Build an Aircraft Data Logging System, p. 26 | Portable Power: A DIY Power Adapter, p. 58 | Thermal Energy Production, p. 66

CIRCUIT CELLAR

THE MAGAZINE FOR COMPUTER APPLICATIONS

November 2010
Issue 244

ANALOG TECHNIQUES

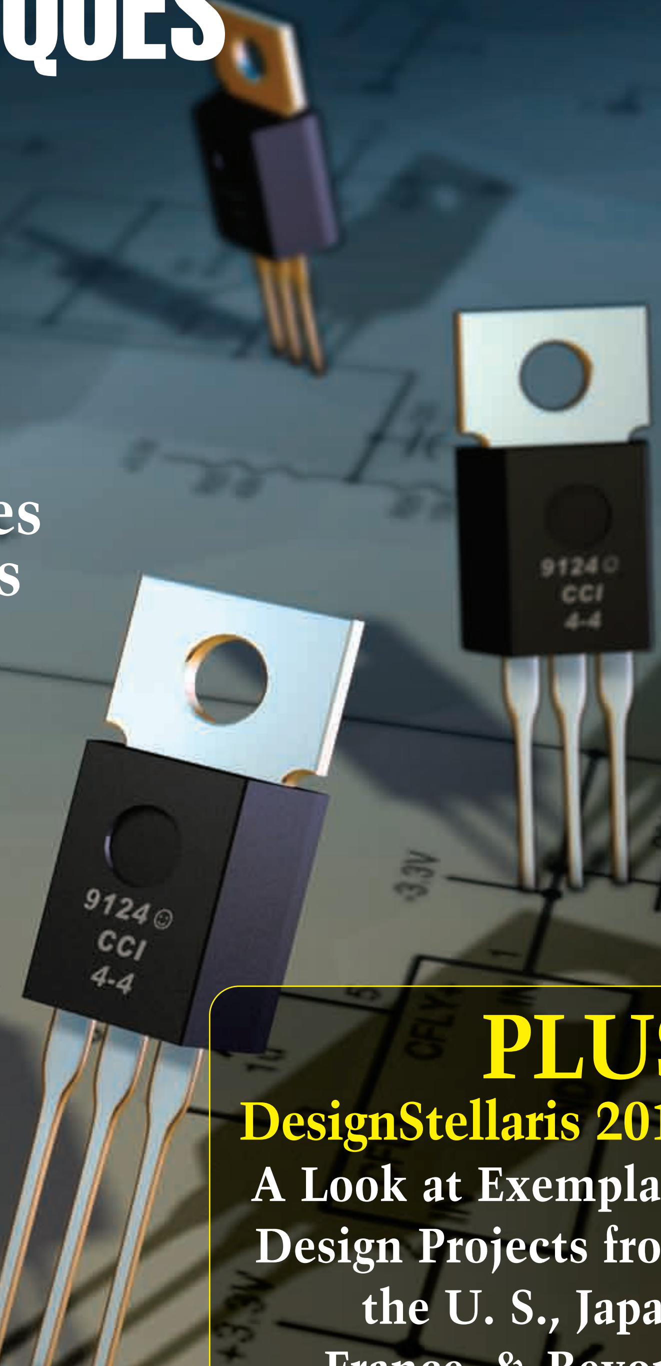
A PVDF Phased-Array Analog Front End

Sound Generation with Three Basic Chips

Solve Connectivity Issues Between Embedded Apps

Digital Audio Spectrum Analyzer

Professional Grounding Rules



PLUS
DesignStellaris 2010
A Look at Exemplary Design Projects from the U. S., Japan, France, & Beyond

$7.50 U.S. ($8.50 Canada)

www.circuitcellar.com

Low-cost Industrial
Serial to **Ethernet** Solutions

NetBurner
Networking in One Day!

- Instantly network-enable any serial device
- No programming is required for serial to Ethernet application
- Customize to suit any application with a development kit

SBL2e Chip

2-port serial-to-Ethernet server with eight A/D converter inputs and optional SPI, I2C, CAN peripheral device support

SBL2e 200

2-port serial-to-Ethernet server with four A/D converter inputs, optional I2C peripheral support and 10-pin header

SBL2e 100

2-port serial-to-Ethernet server with four A/D converter inputs, optional I2C peripheral support and RJ45 connector

SBL2e X

External 2-port serial-to-Ethernet server with RS-232 support

SBL2e XA

External server with up to four A/D converter inputs, up to eight digital I/O and up to two UARTS (one RS-232)

Hardware Features

Up to three serial ports, 10/100 Mbps Ethernet, up to 10 digital I/O, 12-bit A/D converters, operating temperature -40 to 85 C, 32-bit performance

Software Features

TCP/UDP/Telnet/HTTP modes, DHCP/Static IP support, web-based or AT command configuration

Low Prices

SBL2e Chip .... $10.50 (Qty. 10k, Device P/N: SBL2eCHIP-236IR)
SBL2e 200 .... $19.95 (Qty. 1000, Device P/N: SBL2e-200IR)
SBL2e 100 .... $21.95 (Qty. 1000, Device P/N: SBL2e-100IR)
SBL2e X........................... $79.00 (Qty. 1000, Device P/N: SBL2eX-100IR)
SBL2e XA........................ $79.00 (Qty. 1000, Device P/N: SBL2eXA-100IR)

Need a custom solution?

NetBurner Serial to Ethernet Development Kits are available to customize any aspect of operation including web pages, data filtering, or custom network applications. All kits include platform hardware, ANSI C/C++ compiler, TCP/IP stack, web server, e-mail protocols, RTOS, and NB Eclipse IDE.

Information and Sales | sales@netburner.com

Web | www.netburner.com **Telephone** | 1-800-695-6828

TASK MANAGER

Radical Empirical Testing

A reader recently told me his fascination with electronics began in childhood when he was shocked while playing with an electrical socket. Now that's one way to convince yourself that electricity exists! *Nihil in intellectu nisi prius in sensu*,[1] right? Obviously, the reader lived to tell the story, become a professional engineer, and subscribe to *Circuit Cellar*. What's truly interesting is that he wasn't the first engineer to tell me that he first "discovered" electricity through his senses—that is, when it coursed through his body. I've heard a few similar stories during my years here at *Circuit Cellar*. Several readers literally jolted themselves into the field of electronics while sticking all sorts of objects (e.g., paper clips and forks) into sockets or messing with older relatives' electronics.

Here in the Editorial Department, we're well aware of the advantages of empirical research methods, and we definitely advocate the notion that thorough testing and measurement can lead to insight, truth, and innovation in the various sciences and fields of engineering. But please, safety first. As a successful adult designer/programmer, you need not involve all of your senses—like touch and taste—when testing electronics at your workbench. *Or do you*? Due to some recent groundbreaking engineering developments, eating chips for testing purposes might become the norm in the not so distant future for designers working in the bioengineering field. In fact, one inventor has done just that. Proteus Biomedical's CEO and cofounder Andrew Thompson claims to have tasted his Redwood City-based company's digestible chips. In a September 2010 *Bloomberg* article about the technology, Olga Kharif detailed Proteus Biomedical's goal of putting microchips in prescription pills. According to Kharif, the technology involves "chips that can be embedded in pills, turning them into networked, digital drugs."[2] The aim is to alert patients when they miss doses of specific drugs.

The idea of inserting chips in pills isn't new, but the fact that the technology is in the clinical trial phase has the biomedical community buzzing. But what do you think? Will you be inserting chips in medicines, foods, and digestible medical devices?

This month we feature articles about projects—such as an aircraft data logger (p. 26), a digital audio spectrum analyzer (p. 38), and a sound generator (p. 46)—that didn't require testing methods such as absorbing low voltages or tasting chips. But the designs are still innovative. I encourage you to try your hand at a few of the projects. Perhaps the learning process will be your first step toward developing a marketable MCU-based invention of your own. Whether or not that invention will involve radical empirical testing remains to be seen. But be sure to let us know!

---

[1] Nothing in the intellect unless first in the senses (Latin)
[2] O. Kharif, "Innovator: Andrew Thompson," *Bloomberg Businessweek*, September 16, 2010.

cj@circuitcellar.com

C. Abate

CIRCUIT CELLAR®
THE MAGAZINE FOR COMPUTER APPLICATIONS

FOUNDER/EDITORIAL DIRECTOR
Steve Ciarcia

EDITOR-IN-CHIEF
C. J. Abate

WEST COAST EDITOR
Tom Cantrell

CONTRIBUTING EDITORS
Jeff Bachiochi
Robert Lacoste
George Martin
Ed Nisley

NEW PRODUCTS EDITOR
John Gorsky

PROJECT EDITORS
Ken Davidson
David Tweed

PUBLISHER
Hugo Van haecke

ASSOCIATE PUBLISHER
Shannon Barraclough

CUSTOMER SERVICE
Debbie Lavoie

CONTROLLER
Jeff Yanco

ART DIRECTOR
KC Prescott

GRAPHIC DESIGNERS
Grace Chen
Carey Penney

STAFF ENGINEER
John Gorsky

Circuit Cellar's mission is to collect, select, and disseminate need-to-know information around the world in the fields of embedded hardware, embedded software, and computer applications. Circuit Cellar uses an assortment of print and electronic content-delivery platforms to reach a diverse international readership of professionals, academics, and electronics specialists who work with embedded, MCU-related technologies on a regular basis. Our aim is to help each reader become a well-rounded, multidisciplinary practitioner who can confidently bring innovative, cutting-edge engineering ideas to bear on any number of relevant tasks, problems, and technologies.

ADVERTISING
800.454.3741 • 978.281.7708 • www.circuitcellar.com/advertise

ADVERTISING REPRESENTATIVE
Peter Wostrel
Strategic Media Marketing, Inc.
1187 Washington St., Gloucester, MA 01930 USA
800.454.3741 • 978.281.7708
peter@smmarketing.us • www.smmarketing.us
Fax: 978.281.7706

ADVERTISING COORDINATOR
Valerie Luster
E-mail: val.luster@circuitcellar.com

Cover photography by Chris Rakoczy—Rakoczy Photography
www.rakoczyphoto.com
PRINTED IN THE UNITED STATES

CONTACTS

SUBSCRIPTIONS
Information: www.cc-access.com, E-mail: subscribe@circuitcellar.com
Subscribe: 800.269.6301, www.cc-access.com, Circuit Cellar Subscriptions, P.O. Box 5650, Hanover, NH 03755-5650
Address Changes/Problems: E-mail: subscribe@circuitcellar.com

GENERAL INFORMATION
860.875.2199, Fax: 860.871.0411, E-mail: info@circuitcellar.com
Editorial Office: Editor, Circuit Cellar, 4 Park St., Vernon, CT 06066, E-mail: editor@circuitcellar.com
New Products: New Products, Circuit Cellar, 4 Park St., Vernon, CT 06066, E-mail: newproducts@circuitcellar.com

AUTHORIZED REPRINTS INFORMATION
860.875.2199, E-mail: reprints@circuitcellar.com

AUTHORS
Authors' e-mail addresses (when available) are included at the end of each article.

CIRCUIT CELLAR®, THE MAGAZINE FOR COMPUTER APPLICATIONS (ISSN 1528-0608) is published monthly by Circuit Cellar Incorporated, 4 Park Street, Vernon, CT 06066. Periodical rates paid at Vernon, CT and additional offices. **One-year (12 issues) subscription rate USA and possessions $45, Canada/Mexico $60, all other countries $63. Two-year (24 issues) subscription rate USA and possessions $80, Canada/Mexico $110, all other countries $116.** 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.

Making Your Devices Smart & Simple!

Internet Connectivity Wizard

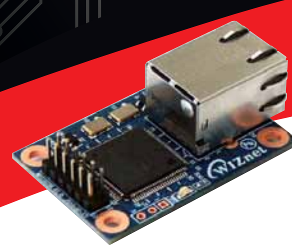
WIZnet's uniquely patented Hardware TCP/IP technology has proven effective in millions of network-enabled devices around the world, allowing users to easily add internet to the devices with greater stability to ensure faster product releases out to the market.

Hardware TCP/IP Offload Technology

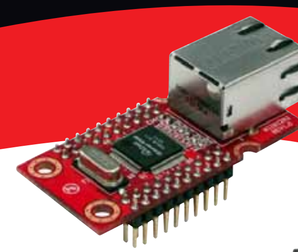
- Offloading Protocol Processing from a System MCU
- Stable Hardwired TCP/IP Logic
- Silicon-proven IP for SoC and ASSP
- High Performance & Low Cost Platform
- No Booting Time
- Simple System Composition
- Best-fits OS-less Device

Applications

- Smart Grid and Smart Meter
- Medical Device Management
- Industrial Control
- Robot Control
- Security System



Serial-to-Ethernet Module



Drop-in Network Module

Wireless LAN Module

External Device Server

Hardware TCP/IP Chip

Get more information at www.wiznettechnology.com

3003 North First Street, San Jose, CA 408-232-5415
sales@wiznettechnology.com

WIZnet TECHNOLOGY

INSIDE ISSUE 244

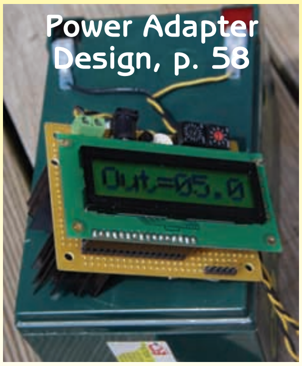
November 2010 • Analog Techniques

Texas Instruments DesignStellaris 2010 Results , p. 12

FIRST PRIZE
Bicycle ABS Brake System
Richard Wotiz
United States

SECOND PRIZE
Camera-Based Eye-Tracking System
Osamu Tamura
Japan

THIRD PRIZE
GPIB Screen Saver
Sylvain Davaine
France

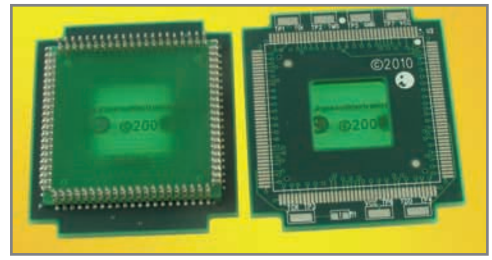



Get Your Hands On What's Next.
Scan
mouser.com
The Newest Products For Your Newest Designs™
MOUSER ELECTRONICS
a tti company
Mouser and Mouser Electronics are registered trademarks of Mouser Electronics, Inc. Other products, logos, and company names mentioned herein, may be trademarks of their respective owners.

XILINX DEVICE CONVERTER ALLOWS RELIABLE UPGRADES

Ironwood Electronics's new Xilinx device converter, the **DC-QFP/PLCC84-J-01**, allows upgraded system-programmable Xilinx CPLDs to be used in previous-generation system boards. These device converters can be soldered directly onto the SMT PLCC pads using standard solder methods.

The DC-QFP/PLCC84-J-01 consists of footprint and pinout conversion multilayer PCB with a true J-lead adapter for SMT processing. The J-leads are constructed with a precise lead frame assembly to accomplish the accurate PLCC package emulation. The DC-QFP/PLCC84-J-01 is designed to receive an XC95216 Xilinx CPLD on the top and converts pin mapping to an XC95108 Xilinx CPLD. This allows a system board of six 36V18 function blocks, providing 2,400 usable gates with propagation delays of 7.5 ns to be upgraded to 12 36V18 function blocks, providing 4,800 usable gates with propagation delays of 10 ns.



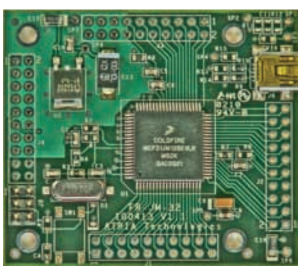
Pricing for the DC-QFP/PLCC84-J-01 is **$136** for single-piece quantities with reduced pricing available depending on the quantity required.

Ironwood Electronics
www.ironwoodelectronics.com

LOW-COST FREESCALE MICROCONTROLLER MODULES

ForeRunner microcontroller modules are based on the Freescale MC9S08 and MCF51 (ColdFire V1) families of microcontrollers. These modules are well designed, simple, support easy access to all the device functions, and provide complimentary 8- and 32-bit microcontrollers with features such as on-chip USB. Communications modules for Bluetooth, RS-232, and USB provide a variety of communications methods to experiment with.

Many of the microcontroller modules are preprogrammed with a BASIC operating system. You develop your project code using only a terminal emulation program such as HyperTerminal. Programs are saved to FLASH and an autorun feature enables your program to start automatically at power on.



Expensive development tools are not required. Free special-edition CodeWarrior development tools are available from Freescale for programming in C. Each microcontroller module provides a connection for a programmer/debugger.

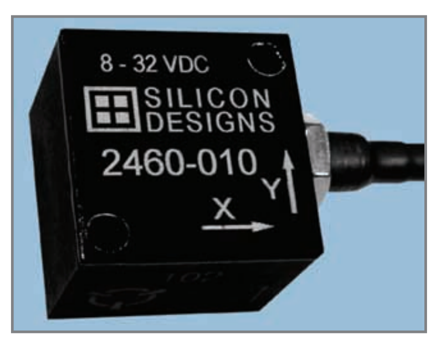
Schematics, datasheets, and downloads are available online. The ForeRunner microcontroller modules start at **$23**. Kits and accessories are also available.

ATRIA Technologies
www.atriatechnologies.com

UNIVERSAL TRIAXIAL CAPACITIVE ACCELEROMETER MODULES

The **SDI 2460 Series** is a new family of easy-to-use universal triaxial capacitive accelerometer modules. Available in eight different g-ranges, these lower-cost sensors are designed to provide high-precision shock, vibration, and acceleration measurement in three orthogonal directions across a broad range of automotive, aerospace, industrial, power generation, and test and measurement applications.

The SDI 2460 series combines three orthogonally mounted accelerometers in an epoxy-sealed anodized aluminum case and is the triaxial version of the 2260, the company's single-axis model. SDI's 100% in-house manufacturing and intense testing processes, along with the incorporation of additive micromachining and integrated circuit technology, produce a highly reliable and rugged capacitive sensor tailored for zero-to-medium frequency instrumentation applications. The innovative technology design—using on-board voltage regulation and an internal voltage reference—make the units relatively insensitive to temperature and voltage changes and eliminate the need for external power amplification and regulation. All modules are easily mounted to a test article or fixture via glue, epoxy, or two #8 or M4 screws.



Carefully regulated manufacturing processes ensure that each sensor is consistently made to be virtually identical, allowing users to swap out modules with little or no testing modifications, saving both time and resources. This also allows test engineers to provide a quick plug-and-play solution for almost any application with total trust in the accuracy of SDI sensors when used within published specifications.

Prices range from **$1,075** to **$946** per piece based on quantity.

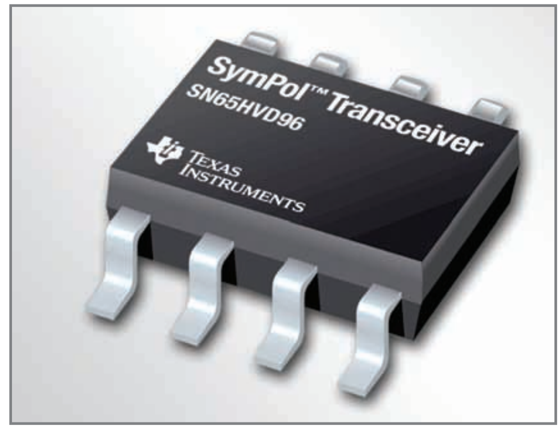
Silicon Designs, Inc.
www.SiliconDesigns.com

NEW PRODUCT NEWS
Edited by John Gorsky

CROSS-WIRE IMMUNITY TRANSCEIVER

The **SN65HVD96** is a new symmetric polarity transceiver that protects systems from communication losses and potential damage should signal wires be inadvertently reversed during installation or maintenance. The SN65HVD96, using the patent-pending SymPol technology, provides high bus fault protection, making it desirable for harsh industrial environments where third-party installers often make the connections. The inverted bus wires are detected internally and corrected automatically, eliminating the need for intervention by the controller or operator without the need for firmware changes.

Applications that can benefit from this technology include heating, ventilating, and air conditioning equipment, surveillance cameras, surveillance IP network cameras, building automation, industrial lighting, and other industrial applications. This device works well with isolation devices such as TI's ISO7241C. TI also provides IBIS models for the SN65HVD96, which designers can use for simulations.



Key features and benefits include reversed-wire immunity, which saves time should a crossed-wire mistake occur, and the same pinout as RS-485, which eliminates the need for board redesigns. The device also has bus pins that can survive faults of –35 to 40 V and a high-input impedance for up to 32 nodes.

The SN65HVD96 is available now in an SOIC (D) package with eight pins and costs **$1.20** in quantities of 1,000.

Texas Instruments, Inc.
www.ti.com

NPN

STATEMENT REQUIRED BY THE ACT OF AUGUST 12, 1970, TITLE 39, UNITED STATES CODE SHOWING THE OWNERSHIP, MANAGEMENT AND CIRCULATION OF CIRCUIT CELLAR, THE MAGAZINE FOR COMPUTER APPLICATIONS. Published monthly at 4 Park Street, Vernon, CT 06066. Annual subscription price is $45.00. Publisher: Hugo Van haecke. Editorial Director: Steven Ciarcia. Editor-In-Chief C.J. Abate. The owner is Circuit Cellar, Inc., Vernon, CT 06066. The names and addresses of stockholders holding one percent or more of the total amount of stock are: Elektor International Media, LLC, 4 Park Street, Vernon, CT 06066.
EXTENT AND NATURE OF CIRCULATION: Average number of copies of each issue published during the preceding twelve months; (A) total number of copies printed, 18,608; (B.1) paid/requested mail subscriptions, 6,329; (B.3) sales through dealers and carriers, street vendors and counter sales, 3,991; (B.4) paid/requested copies distributed by other mail classes, 9; (C) total paid/requested circulation, 10,329; (D.1) samples, complimentary, and other nonrequested copies,1,563; (D.4) Nonrequested copies distributed outside the mail, 1,923; (E) total nonrequested distribution (sum of D.1 & D.4), 3,486; (F) total distribution (sum of C & E), 13,815; (G) copies not distributed (office use, leftover, unaccounted, spoiled after printing, returns from news agents), 4,793; (H) total (sum of F & G), 18,608. Percent Paid Requested: 74% Actual number of copies of a single issue published nearest to filing date: (A) total number of copies printed, 22,905; (B.1) paid/requested mail subscriptions, 5,716; (B.3) sales through dealers and carriers, street vendors and counter sales, 4,176; (B.4) paid/requested copies distributed by other mail classes, 10; (C) total paid/requested circulation, 9,902; (D.1) samples, complimentary, and other nonrequested copies, 1,870; (D.4) Nonrequested copies distributed outside the mail,5,881; (E) total nonrequested distribution (sum of D.1 & D.4), 7,751; (F) total distribution (sum of C & E), 17,653; (G) copies not distributed (office use, leftover, unaccounted, spoiled after printing, returns from news agents), 5,252; (H) total (sum of F & G), 22,905. Percent Paid Requested 56%. I certify that the statements made by me above are correct and complete. Hugo Van haecke, Publisher.

Test Your EQ

Edited by David Tweed

Problem 1—*You have a system comprising three different components, and you know that the MTBFs (mean time between failures) of the components are 100,000 hours, 120,000 hours and 15,000 hours. If any one component fails, then the system is considered failed. How do you estimate the MTBF of the overall system?*

Problem 2—*What are the key simplifying assumptions that make the previous analysis possible?*

Problem 3—*Why is copper wire actually a terrible material to use for a current-measuring shunt?*

Problem 4—*Can you explain the mechanism that allows a series-connected string of decorative lights to remain lit, even when one is burned out?*

Contributed by David Tweed

What's your EQ?—The answers are posted at www.circuitcellar.com/eq/

You may contact the quizmasters at eq@circuitcellar.com

How far will your design take you?

Texas Instruments DesignStellaris 2010 Design Contest Winners

Nothing drives technological advancement in the embedded industry like healthy competition among brilliant engineers. The Texas Instruments DesignStellaris 2010 Design Contest is an excellent example of just how innovative today's top engineers can be when they're given amazing parts, a deadline, and some enticing incentives—such as international recognition and a chance to win a share of $10,000 in cash prizes.

Back in January 2010, engineers were challenged to use the Stellaris LM3S9B96 microcontroller from Texas Instruments with Keil's RealView Microcontroller Development Kit (RVMDK) and SafeRTOS from Wittenstein to create exciting new applications. Texas Instruments launched the contest with a simple question: "Just how far will your design take you?" In response, designers from all over the world stepped up to the challenge and delivered truly inventive designs.

After reviewing all the entries and scoring the projects on their technical merit, originality, usefulness, cost-effectiveness, and design optimization, the judges' results are now final. Congratulations to everyone who participated in this exciting contest!

First Prize

Richard Wotiz
United States | dick601@mystics.org

Bicycle ABS Brake System

The innovative Bicycle ABS Brake System controls brake force to reduce wheel skid. Built around an EKK-LM3S9B96 evaluation board containing a Stellaris LM3S9B96 microcontroller and support circuitry, the system measures the force applied to a bike's brake levers, along with the speed of both wheels. If it detects skidding, it drives a pair of stepper motors that pull back on the brake levers, counteracting the rider's squeezing force on the levers. An adjustment knob enables the rider to set the skid threshold level at which the system activates, allowing for different performance characteristics depending on the trail surface. You can connect a separate unit containing an LCD and menu buttons for diagnostic monitoring.

"My project is an antilock braking system for a bicycle. The system works by measuring wheel speed and brake lever force, and controls the braking action to reduce wheel skid. At first I thought the LM3S9B96 would be overkill, but as the design progressed, I found myself using many more of its features than originally expected. I ended up using most of the MCU's capabilities, while having enough room left over for new features as I improve the design." — Richard Wotiz

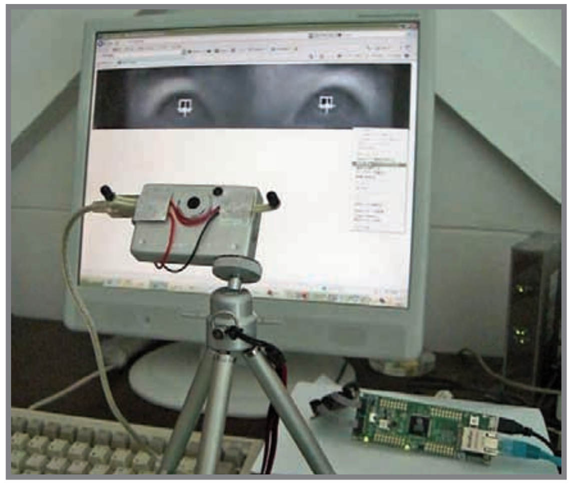
Visit www.circuitcellar.com/designstellaris2010/ for full entries.

Second Prize

Camera-Based Eye-Tracking System

This fascinating eye-tracking system was developed with an EKK-LM3S9B96 evaluation board and a USB camera. The Stellaris MCU works as a USB host and controls the camera. A button on the board is for switching the calibration/tracking modes. In Calibration mode, a captured eye image with detected markers is displayed on a computer screen. In the Tracking mode, gaze points are plotted on a white image. Results can be monitored by multiple PCs on a network.

Osamu Tamura
Japan | tamura@recursion.jp



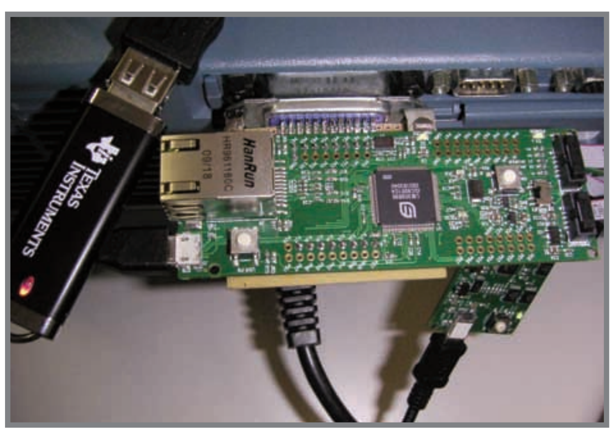
“This system detects eyeball movement and gaze points are plotted. This was my first ARM project. I found the 32-bit chip very easy to use.” — Osamu Tamura

Third Prize

GPIB Screen Saver

Many lab instruments use the GPIB. The handy GPIB Screen Saver is a USB screenshot saver for all GPIB instruments. The LM3S9B96 microcontroller, with its USB host interface, has more than enough resources to make retrieving screenshots easy. The compact design includes a GPIB 25-pin connector. The LM3S9B96 is connected to the GPIB lines without a level shifter or buffer.

Sylvain Davaine
France | davaine_s@yahoo.com



“At my work, we still use some scopes and network analyzers that don’t have USB or SD card capabilities. The only way to get measurements are slow GPIB printers/tracers or floppies. That's why I thought of building the GPIB ScreenSaver! I had a previous experience with a Stellaris controller, so I was happy to use the StellarisWare libraries, which save the developer lots of time by not having to dig into the datasheets in search of the right registers to use. As for the LM3S9B96 in particular, I must say that I didn’t find any bugs and I didn’t have any problems using the controller’s peripherals. I am already developing another application with a LM3S9B92 for a web aquarium monitoring system.” — Sylvain Davaine

TEXAS INSTRUMENTS

Visit www.circuitcellar.com/designstellaris2010/ for full entries.

Honorable Mention

Handheld Pollen Sensor

The handheld Pollen Sensor design is used to detect high concentrations of pollen. It uses an LED to flood a sensing area with light, a large area detector to pick up the fluorescence, and an optical band-pass filter to pass fluorescence wavelength (green) and reject the LED wavelength (violet). Two PWMs in the LM3S9B96 CPU are used. It uses SafeRTOS to improve the processor's ability to track and report the level of real-time particle pulses.

Jim Brady
United States | jimbrady@aol.com

"I built a pollen sensor that measures and reports the level of pollen in the air by detecting its fluorescence. In the past, I have worked with some very expensive small-particle fluorescence sensors, so in this project I wanted to see how well a low-cost approach would work. I liked the TI LM3S9B96 because it has plenty of horsepower and a ton of built-in goodies. Its built-in driver library and RTOS put it a notch above other CPUs. The driver library sped up my code development." — Jim Brady

Stellaris BCU/Battery Module

Design flaws in current lithium-ion systems can be overcome with some additional hardware and the right microprocessor solution. In this creative project, an LM3S9B96 microcontroller is used to implement a battery management controller (BCU) with a safety-certified RTOS with compact and efficient battery monitoring, data reporting, and circuit protection, all in a self-contained battery module assembly.

Marty McLeod
United States |
martymcleod@yahoo.com

"My project is a lithium-ion battery module. It (as typically is done) consists of eight Li-Ion battery cells, battery management electronics, and a user display, assembled together as a complete unit/module. I enjoyed using a nice microcontroller like the TI Stellaris device. It gave me a good opportunity to experience the use of a flexible, powerful device." — Marty McCleod

Color Sensor: A Portable Color Reading Device

The portable Color Sensor is an effective tool for obtaining accurate color readings from both reflected and emitted light. It generates accurate readings in multiple intuitive formats, both on a built-in LCD and on a webpage. An LM3S9B96 controls all operations and communication.

John Peterson
United States | circuit@saccade.com

"The Color Sensor is a device for measuring color, either from emitted light or reflected off a surface. The systems consists of the EKK-LM3S9B96 EVK board with CPU, USB and Ethernet connections, an LCD (connected via SPI), and the color sensor (connected via I2C). I liked the built-in software library for accessing the on-chip peripherals, and the tasking library. This simplified software development significantly. The built-in peripherals and communication were a perfect match for this application. I had little trouble getting the chip up and running." — John Peterson

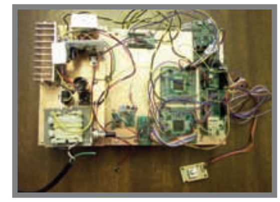
ACC PIN Reader

Users can implement the innovative ACC PIN Reader to increase pin code security. The design is built around an EK-LM3S9B96. The display shows random keypad patterns. The Stellaris microcontroller detects touch, computes touch position coordinates, and converts them to key codes that are then transmitted to an ACC controller.

Aleksander Borysiuk
Poland | alex_priv@wp.pl

Digital Oscilloscope

Digital oscilloscopes generally require a PC, but they don't always have a display built in. This one has a nicely sized screen and can be a great addition to any workbench. The LM3S9B96-based design uses the extended peripheral interface bus and direct memory access to drive an LCD directly without a graphics controller. The internal RAM is used as the display buffer. The embedded RTOS simplifies the software required for the various tasks. The quadrature encoder interface is used for a scrolling knob.

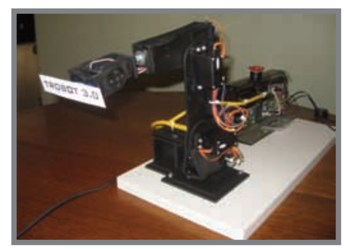


Kevin Gorga
United States | kgorga@stny.rr.com

"My project was to build an LCD oscilloscope with an integral display. I wanted a small standalone benchtop oscilloscope for debugging. I was only able to get some basic functions working in the short time allowed for the contest, but I am in the process of enhancing the capabilities. The LM3S9B96 was a good fit for my project because it had a DMA Controller and Enhanced Peripheral Interface port that I used to interface with the LCD display. It also had other peripheral functions like I2C, large number of I/Os, and Quadrature Encoder interface which I was able to make use of." — Kevin Gorga

TROBOT 3.0: An "ABB Robot Studio" Interface for a Miniature Articulated Robot

The well-designed TROBOT 3.0 is a miniature six-axis robot powered by small (RC-style) servo motors. The design features a Texas Instruments EKK LM3S9B96 evaluation kit, which acts as a servo controller interface between the robot and a PC running ABB's Robot Studio.

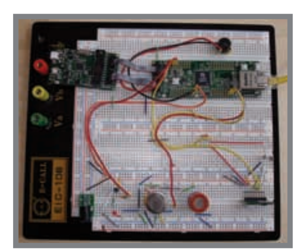


Toby Baumgartner
United States | tbaumg@gmail.com

"The TROBOT consists of six small RC-style servos, was assembled from custom laser cut plastic, and is controlled by a Texas Instruments LM3S9B96 embedded controller running a SafeRTOS. I enjoyed working with the LM3S9B96 processor with the SafeRTOS, it was very powerful, and handles all of the tasks very well." — Toby Baumgartner

Lethal Alert: A Real-Time Gas Leak Detector

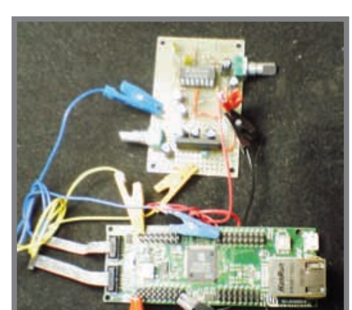
The Lethal Alert unit is a networked real-time monitoring device used for sensing LPG and CO in the air. The sensors are connected to an LM3S9B96's ADC pins. In the event of a gas leak, the device sounds an alarm and passes an error message to a monitoring remote TCP server.



Rajesh Kademada Chittiappa
Canada | rkc.mail@gmail.com

Transceiver Controller

With this easy-to-use LM3S9B96-based design, a user can remotely access and control a transceiver via an Internet-connected PC. The user simply uses buttons on a web browser and a VoIP connection to send and receive data.



Shinji Kasaki
Japan |
takamaro@mtd.biglobe.ne.jp

Honorable Mention

Ethernet Teleoperated Balancing Robot

Users can remotely control this innovative LM3S9B96-based balancing robot system. After gyro and accelerometer data are read and processed, a balancing algorithm is performed and output is sent out to the left and right motors.

Nghia Tran, Victor Tran, Tammy Tran, Andy Tran, and Trucmai Nguyen
United States | txnghia@sbcglobal.net

Hands-Free USB Mouse

The futuristic Hands-Free USB Mouse features a head-tracking system that enables a user to move a mouse cursor around a computer screen. It detects eye motion in order to initiate button clicks. An EKK-LM3S9B96 kit provides the USB hardware to interface to a PC.

Stephan Lubbers
United States | ke8FP@arrl.net

"My hands-free mouse uses a sensor system worn on the user's head to detect motions that are translated into cursor movements on a computer screen. The LM3S9B96 was an easy chip to get to know. I got basic portions of my project up and running quickly. TI did a great job with their contest. The hardware and software worked right out of the box and plenty of software examples addressed how to use the chip. This allowed me to concentrate on my project without getting bogged down in making the basic hardware work. Online help forums attempted to answer all of the crazy questions contestants came up with. I'm looking forward to the next TI contest!" — Stephan Lubbers

Specialized Signal Processing Engine

Featuring an EKK-LM3S9B96 board, this project employs a signal-processing engine to offload demanding sensor processing tasks from a master microcontroller in a multi-processor environment. It's meant to assist a master micro by managing an assortment of specialized sampling tasks.

Matt Pennell
United States | mgpennell@gmail.com

"This project is designed to offload specialized A/D sampling tasks from a master microcontroller. The system hooks together several of the LM3S9B96 peripheral modules to efficiently process analog data. It is easy to get started with all of the example projects in the StellarisWare library. The processor and support libraries are also well-documented, which facilitated development. For my design, I liked the level of control that the processor provides for the A/D converters and the DMA engine. I also liked working with the Cortex-M3 and its interrupt controller." — Matt Pennell

Automatic Pill Dispenser

With this medicine dispenser, users can set predefined pill doses to be ejected on regular schedules. The LM3S9B96-based system is meant for users who need to take pills on a regular basis.

Wolfgang Guettler
Germany | wo_gue@hotmail.com

"While brainstorming about a useful application which could make life easier—at least for some people—and which could be realized using the LM3S9B96, this application came to my mind. Put simply, the LM3S9B96 works like a real-time scheduler that triggers a pill-ejecting mechanism. Once I got familiar with the library functions, it was helpful and time-saving to not have to deal with processor-specific details." — Wolfgang Guettler

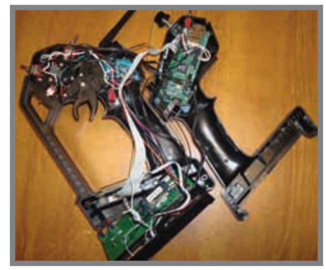
D-Star Adapter: Send and Receive Digital Audio and Data for Amateur Radio

The D-Star Adapter enables a user to transmit digital audio and data for amateur radio communications. With an LM3S9B96 at its core, the design—which encodes and decodes only digital data—can communicate with the APRS amateur radio system.

Fabian de la Fuente
Argentina |
fdelafuente@speedy.com.ar

Five-Channel Programmable RC Pistol Grip Transmitter

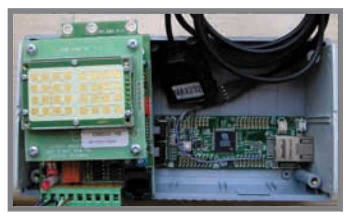
With an LM3S9B96 and some ingenuity, it's possible to upgrade and customize a fairly simple RC system without breaking the bank. The project's purpose is to improve a basic 2.4-GHz RC radio system—comprising a basic two-channel transmitter and five-channel receiver—so it works as well as a pricey high-end RC radio.



Greg Cloutier
United States |
glhs577@hotmail.com

"After creating a project with Texas Instruments's LM3S9B96, it has opened the door to strongly consider the Stellaris family for future efforts. The peripheral library saved valuable time and debugging while the requirement to use the included SafeRTOS worked in my favor. The LM3S9B96 was a great fit for my design due to the amount of I/O, the amount of peripherals, the built-in RTOS, and especially the fast interrupt system." — Greg Cloutier

MuDLi: Multipurpose Data Logger with Internet Connectivity



The multipurpose "MuDLi" is an LM3S9B96-based data logger that users can modify to meet their application goals. It's perfect for storing data as well as transmitting preprocessed data files over the Internet.

Imrich Konkol
Slovakia | ikon@inmail.sk

"Finding a multipurpose Internet-connected data logger with additional features and configurable capabilities for a reasonable price is a challenge. So, I decided to design data logger myself based on the TI LM3S9B96. It is a nice piece, indeed. It has a lot of capabilities, peripherals, enough flash memory, and a reasonable amount of RAM too. The Keil development environment is easy to learn and has powerful features for debugging and fine-tuning project properties." — Imrich Konkol

Electric Motorbike

This 48-VDC to 30-VAC, 7.5-kW inverter project powers a three-phase AC induction motor, which is the main drive system of a motorbike that was converted from gas to electricity. The control system comprises an LM3S9B96, a three-phase driver chip, and 12 n-channel MOSFETs mounted on a custom aluminum heat sink.

Justin Curran, Catherine Woodward, and Tristan Nixon
Canada | jtcurran@shaw.ca

"Our system works by doing an A/D conversion of the analog signal produced by the electronic throttle. This A/D conversion is then used to determine the output frequency of the three-phase AC signal. The LM3S9B96 is then used to generate pulse-width modulated signals to drive the MOSFETs to create the three-phase AC signal. I enjoyed working with the LM3S9B96 microcontroller, I found its computational power to be most adequate and the RTOS was convenient for setting up prioritized tasks." — Justin Curran by Simon Ford

NXP mbed DESIGN CHALLENGE

Introducing mbed

Ready to start prototyping the mbed way? Here you learn how mbed's co-creator Simon Ford took the design from concept to creation! The mbed microcontroller and tools now make rapid prototyping a cinch.

When *Circuit Cellar* and *Elektor* approached me about running a "design challenge" around mbed, I saw an opportunity to do something special. Bringing together the mbed platform we've been developing and a gaggle of innovative engineers should be a recipe for some great results! We're setting out to help make the microcontroller world a better place by laying down a very specific, yet wide-ranging, challenge: What technology can you help unlock for your fellow innovators?

I grew up on magazines like *Circuit Cellar* and *Elektor*, and many of my early experiments were inspired by the projects people had sent in. A few years on, I wrote a couple of articles about projects I had designed, and got them published too. My hope was that the articles would serve to inspire my comrades in the same way. I was still at school at the time, so getting paid to do my hobby was a huge novelty. Interestingly, it is a novelty I have managed to keep going at ARM, and one that I plan to continue! The point: progress is all about standing on the shoulders of giants, but sometimes I wonder if the embedded world missed that memo.

I started mbed as a skunk works project with Chris, a friend from work. We were both volunteering, trying to help people with microcontroller projects, and it was really painful. You can only make so many excuses for the tools, processes, and general fiddliness until you have to step back and ask, "Why does it have to be so hard?" That led to many evenings of research and experiments to understand how we could help people get stuff done. Fast forward a few years and it has become an official ARM project and we've grown an excellent team that is making the idea a reality. But what makes me really proud is the great community springing up around it; everyone has been incredibly supportive of each other, and it is a nice reminder that engineers are fundamentally enthusiastic, inventive, selfless, and helpful.

If you haven't already seen what we are up to, take a look at http://mbed.org. The mbed Microcontroller is the hardware component, taking a top-end ARM microcontroller and packaging it in a 0.1" pitch form factor with built-in USB programmer. The mbed Compiler aims to simplify tools too, making them accessible online just like webmail, so it "just works." And the developer website we're building is helping to provide the support, the resources, and the tools to allow you to share your developments, programs, libraries, and write-ups to help others get the next job done faster. This is where the challenge really kicks in.

We're looking for you to collectively enable as much technology as possible: get components talking, create a killer library, build an innovative reference design, or invent some form of interesting product prototype. This basically means anything that can be shared on http://mbed.org and reused by the next person to inspire and build their prototypes even faster. In reality, this is less of a competition against other developers (although that is very much encouraged!) than it is a challenge against the stranglehold of complexity and obscurity that hangs around microcontroller technology.

I'll be at Elektor Live! in Eindhoven, The Netherlands, on November 20, 2010. So, if you are around, please track me down and say hello. It would be great to hear your plans, problems, and progress first hand.

The most successful projects are ones where you scratch your own itch. Our itch was a full-on allergic reaction to all the barriers to prototyping with microcontrollers. We've tried to get things moving, and I hope you'll take up the challenge to join us. So, get yourself an mbed, get involved in the design challenge, and get building the foundations for enabling the products of the future. Help us set embedded technology free! ■

Simon Ford, co-creator of mbed, is a lifelong electronics and computer engineer. He works at ARM, and before starting mbed was technical lead for the ARMv7/NEON architecture now found in most new smartphones.

To enter the NXP mbed Design Challenge 2010, go to: www.circuitcellar.com/nxpmbeddesignchallenge/

by George Novacek (Canada)

My Analog World

The Significance of Grounding

You likely move between the analog and digital "worlds" on a frequent basis. But no matter what sort of project you're working on, always follow professional grounding rules. There is no such thing as "too good a ground."

Quantum mechanics notwithstanding, our world is analog. And so despite our fascination with everything digital, we need interfaces to provide bridges for our analog reality to cross over to the digital paradigm and then back again. One may ask: Is there a common denominator that binds these two worlds together regardless of their many conceptual differences? In my mind, it is grounding.

WHY GROUNDING MATTERS

When I was 10 years old, I built my first vacuum tube radio. Its 200-VDC plate voltage provided me with a quick, effective lesson on electricity, especially capacitors. The radio worked OK, but the 50-Hz hum—which was almost as loud as the program itself—bothered me. And so I stumbled onto the arcane art of grounding. It took me a while, but eventually, by trial and error, I figured it out and the hum disappeared. I later realized that I had reinvented the wheel, but the lessons I'd learned in the process I never forgot.

RF engineers have always maintained a healthy respect for grounds. But to those of us working in the relatively low-frequency spectrum of control systems and audio, transistors brought significant relief from vacuum tubes and grounding appeared less critical. Until the clock frequencies began to push into RF and microwave, ADCs and DACs moved from 8-bit to 10-, 12-, 16-, and 24-bit resolution and customers began insisting on EMC and ESD immunity.

The arrival of a multilayer PCB with a ground plane alleviated many problems, but we had to rediscover the concept that RF engineers had known all along: circuit topology is just as important as its geometry, and grounding is paramount.

A ground plane is only as good as its designer. It has a finite impedance and is not always a solid plane, as it must accommodate components' solder pads and vias. If the designer is not careful, inevitable ground tracks shared by several circuits could cause problems.

Consider a 0.25" trace

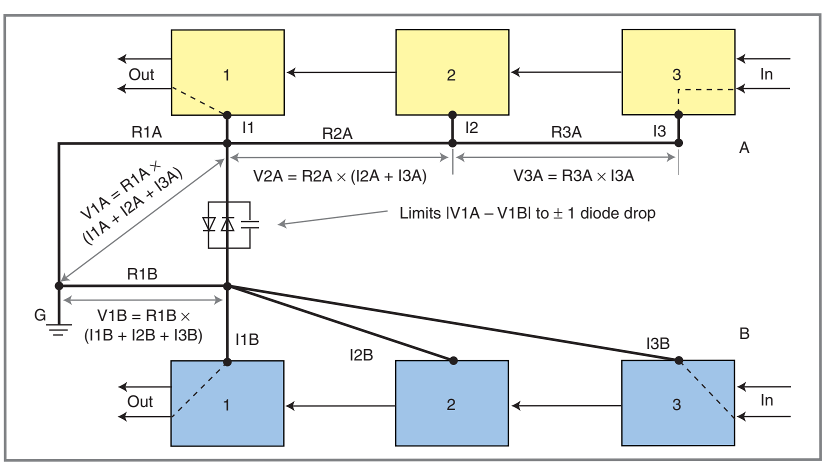


Figure 1—Both sections, A and B, may be on the same PCB with separate ground planes (e.g., analog and digital). The diodes and the capacitor between the planes limit potential differences due to ground bounce, etc. Broken lines inside boxes 1 and 3 indicate ground referenced, non-symmetrical inputs and outputs.

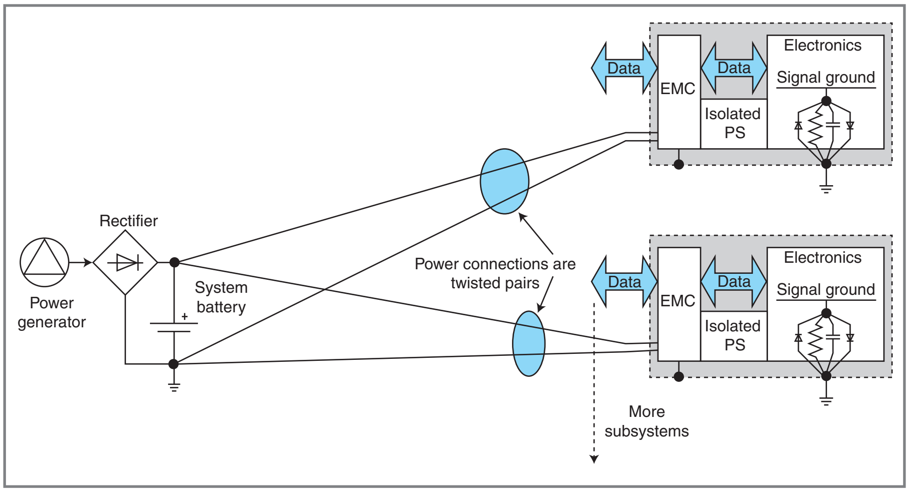


Figure 2—Connecting subsystems to avoid ground loops

of 1-oz. copper, 32 mils wide, running between IC pins to connect two ground plane islands. It is shared by an ADC with some other circuit. The trace will have an approximate 4-mΩ resistance. The least significant bit (LSB) voltage of a 16-bit ADC with 5-V full scale input is 76 µV. Note that 19 mA drawn by another circuit through the same trace will obliterate the LSB. With a 24-bit digitizer just 75 µA will have the same effect. Today's PCBs with 0.5-oz copper and 8-mils-wide traces make the PCB layout even more critical, leaving no room for error.

FUNDAMENTAL RULE

The fundamental rule for grounding is depicted in Figure 1. By "ground" I mean the common 0 V potential to which signals are referenced. The "chassis ground", if grounding conductors had 0 Ω impedance, would also be 0 V—but, unfortunately, it never is. Yet there are still systems that are sufficiently insensitive to ground potential differences. They use the chassis for the signal and power returns. At one time, this was the way cars had been wired.

Figure 1a shows circuits sharing a common ground run. Notice that the output or the highest current drawing stage (1) must be the closest to the common point to minimize the voltage developed by that stage current over the grounding conductor. Also notice that the input signal and its return must be tied to the input block (3). Internal signal returns (grounds) are shown by broken lines. Returning inputs or outputs anywhere else would superimpose the noise from stages 1, 2, and 3 on the input signal.

Figure 1b shows the approach often used in RF equipment. There is no sharing of grounds; they are all individually tied to a single point. Each circuit A and B can occupy its own PCB or they can be on a single PCB, their ground planes separate, such as analog and digital circuits. The grounds come together at the point G, where the chassis is also connected. Where there are a few inches of wire tying the individual grounds together, it is a good idea to insert fast signal diodes and a capacitor as shown between the separate ground runs. Any potential difference developed between the separate grounds due to finite impedance of wiring, as shown in Figure 1, will be attenuated and clamped by the three components. Note that the "capacitor" should in fact be a parallel combination of a number of capacitors, depending on the application, to guarantee performance across the spectrum. The following are typically used: 100 pF, 1 nF, 10 nF, 0.1 µF, and 1 µF.

In safety-critical systems such as aircraft comprising two or more subsystems enclosed in metal cabinets, such as shown in Figure 2, only currents from lightning or other interference suppressed by the EMC blocks is allowed to be returned to the chassis. Power needs to be delivered by twisted pairs and all the returns connected to the chassis at a single point. If the signal grounds of the electronics are not allowed to be connected to the chassis, which depends on the system architecture, a combination of diodes, a capacitor, and a resistor as shown needs to be used to prevent ground loops as well as parasitic feedbacks between the electronics and the metal cabinet.

The subsystem enclosures shown with dotted lines connect to the chassis by mounting screws or straps. Most of today's avionic equipment uses isolated power supplies. There is no galvanic connection between the power return and the internal signal ground to eliminate ground loops, yet the enclosures need to be connected to the internal grounds to eliminate parasitic feedbacks through capacitive coupling; but then they would create ground loops. Some system architectures require signals working with their own ground, isolated from the chassis, to prevent ground loops. To compensate for the parasitic capacitances existing due to the proximity of the metal enclosure to the components and the EMI/lightning protection, the resistor, capacitor, diode combination (as mentioned above) is used, with typical values of resistor about 10 kΩ and capacitance less than 10 µF. The capacitance is again a parallel combination of smaller capacitors to suppress the desired frequency spectrum.

FOLLOW THE RULES

Digital circuits didn't used to be as finicky for grounding as analog circuits. This is no longer true. If you work with CMOS logic or relatively low-gain small bandwidth analog circuitry, you may be able to get away with violating some grounding rules. But I advise against it. In my book, there is no such thing as too good a ground.

George Novacek (gnovacek@nexicom.net) is a professional engineer with a degree in Cybernetics and Closed-Loop Control. Now retired, he was most recently president of a multinational manufacturer for embedded control systems for aerospace applications. George wrote 26 feature articles for Circuit Cellar *between 1999 and 2004.*

by Matt Oppenheim (United Kingdom)

PVDF Phased-Array Analog Front End

Need a board to test algorithms for CDMA signal transmission and processing? You can employ an array of broadband receivers using thin polyvinylidene fluoride (PVDF) sheets as the transducers. Here you learn about the analog electronics required to condition the analog signals for digitization and the subsequent transmission line implementation.

My supervisor appeared at the lab door pulling a large trolley. "This is the single-channel confabulator I made for my PhD," he beamed, showering a pile of valves, wires, and dust onto the floor. "Make it portable, battery-powered, and scale it up to eight channels," he instructed, disappearing behind the growing mound of antiquated technology. As he disappeared he muttered: "It's summer vacation, so I'm off for three months. Have fun!"

In this article, I present an implementation of an array of broadband acoustic receivers using thin polyvinylidene fluoride (PVDF) sheets as the transducers. The board is intended for testing algorithms used in code division multiple access (CDMA) signal transmission and processing. The final board is shown in Photo 1.

What is CDMA? Imagine that you are in a crowded room. Someone shouts loudly from one side to the other. You'll hear and understand him. But if several people start shouting, then the messages will become garbled. What if everyone else is whispering a different message at the same pitch, all together, all at the same time? Furthermore, if they are whispering in a language that you understand, and everyone else is talking in a foreign language, you will probably catch the faint message that you are interested in. Rather than dominate a single frequency, the signal is transmitted over a broad range of frequencies, but at very low amplitude. This means that regular narrow-band transmission is not interfered with by CDMA, nor does it block the signal.

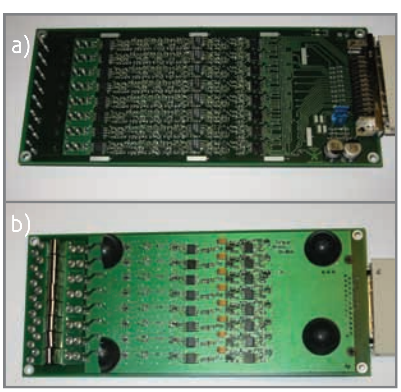


Photo 1—The eight-channel phased-array PVDF board. The top layer (a) houses the transducers. The bottom layer (b) houses most of the electronics and almost all of the analog circuitry. The board is four layers with a dedicated ground and power layer. All components are hand soldered!

This article focuses on the analog electronics required to condition the analog signals for digitization by a 16-bit analog-to-digital converter (ADC) and the subsequent transmission line implementation to enable communication with the processing platform. I'll cover design issues and board layout considerations. The board is being used by a fellow Lancaster University student, Mohammed Alloulah, for his PhD research pertaining to signal processing algorithms using an FPGA.

TRANSDUCERS

The PVDF sheets look and handle remarkably like a small piece of a Kit Kat wrapper, and that's precisely what I used for testing the film mounting. PVDF film is a piezoelectric material. When the film is hit with ultrasound in a frequency range of about 10 to 100 kHz, a tiny current of a few picoamps is produced by the film.

So how do you mount what is basically a conductive plastic film? If you try soldering on leads, you quickly end up with a melted puddle of smoking plastic. We tried using conductive epoxy to stick the film to PCB pads, but found that the supposedly conductive epoxy wasn't (conductive, that is, no matter what the can says).

The most reliable method we found is to sandwich the film between two PCBs—the main board and a clamp board. Slivers of conductive tape are placed between the PVDF film and the pads on the boards. Then the assembly is bolted together using M1.6 thread screws. Getting the film to curve into a hemi-cylindrical shape requires some geometry calculations so that the mounting pads are the correct distance apart for the film to curve naturally into the required shape.

Figure 1 shows the PVDF mounting arrangement. The channel spacing of 8 mm is a requirement of the signal-processing

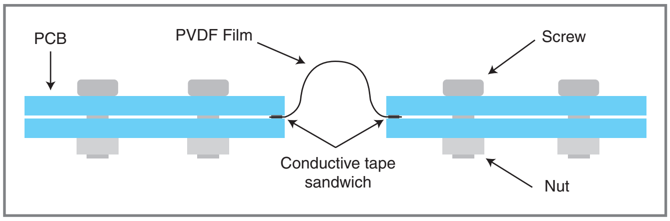


Figure 1—This is the PVDF mounting arrangement.[1] (Source: Mike Hazas)

algorithms implemented on the FPGA board that processes the final signals produced by the receiver board. These algorithms are beyond the scope of this article. The clamp PCBs are attached using two bolts, because with only one bolt they are free to rotate out of position.

ANALOG SIGNAL CONDITIONING

As you would expect, when the transducer produces a current, the first stage of the signal-conditioning chain is a current-to-voltage converter. With such a tiny current to amplify, the input current offset and bias of the op-amp are critical. I settled on the OPA129UB manufactured by Texas Instruments, which is billed as "ultra-low bias current." This is quoted as ±30 fA! The noise figure of any amplification chain depends critically on the first stage, so I used the best part that I could find.

You will notice the "T" arrangement of resistors in Figure 2 around the current-to-voltage converter. This allows much lower values of resistors to be used compared with using a single resistor. I needed the equivalent of at least a gigaohm and wanted to be able to try higher values. The disadvantage of this arrangement (nothing is free!) is that the offset voltage is amplified by the same factor by which you are reducing the equivalent single resistor. So long as this offset does not cause the output signal to approach the voltage rails, this is not a problem as the output voltage signal from the converter is decoupled through a capacitor, which eliminates the offset.

To give your tiny signal the best chance of survival, the layout of components and tracks for the input to this op-amp are critical. The pinout for the OPA129UB is optimized to allow for placing guard traces around the inputs. These guard traces are connected to the "substrate" pin of the amplifier at a single point, which is grounded. Normally, just having a ground plane under the traces is considered sufficient to guard the input lines from interference, but as the input signals are so tiny, I implemented the whole "belt and braces" approach of guard traces and ground plane. The guard traces are connected to the ground plane as close to each end of the traces as practical.

Next, the signal is filtered to remove unwanted high frequencies. Phase information is critical for the final signal processing. So a Bessel filter is used since this does not alter the

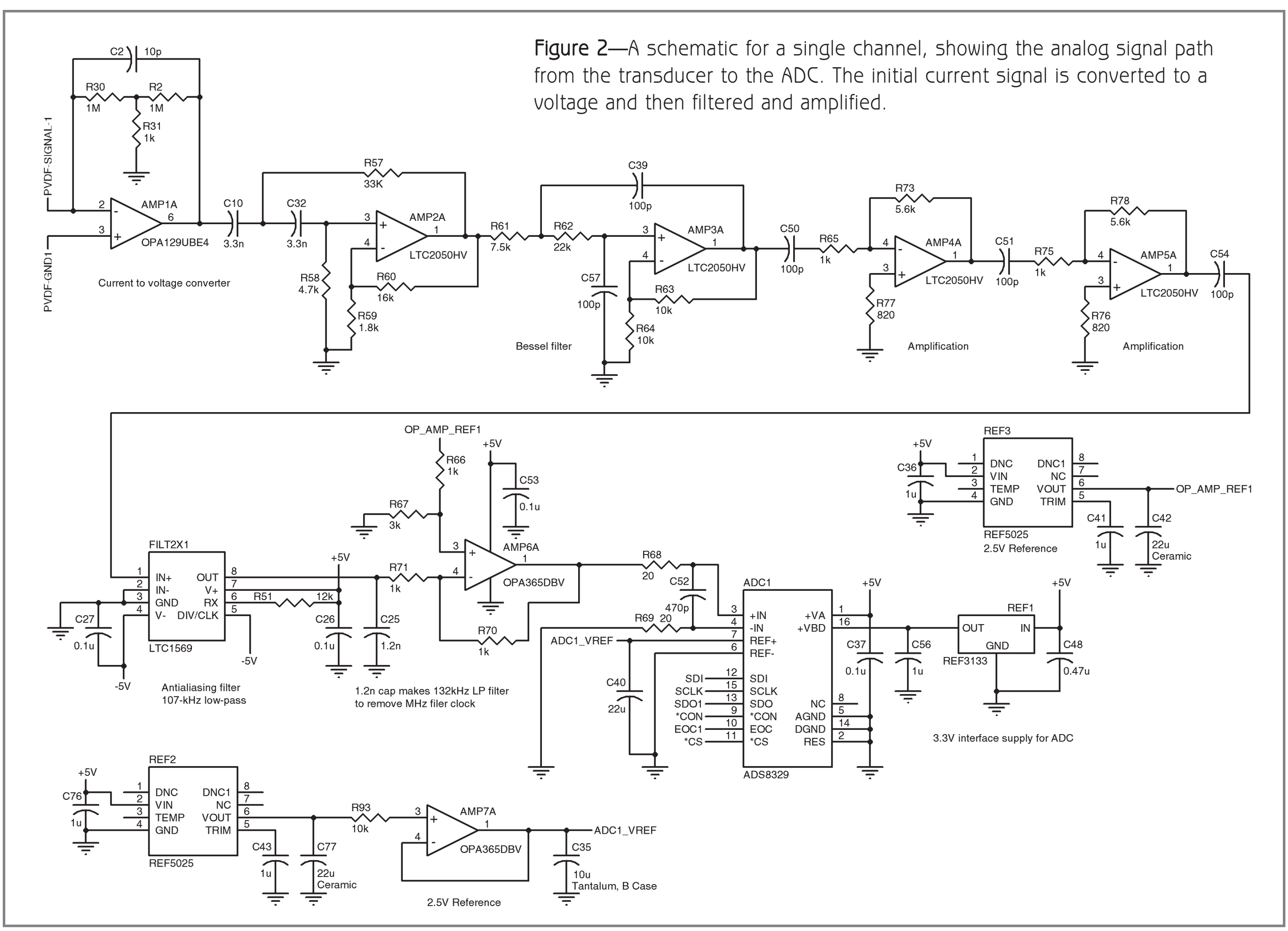


Figure 2—A schematic for a single channel, showing the analog signal path from the transducer to the ADC. The initial current signal is converted to a voltage and then filtered and amplified.

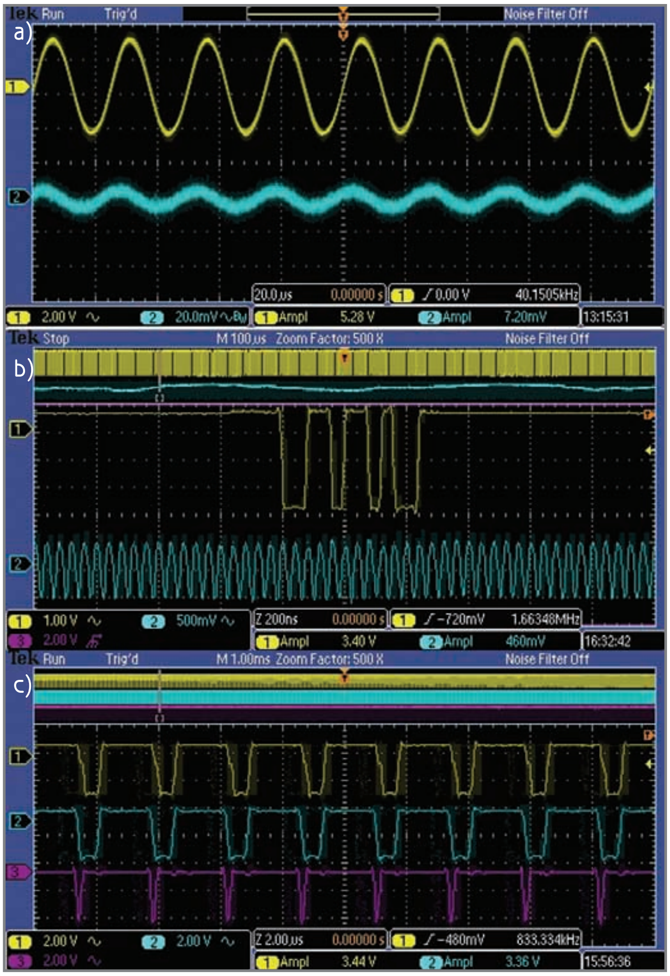


Photo 2—Oscilloscope screen grabs. a—Here you see the test signal after amplification and filtering with the signal immediately after the current-to-voltage conversion. b—This is SPI data from one of the ADCs alongside the SPI clock. c—These are SPI control signals (SDI, !CS, bottom !CON).

phase across the bandwidth of the signal. This is implemented using two Linear Technology LTC2050HV op-amps (AMP2A and AMP3A in Figure 2). Throughout the design, I used single op-amp packages for two reasons. The first was that there is a potential for cross-feed between the op-amp stages in a multiple package. The second was layout space restrictions. I needed each stage to fit into the 8-mm spacing of the PVDF films, which made for some tight routing! Still, it all worked out in the end. I used precision resistors throughout the filters to maintain consistent characteristics across the separate channels.

After the filtering are two stages of amplification (AMP5A and AMP6A in Figure 2). The original design had a single stage. Initial testing indicated that to get good signal amplitude for the following stages, the single stage might approach the gain-pass bandwidth limits of the op-amp. I could have stuck with a single stage and found an op-amp with a higher gain bandwidth, but I settled on staying with the op-amps I knew and having two stages of gain instead. It was a tradeoff, as it is always best to reduce the component count in a design to save space, power, and cost.

I used the classic inverting amplifier configuration since this reduces the component count compared to noninverting. With the noninverting configuration, the input needs to be tied to ground using a resistor; otherwise, the output can have a large voltage offset. To further eliminate the possibility of DC offset, I coupled the stages using a capacitor. I may seem a little paranoid, but what is an extra capacitor between friends? Having the capacitor between each stage also allows for the subsequent stages to be easily isolated during debugging: simply lift the capacitor with some desoldering tweezers (in the unlikely event that the circuit does not perform flawlessly first time).

Photo 2a shows the amplified and filtered output from the final op-amp stage and the output from the first op-amp—the current-to-voltage converter. Note that the amplitude scales differ by a factor of 100. The signals generated from the current-to-voltage converter are tiny! As can be seen, the op-amp amplification and filtering cleans up the signal nicely. The test signal was generated using a 40-kHz piezoelectric transducer connected to a function generator.

ANALOG-TO-DIGITAL CONVERSION

To provide further attenuation for out-of-band frequencies and to ensure that no aliasing occurs, a Linear Technology LTC1569 filter chip is configured as a low-pass filter. This is a versatile tenth-order filter chip in an eight-pin package. The filter cutoff frequency is set using a single external resistor. Refer to the datasheet for details. Linear Technology supplies its own free FilterCAD software to help you with filter design. Naturally, the results use their own range of filter components. C25 in Figure 2 removes the high-frequency (megahertz range) clock noise from the filter's output.

The analog-to-digital conversion circuitry was designed by Mohammed Alloulah in an attempt to limit the damage to his project because I designed the instrumentation that his future thesis depends on. A 16-bit Texas Instruments ADS8329 is used by each channel to convert the conditioned analog signal to digital. An ADC is only as good as the reference voltage used to compare your precious signal to, so a separate voltage-reference IC is used to produce the reference voltage for each ADC (REF2 in Figure 2). This reference voltage is further stabilized using a Texas Instruments OPA365 op-amp (AMP7A in Figure 2). This is implemented to overcome the transient noise that results from the loading effect taking place during conversion—an inherent effect from the architecture of the successive-approximation converter.

Details of this layout and configuration were worked out after spending a lot of time watching a man wearing a cowboy hat explaining the optimum design and layout configuration in Texas Instruments's educational online video series. Even the type and ESR rating of the capacitors that load the output from the reference have a measurable effect when working at 16-bit resolution (C35 in Figure 2). Basically, I copied their design and component recommendations!

The ADC only draws a small current. Since this is a 3-V part and the rest of the board uses ±5-V supply rails, a Texas Instruments REF3133 voltage reference powers each ADC. This adds a further layer of isolation between the ADC and the power supply circuitry. The part does not explicitly require a smoothing capacitor on the VOUT line, but I added

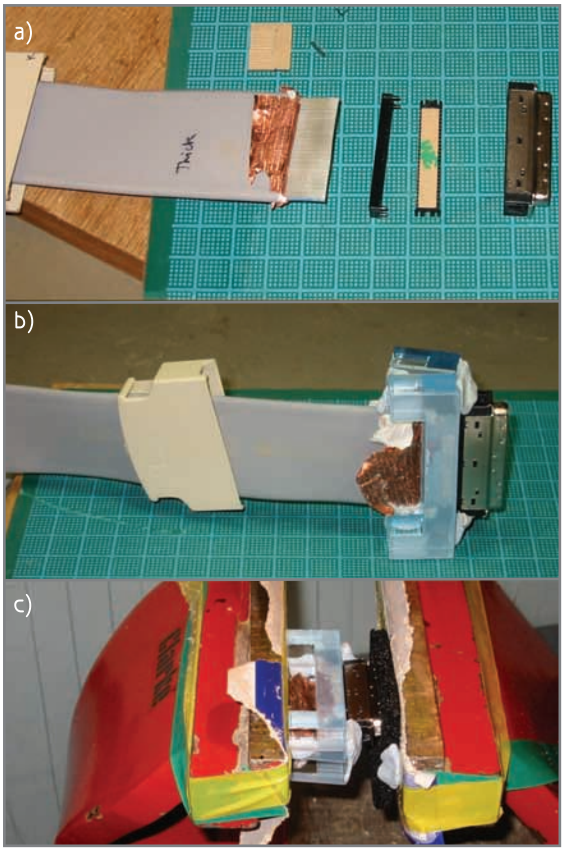


Photo 3—The transmission cable assembly. a—Take a look at the shielded ribbon cable and connector components. b—The connector is ready to press fit. c—I used a vise to press-fit the connector.

one for extra decoupling.

Photo 2b shows the digital data from one of the SPI lines. The lower trace shows a 25-MHz SPI clock signal. Due to the bandwidth limitation of the scope, this does not appear as a nice square wave. Photo 2c shows the other digital control signals for the ADCs generated by the FPGA. Because these are much lower frequency than the clock line, the scope has the bandwidth to display them as square waves.

DIGITAL INTERFACE

It's all analog under the skin! I wanted as fast an SPI as practical to connect the receiver board to a Xilinx Virtex II FPGA development board. Luckily, the communication science lab is adjacent to mine, so I was able to consult the professionals for advice on transmission-line design. As Robert Lacoste often mentions in his excellent "The Darker Side" column, it's all about impedance matching. I chose shielded ribbon cable for the transmission line as it is relatively straightforward to work with and butcher (I use that word accurately) into connectors. Also, I could order a single meter of this cable from my regular supplier, rather than having to purchase an entire reel.

Shielded cable is required, since unshielded cable with a 25-MHz signal pulsing along it acts like an antenna. I don't think the folks using the lab's indoor positioning systems would appreciate having to compete with a homemade transmitter!

With eight channels of SPI coming from the board, and timing control signals going into it, I needed a lot of signal lines. Naturally, I also needed to fit them all into a small enough footprint to not require widening the board since this would result in having to rethink the housing.

The 3M MDR range of connectors seemed suitable for the job as they have a small enough footprint for my board and all of the required genders of connectors are available from my usual online supplier. These connectors are available in both solder cup and insulation displacement connectors (IDC). IDC style has the advantage over solder cups of being much faster to assemble. Naturally, the manufacturer's recommended assembly tool cost a ridiculous amount of money. After carefully perusing the assembly instructions, I figured out a way to press fit all of the parts together using a vise, a pair of simple homemade jigs, and copious amounts of blue tack and insulation tape. I have put details of this assembly technique onto my website. Photo 3 shows the cable and connector being assembled. Photo 4 is the completed cable and connector. This method proved to be straightforward and reliable. I used the lab laser cutter to cut the 10-mm plastic jigs used to press-fit the assembly together. As always, no one was more surprised than I was when it all worked!

The only disadvantage of my cable assembly is that the shielded casing is not available for 50-way flat ribbon cable. I got around this by modifying the round cable-shielded casing with the judicious use of a hacksaw and file. The type of casing I wanted is made for just about any other width of cable!

From the datasheet, the characteristic impedance of the ribbon cable is 50 Ω. Signals originating from the ADC and going to the FPGA are impedance matched using a 50-Ω resistor. Signals coming from the FPGA to the ADC were terminated using a 50-Ω resistor in series with a 0.1-µF capacitor. The theory behind these termination techniques can be found in Ron Schmitt's 2002 book titled *Electromagnetics Explained*.

To eliminate cross-feed between the rapidly switching digital lines, I grounded a wire in between each signal wire on the ribbon cable. If the signals were differential, I would have used a twisted-pair configuration, but SPI signals are not differential.

I built a board that plugs onto the Xilinx development board to allow my cable to connect with the signal pins on the FPGA. This has impedance-matching resistors for the signals that originate from the FPGA going to my receiver board. Photo 4 shows the breakout board mounted on the Xilinx development board. Because all of the signals on the Xilinx board are digital, it is not necessary to add external shielding to this board to protect it from interference. However, having digital pins switching quickly acts as a transmitting antenna, so it would be necessary to case the board for a commercial product to comply with emission standards.

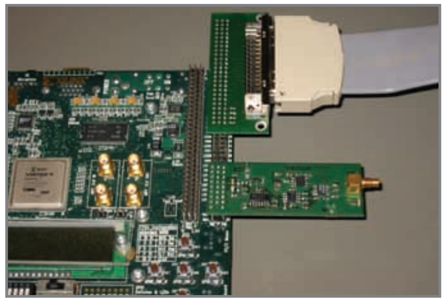


Photo 4—The breakout board and DAC

The output characteristics for the SPI clock line from the FPGA are critical for the interface to work. As always, read the datasheet carefully! We had to play around with the setting for the current limit of the pin to get life from the SPI bus.

CASING & PCB LAYOUT

I found a range of conductive plastic cases rated for EMC protection. The big advantage over using a metal case is the ease with which they can be machined. It's always a good idea to start looking at cases as early on as practical. The case dimensions and mounting arrangement act as major constraints on your PCB layout and dimensions.

I used CadSoft Computer's Eagle v5.6 for the schematic capture and PCB layout. I didn't touch the autorouter and don't advise anyone to do so. I used four layers with a layer dedicated to ground. Optimizing the ground return path is critical to analog design. I did not use separate analog and digital ground layers. I used a separate lobe of ground for each of the analog channels and connected them to a large unbroken area of ground under the digital connector using a neck placed under the ADCs.

I fit each channel into the width of the PVDF film transducer—8 mm. Routing is 90% component layout. It is a false economy to start laying down track until you have every part for the block that you are working on in an optimum position. This took some time, but like many engineers, I find PCB layout engrossing.

Once I had a single channel completed, I found a user language program (ULP) called "duplicate board" that allows a completed block of layout

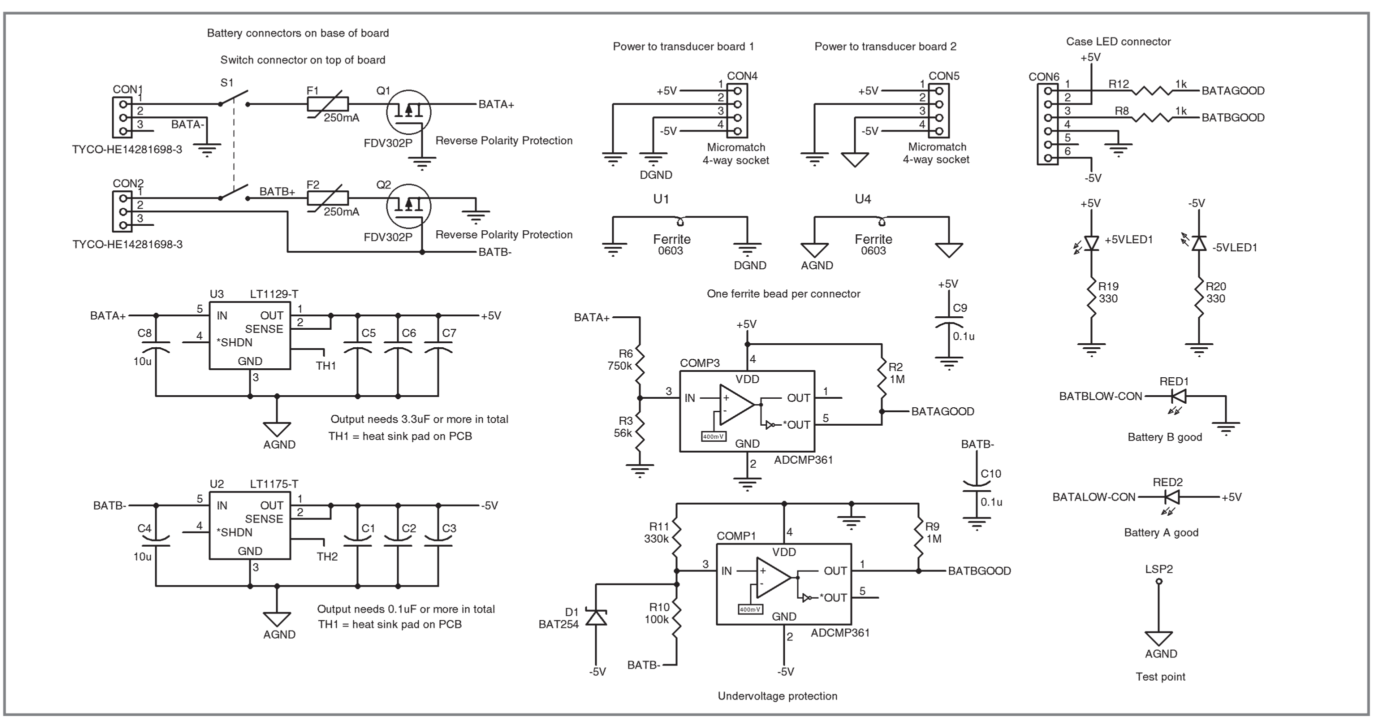


Figure 3—The schematic for the lithium battery power board (±5 V)

to be duplicated with a little fiddling. Look on the Eagle user group notice board for details.

DAC BOARD

We also designed a daughterboard for the FPGA board that enables the FPGA to produce an analog signal. This was originally designed to enable the testing of fancy transmission algorithms beyond the scope of this article. Here we used it for recreating the test signals that are received by the receiver board as a quick check that everything works. Schematics and a board layout are also included on the *Circuit Cellar* FTP site.

POWER SUPPLY

Any ripple or noise on the power lines reduces the chances of maintaining signal resolution through the amplifier and filter stages and compromise the workable range of the 16-bit ADC. Rechargeable lithium batteries are chosen as the power source along with quality Linear Technology linear regulators to produce the ±5-V power rails. Using batteries has the strong advantage of producing an output free of high-frequency ripple.

I added some low-battery-voltage indicators using Analog Devices ADCMP361 comparators. These have a built-in voltage reference. Trying to work out how to hook one up to the negative supply battery and the –5-V output rail to indicate when the battery voltage was flagging took a little figuring!

Capacitance is your friend when it comes to smoothing the supply lines. Batteries can react in the kilohertz range to changes in load. Capacitors can react in the megahertz range. This means that a capacitor attached directly to the supply pins of each IC provides a little power supply adjacent to the chip that can react at the same frequency that the chip is operating at. Extra capacitors adjacent to the power board connectors on the main board further add to this decoupling of the battery supply from the instantaneous demands of the circuitry. At the relatively low frequencies that this board works, capacitor type and placement is not as critical as when working at high frequencies.

I initially used some ribbon cable IDC connectors to bring the power to the main board and doubled up on each pin in order not to exceed the maximum allowed current draw through the sockets. Although this particular set of connectors is excellent for digital signals, I went for a more robust and larger Tyco 0.1″ IDC connectors on the board revision, as they have much larger pins, which gives a tighter connection with a complete overkill of current rating. I checked that I could obtain the connectors from more than one local supplier. While you can often drop a replacement amplifier into a design without having to alter any of the layout, connectors are a pain to replace as they usually have a unique footprint. Figure 3 depicts the power supply's circuitry and layout.

BOARD SUCCESS

The board successfully amplified, filtered, and digitized the signals received by all eight transducers. Clamping the foils is excessively fiddly as it relies on an operator to accurately place the transducers. A more consistent system of transducer clamping needs to be found if the device is ever scaled up.

The transmission cable termination method proved affordable and reliable, showing I do not need to invest in the connector manufacturer's expensive crimping tools for low-volume production. I proved that it is possible to populate and hand-solder a ridiculously complex board. The cost to my sanity may have been excessive, though!

Author's note: MDR Cable assembly notes are available on my website: http://sites.google.com/site/hardwaremonkey/home/cable-assembly. Mohammed Alloulah made a substantial contribution to this design during his ongoing PhD research on signal processing, especially with filter design and ADC circuitry. I hope that the final circuit will contribute to his thesis.

Matt Oppenheim (matt.oppenheim@ gmail.com) holds an MSc in Mechatronic Systems Engineering from Lancaster University. He splits his time working offshore as a Chief Geophysicist for Polarcus onboard seismic survey ships and as a Research Assistant at InfoLab21, Lancaster University. Matt's first love is analog technology, but he can be persuaded to work on digital projects as well. In his spare time, Matt enjoys cycling and drawing cartoons.

PROJECT FILES

To download the project files, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2010/244.

REFERENCE

[1] M. Hazas, "Indoor Location Systems," PhD Thesis, Dept. of Engineering, Cambridge University, UK, 2002.

RESOURCES

H. Johnson, "Multiple ADC Grounding," *EDN*, 2001.

Linear Technology, "LTC1569-7: Linear Phase, DC Accurate, Tunable, 10th Order Lowpass Filter," 1998.

R. Schmitt, *Electromagnetics Explained: A Handbook for Wireless/RF, EMC, and High-Speed Electronics*, Newnes, 2002.

Texas Instruments, "OPA129 Ultra-Low Bias Current Difet Operational Amplifier," SBOS026A, 2007.

SOURCES

ADCMP361 Comparator
Analog Devices, Inc. | www.analog.com

Eagle software
CadSoft Computer | www.cadsoftusa.com

LTC1569 Filter and LTC2050HV op-amp
Linear Technology | www.linear.com

ADS8329 ADC, OPA365 op-amp, and REF3133 voltage reference
Texas Instruments, Inc. | www.ti.com

Virtex-II FPGA Development board
Xilinx | www.xilinx.com

by Eric Ohmit (USA)

MCU-Based Aircraft Data Logging and Telemetry (Part 1)

Data Logger Development

Data acquisition is an essential part of any serious aircraft flight system. This series presents an innovative approach to designing an MCU-based data logging system for a small RC aircraft. The system's hardware is detailed in this article.

Have you ever wanted to know what was going on inside your model airplane while you were flying it? As my electric remote control (RC) models have become more powerful and I've attempted to achieve higher levels of performance, having the ability to collect in-flight data has become more critical. Heat is the enemy of lithium ion batteries, brushless electric motors, and their speed controllers. Lithium batteries are becoming more powerful and capable of delivering much higher currents than ever before (burst currents greater than 200 A). In addition, the cost of these electrical components and batteries makes them a significant portion of the cost of flying electric models. Knowing that my equipment is overheating during flight enables me to better protect my investment.

A few years ago, I designed a system that enables me to acquire in-flight information without having to remove my eyes from the model or take my hands off the RC transmitter. In this article series, I'll detail how I built my aircraft data logging and telemetry system (see Photo 1). In this first article, I'll introduce the hardware and the overall design. In Part 2, I'll describe the telemetry portion of the system.

SYSTEM FEATURES

I used the features of a Microchip Technology PIC24F processor to create the on-board data logger as well as a data downlink module and a base station capable of receiving the telemetered data and translating it to speech/audio

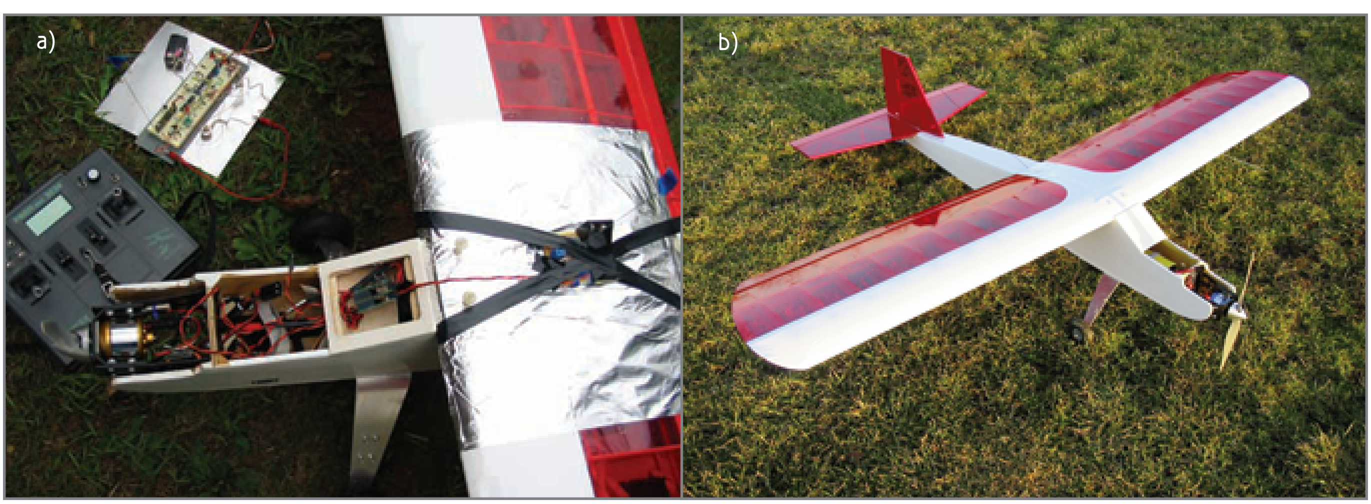


Photo 1a—The data logging assembly installed in an electric Sig Manufacturing LT-40 electric RC model. b—This is a complete view of the RC plane I used for testing.

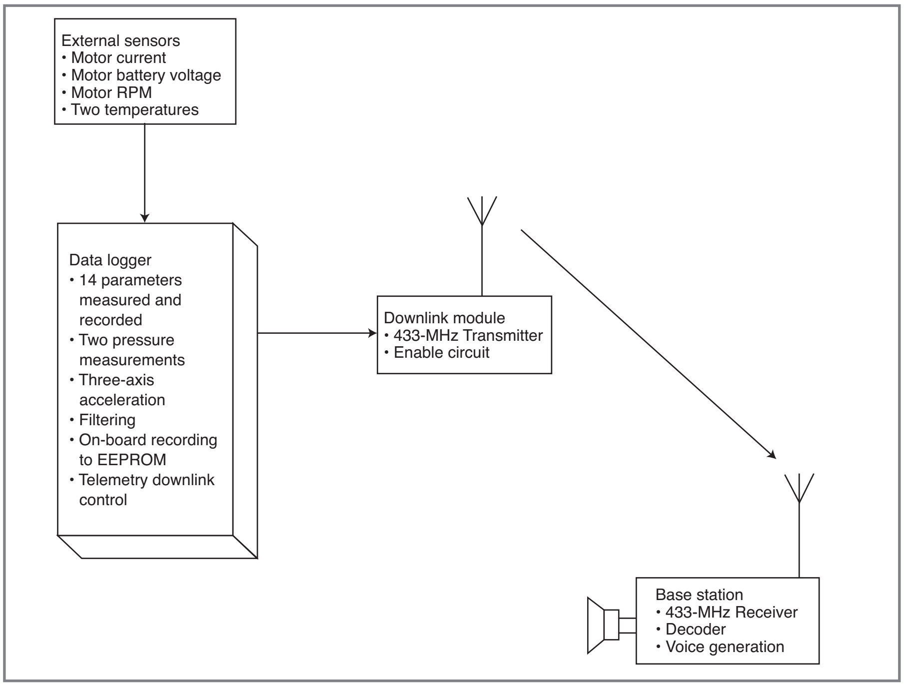


Figure 1—The system includes a data logger, downlink module, base station, and external sensors.

output. The Microchip PIC24F series of processors contains many features not found in previous processor products. Some of the features I used for this project include: a 16-bit Timer2 gated accumulation for measuring RC command pulse widths for control of the data logger and data downlink; a 16-bit Timer3 period register for periodic interrupts for sampling data every 40 ms (25 Hz); a 16-bit Timer5 as a counter for the measurement of electric motor RPM; Peripheral Pin Select pins to assign Timer2 input, Timer5 input, UART, SPI, and PCM (OC1) functions; and an internal 8-MHz oscillator and PLL module to obtain 32-MHz operation.

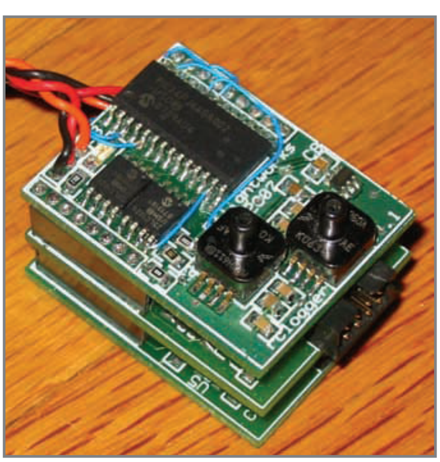


Photo 2—The data logger module

Photo 1a shows the complete system and its location on one of the model aircraft used. You can see the data logger and downlink module using a Radiometrix TXM-433MHz transmitter module mounted on it. The RC radio and base station are also shown. The base station uses the Radiometrix SILRX 433-MHz receiver to obtain the downlinked information.

Figure 1 is a block diagram of the system. The system consists of three elements. The first is the data logger located on the RC model. This is a standalone data logger capable of sensing and logging up to 14 parameters. The measured parameters include static pressure (altitude), total pressure (to compute airspeed), vertical acceleration, longitudinal (fore/aft) acceleration, lateral (side-to-side) acceleration, motor battery current, motor battery voltage, motor or propeller RPM, two remote temperatures, one PCB temperature, one spare parameter, RC radio supply voltage, and RC radio PWM width. These parameters are sampled and stored on EEPROMs. The second is the downlink module. This consists of a Radiometrix 433-MHz transmitter and the necessary circuitry to transmit "on demand" one or more parameters sensed by the data logger. The third is the base station, which is capable of detecting when the transmitter is sending a signal, receiving that signal, and converting it to an audio output consisting of a voice announcing a string of words in the earpiece.

DATA LOGGER MODULE

The data logger module is a self-contained device controlled by the PIC24FJ64GA002 processor in the SOIC form factor (see Photo 2). When assembled, the data logger measures a compact 1.1″ × 1.4″ × 1″ block, which contains the PIC24F processor, two 25LC1024 EEPROMs, sensors (pressure, temperature, and acceleration), and their associated antialiasing filters for noise and amplification where necessary. Interconnection buses to allow each board to communicate with the other are used. In addition, connectors to allow for the use of remotely mounted sensors and an interface to the downlink module are available.

The data logger is also interfaced with the RC radio receiver through an available RC channel, where it draws its power and measures the pulse width of the RC command channel. This measurement is used to determine whether to log data, as well as whether selected data should be sent over the downlink to the base station.

I developed the original layout of the main processor board PCB for the PIC18LF2525 processor (see Figure 2). It already used 3.3-V power, so modifying it for the PIC24 was easy. The boards are designed for surface-mount components, with the majority of the ICs being SOIC and most passive components being 0603, 0805, or 3216 sized. Assembly by hand is possible, although I've successfully used a self-designed solder oven (derived from previous *Circuit Cellar* and *Elektor* articles) using a commercially available toaster oven.

Several differences exist between the power and ground pinouts on the PIC18 and PIC24. As a result, I made several cuts and jumpers on this layout to properly power the PIC24 (seen as blue jumpers on the assembly). Maintaining the connections to all of

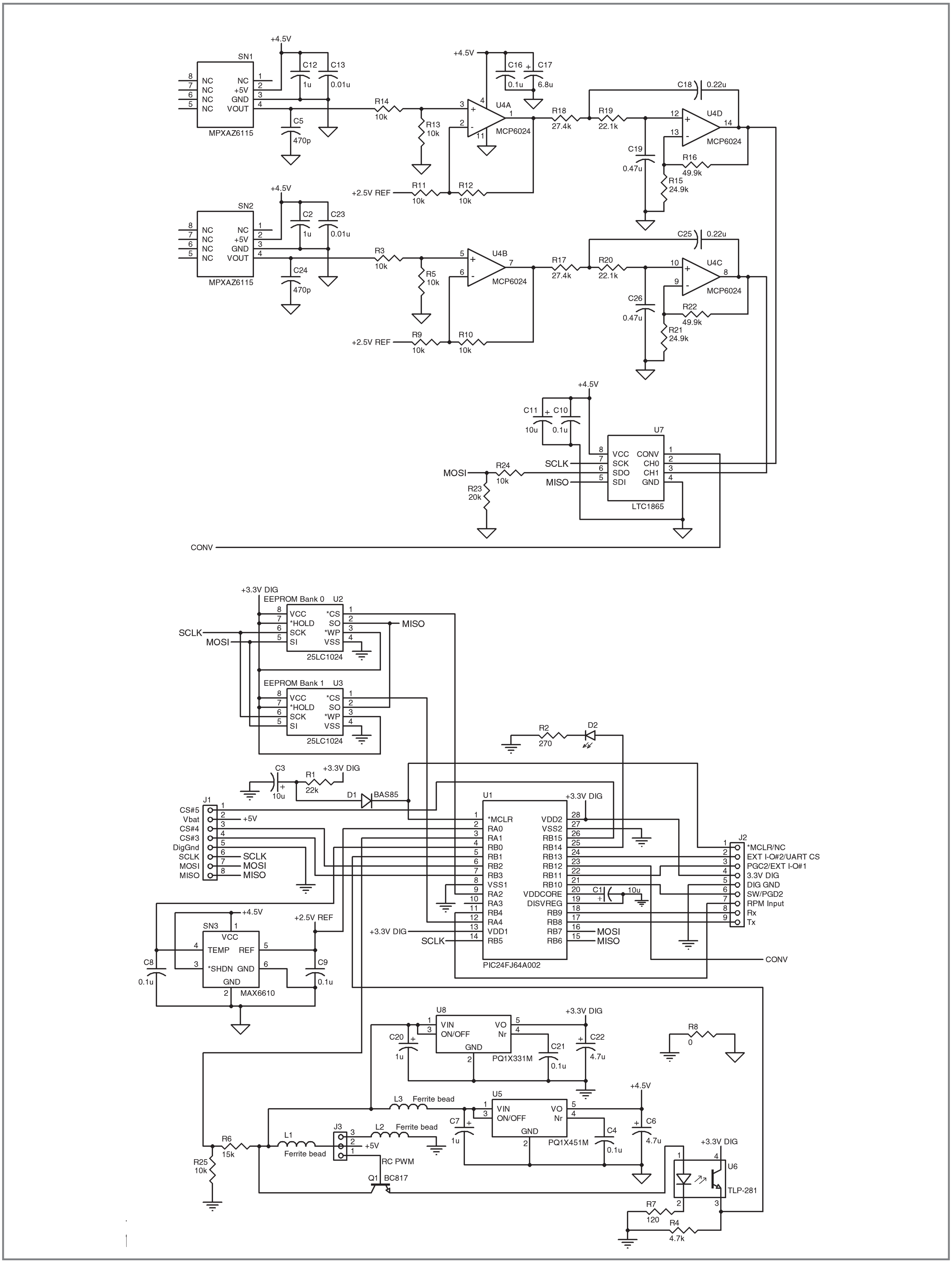


Figure 2—The data logger's main board features a PIC24FJ64GA002 microcontroller.

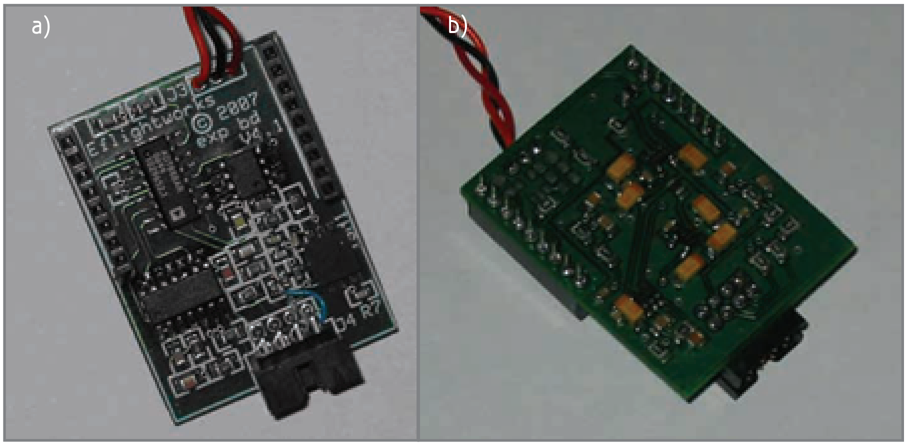


Photo 3—The top (a) and bottom (b) sides of the data logger's expansion board

the other components over the SPI, UART, and chip selects was easier with the PIC24's peripheral pin select (PPS) feature.

The components on the main processor board are the processor, an LED for indication of status, two 25LC1024 EEPROMs for data storage, and two Freescale Semiconductor (Motorola) MPXAZ6115 pressure sensors for measuring static pressure (for altitude) and total pressure (combined with static pressure is used to obtain airspeed). These signals are measured with a Linear Technology16-bit LTC-1865 ADC, which is interfaced to the PIC24 via the SPI bus. A temperature sensor (MAX6610) is located on the board, which also provides the 2.56-V reference used in several parts of the circuitry and analog measurements. Analog low-pass antialiasing filters are used to minimize noise due to sampling as well as adjusting the voltage from the pressure sensors with an offset (–2.56 V) and gain (3×) to improve sensitivity of the measured pressure. This adjustment reduces the useful range of the altitude signal to approximately –2000′ to 10,000′, but this is more than adequate for RC model use and it provides a resolution of less than 0.2′ per bit at sea level. The board utilizes a voltage divider to measure the RC radio's supply battery voltage and the MAX6610 temperature with two of the PIC24's 10-bit ADCs. The MAX6610 2.56-V output is used as the $+V_{REF}$ reference for the PIC24 ADC as well. The nominal 4.8-V RC radio battery powers the 3.3- and 4.5-V voltage regulators. The 3.3-V power is used for the digital portions of the board (PIC24 and EEPROMs) and the 4.5-V power is used for the analog measurements. The Sharp PQ1X331M and PQ1X451M regulators I used have a drop-out voltage of less than 0.3 V at the currents used. Plus, they're small (SOT-23-5) and have excellent noise characteristics. The analog and digital grounds are tied at one location to

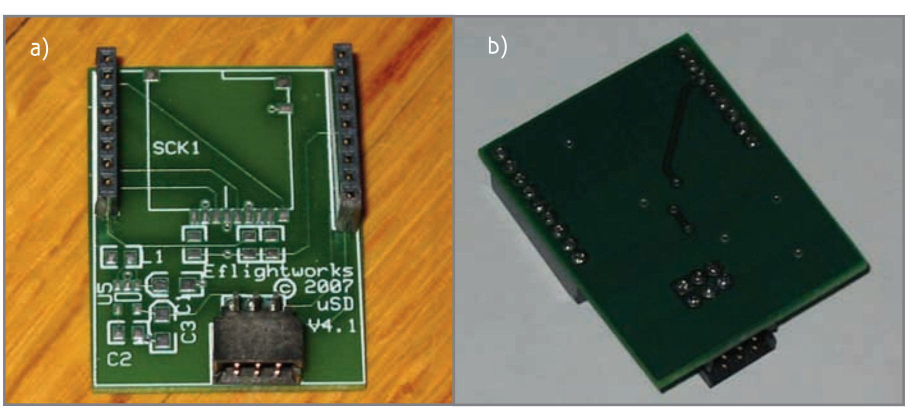


Photo 4—The top (a) and bottom (b) sides of the data logger's microSD and UART interface board

minimize noise. Lastly, the RC radio's command is measured through an optoisolated interface to minimize the potential for interference with the radio's reception. Ferrite beads are used on all power connections to minimize interference and noise as well.

The second board in the data logger stack is independent of the processor (see Photo 3 and Figure 3). This board provides eight additional analog signal measurements and also provides an interface to either a Hall effect (Infineon TLE-4945) sensor or optical sensor (through the comparator) for measuring motor (or propeller) RPM. The comparator and its associated circuitry is not implemented when the Hall effect sensor is used. The output can be directly used with the PIC24F with the voltage divider.

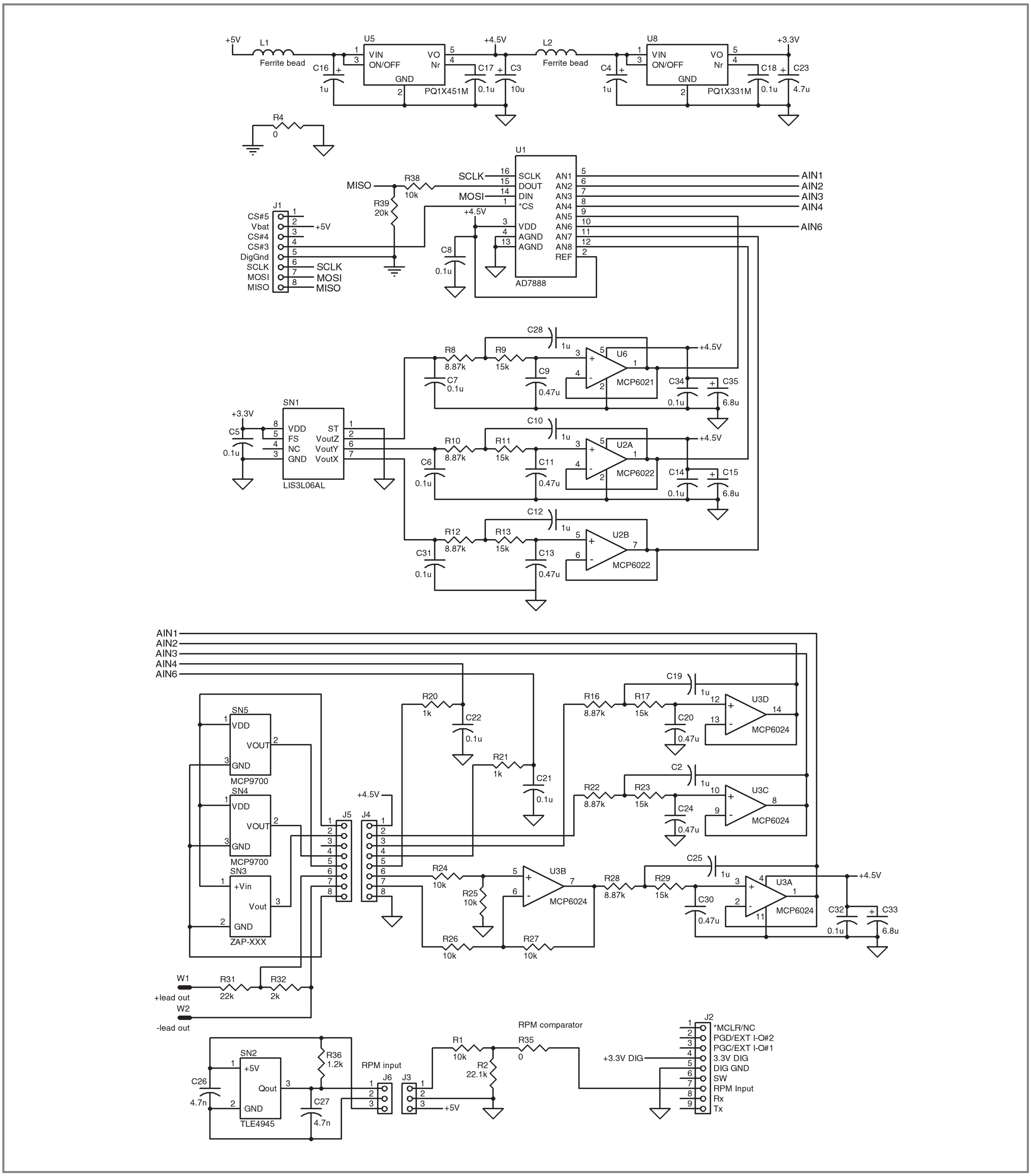


Figure 3—The data logger expansion board

Saelig
UNIQUE PRODUCTS & SUPPORT
www.saelig.com
Color LCD Scopes
Best Seller
2-ch + trigger standalone USB bench scope. $287 / $419
PDS5022S / PDS6062T
2-ch 1GSa/s Scopes
NEW! RIGOL
2-ch 1GSa/s (25GSa/s equiv.) 50/100 MHz scope. $595 / $795
DS1000E
Scope + Analyzer
NEW! RIGOL
25MHz 2-ch /16 logic scope and logic analyzer. $1095
DS1052D
Amazing 7 in 1 Scope! $199
CircuitGear CGR-101™ is a unique new, low-cost PC-based instrument which provides the features of seven devices in one USB-powered compact box: 2-ch 10-bit 20MS/sec 2MHz oscilloscope, 2-ch spectrum-analyzer, 3MHz 8-bit arbitrary-waveform/standard-function generator with 8 digital I/O lines. It also functions as a Network Analyzer, a Noise Generator and a PWM Output source. What's more – its' open-source software runs with Windows, Linux and Mac OS's! Only $199!
"I really like this scope adapter - it's meant for teaching electronic experiments but it's ideal for engineers too."
Alan Lowne
Saelig CEO
CGR-101
www.saelig.com
Handheld Scopes
20MHz / 60MHz rugged handheld USB 2-ch scope. $593 / $699
HDS1022MN / HDS2062M
Pen Scopes
10/25MHz USB powered scope-in-a-probe! Up to 100MS/s. $193 / $280
PS2104/PS2105
XP Emb Touchpanel
Touch-input 10.2" LCD 12V-powered Windows PC. LX800/512MB/4GB/ Ethernet/3xUSB/SD.
CUPC-80
Low-Cost Scopes
2-ch 40/100/200MS/s 8-bit scope range with 5/10/25MHz. $246 +
PS2203/4/5
Scope/Logger
2-ch DSO 16-bit DSO, FFT, VM, logic analyzer, standalone + 24 I/O.
Mephisto
Mixed-Signal Scopes
100MHz Scope, + Spectrum/Logic Analyzer and Signal Generator. $1259+
CS328
microCAM
Compact compressed-serial output camera module for any host system.
uCAM-232 / uCAM-TTL
Digital Microscope
USB PC µScope for PCB inspection, reports, Skype, etc.
PL1012 / PL1216
FREE COFFEE
Call 1-888-772-3544
to get a free Starbucks Card with your >$50 order!
While supplies last - not available with any other offers
Mention offer# SBW
USB Bus Analyzers
Best Value
Packet-Master™ - USB 1.1/2.0 analyzers and generators. $699 +
USB12 / 480+ / 500AG
16-Ch Logic Analyzer
Intuitive full-featured 16-ch 4MB 200MHz sampling memory. $240
LAD-16128U
SPI Bus Analyzer
Protocol exerciser/analyzer for standard SPI and non-standard 4-wire and 3-wire serial protocol interfaces up to 50 Mbps.
SPI Xpress
EMC Spectrum Analyzers
RF & EMF Spectrum Analyzer 1Hz to 7GHz for WiFi, mikes, etc.
EMC RF & EMC Spectrum
6-in-1 Signal Generator
Six signal generators in one small; USB-powered
WGM-201
Thumb-sized USB Tools
10MHz scope, 50MHz counter, 12MHz wfm gen, 100MHz pulse
USB Tools
Waveform Generator
USB2.0 speed 16-bit digital pattern or arbitrary waveform generator.
Wave Xpress
I2C Xpress
Versatile USB 2.0 I2C protocol exerciser and analyzer.
I2C Xpress
Automotive Testing
Kits turn your PC into vehicle-electrics diagnostic tool.
PS3423 / KLARI-MOD
CANminiBOX
Embedded controller series: 2 x CANbus, Ethernet, USB2.0, CF.
emPC-x133
RF Generator
High-res, extremely low-noise, portable 6GHz RF generator.
APSIN6000
RF Testing/EMI Tents
Portable RF test enclosures & shielding tents with external frame.
RF Testing / EMI Tents
Wireless Data Loggers
Log and display temp, hum, volt, event-time or pulse-counting data
RTR-50
Multiparameter Loggers
Mini-logger with built-in temp/hum/ pressure/3-axis accel sensors.
MSR145S
USB Loggers
Standalone USB temp / hum / volt / current loop data logger. $57+
EL-USB-1/2/3/4
SD to USB Adapter
Connect any SD slot (phones/PDAs) to a USB memory stick. Send files without a PC! No drivers needed.
Mobidapter
50MHz AWG
Versitile Function/Arbitrary/Pulse Generator offering precise waveform control demanded by engineers.
TG5011
TorqSense
Configurable, patented USB-output non-contact SAW digital rotary torque transducers with integral electronics.
RWT320
USB to I2C
"Drop-in" solution connects PC to I²C/SMBUS + 32 I/O lines. $89
USBI2CIO
FTDI USB ICs
Lowest Prices
Popular UART and FIFO chips. Upgrade Legacy designs to USB.
FT232RL
CAN-USB
Intelligent CAN connection from PC's USB port. $299
CAN-USB
Serial-Ethernet Cable
Network serial product easily without a PC using this 28" cable. $89
eCOV-110-P
Lorlin Switches
Fantastic array of stock and custom switching devices.
Lorlin
Sound Module
NEW!
14-pin module plays back pre-stored audio files from microSD card
SOMO-14D
Keyboard Simulator
USB board adds 55 I/O and 5 x 10-bit A/D inputs, 1 x 10-bit analog O/P.
PoKeys55T
Instant Ethernet
No OS needed. TCP/IP offload, ICs improve system performance.
WIZ110SR / W5100
Ethernet-IO
UDP/IP-controlled 24 digital I/O board 3 x 8-bit TTL ports.
Ether-IO 24
Custom Meters
Customizable high contrast 2.4" TFT color LCD with built in touch screen
RTG005
Wireless Solutions
Analog input, bluetooth wireless modules 433/868/915MHz.
EmbedRF / Adeunis
Temp/RH Sensors
Novel ambient sensors & modules accurately measure temp/RH.
UPSICAP / DLP-TH1
.NET Board
Tiny open source boards using USBizi chipset: Kits $98
FEZ Domino / FEZ Mini
Easy OLED Display
Compact, economical smart OLED with graphics drive from USB or RS232.
uOLED-96-G1
RF Modules
Simultaneously transmit composite video and stereo audio signals.
AWM6XX TX/RX
RS232 to 422/485
9p-9p or 25p-25p self-pwrd, isolated RS232-RS422/485
KK Systems
1/2/4/8/16 x RS232
Add 1-16 COMports via your PC's USB Port easily.
USB-COM
USB-Serial
A complete CP2102 USB-serial converter in a DB9 shell. $26
CE-USB
Saelig.com
unique electronics
888-75AELIG info@saelig.com
Above are some of our best-selling, unique, time-saving products - see our website for 100s more: WiFi/910MHz antennas, wireless boards, LCD display kits, Ethernet/IO, USB/RS232/485, USB-OTG, instant Ethernet-serial, CAN/LINbus, USB cables/extenders, line testers, logic analyzers, color sensors, motion controllers, eng. software, wireless boards, SMD adapters, I2C adapters, GPS loggers, automotive testing, security dongles, video motion detectors, crystals/oscillators, custom switches, barcode scanners, DSP filters, PLCs, Remote MP3 players, etc. FREE Starbucks card with your $50 order!
Check www.saelig.com often for special offers, bargains, business hints, blog, etc.

This is isolated from the analog circuitry on the board to minimize noise in the analog signals. Among the analog signals measured by this board are three-axis accelerations (measured by an STMicro LIS3L06AL accelerometer in the lower right part of the PCB). The ±6-g measurement range is selected by connecting the FS pin to 3.3 V. The other sensors are interfaced through the eight-pin, 2-mm Millimax connector. In my electric models, I typically use two temperature measurements (by MCP9700 TO-92 form factor sensors), electric motor current (measured by an Amploc ZAP-25, 50, or 100-A Hall effect sensor), motor battery voltage (through a voltage divider to scale voltages from up to 50 V max). One additional signal is available for expansion. Antialiasing low-pass filters are implemented on the board. The signals are sampled by an Analog Devices AD7888 12-bit ADC, which interfaces to the PIC24 (via the interconnection between the boards) over the SPI bus. Separate 4.5-V and 3.3-V regulators (also Sharp PQ1X451M and PQ1X331M) are located on this board. The 4.5-V regulator is used for the ADC reference and power to all analog signals with the exception of the accelerometer which uses 3.3 V. Power to these regulators is supplied by the 4.8-V RC battery on one of the interconnect pins between the boards. Ground, SPI SDI, SDO, SCLK, and chip select are also provided.

The third board in the data logger is only used to interface with the downlink module (see Photo 4 and Figure 4).

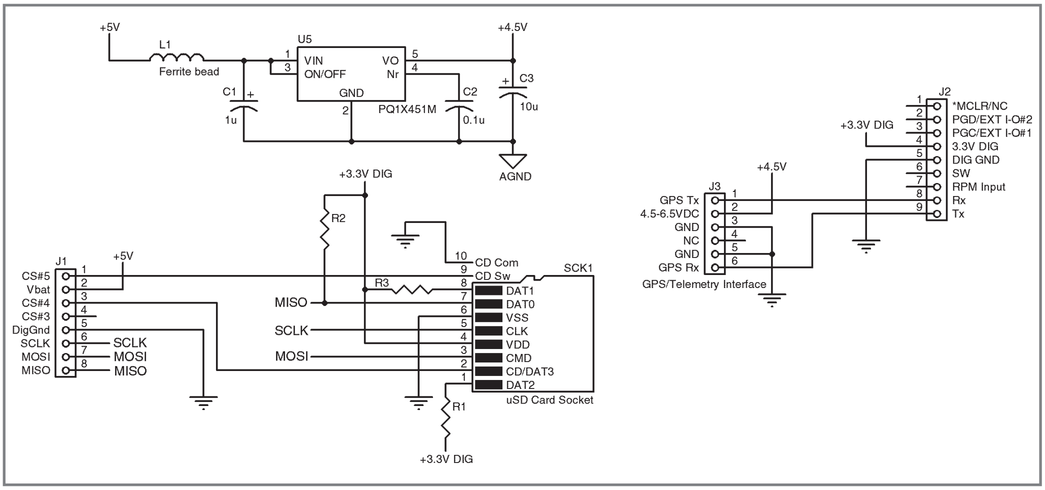


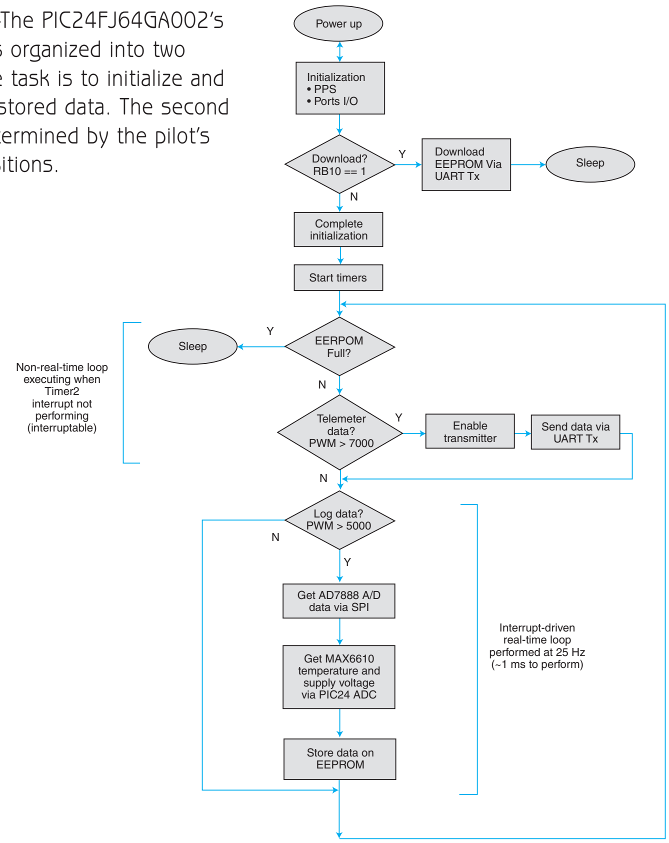
Figure 4—The data logger microSD and UART interface expansion board

Although it is capable of accommodating a microSD card socket, for use as a data storage media and interface with a GPS receiver, I did not use these features.

A software block diagram is shown in Figure 5. Software for the PIC24 in the data logger is organized into two tasks. The first is initialization and download of previously stored data (through a dongle that converts 3.3-V levels to RS-232 levels for interface with a laptop). If PORTB pin 10 is held high during initialization, data is downloaded from the EEPROM. The processor is put in Sleep mode after completing the download. This keeps the unit from potentially overwriting previously recorded data. After initialization (and if no download of EEPROM data is performed), the second task is executed, which is data logging during the real-time loop. Control of this logging is determined by the RC pilot's switch positions. The RC PWM is measured with the Gated Accumulation feature of the PIC24's Timer2. This feature will measure the time between an up-going and down-going pulse based on the processor's clock continuously with no intervention from software except to read the data.

Normally, the pilot leaves the radio switch in the first position when the system is initially powered. (This position is close to the minimum pulse width possible, approximately 1 ms.) This makes the software skip the logging of data. If the switch is put in the second position (near midrange of the pulse width, approximately1.5 ms), the data logging begins and runs until the EEPROMs are filled or the pilot returns the switch to the first position. This stops the logging until the switch is returned to the second position; logging will continue from that point. Moving the switch to the third position (approximately 1.8 ms) causes the logger to enable the transmitter on the downlink module and then send the temperature value (2 bytes) over the UART to the downlink module. In order to repeat the transmission, the switch must be returned to the second position, then to the third position for another transmission. This action has no effect on the data logging and is

Figure 5—The PIC24FJ64GA002's software is organized into two tasks. One task is to initialize and download stored data. The second task is determined by the pilot's switch positions.



outside of the real-time interrupt. When both EEPROMs are filled, logging is discontinued.

With the processor running at 8 MHz, timing checks of the software have shown the entire data logging frame can be completed in around 1 ms. The 25-Hz sample rate provides just over 10 minutes of data with the 25LC1024 EEPROMs.

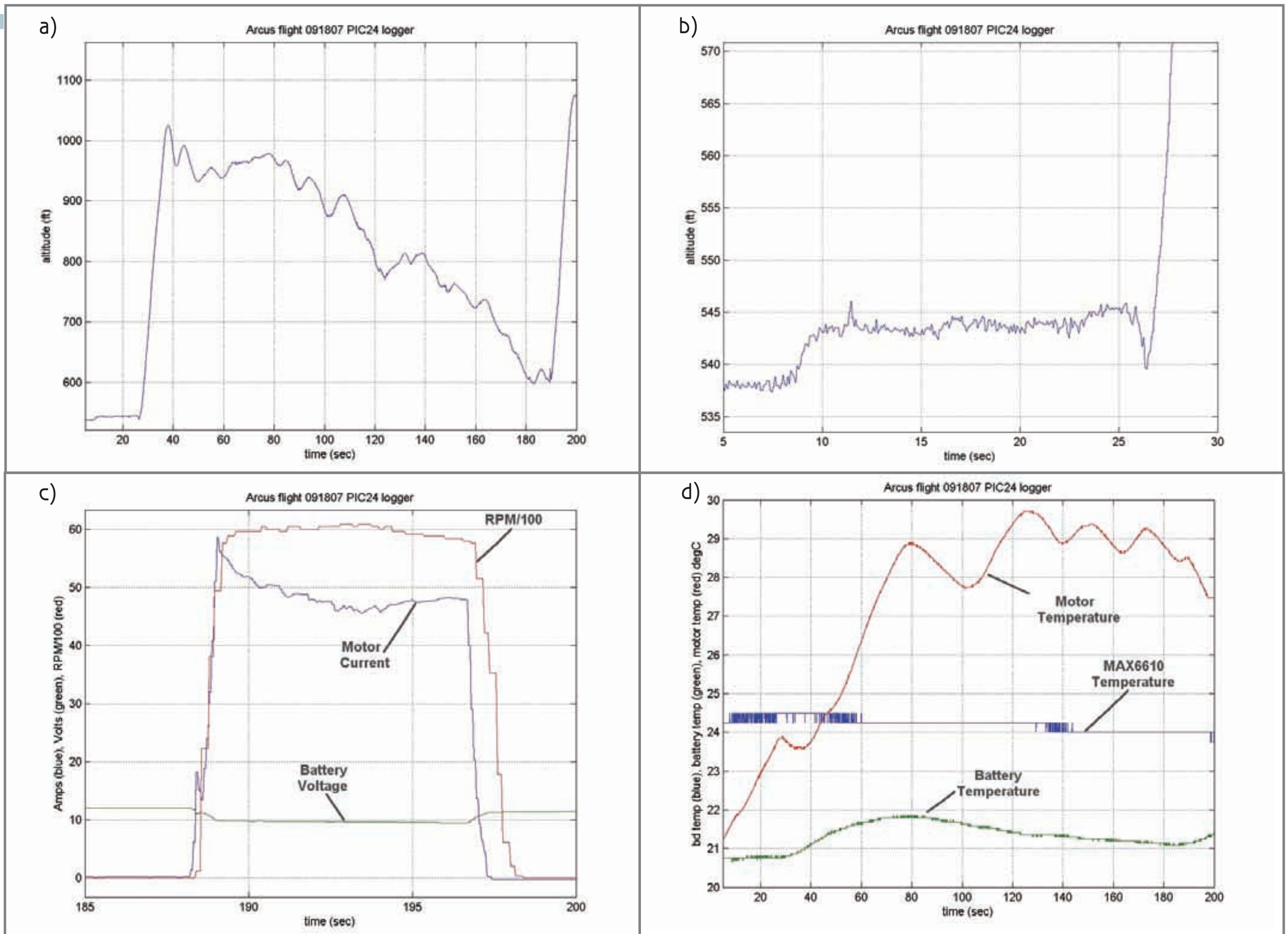
The logger has been used by itself to record data on several model aircraft. I used an RC electric glider (Robbe Arcus). Photo 1b shows a more conventional RC airplane (Sig Manufacturing LT-40) which was used for telemetry testing presented in Part 2. Photo 5a illustrates the altitude data from one of my flights with the electric glider. Note the rapid rate of climb (approaching 3500′ per minute) while the motor is running. After this climb, the motor is switched off and the aircraft glides, shown as a slow rate of descent until the motor is restarted and another climb occurs. Photo 5b shows the first 25 s of this same time history. Note the initial steady altitude, with a slight rise of about 5′, and then a rapid rise. This corresponds to the model resting on the ground with the logger running, the model being picked up and then thrown for the beginning of the flight. This shows the excellent sensitivity and resolution of the altitude signal.

Photo 5c shows the current and voltage of the motor's battery supply during this same flight (note the reduced time scale to better illustrate the signals). Also shown is the propeller revolutions per minute (RPM), note this is shown as RPM/100 to allow its overlay with current and voltage. Photo 5d depicts the three temperatures recorded during this same flight. Lastly, Photo 5e shows the accelerations from the three-axis accelerometer. Sample rates for this data were 25 Hz for the accelerations and 12.5 Hz for the remaining signals except RPM. The system measures RPM by counting the number of times a motor magnet north-to-south transition occurs as it passes the Hall effect sensor during each revolution. It is necessary to gather this data for longer than 80 ms (12.5 Hz) to achieve a good resolution. By accumulating the counts for 200 ms (5 Hz), the resolution was sufficient. The number of motor poles determines how many north-to-south transitions occur for each revolution of the motor shaft when using the Hall effect sensor.

This post-flight data has been very useful to me. It has allowed me to keep track of the performance of critical components (e.g., battery and motor) as well as assess the stress on my equipment. Battery, motor, and speed controller temperatures can all be evaluated during flight. Vertical load factor during maneuvering can be used to determine if I am over stressing the model, to satisfy my curiosity of how much load the model is pulling through an aerobatic maneuver, or to indicate how smoothly I am flying.

ON TO TELEMETRY

In this first installment, I described the airborne data logger I developed using the Microchip PIC24F64GA002. The next installment covers the telemetry capability. I'll also



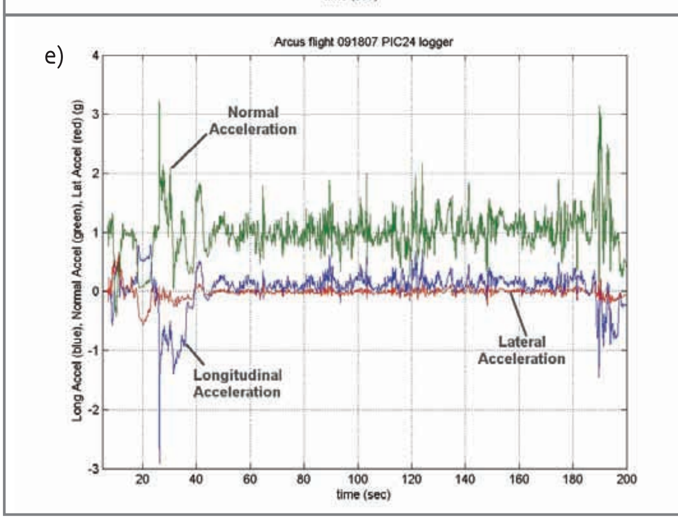


Photo 5a—This is the altitude time history from the data logger on an RC electric glider. b—The first 25 s of the altitude time history from the data logger on an RC electric glider. c—The voltage, current, and RPM/100 time history from the data logger on an RC electric glider. d—Temperature time histories from the data logger on an RC electric glider. e—Acceleration time histories from the data logger on an RC electric glider.

cover the synthesis of a useful audio feature: temperature data is transmitted to the base station, interpreted, and then announced to the pilot via an earpiece.

Eric Ohmit (eohmit@gmail.com) is an Aeronautical Engineer with a BS in Aeronautical and Astronautical Engineering from Purdue University. He has worked in the aerospace industry for almost 30 years, specializing in flight control systems. Eric has been building model airplanes, rockets, and electronics designs for over 40 years. He is also a semi-active Ham, KC2FKV.

PROJECT FILES

To download the project files, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2010/244.

SOURCES

LTC-1865 ADC
Linear Technology | www.linear.com

PIC24FJ64GA002 Microcontroller
Microchip Technology, Inc. | www.microchip.com

MPXAZ6115 Pressure sensors
Freescale Semiconductor, Inc. | www.freescale.com

SILRX Receiver and TXM 43-MHz transmitter
Radiometrix | www.radiometrix.co.uk

PQ1X331M and PQ1X451M regulators
Sharp Microelectronics | www.sharpsme.com

by George Adamidis (Greece)

Digital Audio Spectrum Analyzer

You can embed this digital audio-frequency spectrum analyzer in an audio system or use it as a stand-alone unit. The real-time system computes the distribution of the audio signal energy to 20 specific frequency bands and displays it on a 20 × 20 LED display.

When I was working on a radio station some years ago, I noticed the usefulness of a device called an audio spectrum analyzer. It produces nice visual effects and simultaneously provides an effective observable way to inspect audio performance during playback, recording, or any tonal adjustment (bass, treble, mid-frequency adjustment, or adaptive equalization).

After learning about the advantages of an audio spectrum analyzer, I began thinking about building my own. I wanted a hardware unit that I could easily embed in any audio device or use as a standalone device in my recording studio. To start my project, all I needed was some background knowledge about signal processing, a digital signal processor (DSP), an ADC, a microcontroller, op-amps, and a display. My desire was to build a powerful yet simple and inexpensive device.

I'm glad to say that after searching for information in books and purchasing components on the Internet, I built my own audio spectrum analyzer (see Photo 1). In this article, I'll describe the steps I took to make my dream come true.

SIGNAL PROCESSING CHIP

When I started this project, I was already familiar with PIC programming, so I immediately noticed Microchip Technology's dsPIC 16-bit digital signal controllers while doing some research on its website. I realized that a dsPIC would be a nice match for my low-cost, simple project. Why? The powerful silicon device is both a DSP and a high-performance 16-bit microcontroller in one package.

I chose a dsPIC30F6012A from among the many dsPICs because it has enough I/O pins and internal data memory for my project (see Figure 1). It also contains a 16-channel, 12-bit internal ADC that further simplifies the design.

SPECTRUM BASICS

Before I elaborate on the details of my project, let's consider a few questions. What is a spectrum, and why would you want to analyze one? Your normal frame of reference is time. You can use an oscilloscope to view the waveform of a signal in the time domain. Why do you need a spectrum?

Around the start of the 19th century, theorists discovered that any time-domain electrical phenomenon is made up of one or more sine waves of appropriate frequency, amplitude, and phase. In other words, you can transform a time-domain signal into its frequency-domain equivalent. The main advantage of the frequency domain is that with proper analysis you can perform extremely useful measurements, which are impossible to do in the time domain. Perhaps the most important measurements in the frequency domain are those that reveal the amount of energy present at each particular frequency. This is quite useful information for musicians or sound engineers in the case of an audio signal; it's advantageous for RF technicians in the case of an RF (radio frequency) signal.

Some measurements in the frequency domain require that you preserve complete information about the signal frequency, amplitude, and phase. This type of analysis is called vector signal analysis. However, another large group of measurements can be made without knowing the phase relationship among

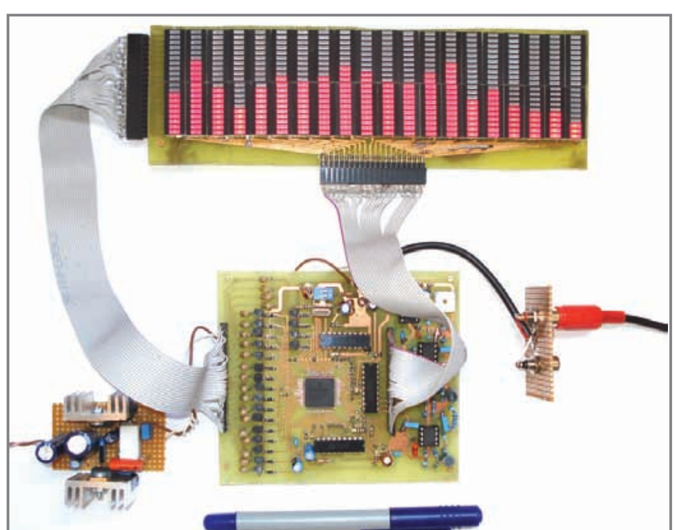


Photo 1—The complete audio spectrum analyzer

the sinusoidal components. This type of signal analysis is called spectrum analysis. The electronic devices used for vector signal and spectrum analysis are called vector analyzers and spectrum analyzers, respectively.

There are mainly two approaches to building a spectrum analyzer or a vector analyzer: the sweep-tuned method and the fast Fourier transform (FFT) method.[1] For my project, I used the FFT method, so my audio spectrum analyzer is a Fourier-type analyzer. As such, it digitizes the time domain signal and then uses DSP techniques to perform an FFT and displays the signal in the frequency domain.

The FFT algorithm is mainly a powerful vector analysis tool. My Fourier-type analyzer is actually a vector analyzer. It retrieves complete information about the signal frequency, amplitude, and phase and stores the information in a digital memory. However, for simplicity, it displays only the signal amplitude in the frequency domain. That's why I call it a spectrum analyzer.

HARDWARE

The first problem I faced during the design process was the display. I had an idea about how to perform the FFT in a dsPIC, but it was difficult for me to decide how to display the frequency spectrum. I had to choose between a high-resolution LCD module and a low-resolution simple LED display. My first thought was to use an LCD module, but I eventually dropped this idea (in hindsight, I regret the decision) and I chose a simple LED display. This was mainly a choice for hardware and software simplicity, speed improvement, and cost reduction. An average LCD module could offer me neither the speed I needed nor the low price.

I decided to build a large 400-LED display with a viewing area of about 20.3 cm × 5.1 cm. I built it using 40 commercially available 10-LED bar graphs, as you can see in Photo 2 and Figure 2. I wanted to drive this display directly from the dsPIC30F6012A as a matrix without using I/O expanders or any specific display drivers. I needed 40 I/O pins for this, which is almost all the available I/O pins of the dsPIC30F6012A!

After my decision about the display, I also had to think about how to plot the frequency spectrum. Just looking at the LED display, I realized that a 20-band bar graph energy plot would be the best option. But another problem came up: what should be the central frequency for each band? I used mathematics and my knowledge of audio perception to find the best answer to this question.

The audible frequency spectrum lies from 20 Hz to 20 kHz, which is about 10 octaves. If you try to split this spectrum

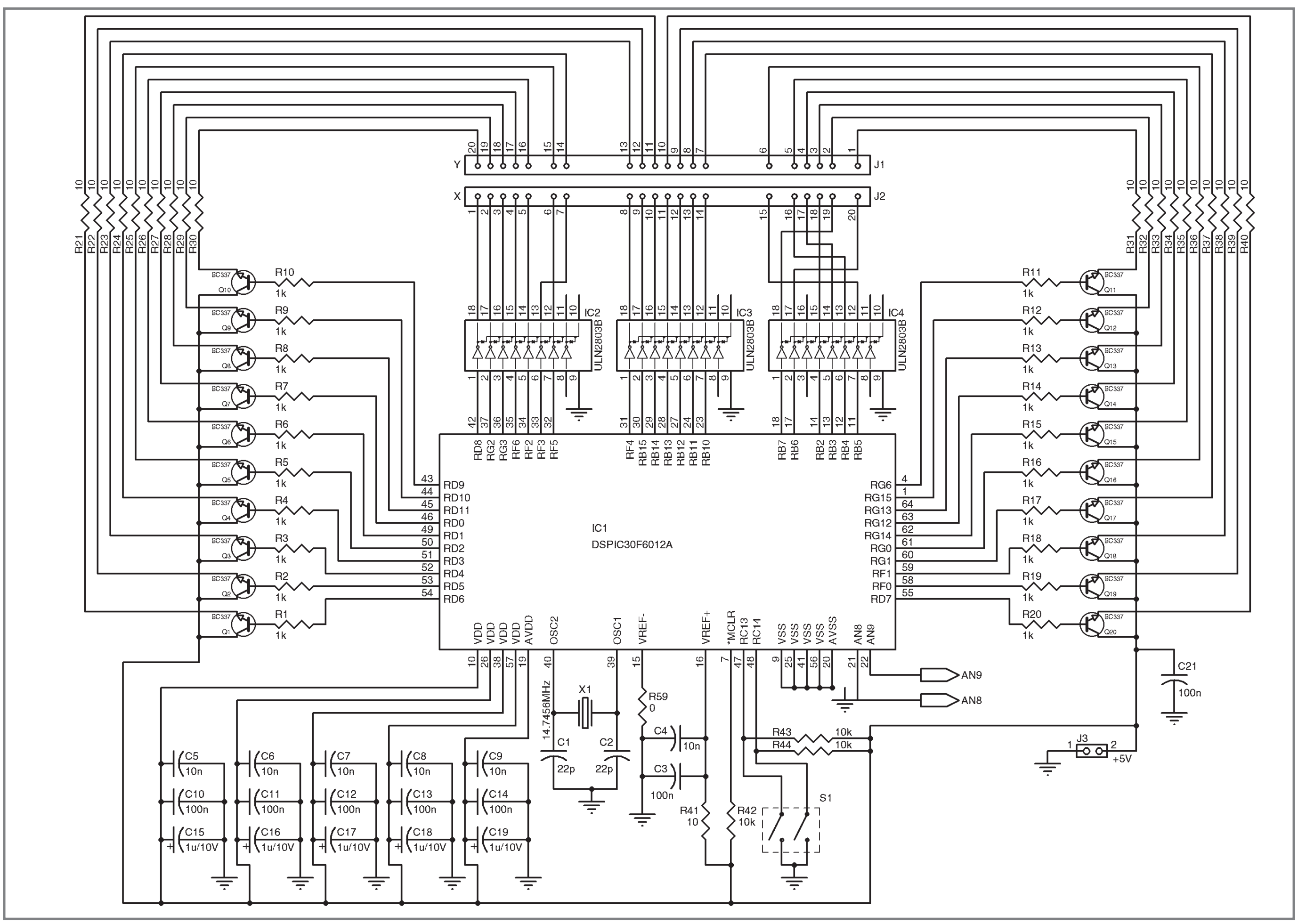


Figure 1—Take a look at the digital portion of the circuitry. The design is built around a Microchip Technology dsPIC30F6012A. You can also see the x and y current drivers.

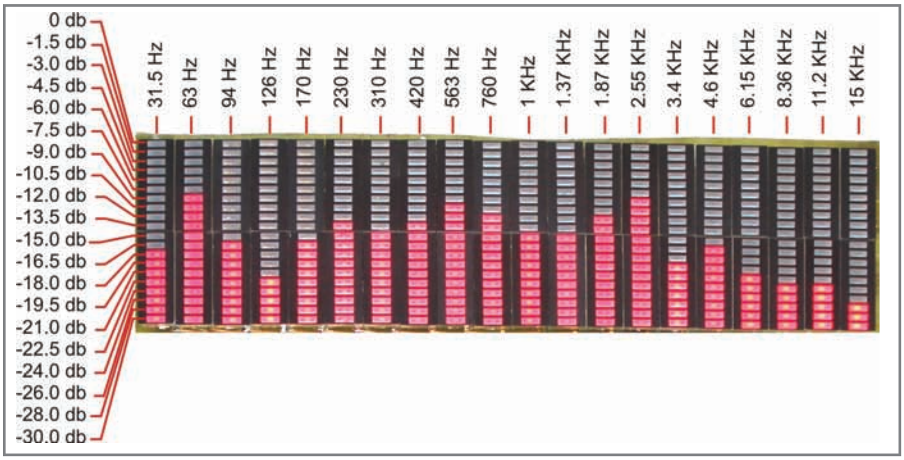


Photo 2—A close-up shot of the display. The spectrum is displayed in 20 bar graphs.

into 20 bands, each band must have a central frequency CF(i), which can be found with Equation 1 by setting a = 10/20 = 0.5 octaves and CF(0) = 20 Hz:

$$CF(i) = CF(0) \times 2^{i \times a},$$
where i is an integer from 0 to 19 [1]

Solving Equation 1 for a = 0.5 and CF(0) = 20 Hz, you get CF(i) = {20, 28, 40, 57, 80, 113, 160, 226, 320, …, 7240, 10240, 14500 Hz}. Looking at these frequencies, it's clear you need a frequency resolution up to 8 Hz in the lower boundary of the audio spectrum. Unfortunately, the maximum frequency resolution I could achieve in my hardware was about 31 Hz, so I needed a more reasonable set of frequencies.

I chose 31, 62, and 93 Hz to be the central frequencies for the first, second, and third bands, respectively. I also chose 15 kHz to be the central frequency of the twentieth band. Then, in order to find the remaining frequencies in the set, I had to calculate CF(19) = 15 kHz and CF(2) = 93 Hz, according to my aforementioned considerations. This solution is:

$$CF(i) = CF(2) \times 2^{0.4314(i-2)},$$
where i is an integer from 2 to 19 [2]

The results are shown in Photo 2.

Then, I had to choose the length of the FFT for my project and also determine the ADC's sampling frequency. I solved my problem. The frequency resolution depends on the FFT's length and the sampling frequency. For an N-point FFT, at a sampling rate SR, the frequency resolution df is:

$$df = \frac{SR}{N}$$ [3]

The sampling rate also must comply with the Nyquist sampling theorem, which indicates that a continuous signal can be properly sampled only if it doesn't contain frequency components greater than half the sampling rate:

$$SR > 2 \times f_{MAX}$$ [4]

For an audio signal, and in a perfect world, the Nyquist limit for the sampling frequency could be as low as 40 kHz if f_{MAX} could be perfectly limited to 20 kHz. But in our imperfect world, in order to avoid using complex antialiasing filters (filters of great order), I chose a sampling frequency much greater than the Nyquist limit. I chose to over-sample at 80 kHz. With this choice, taking into consideration Equation 3 and also that N can be only a power of two, I reached the conclusion that N should be at least 4096 in order to achieve the required 31-Hz resolution.

But a 4096-point FFT cannot be implemented using a dsPIC in real time (due to memory, speed, and fractional nature restrictions). Thus, I had a new problem to solve. I tried to find the solution considering that there is no need for the 31-Hz resolution at frequencies above 126 Hz. By exploiting this observation, I reached the conclusion that if I used two different sampling rates and two antialiasing filters I could reduce the required

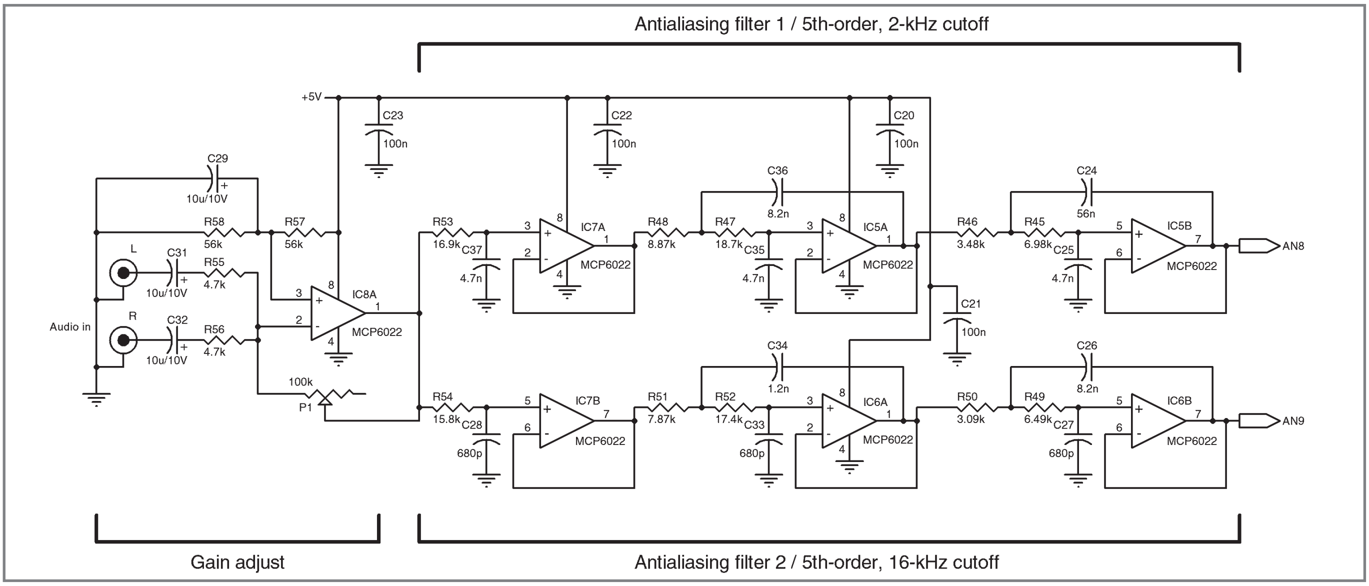


Figure 2—This is the analog portion of the design (antialiasing filters and gain adjust).

FFT's length. Thus, I decided to use two different sampling rates at 8 kHz and 80 kHz, two 256-point FFTs, and two antialiasing filters as shown in Figure 3.

The first FFT can provide the resolution needed for the low frequencies, while the second FFT provides a very good resolution for frequencies above 1 kHz. Of course, this is not the only possible solution, but it works perfectly with the dsPIC. After this exhausting investigation, I was finally able to design the block diagram of my audio spectrum analyzer (see Figure 4).

The antialiasing filters were designed using Microchip's FilterLab software and MCP6022 op-amps.[2] They were designed for a signal-to-noise ratio better than the dynamic range of my spectrum analyzer (which is 30 dB). The filters are quite simple and straightforward because the required filter order is limited to five (due to the oversampling).

You are probably wondering about the current drivers in my block diagram. What are they needed for? The dsPIC30F6012A can do both DSP and displaying in real time, but it cannot drive the LEDs with the current needed. That's why some current drivers must be used. You are probably also wondering how much current is needed for the large 400-LED display (see Figure 5). You'll be surprised that it is less than 600 mA. This is because the LEDs are not powered simultaneously, but are cycled use a scanning technique.

SOFTWARE

I wrote the software in C with Microchip's MPLAB C30 compiler and many routines from the compiler's DSP library.[3] The software implements a frequency analysis technique, which I call a "20-band parallel analysis filter algorithm using FFT." To demonstrate this technique, I must use some mathematics. (Refer to a reliable book on DSP for more details on FFT, complex numbers, and window functions.)

Consider the case when a single

tone, S, of unity magnitude (0 dB), at frequency f, is applied to the input of the device. This tone will be sampled at a sampling rate, SR. This will result in a sampling sequence S(n, f, SR) as described in Equation 5:

$$S(n, f, SR) = \sin\left(2\pi f \frac{n}{SR}\right),$$
$$n = 0, 1, 2, 3, \ldots, \pi = 3.14 \ldots \quad [5]$$

Consider a case where you store the first 256 samples of this sequence in memory and apply to them a 256-point, normalized, 0.5 Hamming window w(n). This results in Y(n, f, SR):

$$w(n) = 0.5\left[0.54 - 0.46 \times \cos\left(2\pi \frac{n}{255}\right)\right],$$
$$n = 0, 1, 2, \ldots 255 \quad [6]$$

$$Y(n, f, SR) = W(n) \times S(n, f, SR) \quad [7]$$

It is necessary to use a window function in order to ensure flat-top filter behavior in the system, according to DSP theory. Afterwards, you apply a 256-point FFT to Y(n, f, SR) (an FFT is actually the digital version of Fourier's algorithm), which results in X(n,f,SR):

$$X(n, f, SR) = FFT\left[Y(n, f, SR)\right] = \frac{1}{256} \times \sum_{k=0}^{255} Y(k, f, SR) e^{i \times 2\pi n \times \frac{k}{256}},$$
where i is the imaginary unit. $i = \sqrt{-1}$ and $e = 2.7182 \ldots$ [8]

Notice that S(n, f, SR), w(n), and Y(n,f,SR) are real signals, and X(n, f, SR) is a complex signal that contains a real (Re) and an imaginary (Im) part. Next, you compute the square magnitude of X(n,f,SR) and take P(n,f,SR), which is the discrete digital representation of the signal energy (or power) in the frequency domain.

$$P(n, f, SR) = \left\{Re\left[X(n, f, SR)\right]\right\}^2 + \left\{Im\left[X(n, f, SR)\right]\right\}^2 \quad [9]$$

Next, you define Bar0(f), Bar1(f), …, Bar19(f), from P(n,f,SR):

$$Bar\ i(f) = \sum_{k=u}^{v} P(k, f, SRj) \quad [10]$$

j, u, v are defined in Table 1 according to i. SR1 and SR2 are the sampling frequencies (7996.53 Hz and 87252.071 Hz, respectively). For example:

$$Bar\ 19(f) = \sum_{k=38}^{50} P(k, f, SR2) \quad [11]$$

Following this, you plot Bar0(f), Bar1(f), …, Bar19(f)—as shown in the "plots" diagram posted on the *Circuit Cellar* FTP site—and voilà: Bar0(f), Bar1(f), …, Bar19(f) are 20 band-pass filters, each one having a central frequency CF(i) according to Equation 2. The 20 filters are overlapping at half-power points (–3 dB below peak). Filters 4–9 and 12–19 are flat-top at –16 dB. Filters 0, 1, 2, 3, 10, and 11 have 1 dB more loss than the others.

After this analysis, you may think I am an expert in mathematics. Of course, this is not true! Actually, I did something simple. I just added subsequent FFT power samples, appropriately. Using the Fourier theory, I wrote the equations in PTC's Mathcad engineering calculation software. Just looking at the plots, I tuned the filters by setting appropriate values for j, u, and v. Anyone could do the same in order to retune the filters or to implement as many filters as he wants.

In conclusion, in order to implement the "20-band parallel-analysis filter algorithm," I wrote code for the following steps. Step 1: Acquire 256 samples of the audio signal at SR1 = 7996.53 Hz sampling rate. Step 2: Multiply by a 256-point Hamming window. Step 3: Compute the 256-point FFT of the resulting vector. Step 4: Compute the square magnitude vector P with the resulting vector X. Step 5: From vector P, compute Bar0, Bar1, Bar2, Bar3, Bar4, Bar5, Bar6, Bar7, Bar8, and Bar9 samples. Step 6: Acquire 256 samples of the audio signal at SR2 = 87252.071 Hz sampling rate. Step 7: Multiply by a 256-point Hamming window. Step 8: Compute the 256-point FFT of the resulting vector. Step 9: Compute the square magnitude vector P, of the resulting vector X. Step 10: From vector P, compute Bar10, Bar11, Bar12, Bar13, Bar14, Bar15, Bar16, Bar17, Bar18, and Bar19 samples. Step 11: Plot each Bar_i on the i^{th} LED bar graph.

The aforementioned steps describe in general the flowchart of the main routine of my software. Fortunately, most of the required tasks are included in Microchip's DSP library. But a small detail is not obvious. In my software, Steps 1 and 6 are executed almost in parallel with the other steps and not sequentially as it appears in the list. This is achieved through an interrupt routine. The ADC works independently from the main CPU and the DSP core. This way, the DSP and the CPU core are free to process data with FFT and display routines at the same time when the ADC is gathering new data. This small detail is actually responsible for making feasible real-time processing and displaying by a single dsPIC.

For Step 11, I wrote four different display routines. Each routine enables the spectrum to be displayed in a different

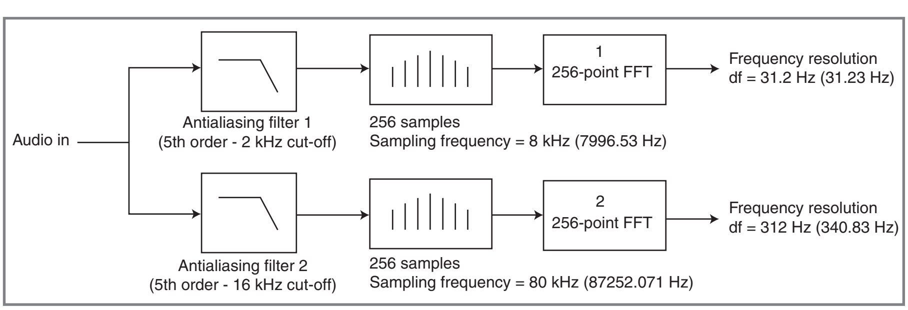


Figure 3—I used two different sampling rates and antialiasing filters.

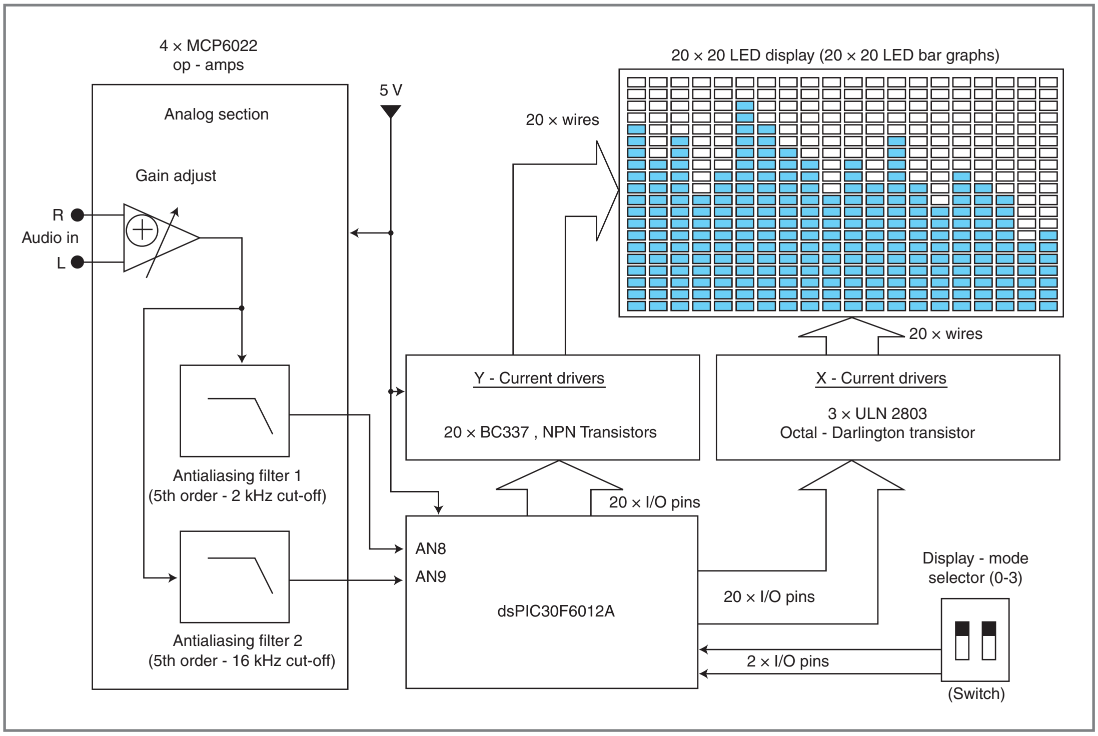


Figure 4—Here you see how each part of the system works with the others.

way (mode). My device supports four display modes. The user can select the display mode using the S1 switch. I named each display mode. Bars is a real-time bar-graph plot. Bars + Peaks is a real-time bar-graph plot with peak holding. Climbing Bars is a slow bar graph plot. (It gives the sense that the bar graphs are climbing to reach the peak. After catching the peak, they fall at a constant rate until the next peak becomes greater than the current one; otherwise, they fall at a constant rate.) Climbing Bars + Rain is a slow bar graph plot with peak holding, which gives the sense of rain. The bars are climbing to reach the peak, and then the bars fall at a constant rate A and the peaks fall at a constant rate B less than A until a greater peak comes up.

Note that the Hamming window must be normalized to 0.5 if you use a fractional DSP such as the dsPIC because Y(n,f,SR) must be less or equal to 0.5 (due to fractional FFT restrictions). Also note that the audio signal is a combination of many simple tones, which have random amplitude and

i	j	u	v
0	1	1	1
1	1	2	2
2	1	3	3
3	1	4	4
4	1	5	6
5	1	7	7
6	1	9	11
7	1	12	15
8	1	16	20
9	1	21	28
10	2	3	3
11	2	4	4
12	2	5	6
13	2	7	8
14	2	9	11
15	2	12	15
16	2	16	20
17	2	21	28
18	2	29	37
19	2	38	50

Table 1—These are the i, j, u, and v factors, which I use in my "20-band parallel analysis filter-algorithm," as per Equation 10.

phase. The FFT algorithm retrieves both the amplitude and the phase information from the input signal. This information is included in the X(n,f,SR) terms, which are complex numbers. For simplicity, my device displays only the Bar_i terms that contain only amplitude (or signal energy) information. Finally, note that Bar_i is the total power included inside the i band. Without the use of a window, you could not achieve flat-top behavior and you would also have too much noise. Any window can be used instead of the Hamming, but this will change the filter's loss and sharpness. The Hamming window offers less possible loss.

TESTING & DEBUGGING

After I soldered the hardware according to my schematic, I tested the analog section (the antialiasing filters) using an oscilloscope and a function generator (see Figure 2). I measured the frequency response of the filters and I found that it was as expected by the simulations in FilterLab.

The dsPIC was programmed and the code was debugged using Microchip's ICD2 in-circuit debugger and programmer. The debugging procedure was mainly focused on the display routines, which I think was the most difficult task in my code. The final tests were made using a HAMEG HM 8030-5 function generator and a PICO ADC-212 Virtual Instrument (oscilloscope and spectrum analyzer). The device was tested in various known-spectrum waveforms (sine, triangle, and square waves in various frequencies), and finally it was tested on real audio signals. You can refer to my online documentation in order to see pictures from the testing procedure and for further information.

RESULTS

I was amazed by the processing performance and speed of the dsPIC. When I applied a real audio signal, I immediately noticed that the audio spectrum was changing so rapidly that I was unable to watch it. That was the first proof of my success in building a real-time device, but it was also a problem! The fast real-time spectrum offers a spectacular visual effect, but it is not very helpful for a user who wishes to retrieve useful information from it. That's why I worked very hard to provide slow (not real-time) display modes in addition to the rapid (real-time) ones.

I measured the useful dynamic range of my device and found it's approximately 30 dB. This range is better than that of the common units found on commercial audio devices, but it's somewhat strange that the dynamic range isn't broader. I use a 12-bit ADC, so anyone would expect a dynamic range of about $20 \times \log 2^{12} = 72.2$ dB. Why didn't this happen as expected? This is because the FFT algorithm results in significant reduction of the dynamic range in a fractional DSP, due to fractional arithmetic restrictions. The only way to overcome this restriction is to use a floating-point DSP, which unfortunately is not now available on a dsPIC.

Since finishing this project, I've made three major modifications. I made the first modification because I discovered that my code could not be compiled in some newer versions of Microchip's MPLAB C30 compiler. This was mainly because some syntax was changed by Microchip since 2007. (I don't know why.) The second modification concerns the main clock frequency. In my original design, I forced the dsPIC to reach its limits running at almost 118 MHz. This wasn't really necessary and caused some significant heat problems. Due to these heat problems, I decided to reduce the speed to one half (59 MHz) without any hardware change and any noticeable performance degradation. I then had to retune my software (mainly

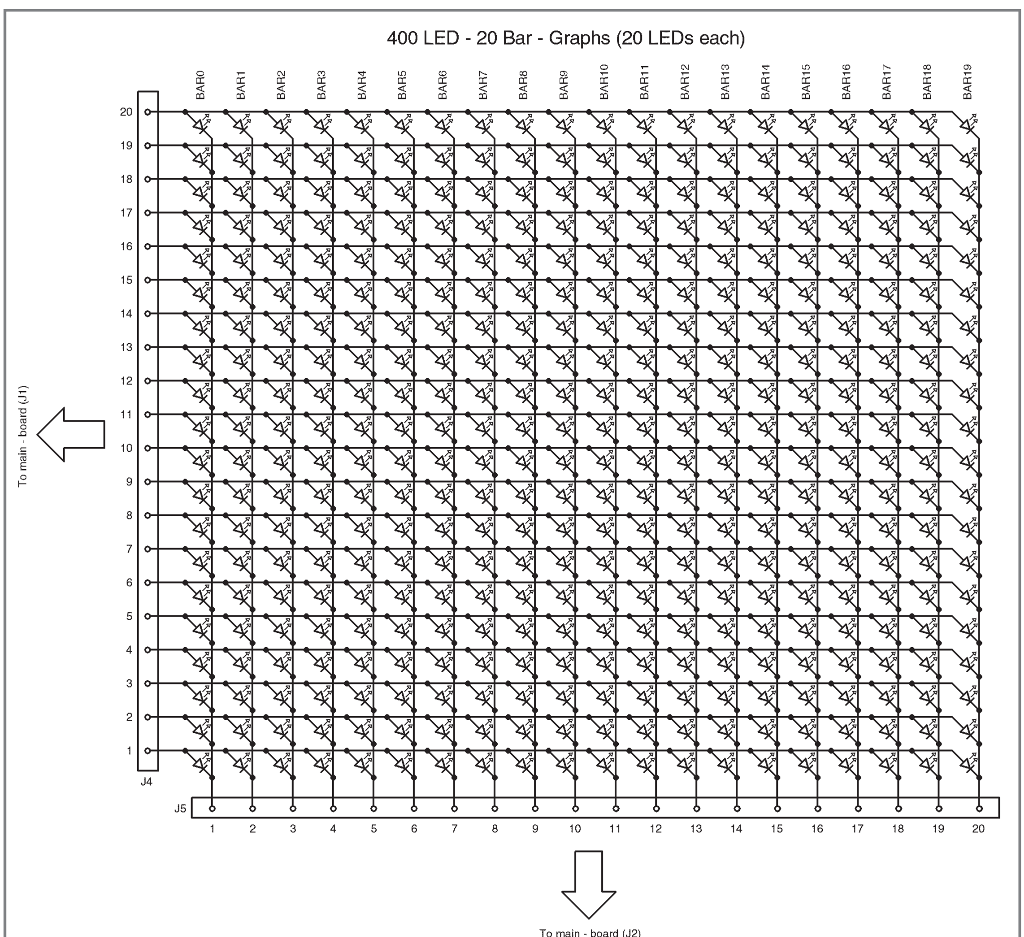


Figure 5—The 400-LED display

the ADC timing routines and the display frame rate). Finally, I changed the display scale step from 1.3 dB to a more convenient 1.5 dB. I also improved the dynamic range by 3 dB using a DSP trick in my code (in line 131 of the main routine).

FUTURE DEVELOPMENTS

I am very satisfied with my inexpensive yet powerful audio spectrum analyzer. I embedded this device in an audio equalizer and it looks great! But I think the main value of my project is not the device itself but in the algorithm I use. My method can be easily upgraded to an "N-band parallel-analysis filter algorithm" by anyone who wishes to build an analyzer with more than 20 bands.

Currently, the 400-LED display supports limited amplitude and frequency resolution. I intend to overcome this limitation in a future design. I hope I'll soon be able to display 32, 64, 128, 256, or 512 frequency bands, in real time, with the use of a high-resolution LCD and three antialiasing filters instead of two. My final intention is to build a professional-looking device with capabilities such as max hold, averaging, marking, and adaptive band configuration.

George Adamidis (sv7fid@yahoo.gr) is a physicist and electronics engineer. He received a diploma in Physics in 2000 and a diploma in Electronics in 2002 from the Aristotle University of Thessaloniki, Greece. George is a lab assistant in the Department of Electronics at the Technological Educational Institute of Crete (T.E.I. of Crete). He also teaches high school Physics and Electronics, and he is an assistant researcher at the Centre of Technological Research of Crete. George's research interests include embedded systems, RF technology, ham radio, and electromagnetic compatibility.

PROJECT FILES

To download the project files, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2010/244.

REFERENCES

[1] Agilent Technologies, "Spectrum Analysis Basics," Application Note 150, 2006.

[2] Microchip Technology, "FilterLab Active Filter Software Design Tool," Product Overview, 2005, www.microchip.com.

[3] Microchip Technology, "16-Bit Language Tools Libraries," DS51456C, 2005.

RESOURCES

L. Di Jasio, *Programming 16-bit Microcontrollers in C*, Newnes, 2007.

M. H. Hayes, *Schaum's Outline of Theory and Problems of Digital Signal Processing*, McGraw-Hill, 1999.

Microchip Technology, "dsPIC30F6011A/6012A/6013A/6014A Data Sheet," DS70143C, 2006, ww1.microchip.com/downloads/ en/DeviceDoc/70143D.pdf.

———, "MCP6021/1R/2/3/4 Data Sheet," DS21685C, 2006.

———, "Reading and Using Fast Fourier Transforms (FFT)," AN681, DS000681A, 2001.

Motorola, "ULN2803/ULN2804: Octal High Voltage, High Current Darlington Transistor Arrays," 1996, www.datasheetcatalog.org/datasheet/motorola/ULN2804A.pdf.

S. W. Smith, *The Scientist and Engineer's Guide to Digital Signal Processing*, Second Edition, California Technical Publishing, 1999.

SOURCES

dsPIC30F6012A DSC, MCP6022 dual op-amp, and MPLAB ICD 2
Microchip Technology, Inc. | www.microchip.com

ULN2803 Darlington transistor array
Motorola, Inc. | www.motorola.com

Mathcad engineering calculation software
Parametric Technology Corp. | www.ptc.com

by Brian Millier (Canada)

Sound Generator

Build a Sound Machine Around Three Basic Chips

As a skilled designer, you can probably build most of the electronics non-engineers have to buy. A sound generator is an example. Here you learn how to build one with a microcontroller, a DAC, and an audio amplifier chip.

One of the most under-appreciated things in life is a good night's sleep, although it becomes more elusive, and thus more appreciated, as you grow older. Technology can't help with your more frequent nocturnal trips to the bathroom, but there are a few other ways it can help.

Two common problems involve the difficulty in getting to sleep in the first place, as well as a heightened sensitivity to extraneous noises, which wake you up during the night. I have two dogs that are inclined to bark when they hear anything unusual. The obvious solution seemed to point to masking out such irregular sounds in some way.

I built a unit that can produce relaxing sounds—such as waves lapping on a beach—using looped sound files in order to produce a continuous, soothing pattern. This can be accomplished by obtaining such sounds in the WAV file format and placing them on a common SD memory card—easily done using a laptop computer. Playing back the sounds requires only three chips: a microcontroller, a DAC, and an audio amplifier chip.

BINAURAL BEATS

After starting the project, I came across the concept of "binaural beats," discovered in 1839 by Heinrich Dove. The various human brainwaves are all in the low-hertz range: much lower in frequency than the human ear can hear. However, it has been shown that if you produce two tones in the frequency range to which the ear responds well, and you make those two tones differ by only a few hertz, then the brain will act as the equivalent of an electronic mixer, and respond to the difference in frequency. By adjusting the frequency difference between the two tones, you can replicate the frequencies used by the various brainwaves. So, in theory, you should be able to reinforce the normal brain patterns involved in relaxation, or for that matter, concentration. I should mention that the effectiveness of this concept depends upon each ear receiving its own tone, so it tends to work best with either headphones or strategically-placed speakers in relationship to one's head orientation.

I decided to incorporate a binaural beat generator in my design. One simple way would have been to mix the two frequencies into all of the stereo sound files. As an amateur musician with a nice recording studio in my basement, this

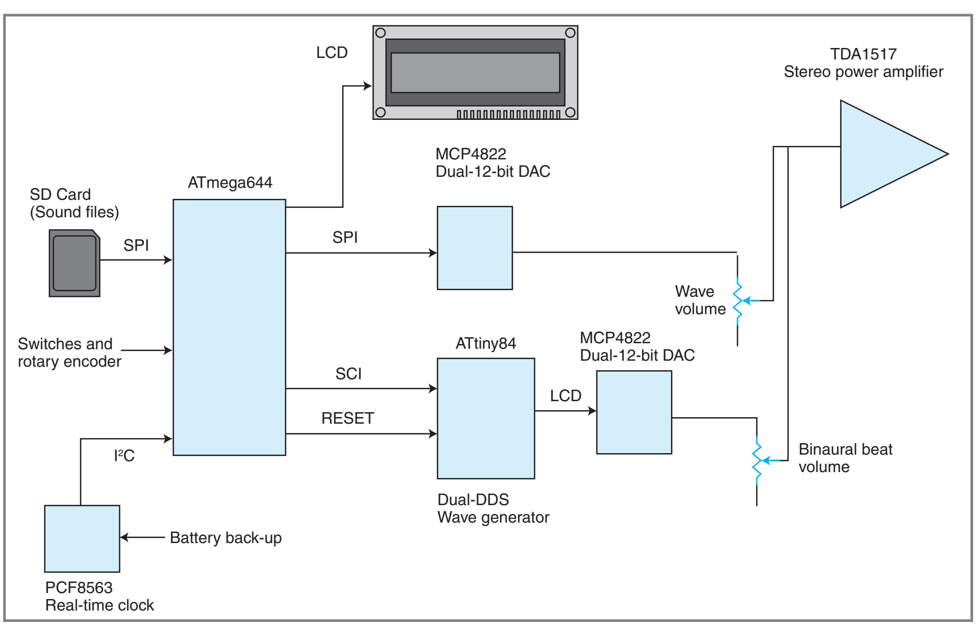


Figure 1—The main parts of the relaxation generator.

would have been easy. It would have lacked flexibility, however. Therefore, I chose to implement a dual-channel DDS synthesizer in my project, capable of producing two audio sine waves at accurately selected frequencies. This sounds fancy, but it required only the addition of a very modest Atmel ATtiny84 AVR microcontroller and a dual 12-bit SPI DAC. Since I was on a roll, I decided to add an alarm clock function to the project. I designed in several features to improve upon our present clock radio.

Even if you you're not particularly interested in building this project per se, you may want to read this article for a synopsis of Franz Voegel's AVR-DOS library. This library allows AVR users to readily interface to IDE hard disk drives as well as various forms of flash memory, under the FAT file system, using DOS-like commands. Also, if you are wondering about where to obtain excellent-quality, relaxing sound files, I cover that in this article as well.

CIRCUITRY

Figure 1 gives you a general overview of the project. Figure 2 depicts the circuitry. The main MCU is an Atmel ATmega644, which has all of the required I/O port functions as well as enough RAM to run the AVR-DOS library. The sound files are stored on an SD card, which plugs into a socket on the main board. SD cards can be interfaced using their own proprietary parallel protocol, or the more common SPI protocol, which is what I chose here.

When the sound is playing, the data from the WAV file is sent out to the Microchip Technology MCP4822 SPI DAC. I chose the NXP Semiconductors TDA1517 6-W stereo amplifier because it requires few external components and doesn't require a strict PCB layout as do the more modern class-D audio amplifier chips.

The binaural beat tones are produced by a dual DDS wave generator. This is implemented using an Atmel ATtiny84 MCU, which then feeds a second Microchip dual 12-bit DAC. The parameters required for the generation of the desired frequency tones are fed into the ATtiny84 via a 1,200-bps serial link from the host MCU. To save a crystal, I enabled the MCU clock output on PortB.1 of the ATmega644, which then provided the 20-MHz clock for the ATtiny84. Note that you have to program a fuse in the ATmega644 to enable this clock output, so make sure to do so (as well as setting the fuse which disables the JTAG interface—required to allow access to PortC for normal I/O purposes).

There was no simple way of incorporating digital volume controls into either the wave sound circuitry or the DDS wave generator. Therefore, I used two mechanical rotary potentiometers to

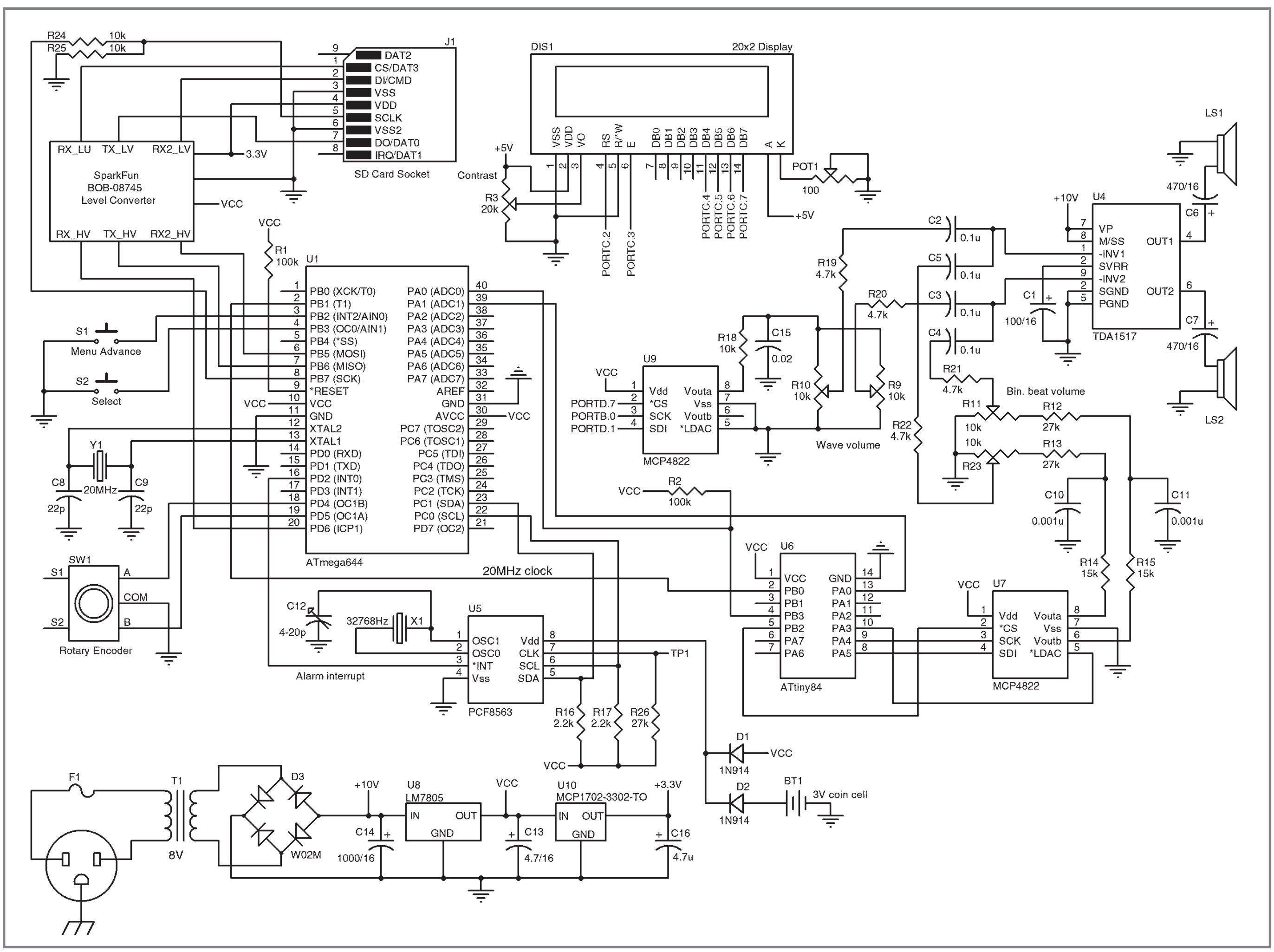


Figure 2—This is the relaxation generator's circuitry.

provide the necessary mixing function. The user interface consists of two push button switches—Menu Advance and Select—as well as a "parameter value" rotary encoder. I used a 2 × 20 LCD panel, with some custom "trickery" to provide the large-font readout for the clock display. All real-time clock functions are looked after by an NXP Semiconductors PCF8563 RTC chip, which is interfaced to the main MCU over an I²C bus.

While not shown in Figure 1, the power supply consists of a 10-V DC power supply that directly powers the audio power amp and is further regulated down to 5 V (for most of the digital circuitry) and 3.3 V (for the SD card).

AVR-DOS

There were several ways I could have approached the issue of storing the relatively large sound files. For a project such as this, it makes sense to use consumer-based flash memory devices such as SD cards or USB flash drives because they are found everywhere and their cost per megabyte is only pennies. Since a PC is where you are going to obtain the sound files in the first place, it makes sense to use a memory device that can be simply plugged into the PC, and then transfer the files to this memory device using standard Windows file functions. That being decided, you have the choice of using either USB flash drives or any of the various card formats used by cameras, cell phones, and so on.

Both categories are now so inexpensive that there is no cost advantage to either one. USB flash drives have the advantage of being totally universal, since every computer since the "dinosaur" models contains USB ports. I chose to go with the SD card format, though. Why did I choose this format?

Most laptop computers, including my own, have a multifunctional card slot which handles the SD format. Interfacing to a USB flash drive can certainly be done by the microcontrollers that I commonly use. This requires a USB host software stack as well as a DOS file-handling library. These are available for many popular MCUs such as the Atmel family that I prefer, as well as Microchip's PIC family. However, these stacks are written in C, and I am far more comfortable using the BASCOM-AVR BASIC compiler. You can get around these software complexities by incorporating the Vinculum VDrive2 module, which is basically a coprocessor module that handles the USB host function as well as the DOS file-handling routines. I used this module in a project outlined in a February 2010 *Circuit Cellar* 235 article, but it is not the best choice when you need to transfer a lot of data quickly.

Several years back, Franz Voegel wrote the AVR-DOS library routines which have been fully integrated into the BASCOM-AVR BASIC compiler. Both the FAT16 and FAT32 DOS file formats are supported. This library is free for personal use. The routines are well-crafted assembly language routines which execute very quickly and don't use up much flash program space or RAM.

I'm basically still "on the grid" (i.e., the 0.1″ grid). I prefer to build my projects on vector protoboards using DIP components whenever possible. In the case of Atmel MCUs in DIP packages, only the ATmega644 contains 4 KB of RAM. All lesser members of the Atmel AVR family contain 2 KB or less. While you can trim down the AVR-DOS library to run on 2 KB, there are speed penalties associated with this, so I didn't pursue this option.

The AVR-DOS file library is written to handle two categories of storage devices: standard IDE hard drives and several format variations of solid-state flash drives. Depending upon your choice here, you must then add a line to your source code to include the appropriate driver file to match your choice (e.g., config_harddisk, configMMC, config_CompactFlash, etc.). I've used both categories in projects myself, but choosing the IDE hard drive option does involve dedicating 20 or so I/O port lines just for the IDE interface itself.

AVR-DOS uses the standard BASIC file handling commands—such as `DIR`, `OPEN`, `CLOSE`, `PRINT #`, `INPUT #`—so it is pretty easy to get used to for BASIC programmers. It supports the three common BASIC file transfer methods: sequential, random access, and BLOAD/BSAVE. While I did not need them in this project, the library also contains functions to read a file's time and date stamp.

AVR-DOS only supports short DOS file names (8.3), but this is a common limitation among the various DOS file-handling libraries and disk coprocessor modules I've encountered. Long file names in DOS are handled in a rather convoluted way in order to preserve compatibility with the legacy 8.3 directory format. However, it isn't only this complexity that discourages the programmers of these libraries. Rather, it is the fact that Microsoft holds patents on the long-file-name format, and there are licensing fees involved in its use.

PLAYING WAV FILES

The first thing I had to determine was how fast my ATmega644 MCU could stream data from the SD card, since that would dictate the maximum sampling rate I could use for the WAV files I wanted to play. I set up a test program to read in 512-byte sectors and looked at the SD card's SPI lines on the scope to see how long it was taking to do this. It turned out it took a bit more than 1 ms to read in a sector, which translated into about 500,000 Bps. This looked encouraging since a standard stereo, 16-bit PCM WAV file, at the standard 44.1-kHz sample rate, requires only a 176,400-Bps transfer rate.

However, loading the WAV data in from the SD card is not the only thing going on simultaneously. The DAC must also be loaded with this WAV data, and to further complicate things, the DAC loading must take place precisely at the chosen sample rate. This can only be done using an interrupt service routine triggered by a timer in the ATmega644 at the chosen sample rate. This is necessary because the time it takes for AVR-DOS to return a 512-byte sector can vary from sector to sector due to DOS files not being contiguous. The location of the next sector must be "looked up" in the file allocation table (FAT).

Handling the DAC update routine in an ISR involves a fair bit of overhead. First off, the WAV data coming in from the SD card, in 512-byte sectors, has to be buffered in a 1,024-byte array. This allows one sector's worth of SD card data to be streamed into one half of the array, while at the same time, the DAC update ISR is feeding the DAC with data from the other half of the 1024-byte array. This involves using something like the "head" and "tail" pointers used in a circular buffer (FIFO).

Another issue is that the WAV file format must be manipulated a bit before it is sent to the DAC. Since the DAC is only 12 bits, the 16-bit WAV data must be shifted right four times. Also, the MCP4822 DAC requires its upper nibble to be set in a certain pattern. Like any ISR, this one requires that many registers be saved and restored each time the ISR is invoked, further slowing things down. Finally, there is the matter of reading the I^2C real-time clock chip and displaying it, in a customized large font, on the LCD.

Given all of the aforementioned factors, and after some experimentation, I decided that my hardware would comfortably handle a monaural 16-bit WAV file at 22,050 samples per second (a standard WAV rate). For the type of background sounds that I was planning on using, this format produced satisfactory sound quality.

In the Sources section at the end of this article, I list the URL for the website I used to learn enough about the WAV file format to allow me to write the necessary code to separate the actual WAV data from the other information found in the header section of the file. I have to say I was stunned to find that the WAV files I looked at contained a 4096-byte "header" at the beginning of each file, almost all of which was just "00" padding. I am sure there are plenty of similar cases of such a waste of space happening in Windows!

BINAURAL BEATS & DDS

As mentioned in the introduction, binaural beats are made up of two discrete sine wave tones, both of which are in a region where human hearing is quite sensitive—that is, the lower range of frequencies that make up the vocal range. Furthermore, there needs to be a frequency difference between these two tones in the fractional hertz to low-hertz range. The idea here is that each ear will respond readily to these tones, and that the brain will act as a "mixer," much like its electronic counterpart. This will produce, amongst other things, a signal whose frequency is the difference between the frequencies of these two tones. This low-frequency difference signal can be targeted to duplicate the various brainwave frequencies. Since different brainwaves are associated with different "moods" (i.e., relaxation, concentration, etc.), the concept is that binaural beats can be used to enhance relaxation or concentration.

What is the most effective way to generate these two tones? The preferred frequencies are in the 150 to 500 Hz range, which is a pretty low frequency to generate by digital means. However, the difference between the two frequencies must be small and accurately selectable. Finally, since these tones are right in the sensitive region of the ear's response, it is important that they be generated as pure sine-wave tones, or else any harmonic distortion will be noticeable and annoying.

All of the above criteria are best met using direct digital synthesis (DDS). In this low-frequency range, it turns out that MCUs can readily perform this task. Refer to Tom Napier's 1998 article titled "Digital Frequency Synthesis" (*Circuit Cellar* 99), in which he describes implementing DDS using a common MCU chip, or my article titled "Analog's High-Flying Direct Digital Synthesizer," which is about a high-frequency DDS chip manufactured by Analog Devices (*Circuit Cellar* 156, 2003).

I originally thought it might be possible to have the ATmega644 perform all of the required tasks: playing the WAV file, the binaural beat DDS, as well as the clock display and alarm functions. However, as mentioned earlier, my tests showed that there wouldn't be enough extra MCU time to reliably perform the dual DDS routines along with everything else. Therefore, I decided to go with the dedicated DDS generator circuit.

I choose the 14-pin ATtiny84 AVR MCU to accomplish the task. (The lesser ATtiny24 or ATtiny44 would have served as well but were not in stock when I was building the project.) It's capable of running at 20 MHz, which I provide using the ATmega644's 20-MHz clock. I used the inexpensive Microchip MCP4822 SPI dual-12-bit DAC here as well.

Although the ATtiny84 has neither a UART port nor a dedicated hardware SPI port, it does contain a Universal Serial Interface (USI). This is basically Atmel's fancy name for a hardware shift register with a bit of support circuitry, but it's capable of providing the SPI function at a data rate up to half the MCU clock, with only a minimum of added support firmware. This was important, since four bytes must be sent out to the DAC at the DDS sample rate, and this would be hard to accomplish if the SPI rate could not be set so high.

The sine wave look-up table is stored in the program flash memory and consists of 256 integer entries. The DDS routine is accomplished by using a 24-bit phase accumulator and selecting the MSB as the index into the 256-entry sine look-up table. At each pass through the DDS loop, the proper increment is added to the phase accumulator. The output frequency of this DDS implementation is expressed by the following equation:

$$F_O = \frac{(\text{sine table index increment}) \times (\text{DAC update rate})}{2^{24}}$$

The ATtiny84's program loop that performs the dual DDS function executes in 21 µs, resulting in a DAC update rate of 46.3 kHz. Solving the above equation results in a table index increment of 362 for every one Hertz of output frequency. Therefore, this DDS implementation has a resolution of 1/362 Hz, which is more than enough resolution to produce the low-Hertz frequency difference needed between the two tones. I'll mention that I used long integers for the two phase accumulators and did long integer math when adding in the phase increment values. This allowed me to program directly in BASIC rather than implement the 24-bit phase accumulator routines in assembly language. Since I was still able to achieve a much higher DAC update rate than was really necessary, this was justified.

The ATtiny84 does not contain a hardware UART/USART function, so it was necessary to implement a bit-banged software UART function to act as a communication link between the ATtiny84 and the host ATmega644. This is easy enough to do since the compiler has software UART functions built-in.

Photo 1—This is the relaxation generator's front panel.

However, there was no easy way for the ATtiny84 to run its tight DDS loop and concurrently monitor the serial link to the ATmega644. To address this issue, I used a spare port line on the host ATmega644 to drive the ATtiny84's *RESET line. When the host wants to update the DDS frequencies, it resets the ATtiny84. The firmware in the ATtiny84 is set up to collect two long integers via the serial link when released from reset. These values are then stored in their respective phase accumulator variables.

Because I was using such a high DAC update rate and getting roughly one hundred 12-bit data points per cycle, the amount of filtering required on the DAC output waveform was minimal. I needed only a single RC filter stage per channel.

ALARM CLOCK

Since I was going to be putting another box on the bedside table, I thought I might as well try to eliminate the clock radio that currently sits there. Adding a real-time clock function to the project didn't seem to be a major undertaking. However, the matter of providing a clearly visible clock readout was somewhat of an issue.

I had already settled on a 2 × 20 LCD for the user interface. This was a reasonable choice as there are five menus needed for the various functions. However, these character LCD panels have only a 6-mm-high font size. That's too small for me to make out in the middle of the night!

What I wanted for the clock display was characters four times the size of the LCD panel's native font. I settled upon the idea of using the "user-defined character" capability of the Hitachi HD44780 LCD controller chip (which drives virtually all small LCD character display panels). I wanted to use two adjacent character positions horizontally plus both the first and second rows combined per clock digit. Given this method, I needed to be able to define 11 different patterns to provide the equivalent of a large seven-segment readout—three more than the eight custom-defined characters allowed by the HD44780 controller.

After some head scratching, I decided to make use of the "L," "|," and "_" characters available within the standard font, and I used the custom characters to make up the remaining eight patterns needed. The end result of this is a numeric font set that is a bit "stylized," but still highly legible. This technique could come in handy for other purposes, so you might want to check out the `Definecustomlcdcharacters` and `Displayclockdigit` subroutines in my source code. You can see the clock display in action in Photo 1.

A major shortcoming of the common garden-variety clock radio is that they fail to sound the alarm if a power failure occurs during the night. Unfortunately, this has been happening to us more frequently over the last few years as our local power company cuts back on preventative maintenance.

I decided to handle this situation by using an I²C real-time-clock chip with a lithium coin cell as a battery backup. I chose the PCF8563 since it contains a built-in alarm function. This makes it much easier for the host firmware. After setting up the alarm time and enabling the alarm flag bits, the host firmware has only to poll the PCF8563's *INT line, which will drop when the alarm criteria is met (alternatively, one could connect the *INT line to an interrupt line on the host and let an ISR take care of things). I mounted C12, a 4- to 20-pF trimmer capacitor, in this circuit. If you have a high-accuracy frequency counter like I have, you can monitor TP1 and adjust the trimmer to achieve exactly 32,768 Hz, which will ensure better accuracy in the timekeeping.

The way I implement the alarm is that I use it to trigger the playing of the song file, "It's a Beautiful Morning," performed by one of my favorite bands from the '60s, the Young Rascals. Should that song finish without being noticed, I then generate a rather rude sawtooth waveform, which gets sent to the wave DAC. It's sure to wake one up!

FILE ACQUISITION

I figured that sound files containing soothing sounds could be found on the Internet, and I wasn't disappointed. A few minutes into my Google search I came across The Freesound Project (www.freesound.org). It has a very wide variety of recorded sound effects content, much of which is recorded with high-quality microphones and recording equipment and generally saved as high-quality WAV files. I used the following five Freesound project sound files for my project: Stream_027, Mountain_Stream_2, Lapping_Waves_and_Seagulls_2, Rain_in_Forest, and Medium_Rolling_Surf. You can perform a search by filename on the Freesound Project's website to access them.

Once you have the files, you have to process them a bit. First you have to make them loopable. This is done in two ways, depending upon the nature of the sound. For steady sounds, such as the Rain Forest sound, all you have to do is edit out the fade in and fade out, when present, so that when the sound file is finished playing and starts over again (loops), there is no break in the sound. For sound files, such as Lapping Waves, which consist of an intermittent sound (i.e., waves crashing with periods of relative silence in between), you have to edit the file to contain a fade in at the beginning, as well as one at the end, and choose starting and ending points in which the intensity of the waves is roughly equal.

Secondly, these WAV files, which are often recorded in stereo at high sample rates (e.g., 44.1 kHz), must be resaved as mono files at 22,050 samples per second. This still provides perfectly adequate sounds, and, at that data rate, the ATmega644 is able to handle the task.

As an amateur musician with a decent recording studio, the aforementioned tasks were easy to perform with my Sonar music recording software. However, most computers today have some form of audio editing software preloaded that will do the trick.

FADE OUT

I have been using my new gadget for a while now. Although such things are highly subjective, I believe it promotes a better sleep, and it has managed to distract our two dogs from hearing every little creak in the house, or every little strange sound outdoors. I like waking up to a good song instead of the beeping of our old clock radio, and the battery-backup feature is sure to come in handy this winter. While it's a small point, the power consumption of this project is only 3 W, which is a bit less than the clock radio it replaced.

I don't have room in this article to go into any detail about binaural beats, apart from the explanation I gave about how I generated them. However, a good starting point for checking them out is the Internet. ■

Brian Millier (brian.millier@dal.ca) was an instrumentation engineer in the Department of Chemistry at Dalhousie University (Halifax, Canada) for 29 years. He continues to run Computer Interface Consultants.

PROJECT FILES

To download the project files, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2010/244.

RESOURCES

B. Millier, "Analog's High-Flying Direct Digital Synthesizer," *Circuit Cellar* 156, 2003.

T. Napier, "Digital Frequency Synthesis," *Circuit Cellar* 99, 1998.

RIFF WAV File format, www.mmsp.ece.mcgill.ca/documents/AudioFormats/WAVE/WAVE.html.

The Freesound Project, www.freesound.org.

F. Voegel, "AVR-DOS for BASCOM-AVR," http://members.aon.at/voegel/.

SOURCES

ATmega644 and ATtiny84 Microcontrollers
Atmel Corp. | www.atmel.com

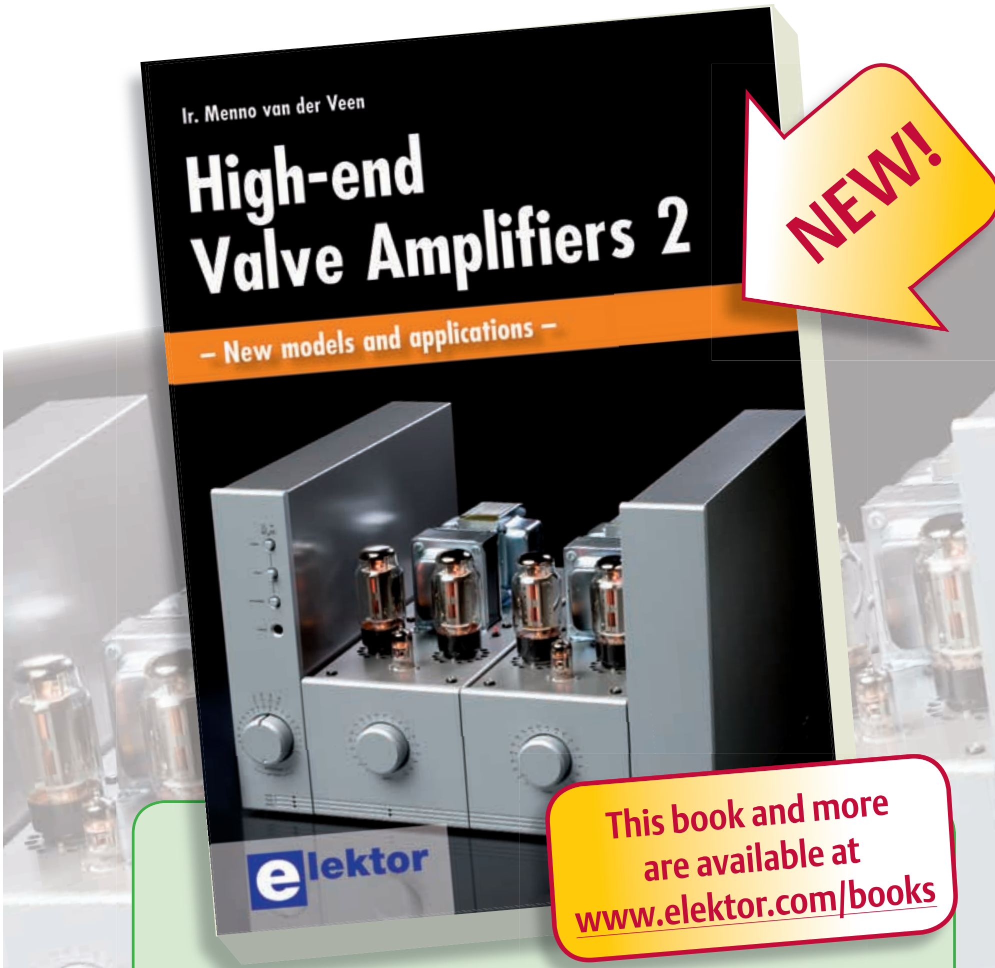
MCP4822 SPI DAC
Microchip Technology, Inc. | www.microchip.com

PCF8563 RTC and TDA1517 amplifier
NXP Semiconductors | www.nxp.com

BOB-08745 Level shifter module
Sparkfun Electronics | www.sparkfun.com

Elektor Shop

The world of electronics at your fingertips!



NEW!

This book and more are available at www.elektor.com/books

Books

New models and applications

High-End Valve Amplifiers 2

Nobody has any doubt that valve amplifiers produce a remarkably beautiful sound. They have a lively, deep, clear, and expressive sound, and dynamically they do not appear to have any limitations. Menno van der Veen investigates, in a systematic theoretical approach, the reasons for these beautiful properties. He develops new models for power valves and transformers, thus enabling the designer to determine the properties of the amplifier during the design process. You will notice in this book that the author not only writes about amplifier technique, but tells about the way the development of valve amplifiers can have an influence on your daily life; even the usefulness of patents is discussed. Summarizing: new theories and solutions for perfect audio with valve amplifiers. Not only the professional and the DIY-er but everyone who wants to understand valve amplifiers will read this book with much pleasure.

420 pages • ISBN 978-0-905705-90-3 • $59.70

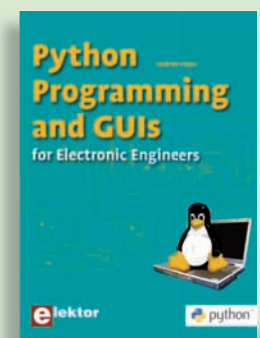
Bestseller!

Use only free or open source software

ARM Microcontroller Interfacing

Learn to interface and program hardware devices in a wide range of useful applications, using ARM7 microcontrollers and the C programming language. Examples covered in full detail include a simple LED to a multi-megabyte SD card running the FAT file system. A chapter on more advanced ARM microcontrollers is included with an overview of some of the newest ARM microcontrollers and their features.

250 pages • ISBN 978-0-905705-91-0 • $47.60



Get started quickly and proceed rapidly

Python Programming and GUIs

This book is aimed at people who want to interface PCs with hardware projects using graphic user interfaces. The programming language used is Python, an object-oriented scripting language; a higher level language than, say, C. The book guides you through starting with Linux by way of a free downloadable, live bootable distribution that can be ported around different computers without requiring hard drive installation. Practical demonstration circuits and downloadable, full software examples are presented that can be the basis for further projects.

224 pages • ISBN 978-0-905705-87-3 • $47.60

Prices and item descriptions subject to change. E. & O.E

CD/DVD-ROMs

See the light on Solid State Lighting

DVD LED Toolbox

This DVD-ROM contains carefully-sorted comprehensive technical documentation about and around LEDs. For standard models, and for a selection of LED modules, this Toolbox gathers together data sheets from all the manufacturers, application notes, design guides, white papers and so on. It offers several hundred drivers for powering and controlling LEDs in different configurations, along with ready-to-use modules (power supply units, DMX controllers, dimmers, etc.). In addition to optical systems, light detectors, hardware, etc., this DVD also addresses the main shortcoming of power LEDs: heating. This DVD contains more than 100 Elektor articles on the subject of LEDs.

ISBN 978-90-5381-245-7 • $46.00

110 issues, more than 2,100 articles

DVD Elektor 1990 through 1999

This DVD-ROM contains the full range of 1990-1999 volumes (all 110 issues) of Elektor Electronics magazine (PDF). The more than 2,100 separate articles have been classified chronologically by their dates of publication (month/year), but are also listed alphabetically by topic. A comprehensive index enables you to search the entire DVD.

ISBN 978-0-905705-76-7 • $100.00

Reign with the Sceptre

This open-source & open-hardware project aims to be more than just a little board with a big microcontroller and a few useful peripherals — it seeks to be a 32-bit ARM7 fast prototyping system. To justify this title, in addition to a very useful little board, we also need user-friendly development tools and libraries that allow fast implementation of the board's peripherals. Ambitious? Maybe, but nothing should deter you from becoming Master of Embedded Systems Universe with the help of the Elektor Sceptre.

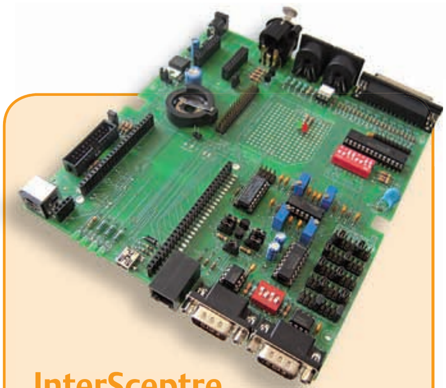
PCB, populated and tested, test software loaded

Art.# 090559-91 • $143.60

Elektor is more than just your favorite electronics magazine. It's your one-stop shop for Elektor Books, CDs, DVDs, Kits & Modules and much more!

www.elektor.com/shop

Elektor US
PO Box 180
Vernon, CT 06066
USA
Phone: 860-875-2199
Fax: 860-871-0411
E-mail: sales@elektor.com

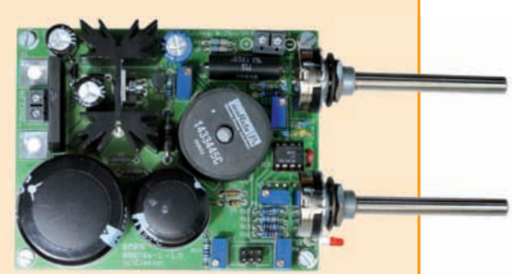


InterSceptre

Recently, Elektor introduced Sceptre, a fast prototyping system fitted with a 32-bit microcontroller. Even on its own, this board will let you produce some great results, but if we add an extension board to make it easier to access all its peripherals, the Sceptre platform becomes downright powerful. What's more, if you fit this extension board into a suitable case, you'll be able right from the start to develop a prototype that you can use 'properly' in a installation, with no trailing wires or bits of sticky tape holding everything together. Now that's what you call fast, convenient prototyping!

Kit of parts, contains PCB and components

Art.# 100174-71 • $ 187.10



UniLab

A power supply with adjustable output voltage and current limiting is part of the basic equipment of every electronics lab. However, the increased complexity of a switch-mode design scares away many potential builders, even though it actually isn't all that complicated if you use a suitable combination of well-known technologies. This circuit is suitable for building a single or dual power supply.

PCB and all components, less power transformer

Art.# 090786-71 • $103.30

Kits & Modules

by George Martin (USA)

Big Changes in the Embedded World (Part 1)

Solving Connectivity Problems

As you well know, most end users require the ability to connect many of their electronic devices to the Internet and other electronics. Simply put: when you are designing a new product, connectivity matters.

My workload slowed down in 2008 as the general economic downturn started. Fortunately, work started to pick back up again at the start of 2009. I suspect my recovery was sooner than most. Much of my business during this recovery has to do with the replacing of obsolete devices. It's not the main focus of my design work, but some of the projects have proved to be extremely interesting. During this slowdown, it's clearly time to look at the services I provide and try to anticipate what will be needed once the economy fully recovers. It's time to move out of my comfort zone and get new technologies under my belt. *Adapt or die!*

I recently thought long and hard about what my customers have been searching for over the past few years. More computing power and memory capacity at better prices were frequently requested. But, as I searched deeper, I found the topic of connectivity (e.g., Ethernet, USB, and wireless) has been one of the most a frequent requests. I think the industry will take care of the increase computing power and memory capacity along with reduced size and power. Moore's Law is still intact and will yield more performance.

It used to be difficult to add Ethernet, USB, and wireless technology to designs. Each interface typically required separate controller integrated circuits and perhaps additional physical modules to convert from the CPU logic levels to the interface levels. These expensive chips were single-sourced and ate up board space and power. This was true even in FPGA designs. I could put an Ethernet controller as intellectual property (IP) internal to the FPGA, but it cost to use that IP and the number of gates the IP consumed pushed the design into larger FPGA devices. That was a very expensive solution for my customer base (lower-volume industrial controls).

The best solution would be to get all this connectivity for free. How free? We'll see.

FIND A DEV SYSTEM

I rely heavily on electronics distributors such as Arrow Electronics for technical support and delivering great pricing and support to my customers. My orders are for one to 10 units and that doesn't get any supplier very excited. My customer's orders are for 100 to 1,000 and more, which is still on the low end, but that gets the distributor's attention. These distributors, along with manufacturers, give technology presentations several times a year. I try to attend the ones about the new computer chips and families. I generally don't attend the switching power supply design seminars.

I attended two interesting presentations so far this year. At the first, Freescale Semiconductor (along with Arrow and Avnet) was talking about the ColdFire CPU. You might recall that I wrote about this processor in my July 2007 article titled "From 'Hello World' to Big Iron" (*Circuit Cellar* 204). In that article, I used one of the NetBurner boards (they are a frequent *Circuit Cellar* advertiser) and pointed out that you had flash memory, USB, Ethernet, RTOS, and much more all built into the unit. This year's Freescale presentation was about its new development system the Tower System. It is composed of horizontal boards with CPUs and other devices all connected with two vertical ladder modules. Any module can go into any location. So, if you're debugging the serial port, put that module on top. If you are debugging the CPU, then put that module on top. Also, as new modules are introduced, they should just plug right into the Tower System.

About 2 hours into the presenter's talk about the modules, the system, the CPU, and the RTOS, he paused and asked if everyone was fluent in C. I'm not sure if he ever looked at the room for a show of hands. I laughed to myself, "C is not dead." Now, if you attended this seminar you got certificated for a free (as in free beer) Tower development system with all the code development tools including the RTOS. Track down the Freescale representative and find a seminar in your area. If there isn't one, get some friends and suggest they have one or at least get you the video of one.

The second presentation I attended was about a month later. Texas Instruments (with Arrow) was presenting Stellaris CPUs. You know all about this line because you likely took part in the Texas Instruments DesignStellaris 2010 Contest (www.circuitcellar.com/designstellaris2010/). If you missed it, well, wait until you see what you missed. Once again, the presenter talked for over an hour and then asked if everyone was fluent in C. He said he could read C but not write it. So, I told him about my C articles published in *Circuit Cellar*.

The Stellaris CPUs also offer flash memory, RAM, Ethernet, USB, and RTOS all built in. There's much more in the block diagram, but the connectivity issues I described are being addressed.

TI took a different approach for the Stellaris development systems. It offers one board for each specific application. TI has one if you want Ethernet, USB, graphics LCD, and RTOS. If you just want Ethernet-to-serial and RTOS, it has one. Each attendee received certificates for development boards and tools at half price.

Listing 1—Simple I/O initialization

```
//*******************************************************************
//
//   Rocker Switch Port J Interface
//
//   Bit 7 6 5 4   3 2 1 0
//       | | | |   | | | |
//       | | | |   | | | |---> SPI Data Out
//       | | | |   | | |-----> SPI Clock
//       | | | |   | |-------> SPI Latch
//       | | | |   |---------> nc
//       | | | |------------< Keyboard
//       | | |--------------< Rocker Switch Down
//       | |----------------< Rocker Switch Up
//       |------------------< Lane Lock
//
//*******************************************************************

#define  RK_OUT       GPIO_PIN_0
#define  RK_CLK       GPIO_PIN_1
#define  RK_LATCH     GPIO_PIN_2

#define  KB_IN        GPIO_PIN_4
#define  RK_DOWN      GPIO_PIN_5
#define  RK_UP        GPIO_PIN_6
#define  RK_LL        GPIO_PIN_7

//**********************************************[ Public ]*********
//
// Rocker Switches Port Interface Sets up the hardware
//
//*******************************************************************
void PortInitRockerSwitches(void) {

// enable the GPIO PortJ
SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOJ);

// Set these pins as outputs
GPIOPinTypeGPIOOutput(GPIO_PORTJ_BASE,
                          GPIO_PIN_3 |
                          RK_LATCH   |
                          RK_CLK     |
                          RK_OUT);

// Set these pins as inputs
GPIOPinTypeGPIOInput(GPIO_PORTJ_BASE,
                          KB_IN |
                          RK_LL |
                          RK_UP |
                          RK_DOWN);

// Set Drive strength and
GPIOPadConfigSet(GPIO_PORTJ_BASE,
                          GPIO_PIN_3 |
                          RK_LATCH   |
                          RK_CLK     |
                          RK_OUT,
                          GPIO_STRENGTH_2MA,
                          GPIO_PIN_TYPE_STD);

}  // end of void PortInitRockerSwitches(void) {
```

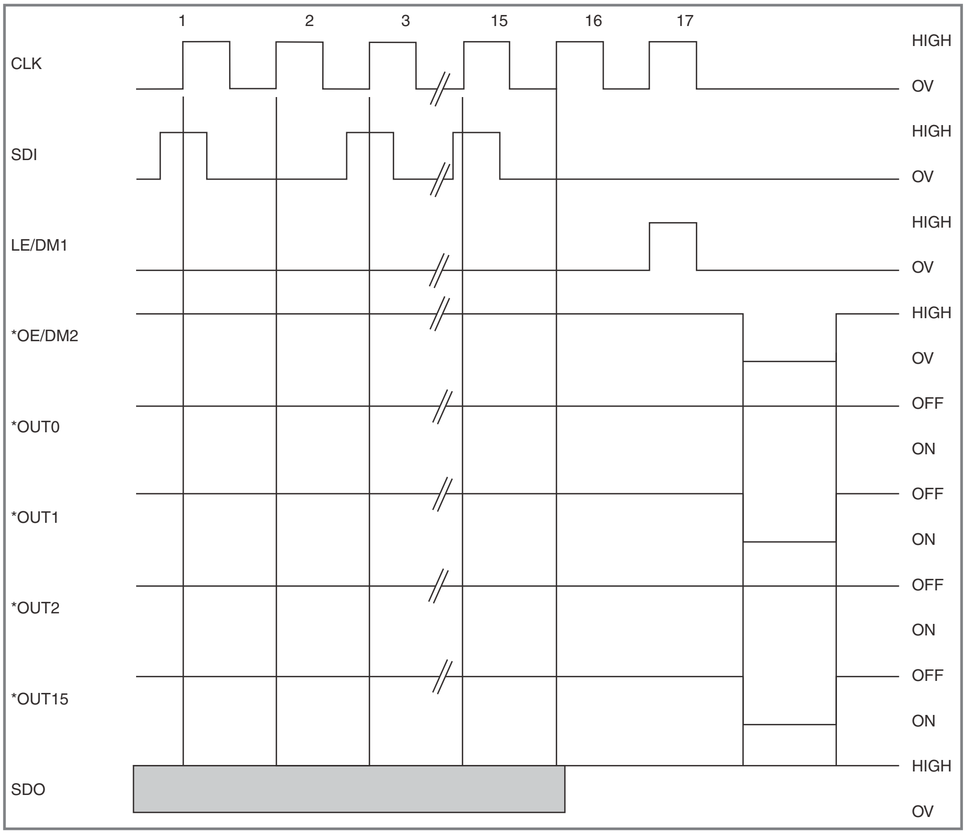


Figure 1—The timing diagram to control the STP16CP05 16-bit constant current LED sink driver

I encourage you to look for these seminars and others from your favorite distributors. They provide great value for your time. Also, Circuit Cellar design contests offer the latest devices and the tools to use them. As I write this, I know there are other suppliers that are also providing a "system on a chip" solution.

START DESIGNING

I now have several monolithic solutions that solve the connectivity problem my customers have described. It's simply a matter of getting one of the low-end evaluation boards and starting to design.

For the purpose of this article, I'm going to start with the board that was used in the TI DesignStellaris Contest. The Stellaris EKK-LM3S9B96 evaluation kit has an LM3S9B96 CPU on a small board with debugging, Ethernet, and USB connectors. All the unused I/O pins are brought out to the eight headers. Power from the USB host powers the board. So, connect one USB cable and two ribbon cables between the debug board to the evaluation board and you are up and running.

With these new highly integrated microcontrollers comes a lot of system software. Each of the blocks in the CPU's block diagram has become complicated. It's so complicated that if you read the hardware manual and tried to configure the block, it would take a while to learn how to do it. What's provided is what I call system software.

Take a look at Listing 1. Here I'm setting up the I/O ports on the CPU to interface to a keyboard. This is an odd keyboard. It's 24 switches, but not arranged in a 4 × 6 matrix. It's arranged in a 24 × 1 matrix. That arrangement is how it's been done for years and I can't change it. This new design will need to plug into existing installations. Also, there are several keyboard matrices on this port and their names are Keyboard, Rocker Switch Up, Rocker Switch Down, and Lane Lock. In order to see if a switch is closed, we drive the 24 "rows" with a pattern of 23 highs and one low. And this pattern (the one low) walks or rotates through all the 24 possible positions. We then read the common line, and if it's pulled low, we know that a switch is closed. I implemented the 24 outputs with an LED driver IC. If you look up the datasheet you'll see that the chip has a serial input that looks like an SPI bus and a parallel output for 16 LEDs. Furthermore, these chips can be daisy-chained to get the 24 bits we need.

I'm using the Port J on the CPU to control these LED driver ICs and read the results from each keyboard. My plan is to bit-bang bits 0, 1, and 2 to run the SPI-like interface and use bits 4, 5, 6, and 7 to read the switch input. Note that this CPU has two dedicated SPIs. I'm not using them here because I have plans to use them elsewhere and the interface to the LED driver ICs is a bit nonstandard. Look at the control signals and you'll see that they are not exactly a standard SPI interface (see Figure 1).

Refer again to Listing 1. You can see I created some `#defines` used to map the names `GPIO_BIT_x` into names like `RockerSwitchUp`. Then, as we enter the routine, you start to see the system routines that I spoke of earlier. The first system call `SysCtlPeripheralEnable(SYSCTL_PERIPH_GPIOJ);` enables Port J. Next we define which bits should be outputs, we then define inputs, and finally we define the drive strength of each bit. Now we could have read the hardware manual and figured what register to write to and which bit to set in each register for

Listing 2—Some of the bit manipulation support routines

```
void RockerClkHi(void) {
GPIOPinWrite(GPIO_PORTJ_BASE, RK_CLK, RK_CLK);
}       // end of void RockerClkHi(void)

void RockerClkLo(void) {
GPIOPinWrite(GPIO_PORTJ_BASE, RK_CLK, 0);
 }      // end of void RockerClkLo(void)
```

Listing 3—I needed to insert delays into the code. I defined a routine to do that.

```
void RockerDelay(void) {
INT16 i;
for (i = 0; i < 100000; i++) {
)
} // end of void RockerDelay(void) {
```

Port J Bit 1 to be an output. But now that you know C, it's easier to read the systems software manual to get the job done. Port J is now configured. Ta-da!

CODE & BIT MANIPULATION

Next let's take a look at the code to manipulate Port J bits 0, 1, and 2 to control the SPI to the driver LED and then read the results on bits 4, 5, 6, and 7. I started by making routines for each output bit.

Listing 2 includes the routines for setting the SPI clock bit high and low. I extracted all this so that I just call `RockerClkHi()` or `RockerClkLo()`. It may seem like a lot of work and additional overhead, but when we get to where we use these routines that code will read much better. Note that we could have used defines something like this:

```
#define RockerClkHi() GPIOPinWrite(GPIO_PORTJ_
  BASE, RK_CLK, RK_CLK)
```

This is should be just a simple text substitution. I have not tested this define construct. Each compiler might perform in a manner other that you expected. I see many of the engineers that I work with using this technique. And now it's time for you to start thinking about your own designing style.

I wrote routines for each output bit and one for a delay. If you look at the data chip for the STMicroelectronics STP16CP05MTR sink driver, you'll see some timing requirements such as maximum clock frequency of 30 MHz. In order to meet all these requirements, we need to insert delays into the code. I defined a routine to do just that (see Listing 3).

I just put in a busy loop for the first implementation of this routine. It's something to keep the CPU occupied just wasting clock cycles. In the example, I incremented the variable i 10,000 times. That provided about 1 ms delay, which was where I wanted to be. This is a very poor way to implement a delay loop, but I had code running on Day 1. When you start using the compiler's optimization levels, this code could get compiled out for run speed. And that's probably not what you'd expect. You've seen me implement this code with timer interrupts and state machines. But for now, let's just use this delay.

Where in the world did I discover routines such as `GPIOPinWrite()`? It's all in the peripheral driver library user's guide. With the Stellaris family of CPUs, TI has the StellarisWare software suite. Go to TI's website to search for and download the latest copy. Realize that it's about 400 pages of information. In this library, you'll find descriptions for routines that work with all peripheral devices in the chip. But wait, there's more. Some of the Stellaris CPUs come with the peripheral library in ROM. When in ROM, these routines do not use any of the flash memory. You can start debugging with the routines in flash memory. Single step through the code and use break points since you have the source. Then, link with the ROM version as your code gets tested.

TURN LOOSE THE RTOS

The code for the entire interface to the keyboards is available on the *Circuit Cellar* FTP site (ReadTheSwitches.c). Look to see how I implemented the timing in Figure 1. There is, of course, more to getting the board up and running. We need to set up the clock and at least blink an LED. For now, get the kit out if you've got one, download the system files, or contact a distributor and get an evaluation board. Look at Freescale's ColdFire, TI's Stellaris, or any other that is a complete monolithic solution.

There's a whole lot more that comes with any of these new highly integrated CPUs. In the second part of this series, I'll describe how to get a program up and running and then turn loose the RTOS.

George Martin (gmm50@att.net) began his career in the aerospace industry in 1969. After five years at a real job, he set out on his own and co-founded a design and manufacturing firm (www.embedded-designer.com). His designs typically include servo-motion control, graphical input and output, data acquisition, and remote control systems. George is a charter member of the Ciarcia Design Works Team. He is currently working on a mobile communications system that announces highway info. He is also a nationally ranked revolver shooter.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2010/244.

RESOURCES

Freescale Semiconductor, "The Tower System," www.freescale.com/webapp/sps/site/overview.jsp?code=TOWER_HOME.

STMicroelectronics, "STP16CP05: Low-Voltage 16-Bit Constant Current LED Sink Driver," 12568, Rev. 10, 2010, www.st.com/stonline/books/pdf/docs/12568.pdf.

SOURCES

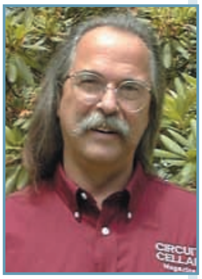
ColdFire Microcontroller
Freescale Semiconductor, Inc. | www.freescale.com

STP16CP05 Constant current LED sink driver
STMicroelectronics | www.st.com

Stellaris EKK-LM3S9B96 Evaluation Kit
Texas Instruments, Inc. | www.ti.com

FROM THE BENCH

by Jeff Bachiochi (USA)



Recharging Portable Devices

A DIY Power Adapter Design

When traveling with multiple portable devices, having the ability to recharge quickly and easily is key. But why buy AC adaptors when you can design your own all-in-one solution? Here's how.

Our constant use of portable devices means frequent recharging. Most products come with an AC charger. For some devices, car charger adapters are available as well. Ten years ago, I needed to stay in touch while spending a week at Boy Scouts of America (BSA) camp ("Sharing Technology with Mother Nature: Out of State with an Internet-Compatible Cell Phone," *Circuit Cellar* 125, 2000). My Motorola StarTAC flip phone had a serial connection, and I brought a laptop with modem (remember those?) so I could get and send e-mail through the phone. I took a used motorcycle battery with a 12-V car socket attached so I could plug in both my phone and laptop and keep them charged.

This July, I went to the BSA's 100th anniversary National Jamboree, which was held, for the last time, at Fort AP Hill in Virginia. This time, I had a laptop, a Droid, a still camera, and a video camera. Sure, I had AC adapters for each piece of equipment, but I would've also needed four separate DC adapters. Instead, I brought a "one size fits all" adapter that I built at my workbench.

PROGRAMMABLE REGULATOR

I doubt there are any circuit builders who are unfamiliar with the three-terminal regulator (see Figure 1a). Using one is a no-brainer. You apply an unregulated voltage between the input and ground terminals, and you get any (lower) regulated voltage between the output and ground terminals. Even if you've used a fixed voltage regulator, you

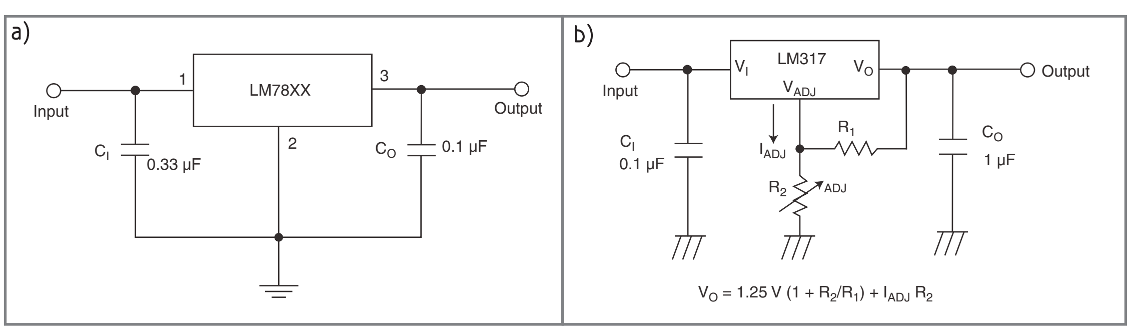


Figure 1a—Here's the standard 78*xx* fixed output linear regulator that's used just about everywhere. It's easy to use and comes in many varieties, including different package styles for low- and high-current applications, and plenty of fixed output voltages to choose from. **b**—The LM1/2/317 adjustable linear regulator is similar to the standard fixed voltage regulator, but it can be set to your choice of regulated voltages from 1.2 up to 37 V (depending on your supply voltage) based on two external resistors.

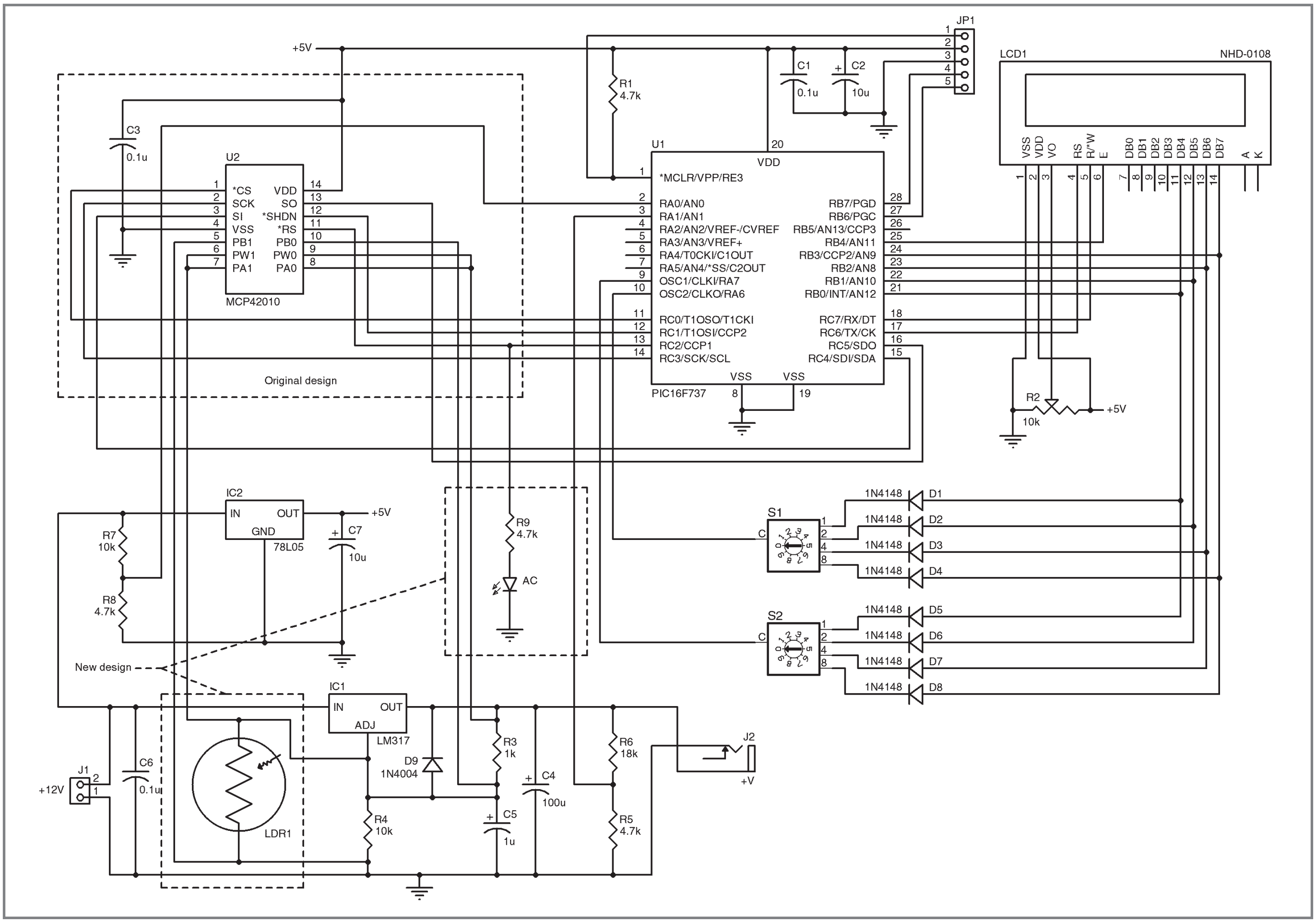


Figure 2—This is a schematic of the project. The original and new design's alterations are outlined in dotted areas. The new design's LED and LDR require the devices to be coupled and placed in a light-proof package.

may not be familiar with its counterpart, the adjustable three-terminal regulator. You can think of the adjustable regulator as a 1.2-V fixed output regulator when the adjustment terminal is connected directly to ground. While the adjustable regulator requires two additional resistors to set the output voltage to something higher than 1.2 V, that's a pretty cheap price to pay for its flexibility.

To increase the output voltage, you add a resistor voltage divider network between the output terminal and ground with the divider junction connected to the adjustment terminal of the adjustable regulator, like the LM317 in Figure 1b. The upper resistor between the output and adjustment terminals creates a current due to the 1.2 V across it. This current also flows through the lower resistor (along with a small bias current provided by the adjustment terminal) creating a voltage across the lower resistor raising the adjustment terminal above ground and the output 1.2 V above that. So, this voltage across the lower resistor, plus the 1.2V across the upper resistor, equals the new regulated voltage.

The upper resistor's value is chosen to give the LM317 a minimum load of about 5 mA. You can see by the formula for calculating resistor values for a required VOUT that with a 5-mA load through the resistor divider, the 50 µA bias current will have very little effect:

$$V_{OUT} = 1.2\text{ V} + \left(1.2\text{ V} \times \frac{R_2}{R_1}\right) + 50\ \mu\text{A} \times R_2$$

A simple calculation will provide a good start for determining R2, the upper resistor: 1.2 V/5 mA = 240 Ω. If you are designing for a 5-V VOUT, the bottom resistor will need to drop 3.8 V (i.e., 5 V – 1.2 V). The resistor will need to be 760 Ω (i.e., 3.8 V/5 mA).

PROGRAMMABLE RESISTOR

At the heart of the project is a programmable resistor. My initial thought was to use a digital potentiometer. But further investigation revealed that the pot was not as isolated from the control circuitry as you might think. The potential on any connections must remain within the supply voltage (usually around 5 V). The circuit I developed has voltages that exceed twice that. So, I couldn't use such devices.

I now can say that I've read the fine print. However, this seemed like such an ideal match for this design, that I saw just what I wanted to see and jumped ahead designing with it. It's a fairly straightforward design. A microcontroller is used to read a couple of miniature rotary BCD switches. These set the desired voltage: one switch for the integer volt value and one switch for the tenths of a volt.

An internal ADC makes it possible to read both battery voltage and regulated voltage. What better way to dis-

ICbank
Gateway to Electronics

3 PORT INTERFACE
RS-485 to Ethernet Converter

Only
$170

RS-485 to Ethernet Converter

Powerful feature

- Protocol converter RS485 between Ethernet
- Offer TCP/IP Communication to Devices with RS485 I/F

Specification

Network	: TCP, UDP, DHCP, ICMP, IPv4, ARP, IGMP, PPPoE, Ethernet, Auto MDI/MDIX , 10/100 Base-TX Auto negotiation (Full/half Duplex)
Serial	: RS485 3 Ports, 1,200~115,200 bps, Terminal block I/F Type
Control program	: IP Address & port setting, serial condition configuration, Data transmit Monitoring
Accessory	: Power adapter 9V 1500mA, LAN cable
Etc	: - DIP Switch(485 Baud Rate setting) - LED: Power, Network, 485 Port transmission signal

ICbank
Gateway to Electronics

www.icbank.com

ICbank Gateway to Electronics

MP3P DIY KIT, Do it yourself

(Include Firmware Full source Code, Schematic)

• myPIC

Only **$160** qty 100 **$220** qty 1

Powerful feature

- MP3 Encoding, Real time decoding (320Kbps)
- Free charge MPLAB C-Compiler student-edition apply
- Spectrum Analyzer
- Application: Focusing for evaluation based on PIC
- Offer full source code, schematic

Specification
Microchip dsPIC33FJ256GP710 / 16-bit, 40MIPs DSC
VLSI Solution VS1033 MP3 CODEC
NXP UDA1330 Stereo Audio DAC
Texas Instrument TPA6110A2 Headphone Amp(150mW)
320x240 TFT LCD
Touch screen
SD/SDHC/MMC Card
External extension port (UART, SPI, I2C, I2S)

• myWave (MP3 DIY KIT SD card Interface)

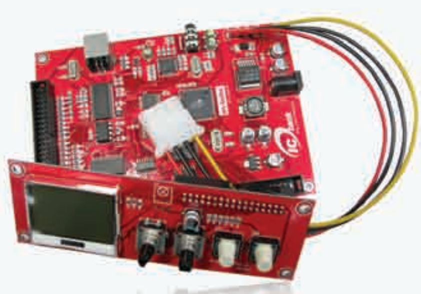
Only **$150** qty 100 **$200** qty 1

Powerful feature

- Play, MP3 Information, Reward, forward, Vol+/-
- Focusing for MP3 Player
- SD Card interface
- Power: battery
- offer full source code, schematic

Item	Specification
MCU	Atmel ATmega128L
MP3 Decoder	VS1002 / VS1003(WMA)
IDE Interface	Standard IDE type HDD(2.5", 3.5")
Power	12V, 1.5A
LCD	128 x 64 Graphic LCD
Etc	Firmware download/update with AVR ISP connector

• myAudio (MP3 DIY KIT IDE)



Only **$180** qty 100 **$220** qty 1

Powerful feature

- Play, MP3 Information, Reward, forward, Vol+/-
- Focusing for full MP3 Player (Without case)
- IDE Interface
- Power: Adapter
- Offer full source code, schematic

ICbank Gateway to Electronics

Youngsan Cho / Senior Account Manager / Sales & Marketing ICbank Inc.

Tel : 82-707-019-3924. Fax : 82-2-6340-1888. Cell : 82-17-268-2078. Email : youngsan@icbank.com

18F Ace High -End 2, 222-14 Guro Digital Valley, Guro-dong, Seoul, 152-050, Korea

Website : http://www.icbank.com/

play these values than by a small LCD? I have a 1 × 8 display that will work well. Figure 2 shows my original design using a digital potentiometer. Can this circuit be saved by a device that does not exceed its maximum ratings?

CADMIUM SULFIDE

My first introduction to cadmium sulfide (CdS) cells came around 40 years ago when I was working for Electronic Music Labs. A new synthesizer being developed was going to use a toggle switch for selecting one of two audio signals. The design's switches were creating audible "pops" when switching between signals. A quiet switch was developed using a combination CdS cell and incandescent bulb prepackaged together. I forget what it was called at the time. An SPDT switch was wired to alternate power to the bulbs of two of these devices. The relatively long time constants of the incandescent bulb and their effect on the resistance change of each CdS cell produces a nice quiet transition between the signals.

The CdS cell has very high resistance (greater than 1 MΩ) when the sensor is in darkness and this is considerably reduced as light is allowed to fall on the sensing surface. As you might guess, when the device has a lot of light and the resistance is small, current flow may become an issue. The physical size of the device has a relationship to its power rating. The largest I purchased have a maximum rating of approximately 0.5 W at about 10 mm in diameter. With 10 V across a resistance of 200 Ω would produce a current of 10 V/200 Ω, or 50 mA. The power would be 10 V × 50 mA, or 0.5 W.

Similar to solar cells, a CdS cell uses photon energy to affect electron movement. However, while a solar cell creates a current via photons giving electrons the energy needed to cross its PN junction, the CdS cell has no junction and the N-type material becomes more conductive due to the increased electron energy thanks to the photon. The CdS cell is essentially a light-dependant resistor (LDR). By placing an LDR across the lower leg of the regulator's resistor divider network, you get a way to change the resistance of this leg and thus the regulator's output voltage.

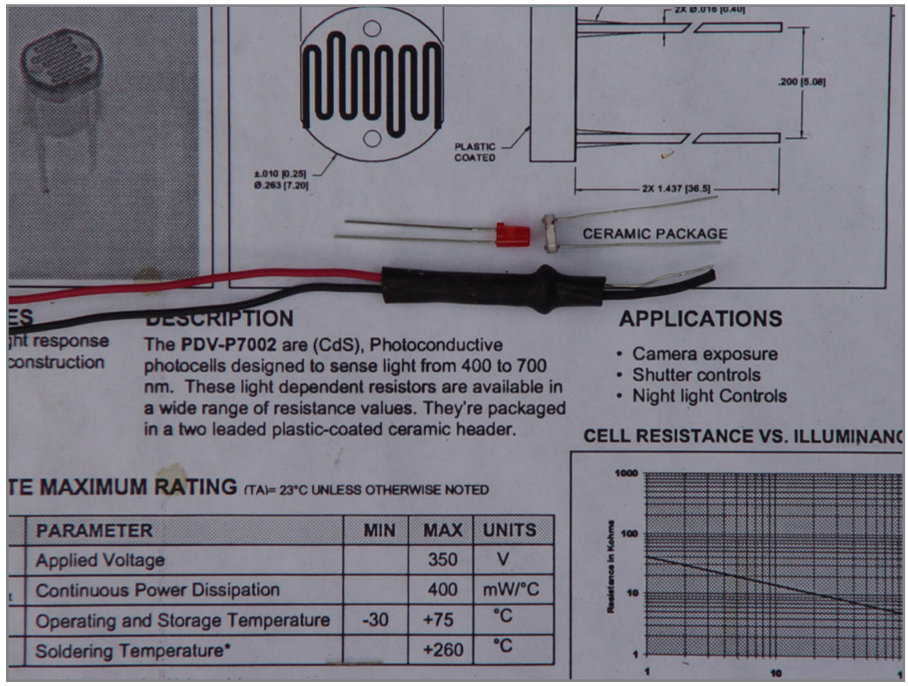


Photo 1—The cadmium sulphide cell, or light-dependant Resistor (LDR), was a saving grace when coupled with a red LED.

To use the LDR in this circuit, you need to control its light source. The micro has a PWM peripheral, so I powered an LED from the PWM's output pin to achieve complete control over its intensity. A series resistor will limit the maximum intensity (at 100% PWM) of the LED, which will keep the LDR from entering a region that exceeds its maximum power dissipation (smallest resistance).

DIGITAL REGULATOR

In the future, you might find voltage regulators that contain their own microcontrollers; but for now, you'll have to design your own. My original design (using the digital pot) was to be table-driven—that is, pick your voltage using the BCD switches and look up (or calculate) the appropriate resistance to set the bottom resistor of the regulator's divider network. This new design won't calculate anything; it will be self-adjusting. If you compare the actual output voltage with the user setting (BCD switches), you can call for more light (higher PWM) if the actual voltage is too high and less light (lower PWM) if the output voltage is too low.

The key to this design is making an LED/LDR module. I began using an LED light source to see how each of

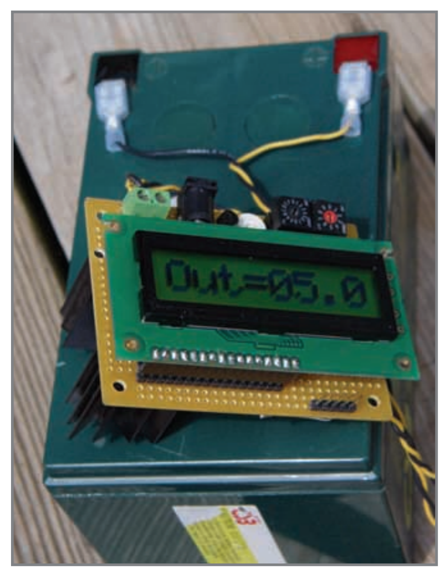


Photo 2—I've dialed in "5" and "0" on the BCD switches, and the microcontroller is measuring and displaying the output voltage. It must compare this voltage with the BCD setting and adjust its PWM output to whatever duty cycle is necessary to drive the connected LED with an intensity level that will keep the LDR at the required resistance, and to produce a voltage that matches the BCD setting.

the LDRs I had purchased reacted to this source of illumination. The spectral range of the LDR is amazingly close to the human eye (approximately 400 to 700 nm). The red LED I had handy will reduce all of the LDRs values down to less than 1 kΩ with approximately 10 mA through the LED. I chose the best alternative as a compromise between lower resistance and the higher wattage (Advanced Photonix PDV-P7002).

It is important to package the LDR and LED in a light-proof container to prevent any stray light from creating an uncontrolled variable in the photons-to-resistance conversion. A short piece of shrink tubing created a stable environment for these two devices holding each in proper alignment (see Photo 1). Now let's see how this all fits together.

IT'S ALL ROUTINE(S)

This design is nothing but routines. Read the user input BCD switch settings. Display status on the LCD. Take analog-to-digital samples (Timer 1 Interrupt). Determine if the PWM requires any adjustment (A/D conversion complete Interrupt). The first two routines happen as time allows, while the final two are handled on an as required basis. The main loop handles reading the user input from the switches and outputting status to the LCD as the lowest of priorities.

Notice that the parallel LCD uses a 4-bit data bus. Also note that this bus is shared with the two BCD switches. The commons for each BCD switch are tied to their own output pin on the microcontroller and the BCD binary switch pins are connected through diodes to the data bus. In order to read to and write from the LCD, the data bus must not be affected by any other device connected to it. Wire-ORing each switch with diodes prevents any data line which is being driven low from affecting another data line presently being shorted together through a switch, as in D0 and D1 when a switch is in position 3 (b'0011'). Connecting each switch's common to an output pin allows the switch to be effectively removed from the circuit whenever a logic high is placed on the common. This way the three devices share the same data bus. Care should be taken to only enable one device at a time or else active interaction will result in bogus data as well as potential device damage.

TAKING YOUR INPUT

I will be treating the two user input switches as the number of tenths of a volt between 0 and 100 tenths (100 tenths = 10 V). Selection is a byte variable whose value is 10 times the upper digit switch reading plus the lower digit switch reading. While storing the value as two BCD digits might make more sense, I want this value to be in the same format as the A/D values. (You'll see what I mean shortly.) In the configuration used, a switch's position is read by writing a low to its common pin, and reading this effect on its binary pins through the data bus on the lower nibble of PORTB. The value read from PORTB is complimented and then ANDed with 0x0F to mask out the upper nibble. While the upper limit (10 V or 100) is just within reach, you know that the lowest output of the regulator is 1.2 V. So, unless you have an LDR that can reach 0 Ω with unlimited power dissipation, the lower values (say, 0 to 30) will be illegal. (I'll let the LCD tell that story.)

Before I explain the display routine for the LCD, I'd like to take a short detour here and interject the A/D calculations as they directly relate to the aforementioned user input switch settings. Having the single byte value SELECTION (from the BCD switch) and OUTVOLTS (from the A/D conversion and calculation) make it easy to determine what must be done. They should be identical for this application.

The status (supply voltage and regulated voltage) can actually be greater than the microcontroller's ADC can handle, so these voltages will be divided (in this case) by three before measuring them. This means that a full-scale conversion will actually mean 15 V and not the ADC's reference of 5 V. While the ADC is capable of 10-bit conversion, I use only the 8 MSBs to keep it simple. With a 5-V reference, each bit will represent 5 V/256, or 0.01953 V. With this application's divider, that would be three times 0.01953 V, or 0.0586 V. I want to eliminate fractional math and only need to show tenths of a volt using a byte variable. Since my design calls for a full-scale input of a maximum of 15 V and I want this to read as a value of 150 (tenths of a volt), I scale everything up by a factor of 256. Therefore, if I multiply the conversion value by 150 and divide it by 256, I get (for full scale) 255 × 150/256 = 38,250/256 = 149 tenths, or 14.9 V. This application requires a simple 8-bit × 8-bit integer multiply for the first part. Choosing a factor of 256 makes all this work out nicely because I need to use only the upper byte of the 16-bit multiply result to divide it by 256!

DISPLAYING RESULTS

There are only two things of importance to this project: the supply voltage and the regulated output. A small three-digit LED display would be adequate. I like using LCDs when possible because I can display more than just numbers. The small NHD-0108, an eight-character single-line LCD has just what I need. If you've been following my column, you know that scrolling long messages on an LCD really removes what at first seems like a limitation of smaller displays.

Besides showing the column and issue number on a power-on screen, I have only two messages to display: "Bat=xx.x" and "Out=xx.x" (see Photo 2). Well, that's not really true, as there are a couple of other messages I added to indicate that your choice on the BCD input switches are illegal. These messages occur in place of the output voltage message if the output cannot be regulated to the required request.

I chose a display time of 1 s for each of the two messages. The variables `Next` (time for next display change) and `Seconds` (running seconds counter) are compared to determine when to increment the `State` variable (which determines the message to display) and recalculate a new `Next`. While this might seem like a bit of overkill for this application, it lends itself well to easily adding additional states for more messaging capabilities.

Each display message consists of a four-character ROM string to LCD routine (i.e., "`Bat=`") to wipe out the previous display's first four characters and leave the cursor positioned for the individual character display of the appropriate value in the format: digit, digit, decimal point, digit. I used an integer divide routine to help turn the byte variables `BatVolts` and `OutVolts` into hundreds, tens, and units digits, symbolizing tens, units, and tenths by the placement of a decimal point in the display format. Both the integer multiply and divide routines used here require less than two dozen instructions each.

PROGRAM INTERRUPTED

Two related interrupts are used in this application. The first is a timer, which provides two functions. It's initialized to interrupt an overflow every 10 ms. The `Count` variable is used to increment the `Seconds` variable every 100 interrupts. The timer also acts to control the initiation of A/D conversions. That's where all the real work goes down. Because the LDR (remember that device) is relatively slow with respect to a microcontroller's execution speed, it can actually be counterproductive to change the controlling illumination faster than it can respond to the control signal. This timer interrupt can be used to slow down the update rate by delaying A/D conversions.

I tested the LDR/LED module by pulsing the LED with a 1-Hz output. With the LDR connected between VCC and a 1-kΩ resistor to ground, a scope showed that the LDR's resistance falls in roughly 20 ms with illumination, but it takes 100 ms for the resistance to rise back up from the lack of the light. With the ADC's present configuration, it can complete a conversion on a new channel every 120 µs. By using the 10-ms timer to begin a pair of conversions they (and the adjustment routine that follows the conversions) won't over run the response time of the LDR.

By varying the PWM's pulse width value Percent, you can effectively control the illumination of the LED from full off to full on. Percent's value is used to set the PWM's duty cycle and is determined by comparing Selection (user's switch setting) to OUTVOLTS (the measured output). If Selection = OUTVOLTS, then no change to Percent is necessary. If Selection < OUTVOLTS, then Percent is decremented, which reduces the light intensity, thereby raising the resistance of the LDR, the voltage across it, and the overall regulated output. If Selection > OUTVOLTS, then Percent is incremented, which increases the light intensity, thereby lowering the LDR's resistance, the voltage across it, and the overall regulated output.

Earlier I mentioned displaying additional messages. The measured output voltage is displayed unless this voltage doesn't match what the you have "dialed in" on the BCD switches. If there is an attempt to decrement Percent below its minimum value, the PWM will already be full off and the output will have reached its maximum voltage. The alternate message "Choose <" is displayed. If there is an attempt to increment Percent above its maximum value, the PWM will already be full on and the output will have reached its minimum voltage. The alternate message "Choose >" is displayed.

SWITCHING REGULATORS

While I've demonstrated this technique on a linear regulator, there are switching regulators that can be substituted. These use the same style of voltage divider network to set the output voltage. They require additional external components and create their own noise source as they rely on switching the supply voltage on and off. However, they do have advantages. They can boost the output voltage above the supply and have a high efficiency at all voltages, unlike linear regulators that make heat (input/output voltage difference times the current supplied). If you are interested, here are a few switchers you might want to check out: Maxim Integrated Products's MAX5096, Linear Technology's LT3430, and National Semiconductor's LM2576 and LM2585.

PORTABLE POWER

I can now keep all of my electronic devices charged while I'm out in the field. If by chance I begin to tax the 12-V gel cell I use to run this project, I'll need to remember to connect my portable solar cell battery charger during the daylight hours to keep the battery ready to do its job. I can also plug this project into the cigarette lighter—ah, excuse me, auxiliary power outlet—in any vehicle so I can charge on the go. Let's see now: phone, camera, laptop, camcorder, gel cell, project, and solar charger. That's it. Is there any room left for a tent? ■

Jeff Bachiochi (pronounced BAH-key-AH-key) has been writing for Circuit Cellar *since 1988. His background includes product design and manufacturing. You can reach him at jeff.bachiochi@imaginethatnow.com or at www.imaginethatnow.com.*

SOURCES

PIC16F737 Microcontroller
Microchip Technology, Inc. | www.microchip.com

LM117/217/317 Regulator
National Semiconductor Corp. | www.national.com

by Tom Cantrell (USA)

Heat Harvester

A Look at Thermal Energy Production

Green is the new black, and the Silicon Wizards are doing their part with chips that do more for less. But maybe there's another answer. Instead of just using less energy, why not use all the energy, and there's a lot of it, that otherwise goes to waste?

At the reservoir in Los Angeles in 1913, William Mulholland, before a crowd of thousands, turned to one of his engineers at the last pump and gave the indication to turn the great wheels, open the floodgates, and down the water came, dancing and sparkling in the sun, and started spilling into the reservoir, and Mulholland turned to the dignitaries and said: "There it is, gentlemen, take it."—T.H. Watkins[1]

Energy is a lot like water. Life is tough without it and it seems there's invariably a "shortage." Considering we're in a seemingly perpetual energy crisis, it's interesting to contemplate all the energy that goes to waste. And a lot of it is heat.

Think of every "heating" process, whether it is a power plant boiler, factory oven, home fireplace, kitchen stove, or for that matter a campfire. Efficiency ranges from bad to a joke sending much, if not most, of the energy out the chimney.

As you're tooling your car down the road, a lot of the expensive energy elixir in your gas tank is lost as heat created by internal combustion and friction. Admittedly, some of it is captured and put to good use when you turn on the heater. As well, you may be surprised to learn there's actually an "Engine Bay Cuisine" sub-culture (well at least a few Wiki entries and webpages) for those cooks that really want to get under the hood. But unless you're a Julia Child wannabe on a roadtrip, I dare-say most of the heat is wasted.

But it doesn't have to be. In a 2004 presentation titled "The Effects of an Exhaust Thermoelectric Generator of a GM Sierra Pickup Truck," Aleksander Kushch, Madhav Karri, Brian Helenbrook, and Clayton J. Richter described a prototype thermoelectric generator design for a standard pickup truck (see Figure 1). Extracting "free" heat energy from the exhaust pipe and coolant, the generator easily delivers upwards of 100 W at freeway speeds. The extra energy translates directly to improved mileage by reducing the alternator load. As fuel prices rise, so does the payoff from the fuel savings, potentially reducing payback time to a matter of months. Even if electric cars render internal combustion moot, don't forget all the surplus heat energy generated by friction. Now we're talking about every motor, shaft, bearing, and gear, virtually anything with moving parts.

There's plenty of heat for the taking, but it's the taking that's the trick. Read on to see why thermal energy harvesting is a truly hot topic.

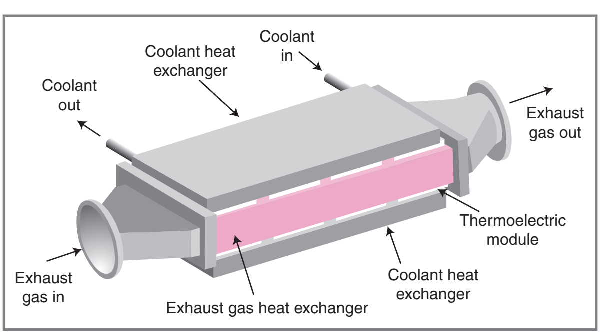


Figure 1—One way to take advantage of all that heat energy going to waste out your tailpipe is this proposed design for a pickup truck thermoelectric generator.[2]

MY GENERATION

MicroPelt is an outfit that's figured out how to use IC-like processes to create arrays of tiny thermocouples on a chip. (For more information about thermogenerators, refer to H. Bottner et al's paper listed in the Resources section of this article.) Although no doubt challenging in practice, in principle it's as simple as fabricating an array of "hot" side legs on one die and an array of "cold" side legs on another die and then squishing them together (see Photo 1).

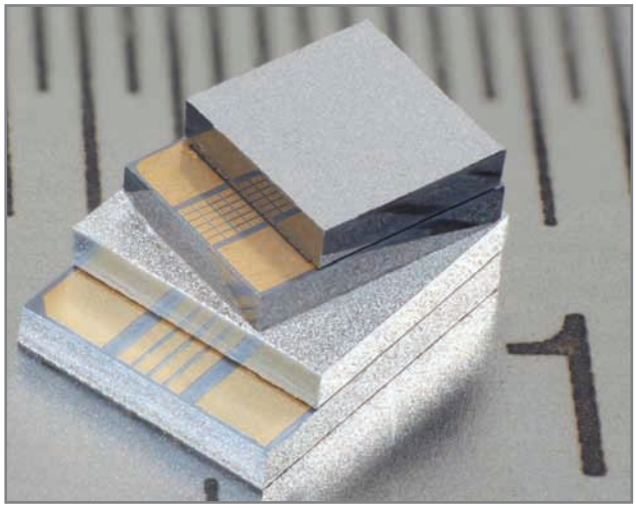


Photo 1—Taking advantage of IC-like fabrication techniques, Micropelt crams hundreds of thermocouples into tiny devices a few millimeters on a side.

As you recall, a thermocouple is formed at the joining of two different conductors. Indeed, the company's name recalls the discovery by Jean-Charles Peltier in 1834 that current flow through a thermocouple causes heat transfer from one leg to the other, the basis for a range of thermoelectric coolers Micropelt offers.

But just as a motor can be a generator, a thermocouple goes both ways. In 1821 Thomas Johann Seebeck discovered that heating a thermocouple generates an electrical potential, which is basis for the thermocouple's familiar role as a temperature sensor. Now, thanks to the miniaturization made possible by their chip-scale process, MicroPelt can cram hundreds of thermocouples on a single device. Noting the voltage generated by a single thermocouple doesn't depend on its size, ganging a bunch of them together delivers a class of Seebeck-on-steroids "Thermogenerators" (aka "TEGs") with the potential to enable a new class of applications (see Figure 2).

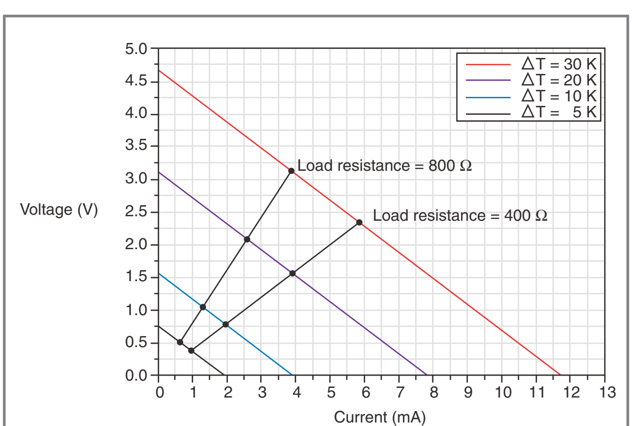


Figure 2—With 540 thermocouples, the Micropelt MPG-D751 TEG can generate enough electrical power from relatively small temperature differentials to do useful things.

Thermal energy harvesting is a supply-side innovation, but only part of the story. There's a demand-side story too, one about ever lower power chips that get better mileage than ever. The convergence of the two trends (i.e., more energy generated, less consumed) makes the opportunity all the more compelling.

HEAT PIPE

Like an electrical circuit, a thermal circuit has an overall resistance (to heat flow) comprising the resistance of each element and the connections between them. For a thermal harvester, the goal is to maximize the difference in temperature between the hot and cold sides of the TEG, which is done in two ways.

The first is to minimize the resistance between the environmental heat source and sink and the corresponding hot and cold sides of the TEG. For example, if the cold side of the TEG is ambient air, typical "heatsink" techniques (e.g., thermal grease, convection) would apply.

At the same time, you want to minimize thermal connections created between the hot and cold sides, ideally leaving the TEG itself as the only thermal path. For instance, metal screws that contact both hot and cold side structures will lower efficiency by letting heat "leak" from one side to the other. Referring again to Kushch et al's presentation, experiments showed that insulating the exhaust pipe between the pickup truck engine and thermogenerator minimized heat leakage between the source (combustion chamber) and sink (ambient air), thereby significantly boosting power output.

Although we refer to a TEG as having hot and cold sides, the "thermal polarity" is reversible in fact. While the difference in temperature between sides determines the magnitude of the voltage generated, which side is which determines the polarity (i.e., direction of current flow). In the most obvious applications (i.e., harvesting waste from a heating process), the polarity is fixed (i.e., in normal operation the temperature on the "hot" side of the TEG will always be higher than the temperature on the "cold" side). But it doesn't have to be so, a fact which represents both an opportunity and a caution.

No doubt there are some applications where it makes sense to harvest in both directions. I'm thinking of a situation analogous to a cave in the desert where the temperature remains semi-constant, cooler than ambient during the day and warmer than ambient at night. Thermal energy could be harvested at nearly all times except for the brief transition periods (one during the day, one at night) when the temperature outside and inside the cave is the same.

But polarity reversal also means applications designed expecting a "normally" hot and cold side must anticipate the possibility of an abnormal event. For example, if a heating process is unexpectedly interrupted, is it possible the "hot" side of the TEG could become colder than the "cold" side? To connected circuits that would

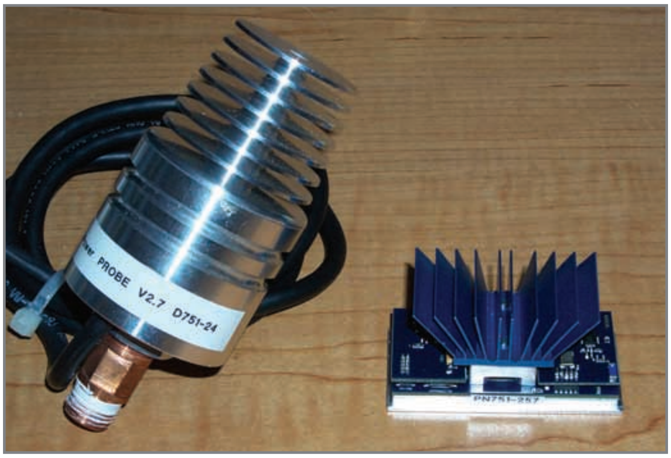


Photo 2—With their fancy metalwork, the "TE-Power PROBE" and "TE-Power NODE" demonstrate that thermal energy harvesting success starts in the machine shop.

be the equivalent of plugging in a battery backwards. Make sure your design has reverse-polarity protection just in case.

FLUXMEISTER

At this early stage, Micropelt works with potential TEG customers individually to help craft application-specific solutions. But if you just want to kick the tires, they do offer some intriguing TEG-based gadgets (see Photo 2) for evaluation. As you can see, a lot of credit for a successful thermal harvester design goes to the machine shop. But the fancy fabrication doesn't come cheap, with prices from $400 to $900 depending on options. Not exactly an impulse buy, but the units are versatile enough to cover a full-range of experiments, prototyping and real applications. Let's take a closer look at the "TE-Power NODE" and you'll see what I mean.

All energy-harvesting designs require certain basic functions. The generator (i.e., the TEG) converts the environmental phenomenon (heat) into electrical power. However, the power output will fluctuate and in any case likely doesn't match that required by the application, so power-conditioning (e.g., rectification, regulation, boosting, inverting, etc.) is required. Finally, any energy harvested in excess of instantaneous demand should be stored for later use.

The problem is applications vary a lot in terms of their specifics, such as the voltage level or amount of storage required. As shown in Photo 3, the kit utilizes a modular approach that accommodates a variety of power conditioning, storage, and application scenarios.

The TEG is instrumented with digital (I^2C) and analog hot- and cold-side temperature sensors (i.e., four temperature sensors total). On the bottom is a baseplate with both tapped holes and built-in magnets as options for attaching to a flat surface. On the top is a finned heatsink. Sandwiched between is a baseboard that brings out the electrical connections (i.e., TEG, temperature sensors) to headers. Although the documentation refers to the bottom plate as the hot side and heatsink as the cold side, the baseboard also includes DIP switches so you can manually change polarity (i.e., baseplate cold, heatsink hot).

If you want to take it from there and implement your own power conditioning and storage, the "Direct Power Module" (DPM) routes the generator electrical connections (i.e., TEG "+" and "–," and the analog temperature sensors) to a header for easy wiring. As an alternative to rolling your own, the kit includes two Micropelt-designed solutions. Job one for the "Power Conditioning Module" (PCM) and "DC Booster Module" (DBM) is to boost and regulate the TEG output voltage to a chip-friendly level. The PCM features a fixed 2.4-V output while the DBM has a trimpot to set the output anywhere between 1.6 and 5 V. A built-in 100-µF capacitor provides a measure of energy storage and there is provision to wire-in an external capacitor for even more. When used with a PCM or DBM, the "Application Interface Module" (AIM) is kind of a break-out box that allows you to jumper any of the signals (i.e., TEG "+" and "–," conditioned/stored power, and both the digital and analog temperature sensors) to headers for easy wiring.

Combined with a PCM, the "Wireless Sensor Module" (WSM) demonstrates a complete example application. It's based on the popular Texas Instruments "EZ430-RF" reference design comprising an MSP430 MCU and CC2250 IEEE 802.15.4 radio chip running under the purview of their "SimpliciTI" protocol software. The WSM talks to your PC via a USB plug-in radio (also courtesy of TI) that's included in the kit.

The WSM comes from the factory loaded with a demo that sends the TEG hot and cold side temperature readings (from the digital temperature sensors) and the power supply (i.e., storage capacitor) voltage to the PC once per second. You can use the kit to develop your own applications since the USB radio also works as an MSP430 programmer and there's provision to connect additional I^2C devices to the WSM. All the source code for the WSM demo (MSP430 "C") and TE-Power SCOPE utility (Delphi) is included on the CD.

LAMP TEST

Getting the demo running is easy. After installing the SCOPE software, just plug in the USB radio, set its virtual COM port address, and the PC side is ready to go.

Now all you have to do is find some spare heat for the

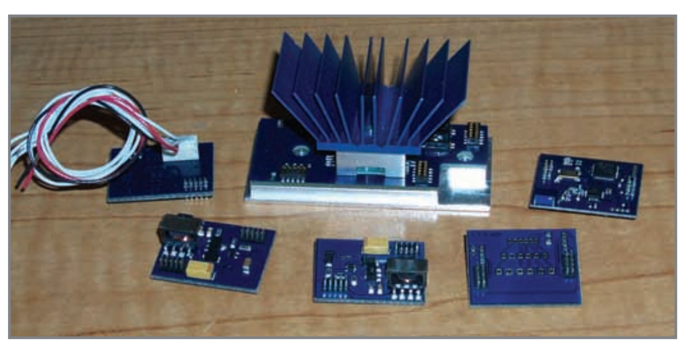


Photo 3—The "TE-Power NODE" is a modular setup that combines the TEG (sandwiched between a baseplate and heatsink) with add-ons for power conditioning/storage, application interface and wireless sensing.

Photo 4—Despite an extremely poor fit (i.e., thermal interface) between the NODE baseplate and curved hood of the desk lamp, this setup easily generated enough power to run the demo.

NODE to harvest. I used an incandescent desk lamp with a metal hood. Once the lamp got hot I just hung the NODE on the hood relying on the built-in magnets (see Photo 4). I anticipated efficiency would be poor as the curved hood on the lamp only made partial contact with the hot-side base, so I was surprised that in less than a minute the NODE came to life and started sending data to the PC. LEDs on both the WSM module and USB radio blink during RF activity (i.e., once per second), a reassuring heartbeat visible at a glance.

Even though there are just three pieces of data in play (hot and cold side temperature, supply voltage), the SCOPE software does a lot with them (see Photo 5). Based on the temperature reports, the software estimates heat transfer through the TEG assuming an average specification for the TEG thermal resistance. In turn, the TEG electrical output can be estimated given an assumption that the load impedance is reasonably matched. The hot- and cold-side temperatures, the difference between them, and the supply voltage are charted in real-time and can be logged to a file for subsequent analysis. The SCOPE software can monitor up to five NODEs, each of which can be given a descriptive name of your choice.

Understandably, the WSM radio transmitter is set to low-gain (+1 dBm) in the interest of minimizing power consumption. And the small module doesn't have room for much in the way of an antenna. No surprise that radio

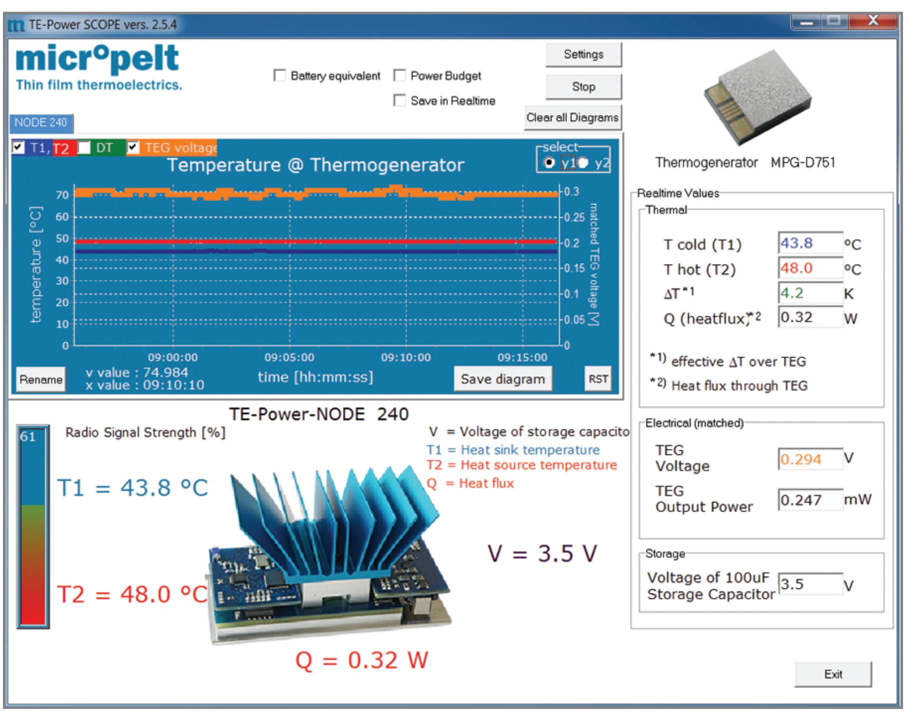


Photo 5—The "TE-Power SCOPE" software tracks the NODE once-per-second radio updates of hot and cold side temperatures and supply voltage and estimates the TEG electrical power output.

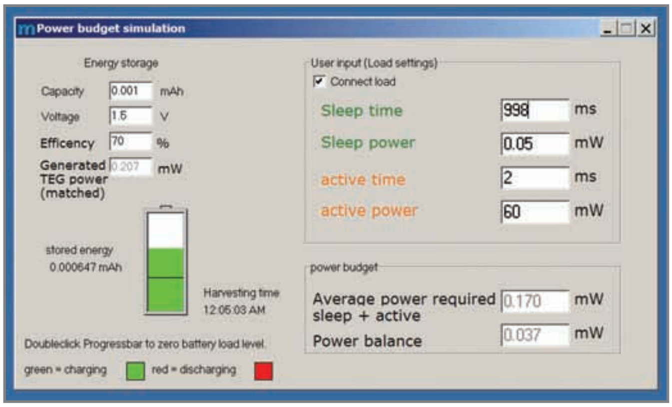


Photo 6—A "Power-Budget" feature of the SCOPE software allows you to play "what if," using the real-time TEG power readings to supply a virtual load and storage device.

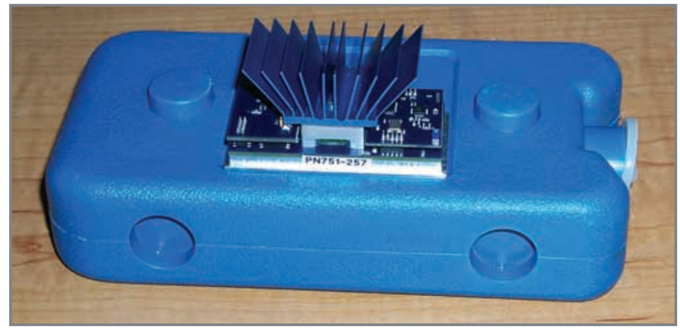


Photo 7—Shaken or stirred, an ice battery is literally a 'cool' application for thermal energy harvesting.

range is limited (the documentation says 5 m) and rather subject to interference. The SCOPE software displays the received signal strength, which is helpful for identifying dead spots and assessing range in a particular setting.

Although the power consumption of the WSM isn't measured directly, you can infer it by watching the supply voltage (i.e., voltage at the 100-µF storage capacitor). When sufficient power is being harvested, the supply voltage sits at about 3.5 V, but it starts to fall when there's a deficit. By moving the NODE to different locations on the lamp, I could adjust the temperature difference to find the transition between surplus and deficit. It turns out that keeping the WSM alive required a mere 0.1 mW or so, an amount the TEG easily generates from a few degrees of temperature difference. The secret to such low power consumption is the fact the WSM is only active for 2 ms during each once-per-second radio update for a duty cycle of just 0.2%. Another key is the efficiency of the PCM power conversion, generally better than 70% according to the documentation.

The SCOPE software also has a "Power Budget" calculator that compares the power being generated by the TEG with demand from a hypothetical system comprising a simulated load and storage (see Photo 6). Like any simulation, true usefulness depends on the accuracy of estimates and assumptions, but it's handy for quickly ball-parking different load, storage, and power conversion scenarios.

ICY HOT

Just for kicks, I thought I'd take advantage of the switchable polarity feature to see how an "Ice Battery" might work (see Photo 7). Along the way, I gained additional insight into thermal considerations designers should keep in mind.

Take a look at Photo 8, keeping in mind hot and cold are inverted. (Micropelt might want to add a "reverse polarity" checkbox to the SCOPE software.) On the left side of the graph, you see what can be a misleading "start-up" phenomenon. When the NODE is first put on ice, the TEG power output jumps quickly to a relatively high level. But then the output begins to fall off, eventually to the point the NODE gives up the ghost.

What the graph shows is the impulse response of the thermal circuit. When initially placed on ice, the cold side (baseplate) temperature transitions quickly downward, while the hot side (heatsink) remains at ambient. That creates a large temperature delta across the TEG and high power output. But over time, heat makes its way from the hot to the cold side across the thermal connections between them, notably including the TEG, reducing the temperature difference and power output.

The thermal resistance of the TEG is a two-edged sword. Keeping it low maximizes the heat transfer and thus power output. But lower TEG resistance also increases the tendency to equalize temperature on each side (zero resistance would have zero temperature difference), which reduces power output. As you can see in the screen shot, over time the hot and cold side temperatures each move towards the other until the entire system achieves thermal stability. Each design will have its own time constants that determine the impulse response, time to achieve thermal equilibrium, and the corresponding sustainable output.

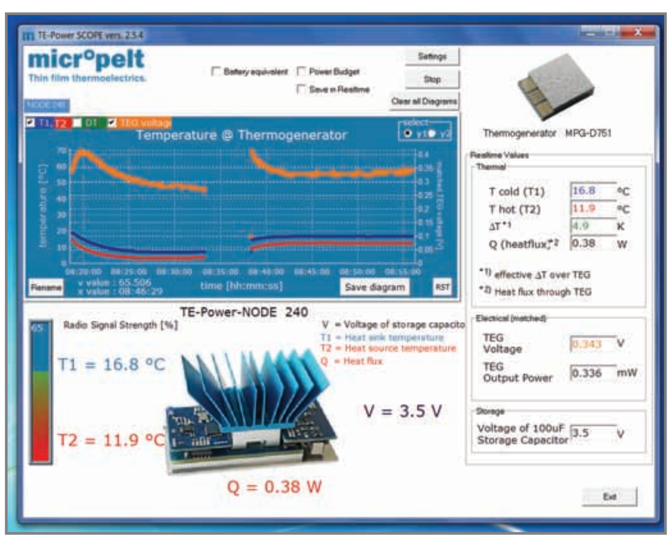


Photo 8—The SCOPE software makes it easy to see the effects of thermal phenomenon such as impulse response (typically at start-up) and convection.

The documentation makes clear that convection is your friend, all the more true when the ultimate heat source/sink has high thermal resistance (as air does). In the second phase of my ice battery experiment (the right side of the graph), I set up a small fan blowing on the heatsink. You can see how,

once again, power rises quickly as the heatsink is immediately warmed by the fan and then tails off as that heat makes its way across the TEG (and other thermal connections) to the cold side. But this time, conveying a continuous source of heat (i.e., a fan blowing ambient air) to the hot side, a temperature differential is maintained that generates enough power to keep the WSM alive.

HOT FLASH

The Micropelt technology is admittedly bleeding-edge and only beginning to make its way into real applications. It's entering the realm of feasibility, but it's still at the point where designers will be challenged to cleverly live within power and price means. Hey, it wouldn't be fun if it was easy, right?

But don't forget that today's bleeding-edge becomes tomorrow's leading-edge and the day after that mainstream. Imagine the possibilities as technology marches on to deliver ever more powerful and efficient generators and smarter, lower-power silicon.

How about health monitors or "smart clothes" powered by the heat of your skin? Or an e-coffee cup that does something amusing when you fill it with steaming brew? Heck, maybe even the ice battery idea isn't as crazy as it sounds.

As we've seen this month, the Wizards are doing their part by creating more (and more efficient) energy-harvesting options and incredibly low-power chips. Now it's time for designers to think outside the box and create applications that take advantage of all the free energy at hand. There it is. Take it. ◼

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.

REFERENCES

[1] PBS, "The West," 2001, www.pbs.org/weta/thewest/program/episodes/eight/takeit.htm.

[2] A. Kushch, M. Karri, B. Helenbrook, and C. Richter, "The Effects of an Exhaust Thermoelectric Generator of a GM Sierra Pickup Truck," 2004 DEER Conference, Coronado, CA, www1.eere.energy.gov/vehiclesandfuels/pdfs/deer_2004/session4/2004_deer_kushch.pdf.

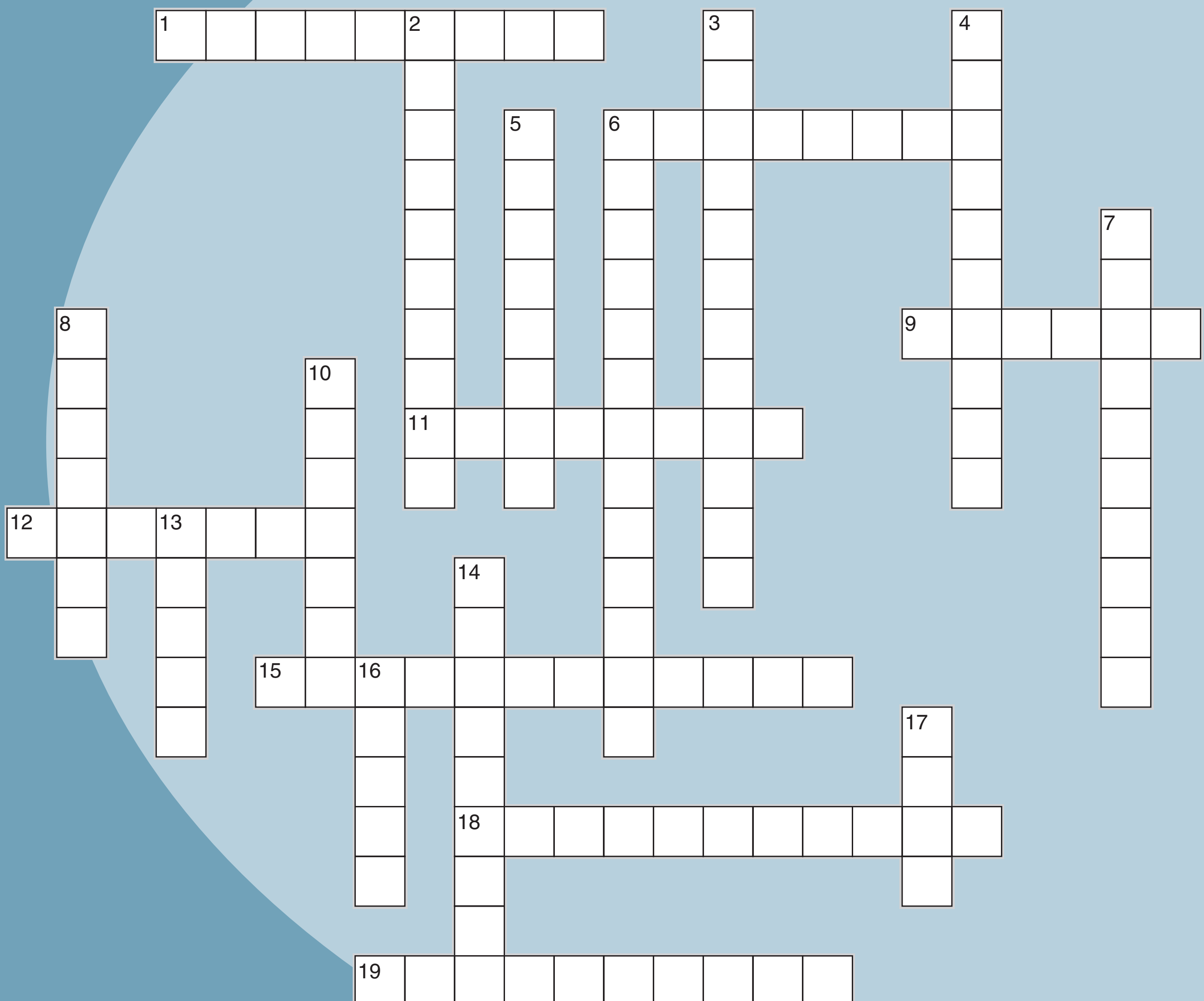
RESOURCE

H. Bottner, J. Nurnus, A. Schubert, and F. Volkert, "New High Density Micro Structured Thermogenerators for Standalone Sensor Systems," www.micropelt.com/down/ict_07_ha_boe.pdf.

SOURCE

Thermoelectric Generator (TEG) and thermal energy-harvesting kits
Micropelt | www.micropelt.com

CROSSWORD



Down

2. Rad : Angle :: H : _____
3. V_Z [two words]
4. The I in AVI
5. Frag
6. “#”; comment designation [two words]
7. P2P [Three words]
8. Eddy
10. Transistor : Active :: Resistor : _____
13. For diverting a current
14. BAT [two words]
16. Modulator + demodulator
17. NRZ is a non-return to?

Across

1. AUX
6. A “sweet” trap for hackers
9. OS core
11. A relay is an electromechanical switch that closes and opens these
12. Measure temperature, pressure, acceleration
15. 1/1,000,000,000,000,000 s
18. Fujio Masuoka; Toshiba [two words]
19. Linux, Apache, Thunderbird [Two words]

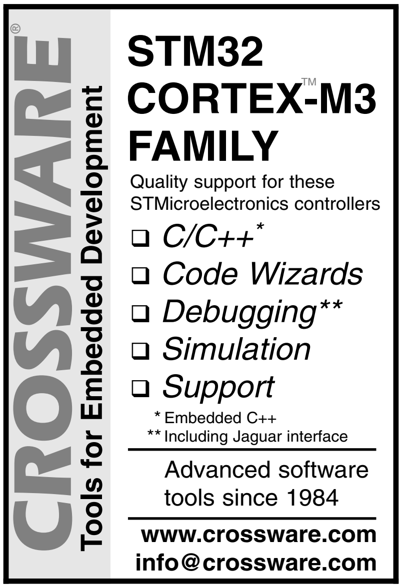
The answers will be available in the next issue and at www.circuitcellar.com/crossword.

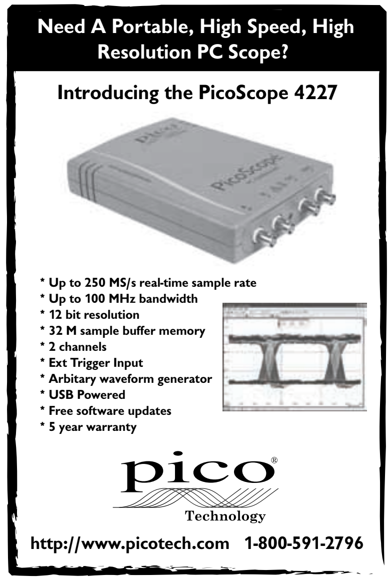
IDEA BOX

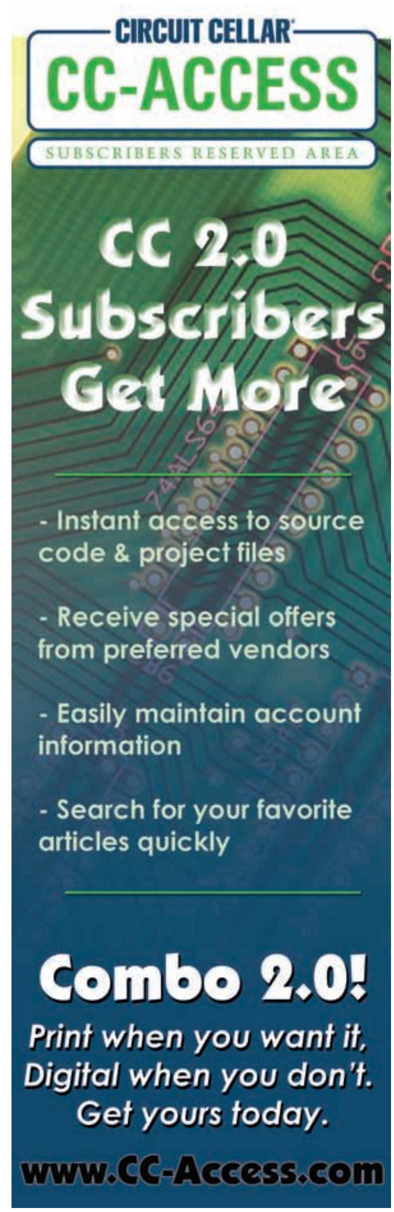
THE DIRECTORY OF PRODUCTS AND SERVICES

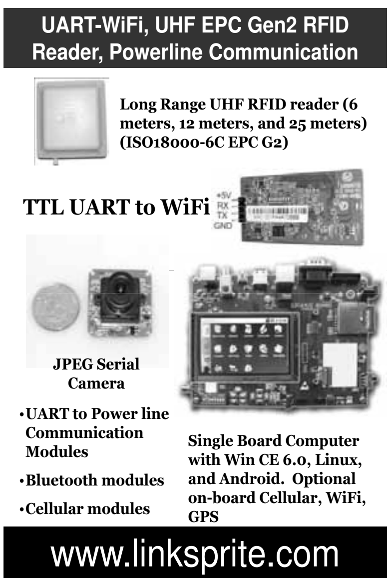
AD FORMAT: Advertisers must furnish digital files that meet our specifications (www.circuitcellar.com/advertise). ALL TEXT AND OTHER ELEMENTS MUST FIT WITHIN A 2" x 3" FORMAT. E-mail adcopy@circuitcellar.com with your file or send it to IDEA BOX, Circuit Cellar, PO Box 180, Vernon, CT 06066. For current rates, deadlines, and more information contact Peter Wostrel at 800.454.3741, 978.281.7708 or peter@smmarketing.us.

The Vendor Directory at www.circuitcellar.com/vendor/
is your guide to a variety of engineering products and services.

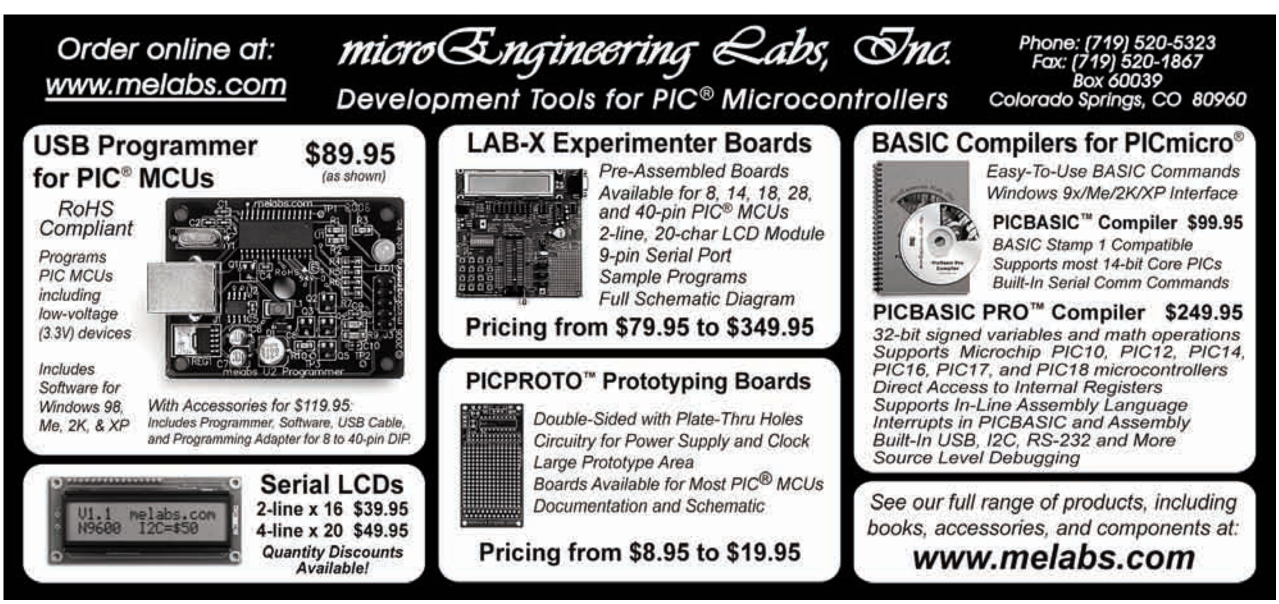

CROSSWARE®
Tools for Embedded Development
STM32
CORTEX™-M3
FAMILY
Quality support for these STMicroelectronics controllers
❑ C/C++*
❑ Code Wizards
❑ Debugging**
❑ Simulation
❑ Support
* Embedded C++
** Including Jaguar interface
Advanced software tools since 1984
www.crossware.com
info@crossware.com


Need A Portable, High Speed, High Resolution PC Scope?
Introducing the PicoScope 4227
* Up to 250 MS/s real-time sample rate
* Up to 100 MHz bandwidth
* 12 bit resolution
* 32 M sample buffer memory
* 2 channels
* Ext Trigger Input
* Arbitary waveform generator
* USB Powered
* Free software updates
* 5 year warranty
pico®
Technology
http://www.picotech.com 1-800-591-2796


CIRCUIT CELLAR®
CC-ACCESS
SUBSCRIBERS RESERVED AREA
CC 2.0
Subscribers
Get More
- Instant access to source code & project files
- Receive special offers from preferred vendors
- Easily maintain account information
- Search for your favorite articles quickly
Combo 2.0!
Print when you want it,
Digital when you don't.
Get yours today.
www.CC-Access.com


UART-WiFi, UHF EPC Gen2 RFID Reader, Powerline Communication
Long Range UHF RFID reader (6 meters, 12 meters, and 25 meters) (ISO18000-6C EPC G2)
TTL UART to WiFi
JPEG Serial Camera
•UART to Power line Communication Modules
•Bluetooth modules
•Cellular modules
Single Board Computer with Win CE 6.0, Linux, and Android. Optional on-board Cellular, WiFi, GPS
www.linksprite.com


CCS
Optimized C Compiler for PIC® MCU
• Decrease code size by 40%
• Use less expensive MCU with smaller program memory
• Save thousands of dollars in component costs over product lifetime
CCS Compilers Save You Money
Compilers Starting at $150
www.ccsinfo.com/SAVE
sales@ccsinfo.com
262.522.6500 x35

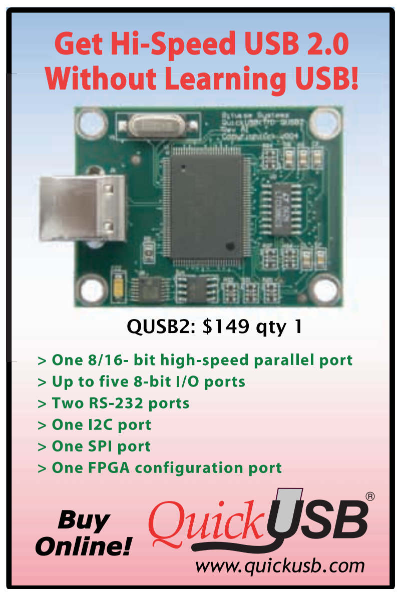

Order online at:
www.melabs.com
microEngineering Labs, Inc.
Development Tools for PIC® Microcontrollers
Phone: (719) 520-5323
Fax: (719) 520-1867
Box 60039
Colorado Springs, CO 80960
USB Programmer for PIC® MCUs
$89.95
(as shown)
RoHS Compliant
Programs PIC MCUs including low-voltage (3.3V) devices
Includes Software for Windows 98, Me, 2K, & XP
With Accessories for $119.95:
Includes Programmer, Software, USB Cable, and Programming Adapter for 8 to 40-pin DIP.
LAB-X Experimenter Boards
Pre-Assembled Boards
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 $79.95 to $349.95
BASIC Compilers for PICmicro®
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
32-bit signed variables and math operations
Supports Microchip PIC10, PIC12, PIC14, PIC16, PIC17, and PIC18 microcontrollers
Direct Access to Internal Registers
Supports In-Line Assembly Language
Interrupts in PICBASIC and Assembly
Built-In USB, I2C, RS-232 and More
Source Level Debugging
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
Serial LCDs
2-line x 16 $39.95
4-line x 20 $49.95
Quantity Discounts Available!
See our full range of products, including books, accessories, and components at:
www.melabs.com


MaxSonar
Ultrasonic Ranging is EZ
LV-MaxSonar-EZ
• Power-up calibration
• Smallest MaxSonar
• 1 inch resolution
•Low power, 2.5V-5.5V
MaxSonar-WR (IP67)
• Industrial packaging
• Weather resistant
• Standard ¾" fitting
• Quality narrow beam
XL-MaxSonar-EZ
• Beam pattern choice
• Tiny size
• Real-time calibration
•High acoustic power
MaxSonar-WRC (IP67)
• Compact packaging
• Weather resistant
• Standard ¾" fitting
• Quality narrow beam
www.maxbotix.com


embedded adventures
New modules available!
Learn about USB with our completely hackable USB2Serial module.
www.embeddedadventures.com

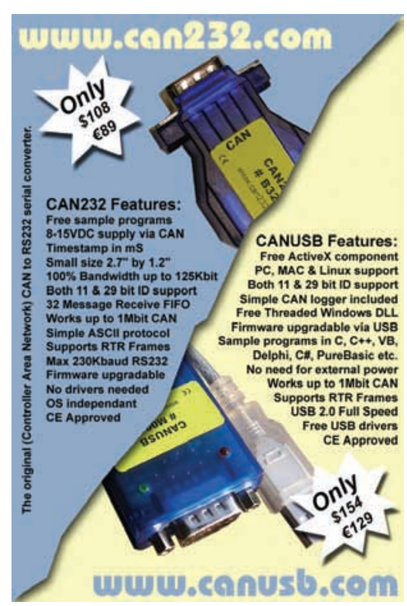

Disk On Chip
Ready for Delivery
From 16MB to 128MB Available!
Call 530-297-6073 Sales@jkmicro.com
www.jkmicro.com
JK microsystems, Inc.
International Orders Welcome

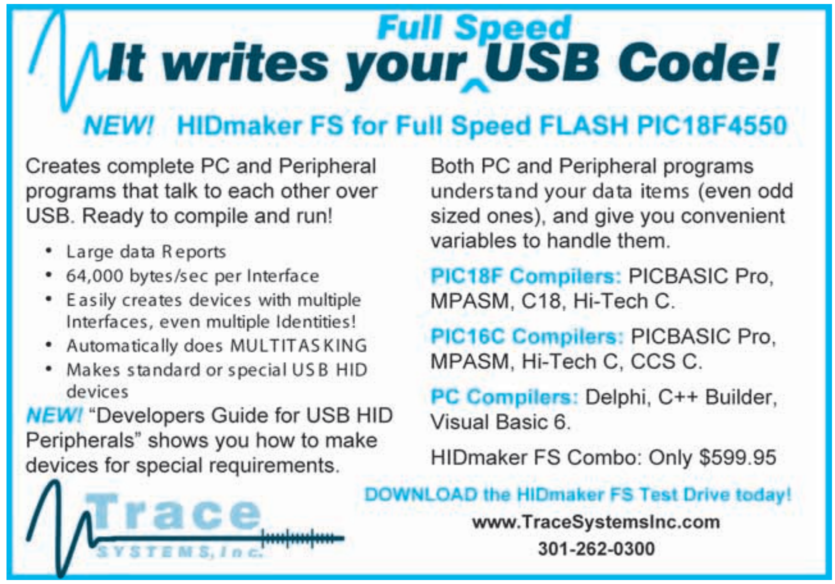

Fine Circuits Inc.
ITAR
ISO 9001:2000
Sales@finecircuits.com
630*213*8700
PROUDLY PRESENT:
Instant Online Order
SPECIAL OFFERS
2 LAYERS @ 11/EA
4 LAYERS @ 33/EA
FREE GROUND SHIPPING
Free 1st article with production order...
Quality supplier of PCB with wide range of rigid and flex manufacturing capabilities include.. 1-30 Layers same day to standard lead time, 4mil laser drill 3/3 line spacing hybrid and Teflon PCB, Noflow prepregs and dielectric thickness, Impedance control with polar report. Stock Material: FR4 TG 140-180 370HR/IS410 poly Nelco, Rogers, Lead Free
Www.finecircuits.com

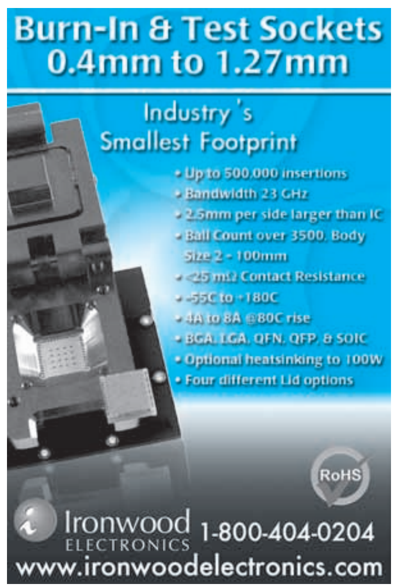

Get Hi-Speed USB 2.0
Without Learning USB!
QUSB2: $149 qty 1
> One 8/16- bit high-speed parallel port
> Up to five 8-bit I/O ports
> Two RS-232 ports
> One I2C port
> One SPI port
> One FPGA configuration port
Buy Online!
QuickUSB®
www.quickusb.com


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


Display with Touch Screen
• Embedded, Standalone, C/C++ Programmable...
• Low Cost User Interface with Ethernet, USB, CF, RS232, ADC, DAC, I/Os, Relays...
• U-Drive™
5.7" QVGA TFT, CAN, Ethernet, CF, ADC, DAC, ...
start at $379 OEM
60+ Low Cost Controllers with TFT, ADC, DAC, UARTs, 300 I/Os, solenoid, relays, Compact-Flash, LCD, Ethernet, USB, motion control. Custom board design. Save time and money.
TERN INC.
1950 5th Street, Davis, CA 95616 USA
Tel: 530-758-0180 • Fax: 530-758-0181
www.tern.com • sales@tern.com


www.can232.com
Only $108 €89
The original (Controller Area Network) CAN to RS232 serial converter.
CAN232 Features:
Free sample programs
8-15VDC supply via CAN
Timestamp in mS
Small size 2.7" by 1.2"
100% Bandwidth up to 125Kbit
Both 11 & 29 bit ID support
32 Message Receive FIFO
Works up to 1Mbit CAN
Simple ASCII protocol
Supports RTR Frames
Max 230Kbaud RS232
Firmware upgradable
No drivers needed
OS independant
CE Approved
CANUSB Features:
Free ActiveX component
PC, MAC & Linux support
Both 11 & 29 bit ID support
Simple CAN logger included
Free Threaded Windows DLL
Firmware upgradable via USB
Sample programs in C, C++, VB, Delphi, C#, PureBasic etc.
No need for external power
Works up to 1Mbit CAN
Supports RTR Frames
USB 2.0 Full Speed
Free USB drivers
CE Approved
Only $154 €129
www.canusb.com





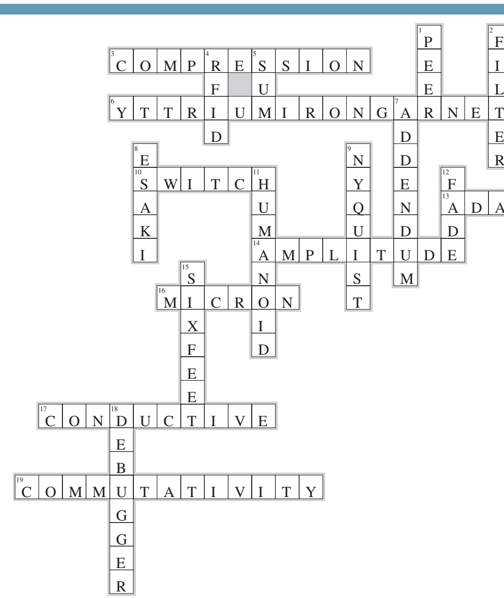
CROSSWORD ANSWERS from Issue 243

Across

3. **COMPRESSION**—5 MB to 3 MB
6. **YTTRIUMIRONGARNET**—Yig [three words]
10. **SWITCH**—Disengage or complete a circuit
13. **ADA**—Embedded computing language developed by the DOD
14. **AMPLITUDE**—Damping decreases a vibration's _____
16. **MICRON**—µm
17. **CONDUCTIVE**—Trace line on a PCB
19. **COMMUTATIVITY**—A different arrangement, but same result; 6 + 5 = 5 + 6

Down

1. **PEER**—Same-to-same
2. **FILTER**—Finite impulse response, infinite impulse response, fast Fourier transform
4. **RFID**—Technology behind "e-tolling"
5. **SUM**—Computed by an adder
7. **ADDENDUM**—Content added to the end of a datasheet
8. **ESAKI**—Fast diode; electron tunneling
9. **NYQUIST**—Sampling theory; Harry
11. **HUMANOID**—A robot that looks and moves like a human
12. **FADE**—Signal reduction
15. **SIXFEET**—Fathom [two words]
18. **DEBUGGER**—An error-finding app



INDEX OF ADVERTISERS

PREVIEW of December Issue 245

Theme: Programmable Logic

MCU-Based Aircraft Data Logging and Telemetry (Part 2): System Development & Results

Interactive USB Module

Mobile Application Development (Part 1): Program a Sound-Monitoring Prototype

THE CONSUMMATE ENGINEER **Good Old PALs**

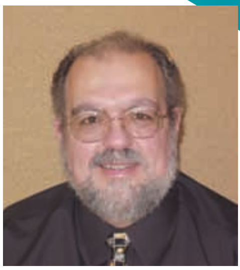
THE DARKER SIDE **Parasitic Components:** When Capacitors Become Inductors

ABOVE THE GROUND PLANE **Real-Life VHF Data Transmission**

FROM THE BENCH **Encoder Essentials:** How to Close the Loop with Hardware

SILICON UPDATE **What's My Line?:** Moving Toward a General-Purpose Chip

PRIORITY INTERRUPT



by Steve Ciarcia, Founder and Editorial Director

Low-Tech Success

I used to do a lot of engineering consulting years ago when I owned a manufacturing company. As you might guess, since then, I've thought it is easier to publish your engineering achievements than originate them all myself. ;-) Don't get me wrong, consulting helped me keep my skills tuned, but too often these consulting jobs turned out to be corporate witch hunts—management hires someone to evaluate their resources and take the blame when the hammer drops.

So, after years of consulting retirement, I was recently helping a friend determine if his engineering team was on the right track. After introducing me (but not my function), he turned the meeting over to the lead test engineer—a 20-something recent graduate—and said he'd be back in a while to check on us. Basically, the team was required to design an engine control simulator with a single output that switches between two common-ground, low-current, externally supplied voltage standards, 1.00 VDC (±0.1%) and 8 V at 4 Hz. They were selected in response to two isolated input control signals. The normal output was "reset" (1.00 VDC) and the other was "set" (8 VAC). The available test-stand power supply voltage was 48 V 400 Hz. Most important, because of the expense of the turbine under test, the simulation switcher couldn't lose its marbles! When the switcher was in the "set" position, this couldn't change until there was a true Reset signal to switch it back to DC output. The AC sweep had to remain on during a power interrupt even until a complete test-stand shutdown (up to 8 hours). And, oh yeah, the operating temperature was –30°C to +65°C, and any circuit had to be static discharge protected.

The four test engineers at the table looked at each other and voiced a silent but collective "wow." It wasn't hard to see that the gears were turning and they were considering all the tradeoffs. Since only 20 test-stand switchers had to be made, high-volume production pennies wouldn't be an issue. As I listened, it reminded me why I started my own business. ;-)

The lead engineer cleared his throat and went to the marker board. "Let's see. We'll need an uninterruptable power supply, but we're probably only talking a few hundred milliamps to run the actual controller. We could design our own 400-Hz input DC switcher for 5-V or 12-V system power and add a pile of extra batteries, but I suppose we can also use an off-the-shelf UPS and a commercial DC power supply. My guess is that a 1000-VA UPS has the capacity to last 8 hours."

One of the other engineers quickly piped in: "I don't think you can use a regular commercial UPS. The electronics is tuned for 60-Hz back-up switching and may not run on 400 Hz. We'll probably have to order a mil-spec UPS or use a 400-Hz motor-generator, especially for that temp range." He sat down with a satisfied look on his face.

"OK, that's done." The lead engineer made a check on the board. "We also have to edge-detect two isolated inputs and basically do a SPDT (single pole double throw) voltage switch with maybe FETs or something …"

I raised my hand and politely interrupted, "Do the set/reset pulses ever overlap? What is their duration? And, what is their voltage level and load capacity?"

The engineer looked down at the spec, flipped through a few pages, and replied: "No, 80 ms, and 4.9 VDC at 53 mA max current." He wasn't impolite, but I got a distinct impression that he wondered why I was asking. "Anything else? Nope. OK, let's continue. I think a simple microprocessor should handle the task. We'll use a couple optoisolators on the inputs. Ordinarily, I'd use a couple triacs on the output to switch the voltages, but it has too much loss for 1 VDC. We'll have to use FETs or add a power driver and run a mechanical relay to avoid switching losses. I suppose we could design our own hardware and PCB using an NXP or Microchip controller, but I'd rather concentrate on the software design with so few to implement. Considering the electrical and environmental issues, perhaps we should go with VME or PC-104 and concentrate on the software …."

I just couldn't let this go on any longer without saying something. Certainly they were all bright engineers and using a hundred pounds of technology to do a one pound job is fine as long as you don't mind wasting 99. Perhaps we have become so high-tech-oriented that we never look for simple solutions anymore? As carefully as I could, so as to not hurt any feelings, I spoke up: "I appreciate that sometimes there is greater political prestige showing management a half ton of electronics you skillfully wrangled together to execute their order (fully knowing they haven't a clue what it is, but it looks good). Given the tight economy, however, do you think there is equal political capital showing up with a ridiculously simple solution?"

I didn't wait for the "I suppose so" and tossed a schematic on the table. It contained one $9 component: a 5-V dual-coil, polarized latching relay with SPDT contacts. "Simply attach the set and reset signals, one to each isolated coil, and the two voltage standards to the SPDT switch. Pulse on, pulse off. No power supply, industrial temp, surge protected, and it stays set forever until a reset pulse." I chuckled and said, "Think analog for a change."

Just then, my friend walked in the door and asked about progress. I jumped in before anyone could answer: "The team has discussed a variety of methods from elegantly simple to completely IT-department-compatible. They can explain the options to you, but I'm sure you won't have any trouble selling whatever it is to management." ;-)

As I was getting up to leave, the lead engineer leaned over and whispered, "Thanks."

Steve

steve.ciarcia@circuitcellar.com

GPS

Parallax GPS Module	PMB-648 GPS SiRF Internal Antenna	PMB-688 GPS SiRF Internal/External Antenna	RXM-SG GPS Module w/External Antenna
#28146; $79.99	#28500; $34.99	#28501; $39.99	#28505; $79.99
Built-In Patch Antenna	Built-In Patch Antenna	Built-In Patch Antenna or optional External (#28502)	Included External Antenna
Breadboard-friendly 4-pin SIP 2 modes of operation: RAW - Unformated data Smart - Syntax Formatted	Included cable connects to breadboard or prototype with stripped wires	Included cable connects to breadboard or prototype with stripped wires	Four general purpose I/O pins provide expansion for pin intensive projects
TTL/CMOS I/O	TTL/CMOS, RS232 Outputs	TTL/CMOS outputs	3.3 V CMOS asynchronous serial @ 9600 baud default for controllers, or USB for PC
Custom ASIC GPS Controller	SiRFstar III GPS Controller		
Tracks up to 12 satellites	Tracks up to 20 satellites		
Position: 2DRMS ~ 2m, WAAS support Velocity: 0.1 m/s Time: +/- 1 us	Position: 2DRMS ~ 5m, WAAS support Velocity: 0.1 m/s Time: +/- 1 us		
Baud Rate: 4800 bps	Baud Rate: 4800 bps (Optional 9600, 19200, 38400)		Baud Rate: 9600 bps (Optional 4800, 19200, 38400, 57600)

Order **GPS Modules** at www.parallax.com or call Sales toll-free at 888-512-1024 (Mon - Fri, 7:00 a.m. - 5:00 p.m., PST).

Friendly microcontrollers, legendary resources.™

PARALLAX INC

www.**parallax**.com

"ParallaxInc" on Twitter, Facebook, and YouTube