Com have stuck their noses into the tent with their own proprietary PoE schemes. When you're talking about hundreds and thousands of nodes, the cost savings from centralizing power distribution are compelling.

Most recently, PoE has gotten an extra push with the emergence of so-called voice-over-IP (VoIP) phones. Simply put, VoIP phones digitize the voice at one end, ship it over the packet-switched Internet, and then reassemble the packets and convert back to analog at the other end. Although the earliest VoIP phones struggled with technical challenges (e.g., latency) and a lack of service offerings, those problems appear to be fading fast.

Meanwhile, and perhaps most importantly, the VoIP industry has been graced with a regulatory free pass from obligations like 911 service and rural subsidy fees. If you can't beat 'em, join 'em. And now even traditional phone companies like AT&T are hopping on the VoIP bandwagon.

There's only one problem: a real (traditional analog) phone gets its power from the phone lines. Yes, it may come with a wall wart, but that's just for the fancy extras like an answering machine. It's comforting to know that even if power fails, old Ma Bell is still there for you.

So, a VoIP phone is going to go out when the lights do? Or everybody needs a little UPS to go with their VoIP phone? And users get to tie themselves in knots with not one but two cables (Ethernet and power)? No way. Instead, PoE, with centralized power distribution amenable to battery backup, allows VoIP phones meet their analog counterparts head-on.

WORLD WIDE WIRE

Of course, having multiple proprietary PoE standards would be far worse than having no PoE at all. Every time you jack in, it's a smoke test? No thanks.

Fortunately, everybody involved saw the light and managed to reach a peace agreement that was ratified in the IEEE 802.3af standard, which defines PoE enhancements for new and existing 802.3 networks. Let's start with the view from 50,000′ and then dive in to take a closer look.

The standard defines two distinct roles for PoE players: power-sourcing equipment (PSE) and powered device (PD). As the names imply, PSE is responsible for delivering power for use by PDs.

As a practical matter, most designers will find themselves in the PD camp, with PSE mainly under the purview of the infrastructure providers, IT managers, and building contractors. It's kind of analogous to an AC outlet; a relatively small number of utility providers (PSE) deliver power to a myriad of plug-ins (PDs).

Two types of PSE—midpoint and endpoint—are defined (see Figure 1). Like the homebrew schemes, midpoint power injectors and taps sit between existing non-PoE devices and use the two (typically) unused pairs in the four-pair Cat 5 cable. Midpoint PSE is a good solution for initial retrofitting or situations where only a relatively small percentage of ports need PoE. In the long term, market dynamics and customer demand will fuel inexorable migration to endpoint solutions, especially in new commercial installations. Ultimately, it's not hard to imagine (and many

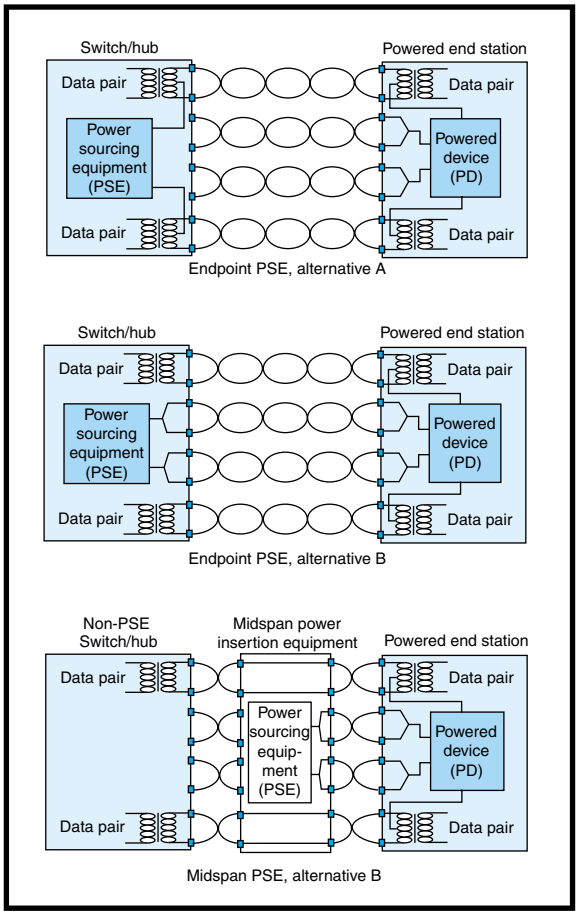


Figure 1—*IEEE 802.3af supports midspan and endpoint configurations. Note that the former only allows sending power over the spare wire pairs, while the latter also supports the option of sending power over the data pairs. Another difference is that endpoint solutions can serve 10, 100, and 1000 Mb-per-second links, while midspan is limited to 10 and 100 Mb.*

PoE advocates hope for) a time when practically every Ethernet jack offers power to whatever plugs in.

Another key point is the standard's choice of a 48-V supply (44- to 57-V spec). At first glance this seems like overkill, but, on further thought, it makes a lot of sense. All else being equal, a higher voltage delivers a given amount of power with less wiring loss (i.e., less current) than a lower voltage. At the same time, 48 V is low enough to slip under the radar of various safety regulations that kick in at higher voltages. Finally, matching the voltage commonly used in existing telecom infrastructure makes for less regulatory hassles and an easier sale and transition to the new gear.

The spec limits a single PSE port to delivering a maximum of 12.95 W to a PD. I've read some complaints that this may not keep up with the evermore power-hungry devices, notably laptop PCs, but otherwise that seems like more than enough power for the typical Ethernet add-on.

PSE is burdened with overcoming any issues associated with existing incompatible wiring (e.g., an analog phone on one of the ostensibly spare pairs) as well as the possibility of bumping into existing proprietary PoE gear. To achieve that goal, in addition to using the spare pairs, endpoint PSE has the option of delivering power over the data lines themselves. Fortunately, because Ethernet links are already electrically isolated (e.g., typically via transformer) in the interest of safety, carrying this so-called phantom power (i.e., DC offset) on the data lines isn't a problem. Note that an endpoint PSE can offer power on the data or spare pairs but not both at the same time.

DISCOVERY CHANNEL

I use the terminology that a PSE can "offer power" intentionally. It's like the old saying: PSE can lead a PD to power, but can't make it drink.

Because of the aforementioned possibility of incompatible wiring, not to mention the legion of existing non-powered devices, PSE cannot simply fire and forget the 48 V. Instead, PSE periodically probes its Ethernet links to see if a PD that requires power has jacked in. Let's go through a second in the life of PSE and a PD to illustrate the process.

Considering various real-world scenarios, the 802.3af experts came up with a clever way to sort the wheat from the chaff. The scheme simply requires that the power requesting PD exhibit a particular signature resistance across the power lines.

During the detection phase, PSE relatively slowly (0.1 V/ms) ramps up the power from 0 to 10 V. During this time, the current allowed to flow is limited to 5 mA in case there's a short or other problem. As the voltage rises, the PSE measures the current flow at two levels (at least 1-V apart) to determine the possible presence of the detection signature resistor in a PD at the other end. Two measurements (slope) are required to factor out the DC offset (cable loss) between the PSE and PD. If the PSE detects an open (i.e., nothing or a non-powered device plugged in), short, or anything other than the specified detection signature resistance, it shuts off the power.

Assuming that a valid PD (i.e., detection resistance) is found, the PSE continues ramping the voltage. As it passes between 15.5 and 20.5 V, the PSE and PD enter what's known as the classification phase, in which the PD is given the opportunity to request a particular amount of power (see Table 1). Once again, the technique involves the PSE measuring the current flow, this time across a classification resistance. Note that from a PD's perspective, the classification phase is optional with full power the default. However, a PD that volunteers to accept a lower power level will be welcomed by multiport PSEs that must allocate their maximum power delivery capability across many ports. Knowing now that there is a PD requesting power (and how much to deliver), the PSE ramps the voltage and current limit to their full specifications.

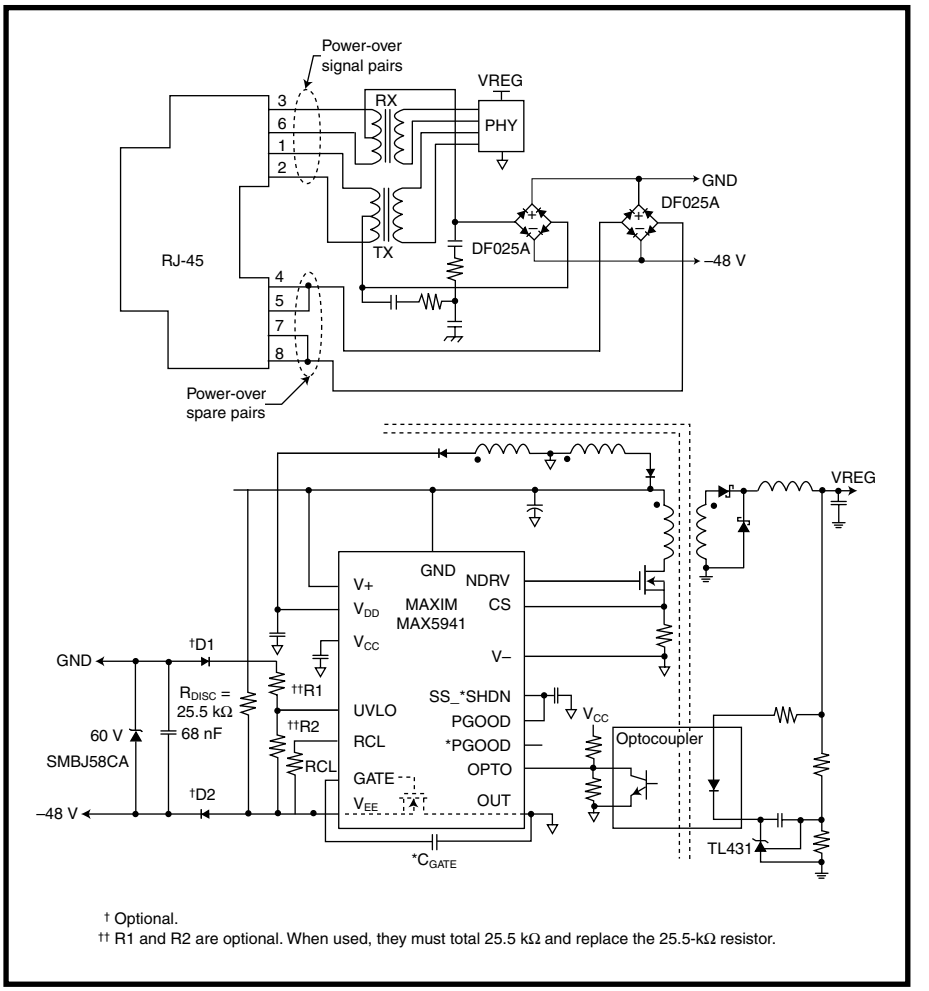


Figure 2—*Chips like the Maxim 5941 include both 802.3af features and power supply-related logic such as an oscillator and closed-loop PWM controller.*

Total UART to RF Solution

112dB Link Budget

Radiotronix ©

Embedding the wireless future...

84 channels



DTS spread spectrum

Wi.232DTS Module

5 volt tolerant I/O

Delivering the latest with the greatest.

The Wi.232 module is a state-of-the-art communications system encapsulated in a tiny, surface mount package. As the catalog distributor focused on the needs of engineers, Mouser is an all-in-one package. We not only deliver the latest technologies, but the great service and support you need.

SEMICONDUCTORS | PASSIVES | INTERCONNECTS | POWER | ELECTROMECHANICAL | TEST, TOOLS & SUPPLIES



Supplier Authorized Distributor

New Products,
New Suppliers,
New Technologies,
New Catalog Every 90 Days!

MOUSER ELECTRONICS tti

(800) 346-6873 www.mouser.com

Mouser® and Mouser Electronics® are trademarks of Mouser Electronics, Inc.

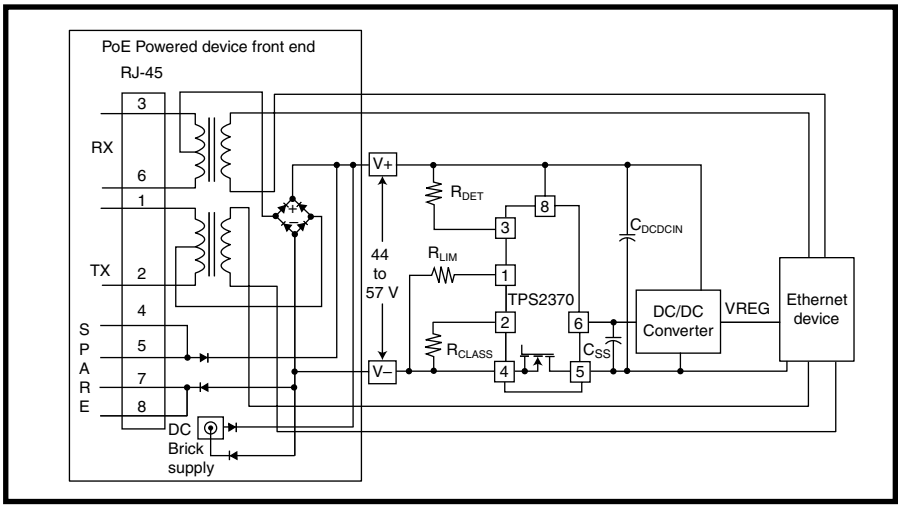


Figure 3—*The simplest PoE solutions combine a chip like the TI TPS2370 that handles 802.3af features (e.g., detection and classification) with an off-the-shelf DC-DC converter module.*

DOUBLE DUTY

During detection and classification, the PD has little responsibility other than to present the appropriate resistance (detection and classification) to the PSE at the appropriate time (i.e., as the voltage ramps through the detection and classification levels). Remember, though, the PD must be able to accept power on either the data or spare pairs (but never both at the same time) at the PSE's pleasure. The IEEE standard specifically disallows PDs that don't accept either power-delivery option.

Generally speaking, the PSE handles current limiting. It will shut off power if the PD exceeds the allocated, or maximum, current draw. However, if the PD presents a large capacitive load (greater than 180 µF), the IEEE spec requires that it assume responsibility for inrush current limiting.

The PD also has housekeeping to attend to during normal operation. The spec requires that the PD monitor the input voltage and turn on the load after it reaches 42 V. Similarly, the PD must disconnect the load, a so-called undervoltage lockout (UVLO), if the voltage falls below 36 V. The 6 V of hysteresis prevents the power supply from chattering on and off under marginal conditions.

The subject of disconnection is a little deeper than you might think. During operation, the PD is required to present two distinct "maintain power signatures" (MPS)—one AC and one DC.

The AC MPS scheme is similar to the detection phase in that the PSE probes the link periodically with a ripple voltage riding on the output looking for some indication that a valid PD is still present. This handles the typical case of a physically disconnecting PD. If the PSE suddenly finds a high resistance—greater than 1.98 MΩ—it presumes that the PD's plug has been pulled and turns off power.

The DC MPS component is a little trickier. Essentially, the spec says that if the PD fails to consume enough power (e.g., 10 mA) enough of the time (e.g., 75 ms every 325 ms), the PSE is required to cut it off. To complicate matters further, although PDs are required to present both AC and DC MPSs, PSEs are allowed to choose whether they monitor one, the other, or both.

Although the vast majority of PDs will likely need more power, be advised that if a PD happens to draw less than 10 mA, it seems to me there could be a problem.

PSE probes and detects the PD,
PSE gives the PD power
PD doesn't use enough power,
fails DC MPS
PSE cuts off the power
Repeat…

Of course, the easy way out is to hang a dummy load (e.g., a resistor or LED) on the thing and be done with it, but that's a hard pill to swallow for any efficiency and environmentally-minded engineer.

A similar question is how one might go about turning a PD completely off while leaving it plugged in. Maybe it's as simple as a double throw on/off switch that disconnects both the AC MPS detection resistor and the DC load; but, I have to admit, I'm still scratching my head a bit.

POWER TRIP CHIPS

Reflecting grassroots support for 802.3af and a positive market outlook, a lot of major IC players are rolling out compatible products, even as the ink on the standard is barely dry. As mentioned earlier, most of the action is in chips for PDs given the universe of PoE plug-ins they will enable.

As a would-be PD designer, your first choice is whether to make or buy the DC-DC converter that ultimately takes the 44 to 57 V and turns it into the lower voltage(s) (1.8 to 12 V) that most digital systems require. Chips are available that support both the make and buy options.

For those who have the time and inclination to design their own power supply, a chip like the Maxim 5941 is the way to go (see Figure 2). In addition to handling the PD detection and classification phases, it incorporates the PWM controller that serves as the brains of a switch-mode power supply.

For those who prefer to take the easier, albeit more expensive, route of buying the DC-DC converter, a solution based on the TI TPS2370 is blessedly simple (see Figure 3). Beyond the



Photo 2—*If things go as PoE proponents plan, the power outlet of tomorrow will look something like this NJ100-powered network jack from 3Com.*

STAMP VIDEO !!

LOW COST SINGLE CHIP VIDEO CONTROLLER



$25 EVAL (1)

$7.10 OEM (1K)

- Works with Stamp, z80, AVR, 8051, x86, PIC, etc.
- 8 col x 6 row alpha-numeric or 640 pixel graphic
- Includes high speed serial and parallel interfaces
- Drive CRT, TV, LCD with NTSC or VGA signal
- Source code examples in C, BASIC, Assembly

SINGLE CHIP COMPUTER $7

eval kit oem (1k) $1.99

-ZERO external components
-Built-in BASIC / Assembly
-RS232 program download
-1K flash,64ee,3irq,2timers
-15 I/O bits, A/D comparator
-20mips,faster than pic/8051
-20 pin DIP part #MV1200



NEW! 8K SUPER CHIP
40x the BASIC program space!
-32 i/o, 12 irq, 3 timers, Bus
-8K flash, 512ee, 512nvram
-Watchdog with internal osc.
-40 pin DIP part #MV8515
oem(1k) $7.10 eval(1) $25.00

CREDIT CARD COMPUTER 2



New PLUG-N-GO, no cables/power supply to buy!
Lo-power RISC cpu 10x faster than PIC, Z80, 8051
4meg NV mem,ser,par,RTC,4ch 12bit ADC,ISA bus
Built-in BASIC, Assembly, C compiler included
Friendly instruction set, unlike PIC or 8051
CCC2 eval(1)$75/oem(1k)$21 CCC1 $50/$14



LO COST MINI-PC

Includes DOS,NV mem,ADC,RAM,clock,ISA bus
Ports for serial, parallel, LCD, keyboard
Program in Turbo C, BASIC, MASM, etc.
Complete, no costly development kits required
Lowest power lowest cost PC compatible available
XT: oem $27 eval $95 AT: oem $55 eval $195

SERIAL MINI-TERMINAL

RS232 terminal for Stamp, PC, Z80, AVR etc.
-super low-current, powers from serial line
-LED backlit LCD, visible in all conditions
-115.2kbps, DB9 conn, simple commands
-specify 20 customizable or 16 tactile keys
eval(1) $75,oem(1k) $21.30,w/BASIC cpu $27



640x480 VGA LCD $27

Controller for most single/dual scan LCDs
Works with lo-res (160x120, 320x240,etc.)
Use with PC or SBC, standard VGA BIOS
Source code demo shows VGA initialization
Adaptable for other CPUs (i.e. Z80, HC11)
oem(1k) $27 evalkit(1) $95 w/10"LCD $195



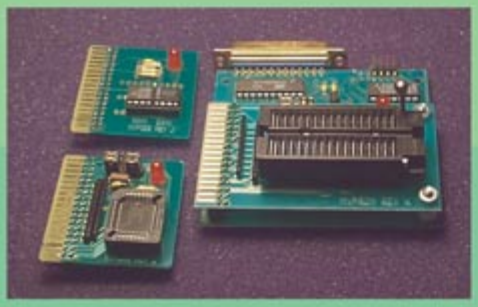
STAMP DRIVE



Read / Write PC compatible hard disk, PCMCIA, & Compact Flash on stamp, 8051, pic, avr, z80, x86
-up to 4 gigabyte capacity
-low power, simple commands
-RS232 @ 9600, 115.2k, 667k
-$95eval $27oem, IDE version$14

LIMITED TIME OFFER!
- FREE SERIAL EE
- FREE PLCC ADAPT.
- FREE ZIF SOCKET

$95 PRINTER PORT PROGRAMMER



LOWEST COST, FASTEST, EASIEST TO USE!

Does FLASH, EE, NVRAM, EPROM to 8m, (27080). Adapters for micros.
Super small size! Perfect for notebooks or drive bay of desktop computer.
WINDOWS and DOS software with script file ability to simplify operation.
Higher yield and quicker programming than others. "Gang Bang" option.

WWW.STAR.NET/PEOPLE/~MVS

MVS Box 803
Nash.,NH 03060
(508) 792 9507

.MVS

5yr Limited Warranty
Free Shipping
Mon-Fri 10-6 EST

SERVING THE EMBEDDED COMMUNITY SINCE 1979!

Save months of development time by using our ready-to-run embedded single board computers with your choice of operating system...

VIPER single board computer

- 400MHz Intel® PXA255 XScale RISC CPU
- Ultra low 1.6W operating power
- 64MB DRAM, 32MB Flash, 256KB SRAM
- Flat panel display controller (TFT/STN)
- CompactFlash, USB, serial ports, RTC
- GPIO, audio, PC/104 expansion

Embedded Linux Development Kit

- Based on 2.4 kernel
- Uses Journalling Flash File System (JFFS2)
- OpenSSH for improved communication security
- Optional embedded Java™ technology

Windows® CE .NET Development Kit

- Flash memory management using Intel Persistent Storage Manager
- Standard build includes comprehensive set of Windows CE components
- Powerful GUI capabilities
- Includes Embedded Visual Tools C++ Tools

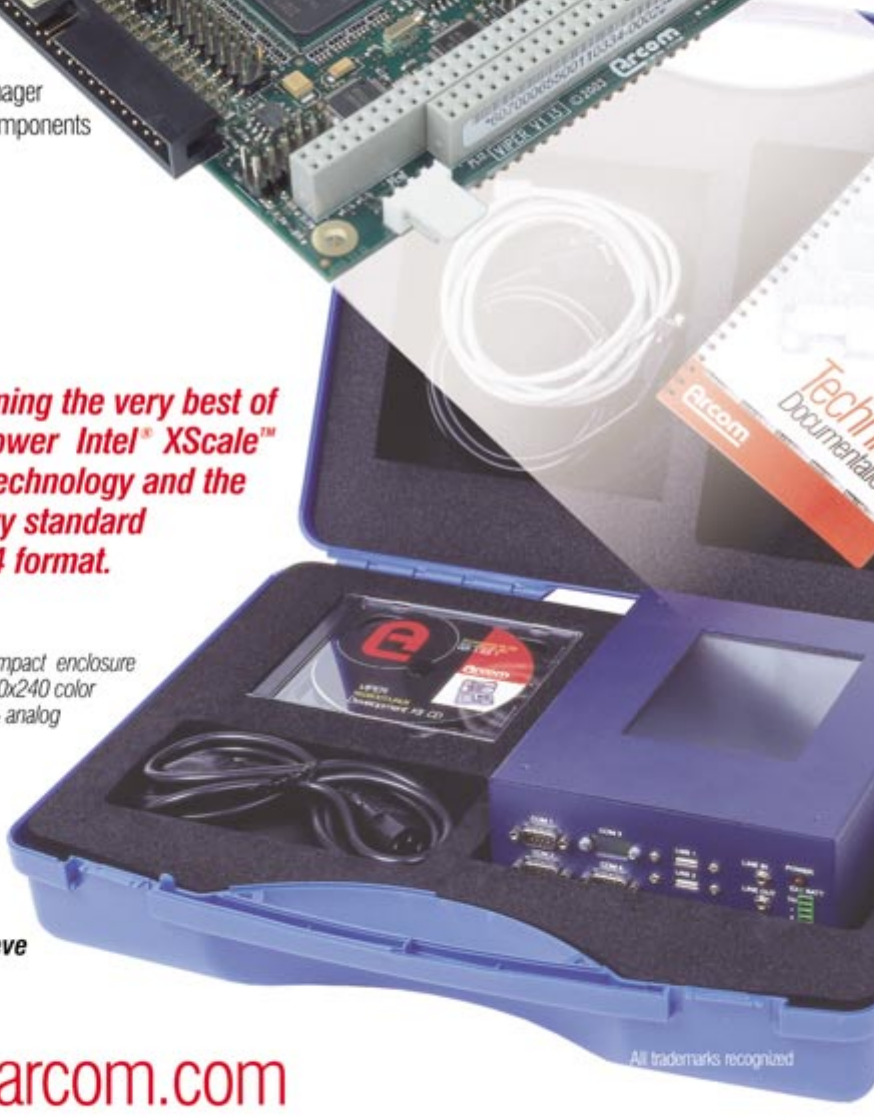
VxWorks® Development Kit

- Support for VxWorks 5.5 real time operating system
- WindML 3.0 multi-media library
- Automatic Tornado II installations



Combining the very best of low power Intel® XScale™ RISC technology and the industry standard PC/104 format.

Industrial compact enclosure (ICE) with 320x240 color TFT display & analog touchscreen.



CUSTOM DESIGN & BUILD ARCOM

Use our Design & Build service to achieve cost effective custom design solutions.

Check our website www.arcom.com

Development Kits | Embedded Boards | PCs & Enclosures | Industrial Networking | Design & Build

Arcom

Call now on 888-941-2224
E-mail @ us-sales@arcom.com

All trademarks recognized

standard 802.3af front-end, design-in involves little more than choosing the appropriate detection, classification, and current limit resistors.

There are plenty of other 802.3af chips from all the usual suspects. However, as you go through the datasheets, be on the lookout for possible gotchas related to different interpretations of the spec. For instance, Supertex makes a part, the HV110, that's similar to the aforementioned TPS2370. However, unlike the TI part, the HV110 includes a semi-equivalent feature to the DC MPS that normally resides in the PSE. If the DC current draw falls below 20 mA, the HV110 itself will cut off power. The datasheet argues that this provides insurance should an HV110-based PD encounter a noncompliant PSE. But whether or not this is actually a feature or hindrance will depend on the specifics of a particular PD application.

Other detailed differences include the fact that some chips are smart enough to switch out the detection and classification resistors after those phases are complete in order to eliminate the unnecessary load. Finally, it's one thing if you presume your PD will be powered only by a PoE jack, period. But what about the situation in which a PD is plugged into a non-powered Ethernet, not to mention the option of standalone operation (i.e., not connected to Ethernet)? The fact is, many PDs will still need a DC power jack and wall wart to handle these situations. The easiest solution is to use a high-voltage (e.g., 44 V) DC supply and feed it through the 802.3af front-end, taking advantage of the current limiting, undervoltage lockout, and other PoE features. Alternatively, a lower DC voltage can be inserted pre- or post-DC-DC converter.

CLOSE SHAVE

I'm pretty optimistic about the outlook for 802.3af and PoE. With a standard in place, I suspect the infrastructure providers, large commercial installations, and VoIP advocates will move quickly to proliferate powered sockets (see Photo 2). As that happens, suppliers of plug-ins will have the opportunity, and ultimately competitive necessity, to take advantage of the "free" power.

Heck, maybe PoE will even have an effect beyond the world of ones and zeros. Without really trying, 802.3af also happens to define a single worldwide standard for power delivery, something even the United Nations can't pull off.

Indeed, I heard about a trade show PoE demo that had folks shaving with an electric razor plugged into an Ethernet jack. Not that I'm an advocate of desktop toiletry, but the market will no doubt show a lot of creativity when the time comes. For example, airlines might find PoE a great way to accommodate both the power and communication needs of their electronics-laden passengers.

If you're designing a gadget that plugs into Ethernet, I strongly recommend that you get up to speed on 802.3af. A chicken in every pot—and 48 V in every RJ-45—is the battle cry of this twenty-first century version of "power to the people." ◙

Tom Cantrell has been working on chip, board, and systems design and marketing for several years. You may reach him by e-mail at tom.cantrell@circuitcellar.com.

RESOURCE

IEEE, Inc., "802.3AF-2003 IEEE Standard for Information Technology—Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications Amendment: Data Terminal Equipment (DTE) Power via Media Dependent Interface (MDI)," 0-7381-3696-4, 2003.

SOURCES

MAX5491
Maxim Integrated Products, Inc.
(408) 737-7600
www.maxim-ic.com

HV110 PD Controller IC
Supertex, Inc.
(408) 222-8888
www.supertex.com

TPS2370
Texas Instruments, Inc.
(972) 644-5580
www.ti.com

USB
Add USB to your next project—it's easier than you might think!
USB-FIFO up to 8 mbps
USB-UART up to 3 mbps
USB/Microcontroller boards pre-programmed with firmware
Absolutely NO driver software development required!
www.dlpdesign.com
Design services available
DLP Design

USB DAQ
LabJack U12
Available now for only ...
$119 qty 1
($99 qty 10+)
USB Data Acquisition & Control
* Built-in screw terminals
* Easy-to-use USB
* Everything included
* Use with C, VB, LabVIEW, etc.
* Windows 98SE/ME/2000/XP
* 12-bit analog inputs (8)
* 10-bit analog outputs (2)
* 20 digital I/O
* 32-bit counter
Made in Colorado, USA, by LabJack Corp.
info@labjack.com, (303) 942-0228
www.labjack.com

Put Your Devices on the Internet
(and in local networks too)
Networking isn't just for PCs. Learn how to design and program small devices that exchange data via TCP/IP, host Web pages, communicate using e-mail and FTP, and much more!
Embedded ETHERNET AND INTERNET COMPLETE
Designing and Programming Small Devices for Networking
JAN AXELSON
Embedded Ethernet and Internet Complete
by Jan Axelson
ISBN 1-931448-00-0 $49.95
Lakeview Research www.Lvr.com
From the author of USB Complete

IDEA BOX

THE DIRECTORY OF PRODUCTS AND SERVICES

AD FORMAT: Advertisers must furnish digital submission sheet and digital files that meet the specifications on the digital submission sheet. **ALL TEXT AND OTHER ELEMENTS MUST FIT WITHIN A 2″ × 3″ FORMAT**. Call for current rate and deadline information. Send your disk and digital submission sheet to: IDEA BOX, Circuit Cellar, 4 Park Street, Vernon, CT 06066 or e-mail kc@circuitcellar.com. For more information call Sean Donnelly at (860) 872-3064.

The Suppliers Directory at www.circuitcellar.com/suppliers_dir/
is your guide to a variety of engineering products and services.

PHYTEC

phyCORE®
New Generation
Single Board Computer



- Insert-ready Single Board Computer modules in sub-miniature dimensions (as small as 47x55 mm) populated by:
  - 8-bit: ADuC812, AT89C51CC01, DS80C390, P87C591, P89C51Rx2, P89C66x, P89C51MX
  - 16-bit: C167CR, C167CS, XC161CJ, XC167CI, PXAC3, PXAG49, ST10F168, ST10F269
  - 32-bit: AT91M55800A (ARM7TDMI core), Elan SC520, MPC555, MPC565
- applicable controller signals extend to standard pin header (2.54 mm) or high-density Molex (0.625 mm) connectors, enabling SBC to be plugged like a "big chip" into OEM targets
- Low EMI design achieved via GND circuitry, multi-layer PCB, by-pass capacitor grid and short signal traces achieved via small footprint and use of 0402 SMD passive components
- 32 KB to 64 MB external SRAM and Flash (controller-dependent)
- CAN, Ethernet, RS-232 and RS-485 interfaces; ADC, DAC (controller-dependent); freely-available /CS and I/O lines
- available in Rapid Development Kit including Development Board, AC adapter, serial cable and SPECTRUM CD with eval software tools, electronic documentation and demos
- Stick It In!: insert-ready PHYTEC SBC modules accompany you from evaluation through protyping to end production... accelerating your design cycle and reducing your time-to-market

PHYTEC America, LLC ■ 203 Parfitt Way SW, G100 ■ Bainbridge Island, WA 98110 USA
www.phytec.com (800) 278-9913

BOB

eXperience XBOB
turn serial data into video text

www.decadenet.com

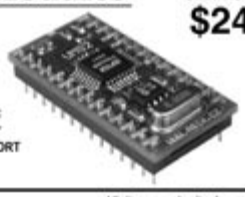
DECADE ENGINEERING
503-743-3194 Turner, OR, USA

LOW-COST PIC TOOLS

CAPR-PI
ICSP PROGRAMMER FOR PIC MCU's
$28
SOFTWARE INCLUDED

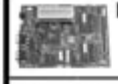


USB Modules
STARTING AT: $24



CONVERT PC USB PORTS TO:
• SERIAL PORT
• PARALLEL PORT

BayTech SALES
Visit our web site for more great values in PIC tools!
www.baytechsales.com
(978) 692-5320



I2C, MDIO, CAN, SPI, XA Kits, & ISP FLASH



PC-ISP-FLASH programs 80C51, XA and AVR ISP/IAP FLASH micros from Philips and Atmel with user upgradeable Windows software. ISP interface cable, software and documentation included for **$199.00**.

PC-I2C-KIT allows bi-directional communications between any PC and I2C, SPI or MDIO based peripherals. Windows software, board, interface cable and documentation included for **$249.00**.

PC-I2C-DEV lets the user write programs to control the I2C/SPI/MDIO interface board directly using Microsoft Visual C/C++, Borland C/C++, Microsoft Visual Studio, etc.. Includes a software DLL for custom user applications and a complete PC-I2C-KIT for **$499.00**.

XTEND Kits allow the user to debug or prototype any Philips XA based design. Kits include XA Dev Board with Monitor and Debugger, Software, Documentation, Schematics, Power Supply and Serial Cable.

XTEND-Gxx (G3 & G49) or **XTEND-Sxx** (S3) for **$249.00**
XTEND-C3 or **XTEND-H4** for **$299.00**

FDI
www.teamfdi.com
VISA/MC/Amex

Future Designs, Inc.
2702 Triana Blvd
Huntsville, AL 35805
(256) 883-1240
Fax (256) 883-1241

MICROSTEP DRIVER FOR TINY MOTORS



- For motors to 300 mA/ph • 3,200 µsteps/rev
- Micro-Full-Half Step • PCB Mount or Cable
- 12-42+ VDC • Opto-Isolated Interface
- Bipolar Chopper • Compact Package

Order # CY-20
$108 in qty 1 – $67 in qty 1,000

800-328-3938 CyberPak company
www.CyberPakCo.com
email: sales@CyberPakCo.com

AVR Pocket Programmer

- supports standard, tiny & mega devices
- portable

USE FOR:
- field upgrades
- product configuration
- lab



starter kit $179.95
pocket programmer $79.95

npe
www.npe-inc.com
quantity pricing! Order online!

Value CAN

The High Value Tool For Controller Area Network

- USB to CAN
- Simple software analyzer included
- DLL with examples for custom applications
- PC isolated from CAN
- 100% bandwidth at 500Kb

Only $249

ICS RADDAQ

www.intrepidcs.com/vcan

PC KEYBOARD EMULATION

Interface Keypads, Switches, or RS-232 to your PC Keyboard Input

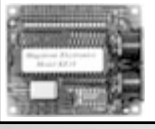
MODEL KE24 ONLY $99.95



• Up to 12 x 12 matrix • Programmable • RS-232 Port • Macro Capability

The KE24 is the ultimate in flexibility. Inputs or serial data can emulate any of the 101 keys from a standard keyboard.

MODEL KE18 ONLY $44.95



• Up to 9 x 9 matrix • 2.5" x 3.0" Size • PC Keyboard Port • PCXT, AT Compatible

The KE18 combines a multitude of features with small size at an economical price. Custom units available.

HAGSTROM ELECTRONICS
11 Fiddlers Green
Lansing, NY 14882

Toll Free: 888-690-9080
Phone: 607-533-4441 Fax: 607-533-4443
www.hagstromelectronics.com

VISA MasterCard

Developing for Rabbit?



- ANSI C Compiler with Linker and Debugger
- Integrated Full TCP/IP Stack
- Far Pointer and Far Data Support
- Ultra-Compact Code Generation
- **Download Free Demo!**

Softools
www.rabbittools.com
860.236.4201

Don't Compromise!

Give your programmable instrument the command language it deserves

SCPI *used to be the preserve of the major manufacturers...**NOT ANY MORE***

With the JPA-SCPI Parser C library, you can support full SCPI with ease

It interprets and parameterizes all the commands received, allowing you to focus on making your product great!

The JPA-SCPI Parser C source library is ANSI-compliant and works with most platforms and ANSI C compilers.

No runtime fees. Site license: $495

For full information, a downloadable demo or to order, visit:

www.jpacsoft.com

JPA Consulting *accelerating development*

Gamoto

The *Smart Motor Controller**



* Closed Loop PID Control
* 8 trapezoidal motion profiles
* I^2C bus, Cricket bus, or Serial
* Motor current (torque) feedback
* Turns any motor into a servo
* 32-bit position registers

www.gamatronix.com

It writes your USB code!

NO Need to be a USB expert!

HIDmaker ($399) – creates ready to compile PC & PIC programs that talk to each other over USB.

Choose your favorite languages!
PIC: Pic Basic Pro, CCS C, Hi-Tech C, MPASM. **PC:** VB6, Delphi, C++ Builder.
Single chip solution: PIC with built-in USB

HIDmaker Test Suite ($149)

USBWatch – shows your device's USB traffic, even during 'enumeration', without expensive equipment.

AnyHID – Test any USB HID device. See what data it sends, even what the data is used for.

Trace SYSTEMS, Inc.

301-262-0300
WWW.TraceSystemsInc.com

PROTOTYPE
CIRCUIT BOARDS
$85 US for two 5" x 6" 2-layer boards
Shipped NEXT BUSINESS DAY if data is received by 1:00 pm EASTERN
www.apcircuits.com
AP Circuits
(403) 250-3406
staff@apcircuits.com

ActiveWire® USB
Simple USB Interface!

• Works with MacOS 8/9, Win98/2K/ME/XP FreeBSD and Linux!
• 24Mhz CPU core with USB
• Firmware downloadable via USB
• 16 bit parallel Input/Output
• See web-site for add-on boards
• All drivers, manuals, demos are on our web-site for immediate download!
$59 plus shipping
• Inquire about custom designs/products
ActiveWire, Inc.
www.activewireinc.com
ph +1.650.465.4000

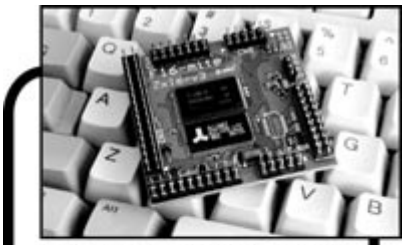
China Connection...
PCB Production, Assembly, Turnkey
Prototype ~ Production
D/S PCB ~ 12¢ per in²
4-L PCB ~ 20¢ per in²
10-Day Delivery w/solder Mask
NRE not included
UL Approved
MYLYDIA INC.
1-800-Mylydia
sales@mylydia.com
www.mylydia.com


Low Cost OEM Handheld Computer
- Motorola DragonBall 16MHz EZ CPU
- 160x160 pixel LCD with Touch Panel
- RS-232/IrDA interfaces
- MOQ pricing USD 40.00
- OEM pricing available
- OEM customizations possible
- Complete s/w, h/w SDK available
- Our web site: www.zexus.com.hk
Tel: 852-2964 1771
Fax: 852-2546 3806
Email: sales@zexus.com.hk

Embedded Ethernet
ONLY $98

NEW! picoFlash
• 10Base-T Ethernet
• 186 Processor @ 40 MHz
• DOS w/ Flash File System
• 16 Digital I/O • 5V DC
• Console / Debug Port
• Hardware Clock/Calendar
• Socket for DiskOnChip
• 512K DRAM & 512K Flash
• (2) Serial Ports
• (2) 16-bit Timers
• Watchdog Timer
• 3.75" x 2.50"
$129 Development System
Development Kit Includes:
• picoFlash Controller
• Borland C/C++ 4.52
• TCP/IP, PPP & Web Server
• Serial Driver Library
• AC Adapter and Cables
Call 530-297-6073 Email sales@jkmicro.com
On the web at www.jkmicro.com
JK microsystems

Circuit Cellar
back issues available as
Searchable archives on CD-ROM
Get the ones you missed!
CD-ROM #8 2003 Issues 150-161
CD-ROM #7 2002 Issues 138-149
CD-ROM #6 2001 Issues 126-137
Order online
www.circuitcellar.com
or call (860) 875-2199


Tiny, powerful, inexpensive microcontrollers
39 I/O + 8 ch 10 bit A/D
25MHz 16 bit Moto CPU
128K SRAM + 128+K flash
LCD/Keypad interface
Affordable C compilers
Background debugging
Comprehensive C library
Snap-off proto area
Ton of capabilities...
...yet fits in tight spaces and budgets
www.steroidmicros.com

Everything you need to build your own...
MOBILE ROBOT

➤ SONAR UNITS
➤ VISION SYSTEMS
➤ MOTOR DRIVERS
➤ OPTICS
➤ MICROPROCESSORS
➤ ARITIFICIAL INTELLIGENCE
➤ WEB CONTROLS
Zagros Robotics
PO Box 460342, St. Louis, MO 63146
(314) 768-1328 · info@zagrosrobotics.com
www.zagrosrobotics.com

Everything You Need For Instrument Control!
The QVGA Controller™

A C-programmable embedded computer with a built-in GUI. Use it for machine control, embedded systems, scientific instruments, and OEM applications - wherever you need an I/O rich computer and a smart user interface!
Ready-Made Graphical User Interface
● Bright, High Contrast 1/4 VGA (320x240 pixel) Electroluminescent (EL) or Monochrome Display
● High-resolution touchscreen
● Precoded menu managing software
Powerful Controller
● Programmable in C and Forth
● Real-time Multitasking Executive

All the I/O You Need
● 48 Analog & Digital I/O Lines
● Expansion I/O modules for any need
Plenty of Memory
● Up to 768K Flash, 640K RAM, 64Mbyte Mass Memory
Mosaic Industries Inc.
tel: 510-790-1255 fax: 510-790-0925
www.mosaic-industries.com

IDE, Compilers, ICD, Simulators, Programmers...
Emulators
8051 PIC® AVR®
80196 XEMICS
MSP-430 SENSORY
Phyton 718.259.3191
www.phyton.com

DEVICE PROGRAMMER, USB ROM Emulator, EPROM ERASER
TopMax ChipMax
TopMax : Programs over 7000 devices, 48 Pin ZIF Socket ,Tests TTL/CMOS Logic IC, Low voltage, On-board processor and built-in Universal AC, 9X/NT/2000/XP,CSA, UL, Made in USA, Lifetime free software, 2 year warranty, Free software update for life time
TM-8G - DP | 8-gang (32-pin DIP) adapter for E(E)PROM | $325
TM-8G - PL | 8-gang (32-pin PLCC) adapter for E(E)PROM | $495
$995 $345
ChipMax : Reliable and low price, Programs over 4000 devices, EPROM, FLASH Memory, Microcontrollers, PIC, Serial EPROM, EPLD, GAL, PALCE, 40 Pin ZIF Socket, Parallel, 9X/NT/2000/XP, Made in USA ,1 year warranty, Lifetime free software, 110 or 240 AC Adapter. CM-4G: 4-gang adapter 32-DIP $115
EeRom-8U
$495
"USB Interface"
EeRom-8U : 8Mbit-45ns EPROM/Flash Emulator, High speedy USB port interface (Plug & Play)
• Fast download (8Mbit per second), Lower Voltage, 8bit, 16bit, and 32bit data split
• High Speed access time (built-in 8M SRAM 15ns)
• No external power is needed, 9x/XP/NT/2000
• Accepts all standard Hex formats and binary files
• Jumper less Emulation size configuration
LA6T-110
$115
LA6T-110 : UV ERASER for Personal user, Fast and efficient, Molded Plastic Case, Hold 9 of 28-pin or 6 of 40-pin EPROM. Timer(15-40 min),
Programming Adapters(low price) : 8-SOIC($70), 32-PLCC($70), 44-PLCC($115), 48-TSOP($135), 44-QFP($65), 52-PLCC($165), 68-PLCC($165) ,
EPROM Emulator Adapter, 32/28-pin($95) 44-pin($115) DIP-to-PLCC
EE Tools
E.E. Tools, Inc.
549 Weddell Drive Sunnyvale, CA 94089 www.eetools.com
Tel.408-734-8184 Fax.408-734-8185

Nobody beats our PCB Prototypes !
These are Per Board Prices for 5 day turn
Compare our prices
Compare our turn time
Compare our ISO-9002 quality

Board Sizes	2Layers	4Layers	6Layers	8Layers
20 sq. inches	$13	$33	$48	$
40 sq. inches	$18	$43	$67	
60 sq. inches	$23	$47	$77	
80 sq. inches	$33	$73	$87	
90 sq. inches	$53	$87	$97	
100 sq. inches	$68	$107	$117	$

Now Introducing 2 Layer BARE PCBs
No Mask - No Legend - ALL SAVINGS!
visit us online for details
These are our regular 5 day turn prices offered online. And you DON'T have to buy 20 pieces to get these low prices.
Our minimum is just 5 pieces! Call us if you need faster turns or higher technology.
Includes Tooling, Mask and Legend..
Order Online at
PCB FABEXPRESS
High Quality PCBs @ Low Impact Prices
www.PCBFABExpress.com
eMail: support@pcbfabexpress.com Phone: (408) 857 0039

ASH WARE Inc.
proudly announces its
eTPU Stand-Alone Simulator
TIMING IS EVERYTHING™
With 32/64 channels of I/O the Simulator is used with Byte Craft Limited's eTPU_C Compiler to solve complex timing problems – including engine and motor control and communications – in both time and angle domains.
ASH
www.ashware.com

ETHERNET ON A CHIP
• System on a Chip • FREE RTOS/TCP/IP Binary
• µP, 256K SRAM, Ethernet
• CAN, SPI, Serial, Profibus
• Webserver, Protocols Available
DSTni™ DSTni-LX-001 1023120 0125 KOREA
Just $299 to get started order on-line
AVAILABLE THROUGH MASTER DISTRIBUTOR
630-245-1445 • 630-245-1717 fax
www.gridconnect.com
gridconnect. SYNERGETIC
Synergetic and DSTni are trademarks of Lantronix.
ORDER AN EVAL KIT TODAY!
Includes Paradigm C++ Eval

PC/104 PERIPHERALS

SCIDYNE Offers a Full Line of Innovative Modules for PC/104 Applications

ADIO-104 This Module Does it all!
•16 Analog Inputs •5V Power
•8 AnalogOutputs •Pulse Accumulator
•24 Digital I/O lines •Open-Drain Outputs

DIO96-104
•96 Bi-Directional Digital Channels
•Input, Output and Strobed I/O functions
•Uses Familiar 82C55A Chips

XIO-RO8
Add eight High-Power relay outputs to any digital port.

Call or Visit us on the web
SCIDYNE® 1-877-724-3963
www.scidyne.com

USB & Ethernet Serial Connectivity



USB to Serial
- 1,2,4, and 8-port models
- RS-232, RS-422/485, and RS-232/422/485 versions
- Supports data rates up to 921K bps

Ethernet Serial Servers
- 1,2,4, and 8-port models
- Ethernet 10/100Base-T (autosense)
- RS-232, RS-422/485, and RS-232/422/485 versions
- Easy-to-use software included

LIFETIME WARRANTY

SEALEVEL
www.sealevel.com
sales@sealevel.com 864.843.4343

okwelectronics.com

RF Remote Control Systems
FM 315 MHz
'Mix-n-match' Encoders & Decoders

102 SERIES HAND-HELD ENCODERS
- Transmitter encoders for OEM systems
- Secure KEELOQ protocol
- FM 315 MHz frequency
- 1,2 or 4 push button switches
- LED transmission indicator
- Range up to 660ft
- Auto shut-off
- ABS enclosure
- 9V Battery supplied
- FCC compliant

NEW!

008 SERIES FM DECODER
- FM remote Rx/decoder unit
- 'Easy learn' Tx encoder feature
- Secure KEELOQ protocol
- FM 315 MHz frequency
- Four relay outputs
- Range up to 500ft

More info. @ www.okwelectronics.com
OKW Electronics, Inc.
Tel: (800) 965 9872 Fax: (412) 220 9247
300 Old Pond Road, #203, Bridgeville, PA 15017
Email: sales@okwelectronics.com

R2 Controls provides Microcontroller *Solutions* that Deliver High *Performance, Reliability &* Exceptional *Flexibility!*

Available Now!

Customized Controller Design and Manufacturing Services

To complement our established family of Low Cost Full-Featured Microcontroller Solutions. We provide quick delivery of customized modifications to our Standard Controllers, or a completely new product designed to meet your needs and reduce your time to market! Call us today for details on our extensive custom design and quality manufacturing services!

New features in our standard products include Relay Outputs, Opto-isolation, Ethernet, Modbus and more! Check our website for new product announcements and pricing!

R2 Controls
Contact us at:
www.r2-controls.com or 866-797-9229

RF MODULES

NEW!

AT-XTR-903

- Multi Channel Transceiver
- 3V operation
- Up to 38.4Kbps
- Virtual wire Operation
- Wafer thin, Tiny Footprint
- Low power
- 433, 868 & 900MHz
- Standard serial I/O
- Programmable output Power
- Programmable Freq. Channels

ABACOM Technologies
www.abacom-tech.com
Tel: (416)236-3858 Fax: (416)236-8866

QKITS.COM

USB PIC Programmer Kits
RFID Proximity Card Reader Kits
PC Function Generator Kits
Most kits available assembled.

Oscilloscopes:
50MHz Digital Storage
40MHz Hand Held Scopes
1-Channel Digital PC Kits

Mini Kits, Wooden Motorized Non Solder Kits, PWM kits, Stepper Motor kits, Dimmer Kits, PIR Sensors. VU Meter kits, FM Transmitter kits, Audio Amp kits, PC Interface kits, Relay Card Kits, Test Equipment.

See our Solar Products online! 613 544 6333
49 McMichael St • Kingston, ON, K7M 1M8 CANADA
TOLL FREE 888 464 5487

CANtrol your FUTURE

Controller Area Network solutions to control and monitor your application process...catch the CAN bus and save money.

- CAN development kits from - $99
- Atmel AVR boards with CAN - $49
- ATmega 128 boards - $69
- ATmega 103 boards - $39
- AVR ISP programmer - $15
- Embedded Modem 2400 bps - $39
- MSP430 development boards - $49
- CAN Motor Controller kits from - $49

Akida LLC
3725 E. Roeser Rd
Phoenix, AZ 85040
Tel. 602-454-2654

Visit our website at www.Akida.com

ANSI C software development tools for:
CypressM PSoC
AVR & tinyAVR
HC08,11,12,16
TI MSP430

IMAGEcraft

Check it out:
Version 6 Tools
Full-featured 30-day demos on our website!

hardware development kits:
Elektronikladen - Motorola HC08, HC12 reference design & starter kits
CANDIP - AVR/CAN on a chip!
Ethernut - Complete AVR TCP/IP solution
Atmel - ICE-200, STK-500, EITI etc.

high powered development, garage powered prices!

www.imagecraft.com
info@imagecraft.com
706 Colorado Ave. Palo Alto, CA 94303
(650) 493-9326 • FAX (650) 493-9329

VISA / MC / AMEX
Check / Money Order

...say 'NO' to JURASSIC PRICES!

Network Boot for Embedded Devices

Choose emBoot for:
- Off the shelf or custom implementations:
  •Option ROM •BIOS-integrated •Disk on Module, Compact Flash
- Special requirements (e.g. custom fail-over on dual NIC boards)
- Client firmware and server components for end-to-end solutions
- Support for most LAN controller chips and boot protocols: PXE, DHCP, BOOTP, RPL
- Over 50 years of combined expertise since 1985

emBoot™ Inc.

(905) 629-0200 www.emboot.com sales@emboot.com



4 Channel RF Transmitter and Receiver Board with Auto Roll.

Available as a kit or assembled and tested unit. Transmitter and Receiver Kit $39.95

www.ElectronicKits.com

Hundreds of Electronic Kits, Robots and Cool Gadgets.
sales@electronickits.com

RD™ Low Cost, High Performance, Versatile Controller

80 MHz

NEW



- 3.9 x 3.6" C/C++
- 20 Solenoid Drivers
- Ethernet, 3 RS232, 16 ADCs, 8 DACs and 40+ IO
- 1 GB CF mass data storage with file systems
- 80 MHz CPU, 16-bit Flash and SRAM

OEM $99

40+ Low Cost Controllers with ADC, DAC, solenoid drivers, relays, file system with CompactFlash, LCD, DSP motion control, 11 UARTs, 300 I/Os. Custom board design. Save time and money.

TERN INC. 1724 Picasso Ave., Suite A, Davis, CA 95616 USA
Tel: 530-758-0180 • Fax: 530-758-0181 www.tern.com • sales@tern.com

IN CIRCUIT emulators

from $2,600
• LPT
• Serial



8051
80186
80196, Z8

Atmel
Dallas
Infineon
Philips
TDK

11992 CHALLENGER CT.
MOORPARK, CA 93021
PHONE 800 • 838 • 8012

SIGNUM SYSTEMS
www.signum.com

PICMICRO TOOLS

MBasic

MBasic is the easiest way to program and experiment with PICmicro® MCUs. MBasic allows you the flexibility you've never had before, choose the microcontroller that best suits your needs. MBasic quickly and easily converts simple BASIC language programs into files that can be programmed into almost any low-cost PICmicro® MCUs.

Starting at $99.95

MBasic features include:
32 Bit math
32 Bit Floating Point
Dallas One Wire Support
If..Then..Else..Elseif..Endif
Hardware PWM
Expanded LCD command
Basic Interrupts
Hardware Timers

Updated In Circuit Debugger
New Oscilloscope Software built in
Free Boot Loader Builder Included
Free Boot Loader Interface Included

ISP-PRO 3.0
All-in-one programmer
$59.95 Complete
The ISP-PRO Programmer is designed as a professional grade PICmicro programmer for hobbyist and professionals. RJ-11 adapter included for Microchip ICD compatibility along with our standard 10 pin header.

- Uses PC Serial port (USB w/ Adapter)
- Very Simple to use
- Free Software updates
- Supports PIC, I2C and more
- Easy In-circuit programming
- Complete with Windows IDE
- Firmware Upgradable
- RJ-11/10 Pin Header
- Includes RJ-11 Cable

Development Boards
Develop your projects the easy way
Ideal for quick prototyping with PICmicro MCUs®. Easy to move from prototyping to finished design. With in circuit programming (ISP) you can test your software/hardware design on-the-fly.
Starting at $49.95

- Completely Assembled Board
- In Circuit Programmable (ISP)
- Solderless Bread Board
- Built-in RS232 (w/ Max232)
- Built-in Power Connector
- Removable Oscillator
- Auto Disconnect for RB3, RB6 &RB7
- Available in Several Models

BASIC MICRO
Visit us online at WWW.BASICMICRO.COM to see our complete product line or call toll free at 1-800-869-5095

CUSTOM MEMBRANE KEYBOARDS / SWITCHES

- 1 TO 2 WEEKS TURNAROUND
- VERY COMPETITIVE PRICING
- Ex.: (5) 4-switch keyboards for $395.00
- PCB backed switches
- Custom metal backplates/assemblies
- Electronic assemblies/graphic overlays
- Electronic file transfer capabilities

Picofab Inc.
4780B Blvd. Henri-Bourassa
Charlesbourg, Quebec, Canada G1H 3A7
Tel: (418) 622-5298 ● Fax: (418) 622-9996
Email: sales@picofab.net

Adapt easily to Motorola's new 9S12E!

Technological Arts

Order online!
Eval Package $133!
OEM Modules from $54
(quantity 100)

www.technologicalarts.com

PROBLEM SOLVED!

Custom Hardware
&
Software Design

- Rapid Prototypes
- Cost-effective Design
- Free Consultation and Quotes

MJS CONSULTING INC.
Mark J. Samarin M.S.E.E.
http://www.mjsconsulting.com
253-858-6844

Windows® CE Embedded Controllers

Windows & Microsoft Are Registered Trade Mark Of Microsoft Corp.

6.5" 640x480 VGA Color LCD Touch Screen

XScale-Plus

Windows CE.NET Visual Basic C#, and Embedded Visual C++

- Available Head-Less, Or With LCD Color Touch Screen Display
- We Make It Easy For You To Program And Easy to Interface I/O
- Visit Our Web Site For More Detailed Information & Pricing

Intel XScale 400 MHz

- Solid State FLASH Disk
- Serial Ports, Parallel I/O
- Analog I/O, Custom User I/O
- PCMCIA Slot, Compact FLASH
- USB, PS/2, Audio Out

RLC www.RLC.com
R.L.C. Enterprises, Inc.
Toll Free 1-888-RLC-TECH

I²C™/SMBus

- Bus Monitors
- Protocol Analyzers
- Host Adapters
- Multiplexers
- Battery Applications
- Software Tools

MCC
Micro Computer Control

I²C is a trademark of Philips Corporation

www.mcc-us.com

The Pocket Programmer$149.95



The portable programmer that uses the printer port instead of a internal card.

Easy to use software that programs E(E)prom, Flash & Dallas Ram. 25/27/28 & 29 series from 16K to 8 Megabit. Adapters available for 874X, 875X MCU's, Pic, Serial Eprom's, 40-Pin X 16, Proms and More.

Same Name, Address & Phone # for 19 Years... Isn't it Amazing ?

Intronics, Inc.
Box 13723 / 612 Newton St.
Edwardsville, KS 66113 Add $7.00 COD
Tel. (913) 422-2094 Add $6.00 Shipping

WWW.IN-KS.COM Visa / MC/Amex/Disc

ALL ELECTRONICS CORPORATION

Electronic and Electro-mechanical Devices, Parts and Supplies. Wall Transformers, Alarms, Fuses, Relays, Opto Electronics, Knobs, Video Accessories, Sirens, Solder Accessories, Motors, Heat Sinks, Terminal Strips, L.E.D.S., Displays, Fans, Solar Cells, Buzzers, Batteries, Magnets, Cameras, Panel Meters, Switches, Speakers, Peltier Devices, and much more....

www.allelectronics.com
Free 96 page catalog
1-800-826-5432

Data Acquisition & Control
via RS232 or RS485 Ports

The ADR interfaces connect to any serial port and allow data acquisition and control using ASCII strings. Easy to use with BASIC, Visual Basic, C, TestPoint or Labview



ADR 2100 - $199

Many Versions are available with **A/D, D/A, DIO** and **RELAY I/O.**

Visit our web site for complete product list, pricing, applications, and programming tutorials.

FEATURES - 32 digital I/O (20mA); 4, 10-bit analog inputs; 2, PWM ports; 2 event counters; 4 interrupts; 1 auxillary output; Dual stepper motor indexer; RS232 to RS485 converter; Daisy-chain up to 10 units; Regulated 5VDC out@100mA

ONTRAK CONTROL SYSTEMS
PH: (705) 671-2652
Fax: (705) 671-6127
www.ontrak.net

PLCs for OEMs
Program like a PLC (Ladder + BASIC)
Low Cost & Flexible like a Micro-p

16 to 192 Digital i/o
4 to 8 Analog i/o
10A PWMs @ 24V
LCD Port
Keypad/Expansion
Real Time Clock
Stepper pulsers
Interrupts
Pulse Measures
MODBUS Ports
RS232 + RS485



$199 Qty 1

Incredibly Easy to Program!
Our software is used by many colleges for teaching PLCs!

Get Free Ladder Logic Simulator:
www.tri-plc.com/cci.htm

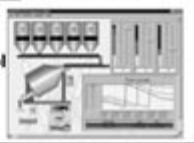
Toll Free: 1-877-TRI-PLCS (1-877-874-7527)

Internet Programmable!
Program / Control your PLC anywhere via the Internet using your web browser!

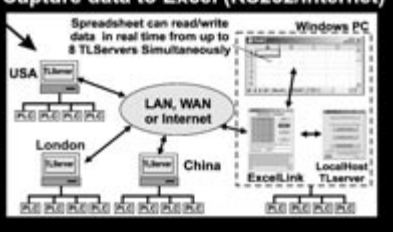


KEYPAD (16 keys) **+ LCD** (4x20) **$119** (Qty 1)

SCADA / HMI
MODBUS Protocol
Master or Slave
(ASCII & RTU)



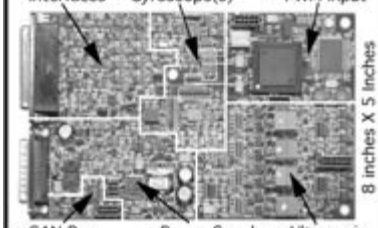
Capture data to Excel (RS232/Internet)



A Single Board Solution For Robot Navigation and Motor Control

NavDrive

- Motor Control Interfaces
- 3 Axis Compass
- Accelerometer(s)
- Gyroscope(s)
- MPC555
- 4 MB Ram
- PWM Input



8 inches X 5 inches

- CAN Bus
- RS-232 Host Interface
- GPS Interface
- Power Supply 8V to 24V In 370mA@12V
- RAM Backup
- Ultrasonic Driver and Receiver System
- C++ Control Framework using GNU Tools
- Debug, Control, and Sensing Applications

Integrated Knowledge Systems
www.iknowsystems.com

IDA Pro Disassembler



The Best tool to Reverse Engineer

From Atmel to Zilog, hundreds of µProcessors and µControllers supported. Older Unlimited Freeware version available for 80x86 and z80. Free Evaluation Version. The PC de-facto standard.

PC Magazine 2001 Technical Excellence Awards Finalist

www.datarescue.com

Give your PC a high speed CAN connection

CAN-4-USB

(Full PC source code included)

If we can provide all the HW and SW for a 4 wheel drive, WALKING, hydraulic vehicle



Photo Credit DRES

Imagine what we can (CAN?) build for you!

Zanthic Technologies Inc.
Phone 403-526-8318
email: zanthic@zanthic.com
web: www.zanthic.com

NOW YOU CAN HAVE IT ALL!

- *Easy to Use*
- *Low Cost*
- *Low Power*
- *Tiny Size*
- *High Speed*
- *DOS Environment*

with the



DOS Stamp™

Easy Software Development: Use your C/C++ or Basic compiler to produce DOS EXE. Download EXE to flash disk via serial port. EXE runs on power up.

Standard Features: BIOS & DOS-ROM, 128K flash disk, 512K SRAM, 40 MHz CPU, 16 digital I/O (opto rack interface), 2 RS-232, 2 timer/counters, simple bus interface, real-time clock

Options: 8-ch 12-bit ADC, flash disks up to 288 MB, 1 RS-485, TCP/IP

Tiny Size, Low Power: 2"x2.6", 5V @ 200 mA at full speed, 40 mA at lowest speed.

Visit http://www.bagotronix.com for info, prices, and FAQs

BAGOTRONIX
2900-1 Crescent Drive
Tallahassee, FL 32301
850-942-7905 phone & fax

NAND FLASH FILE SYSTEM

- Wear Leveling, Garbage Collection
- Power-Fail Safe
- ECC and Bad Block Handling
- File System with Std C Lib API
- FAT File Structure
- Page Caching to Maximize Performance and Minimize RAM
- 20KB Code Size
- RAM Required < 1% Flash Size

www.smxrtos.com/ffscc.htm

Micro Digital Inc
RTOS Innovators

800.366.2491 sales@smxinfo.com

WWW.CONITEC.NET

$333 COMPLETE KIT

GALEP-4 UNIVERSAL PROGRAMMER

Introducing a pocket programmer with true **Universal Output**



Latest generation Galep-4 uses ASIC universal pin technology for each pin of 40 pin ZIF-socket. **5200+ device library** / lifetime free updates. Programs 8/16-bit EPROM'S, EEPROM's, 0-Power RAM, FLASH, Serial EEPROM's, GAL, PALCE, microcontrollers such as 87/89xxx, PIC, AVR, ST62, etc. Low voltage devices down to 1.3V. No adapter required for DIL devices. 8 Hrs. operation on battery (AC charger included). Runs **WIN 98,NT,ME, 2000,XP** with Hex/Fuse Editor. Remote control from other apps, (e.g. VisualBasic). Substitutes high priced universal programmers e.g. ALL-11 (HILO) or LAB-TOOL-48 (ADVANTECH) Providing virtually matching performance at only 1/3-1/5 the price. Info, orders, softw : **619-702-4420**

ISO 9001 CERTIFIED PROFESSIONAL TOOLS

info, downloads: www.conitec.net

CONITEC DATASYSTEMS - 1951 4TH AVE, SUITE 301 - SAN DIEGO, CA 92101 - TEL: 619-702-4420 FAX: 619-702-4419

NEW

RS232 to TCP/IP

- *TCP/com*™ v2.0, RS232 to TCP/IP software. Plus TCP/IP to RS232.
- *WinWedge*™, RS232 or TCP/IP data direct into any Windows app. - Excel, Access, etc.

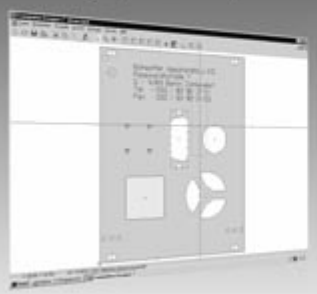
TALtech

Free 30 day evals at
www.taltech.com

Front Panels?

Unrivaled in price and quality for small orders

Download the free ›Front Panel Designer‹ to design your front panels in minutes





Order your front panels online and receive them just in time

www.frontpanelexpress.com

Order online at:
www.melabs.com

microEngineering Labs, Inc.

Development Tools for PIC Microcontrollers

Phone: (719) 520-5323
Fax: (719) 520-1867
Box 60039
Colorado Springs, CO 80960

Serial Programmer for PICs

$119.95



Includes:
Programmer
9-pin Serial Cable
AC Power Adapter
ZIF Adapter for 8 to 40-pin DIP
Software for Windows 98/Me/NT/2K/XP



Low-Cost EPIC Programmer Only $59.95

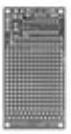
LAB-X Experimenter Boards



Pre-Assembled Board
Available for 8, 14, 18, 28, and 40-pin PIC MCUs
2-line, 20-char LCD Module
9-pin Serial Port
Sample Programs
Full Schematic Diagram

Pricing from $69.95 to $349.95

PICProto Prototyping Boards



Double-Sided with Plate-Thru Holes
Circuitry for Power Supply and Clock
Large Prototype Area
Boards Available for most PIC MCUs
Documentation and Schematic

Pricing from $8.95 to $19.95

BASIC Compilers for PIC MCUs



Easy-To-Use BASIC Commands
Windows 9x/Me/2K/XP Interface

PicBasic Compiler $99.95
BASIC Stamp 1 Compatible
Supports most 14-bit Core PICs
Built-In Serial Comm Commands

PicBasic Pro Compiler $249.95
Supports all PICmicro MCUs
Direct Access to Internal Registers
Supports In-Line Assembly Language
Interrupts in PicBasic and Assembly
Built-In for USB, I2C, RS-232 and More
Source Level Debugging

See our full range of products, including Books, Accessories, and Components at:
www.melabs.com

PRINTED CIRCUIT BOARDS

QUALITY PRODUCT
FAST DELIVERY
COMPETITIVE PRICING

PROTOTYPE through PRODUCTION

- UL approved
- Single & Double sided
- SMOBC
- LPI mask
- Through hole or SMT
- Nickel & Gold Plating
- Routing or scoring
- Electrical Testing
- Artwork or CAD data
- Fast Quotes
- Flex Circuits

SPECIAL OFFER:
10 pcs (3days) 1 or 2 layers $249
10 pcs (5days) 4 layers $695
(up to 30sq. in.ea.) includes tooling, artwork, L.P.I. mask & legend

PULSAR, INC

9901 W. Pacific Ave. Franklin Park, IL 60131 • Phone 847.233.0012
Fax 847.233.0013 • Modem 847.233.0014 • www.pulsar-inc.com • yogii@flash.net

CCS C Compiler
for PIC® Microcontrollers

Command line C Compiler start at just $125

Software Development Kits
Include full IDE Compiler & In-Circuit Debugger/Programmer

Embedded C Language Development Kit
w/ PCW 12 & 14-bit Compiler----------$424
Board Options:16F877A or 12F675
w/ PCWH 12, 14-bit & PIC18 Compiler-----$499
Board Options: 18F452, 18F6720 or 18F8720
CAN Bus Development Kit w/ PCWH--------$549

Use code CCIB2 for $10 discount

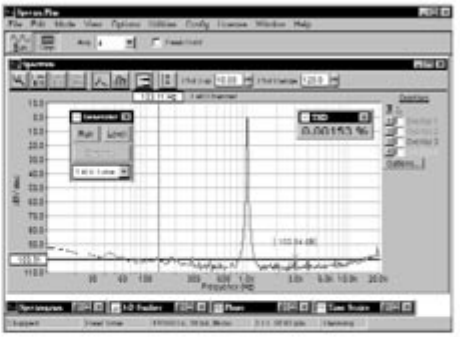
www.ccsinfo.com/cc
262-797-0455 sales x 35

PIC® and PICmicro® are registered trademarks of Microchip Technologies Inc. in the USA and other countries

SpectraPLUS - Real-Time FFT Spectral Analysis

Features
- 20 kHz Real-Time Bandwidth
- Signal Generation
- Dual Channel Analysis
- Distortion Measurements
- Octave Analysis
- Triggering, Decimation
- Transfer Functions, Coherence
- Time Series, Spectrum, Phase, and 3-D Surface plots

Applications
- Frequency Response Testing
- Audio Circuit Testing
- Signal Processing



Pioneer Hill Software
24460 Mason Rd.
Poulsbo, WA 98370
a subsidiary of Sound Technology, Inc.

DOWNLOAD FREE 30 DAY TRIAL!
www.spectraplus.com
from $295

Sales: (360) 697-3472 *Fax: (360) 697-7717* *e-mail: pioneer@telebyte.com*

PIC Programmers



Super Value!
from $16.95 +S&H $4.95*
The P16PRO: can program up to 40 pin PICs including the popular **16F84** & **12C508** ● Needs software (extra $20) ● Available assembled or starting from **$16.95** for the kit. See **www.electronics123.com** for more info!
The **PICALL** programmer can also progam **Atmel AVRs** in addition to the PICs it can program ● Free software ● PICALL programmer kit at **$69.95** See **www.electronics123.com** for more info!

Digital Camera module



1/4" CMOS Camera Module, black & white, with digital output. Size: only 1.38"x1.14". Lens: f7.4mm, F2.1 FOV 32 deg x 24 deg. Pixels: 384x288. Scanning: progressive. Video output: digital 8bit. Programmable features. Shutter: 1/50 to 1/5000 sec. Power Supply, 5VDC. Standby Current <10mA. Needs regulated power supply. *S&H to Canada is $7.95

$57.95 +S&H $4.95* Code: CM4881

Toll Free: 1-888-549-3749 (USA & Canada)
Tel: (330) 549-3726. Request a FREE catalog or visit us at: **www.electronics123.com** for more.
Amazon Electronics, Box 21 Columbiana OH 44408

Weeder Technologies

www.weedtech.com DATA SHEETS

RS-232 Stackable

1710-B Brighton Cove, Ft Walton Beach, FL 32547 850-863-5723



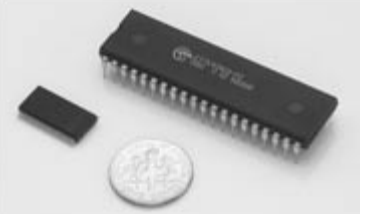
Digital I/O • Analog Input • Analog Output • Solid State Relays • Stepper Motor Controllers • Pulse Counters • Pulse Timers • Frequency Measurement • RPM Measurement • Duty Cycle Measurement • Period Measurement • Projectile Velocity Measurement • Third-Party Peripheral Interface • Data Logging

Plug end-to-end
Stack 32 modules on the same RS-232 cable.

Download Windows software

Data Acquisition & Control
...simple as a laptop.

128 KEY MATRIX IN TTL SERIAL OUT



- 8x16 matrix for up to 128 keys/buttons.
- 2400 bps TTL serial data output.
- Debounce and ghost key elimination.
- Crystal is **only** external component!
- Used from USA to England to Argentina.
- **$6** (qty 100), **$4** (qty 1K), **$3** (qty 10K).
(prices subject to change without notice)

www.motionencoders.com

INDEX OF ADVERTISERS

The Index of Advertisers with links to their web sites is located at www.circuitcellar.com under the current issue.

Page		Page		Page		Page	
88	Abacom Technologies	31	Earth Computer Technologies	26	MaxStream	31	Remote Processing
86	ActiveWire, Inc.	87	EE Tools (Electronic Engineering Tools)	90	MCC (Micro Computer Control)	23	Renesas Technology Corp.
21	AeroComm, Inc.	61	EMAC, Inc.	92	Micro Digtal Inc.	90	RLC Enterprises, Inc.
89	Akida LLC	89	emBoot, Inc.	92	microEngineering Labs, Inc.	28	Rogue Robotics Corp.
91	All Electronics Corp.	16	ExpressPCB	90	MJS Consulting	60	Saelig Co. Inc.
93	Amazon Electronics	84	FDI-Future Designs, Inc.	79	Mouser Electronics, Inc.	88	Scidyne
86	AP Circuits	92	Front Panel Express	93	Motion Encoders	3	Scott Edwards Electronics Inc.
82	Arcom Controls	85	Gamatronix	86	Mosaic Industries, Inc.	88	Sealevel Systems
87	AshWare, Inc.	87	Grid Connect	81	MVS	34	Sensors Expo & Conf.
7	Atmel	85	Hagstrom Electronics	86	Mylydia Inc.	5	Sierra Proto Express
19	AVR 2004 Contest	95	HI-TECH Software, LLC	C2	NetBurner	89	Signum Systems
91	Bagotronix, Inc.	13	Holmate Semiconductor	84	NPE, Inc.	29	Silicon Laboratories, Inc.
90	Basic Micro	55	ICOP Technology Inc.	88	OKW Electronics, Inc.	85	Softools
84	BayTech Sales	63	Imagine Tools	91	Ontrak Control Systems	92	TAL Technologies
67	Bellin Dynamic Systems, Inc.	89	IMAGEcraft	50	PCBpro	C3	Tech Tools
40	CadSoft Computer, Inc.	86	Intec Automation, Inc	8	PCB123	72, 73	Technologic Systems
89	Carl's Electronics	91	Integrated Knowledge Systems	42	PCBexpress	90	Technological Arts
93	CCS-Custom Computer Services	85	Intrepid Control Systems	87	PCB Fab Express	89	Tern Inc.
92	Conitec	91	Intronics, Inc.	C4	Parallax, Inc.	85	Trace Systems, Inc.
84	Cyberpak Co.	71	Jameco	84	Phytec America LLC	91	Triangle Research Int'l, Inc.
1	Cypress MicroSystems	32, 86	JK microsystems, Inc.	87	Phyton, Inc.	61	Trilogy Design
33	CWAV	85	JPA Consulting	90	Picofab Inc.	67	Vantec
28	Dallas Logic Corp.	67	JR Kerr Automation & Engineering	93	Pioneer Hill Software	93	Weeder Technologies
91	DataRescue	83	LabJack Corp.	93	Pulsar, Inc.	86	Zagros Robotics
84	Decade Engineering	83	Lakeview Research	88	Quality Kits & Devices	91	Zanthic Technologies Inc.
83	DLP Design	51	Lemos International	88	R2 Controls	86	Zexus Technologies Ltd.
9	DPAC	2	Link Instruments	57	R4 Systems Inc.	47	Zilog
56	Dynon Instruments, Inc.	17, 42	Linx Technologies	11, 41	Rabbit Semiconductor		

Preview of May Issue 166
Theme: Communications

Programmable IR Receiver for PCs

USI-Based I²C Slave

ZRT Real-Time Operating System

Simple Bluetooth Integration (Part 1): Implementing Bluetooth Modules

Get Moving with the MC34921FTA Power System Control IC

APPLIED PCs Radio Roundup

FROM THE BENCH USB in Embedded Design (Part 2): HIDmaker Converts an Application

SILICON UPDATE The Heat is On

ATTENTION ADVERTISERS

June Issue 167 Deadlines

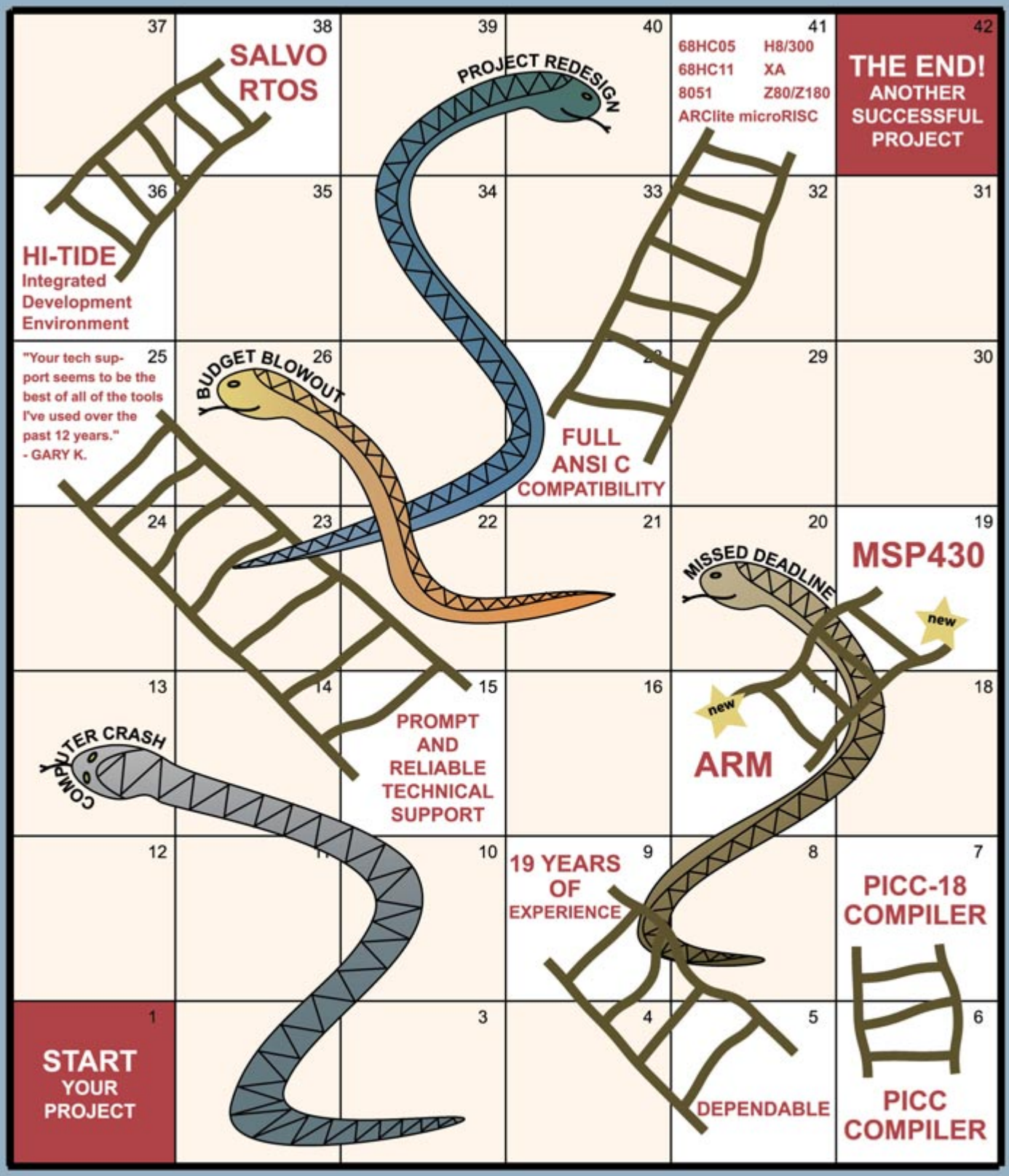
Space Close: April 9

Material Due Date: April 19

Theme:
Measurement & Sensors

BONUS DISTRIBUTION
Sensors Expo

Call Sean Donnelly now to reserve your space!
860.872.3064
e-mail: sean@circuitcellar.com



EMBEDDED SOFTWARE DEVELOPMENT HAS ITS UPS AND DOWNS. CONQUER OBSTACLES WITH OUR FIELD-PROVEN ANSI C COMPILERS AND RTOS. HI-TECH SOFTWARE HELPS YOU CLIMB THE SUCCESS LADDER!

6600 Silacci Way
Gilroy CA 95020
ph: 800 735 5715
fax: 866 898 8329
email: sales@htsoft.com
web: www.htsoft.com

HI-TECH SOFTWARE



PRIORITY INTERRUPT

by Steve Ciarcia, Founder and Editorial Director

Beach Days

Editorials aren't usually two or three parts, so let's just call this a continuing saga. When I left you last month, I had just finished describing how Don, my alarm system professional, had succeeded in totally destroying all concept of home automation in the Circuit Cellar. In the process of installing a new commercial alarm system, he succeeded in ripping out all the alarm-to-HCS connections throughout my house (from motion detectors and sensors).

The bad news was that for the first time in 24 years I had manual lighting again. That isn't a problem for most people, but around my house it means you bump into lots of things in the dark. It has been so long since I've had to use the light switches that I don't even know where most of them are, and my unconventional floor plan doesn't give any hints about where to look. Worse yet, in the course of 20-plus years of renovations and additions, I implemented a few lights with no manual switches at all. (They're X10 or SSR-controlled from the HCS only.) Basically, there is no such thing as "manual" in my house. I haven't solved this problem. And dreadfully, Don hasn't finished yet either.

The day I wrote last month's editorial, I was preparing to leave on a trip at 4 a.m. the following morning. At 10 p.m. that evening, while Don was still doing his thing, I demanded a ceasefire. Six hours later I entered a four-digit activation code and started on an 1100-mile drive down the coast. The motivation to take the trip was that I could combine business and pleasure. A magazine run by professional people doesn't require supervision. I spend most of my day e-mailing people down the hall or from home as it is. I figured I could manage contests, review editorial stuff, and do virtually everything else I normally do while sitting on a beach far away from snowbound Connecticut.

My fantasy was prompted by thoughts that I could telecommute. At one time people actually went to an office, but these days everyone in Silicon Valley stays home and meets in cyberspace, right? If it's nothing more than using a laptop on the road and sending e-mails back and forth, then I should be able to pull that off for a month, right? Well, after trying it for the last four weeks, I have concluded that that concept is still worthwhile, but the execution has a few bugs. In my experience, successful telecommuting is entirely dependent upon your Internet connection and the intelligence of the IT guy back at the office.

My first rude awakening was that vacation islands in Florida are definitely not Internet communication hubs. After being confronted by one of the neighbors, I explained that I wasn't some weirdo stalking the neighborhood. I was just out sniffing Wi-Fi (which sounded just as bad, I'm sure). It was a nice beach but it was a black hole for communications. There were no cable modems and no DSL service, and the nearest Starbucks was 10 miles away. Telecommuting meant using a 56K dial-up to a single available local AOL line. If that line was out or busy, the only alternative was 28K at $8 per hour on an 800 line. Given my current experience, I was lucky even having 56K.

Reality struck the second day after my arrival. The Atmel AVR contest had just started and I received over 1000 sample requests in just three days. Worse yet, they were all e-mailed to me! I couldn't use the automatic e-mail-to-Excel program on my office PC that would have filtered and sorted them in 2 minutes. Instead, I had to spend six days downloading all the messages and individually pasting them into the spreadsheet. After that, it seemed like every business correspondence for the rest of the month involved downloading 20-meg files. It was never a case of quickly reading e-mail and switching offline. It was always a question of how many hours before I could do anything else but send or receive correspondence.

Probably the greatest revelation was the importance of something that many of us take for granted: e-mail archives. When I use Outlook at the office, everything ends up in folders and I don't have to worry about losing what I've sent or received. When I'm outside and not using Outlook, it can be a very different story. I thank heavens that John Gorsky had the foresight to switch our e-mail system from a POP server to an IMAP server recently. While I'm not entirely clear on the details, all I know is that I was able to keep all of the correspondence for the whole month and then easily synchronize everything on my office PC when I returned. The older system would have been a nightmare.

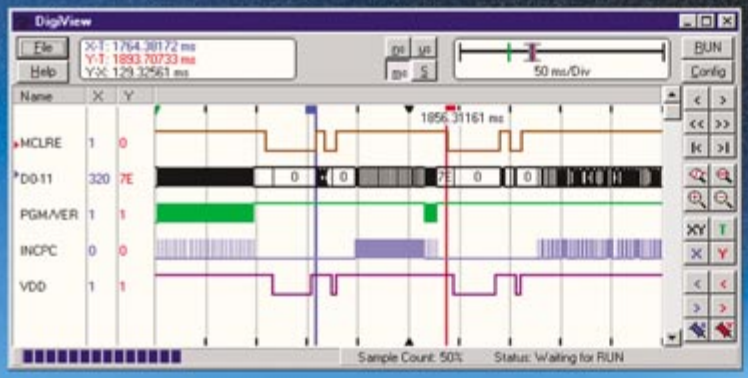
Certainly, companies that handle a lot of telecommuting have elaborate systems with many more features. My experience was merely a toe in the water. With the exception of the connection bandwidth, I think it has potential. Before I plant myself on another pile of sand in Florida, I definitely need to solve the bandwidth issue. I installed XM Radio so I wouldn't have to surf radio stations while driving. I know there are satellite connections for the laptop too. If you have any firsthand experience using this, I'd appreciate hearing the good, the bad, and the ugly about doing it. Forewarned is forearmed.

steve.ciarcia@circuitcellar.com



Portable and Affordable Logic Analyzer

uses real-time Hardware Compression



- Save up to 5 minutes of Data @ 10ns.
- Trigger on rising, falling, or transitions.
- Trigger on Pattern Matches.
- Real-time Hardware Compression.
- USB Powered for Portability.
- Data Transfered in 1 second.
- 18 Channels
- 100 MHz

Software Compatible with:
WIN 98SE, ME, NT, 2K, XP, XPpro

DigiView

100 MHz, 18 Channel, Billions of Samples @10ns!



only $499.00



ROM TOOLS





See Our Other Embedded Development Tools



TechTools

www.tech-tools.com

(972) 272-9392 • sales@tech-tools.com

PICmicro® MCU TOOLS

Copyright © 2003 TechTools • DigiView, FlexROM, EconoROM, QuickWriter TDE and ClearView are trademarks of TechTools • PICmicro is a registered trademark of Microchip Technology Inc.

ROBOT ROUNDUP!

The **BASIC Stamp 2** is the cornerstone for all Parallax robots. Ease of programmability, high quality manuals, and robust robotics hardware are attributes of each robot we offer. The 2004 Roundup gives you a glimpse of our 4 complete kits. The **Boe-Bot** and **Toddler** (also available in gold) are primed for educational purposes since the **Stamps in Class (SIC)** text is included with each kit. The SIC texts include step-by-step instructions, activities, challenges, and highly developed source code programs.

Fun to build and easy to program!

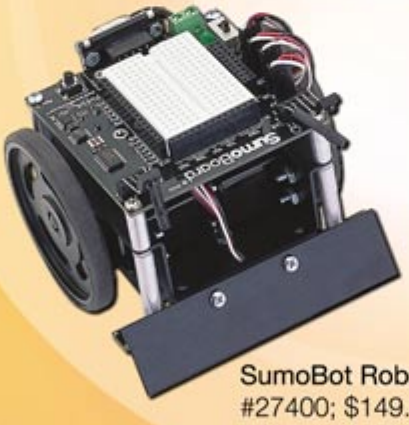


Boe-Bot Robot Kit;
#28132; $229.00

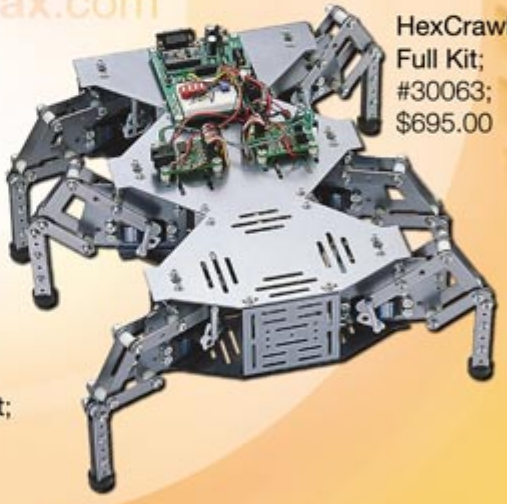


Blue
Toddler
Robot Kit;
#27311;
$249.00

www.parallax.com



SumoBot Robot Kit;
#27400; $149.00



HexCrawler
Full Kit;
#30063;
$695.00

The **SumoBot** and **HexCrawler** are designed for the roboticist. The texts are crafted to walk you through the assembly process and entry level source code programs. These 2 are ideal for customization based on your particular interests, whether it's a Mini Sumo Competition or if you want the most deluxe 6-legged robot with multiple sensors. Visit the **ROBOTICS** section at **www.parallax.com** for a comparison chart and a robot video gallery with all 4 in action. While you're online, take a look at the wide variety of sensors and add-on accessories to upgrade your robot.

Visit **www.parallax.com** to order online 24 hours a day.
Or place your order by calling our sales department
toll-free at **888-512-1024** (Mon-Fri, 7am - 5pm, PST).

PARALLAX inc

BASIC Stamp and the Parallax logo are all registered trademarks of Parallax, Inc.