

#205 August 2007

www.circuitcellar.com

CIRCUIT CELLAR®

THE MAGAZINE FOR COMPUTER APPLICATIONS

EMBEDDED DEVELOPMENT

Build a Handheld Scope

PHP Components Database

Understanding EMI

Silicon Secrets



Announcing a complete hardware and software solution from NetBurner

The NetBurner MOD5234

ETHERNET CORE MODULE with eTPU



Only **\$99**
Qty. 100

NetBurner MOD5234 Ethernet Core Module Features

INDUSTRIAL TEMPERATURE RANGE

| -40°C to +85°C

PERFORMANCE AND MEMORY

| 32-Bit CPU | Freescale ColdFire MCF5234 147 Mhz
| 2MB Flash Memory | 8MB SDRAM

DEVICE CONNECTIVITY

| 10/100 Ethernet | 3 UARTs | 16-channel eTPU | I²C | SPI | CAN
| 47 Digital I/O | 16-bit Data Bus | SD/MMC Flash Card Support

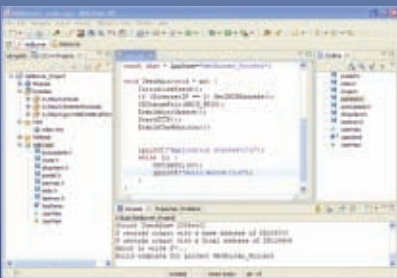
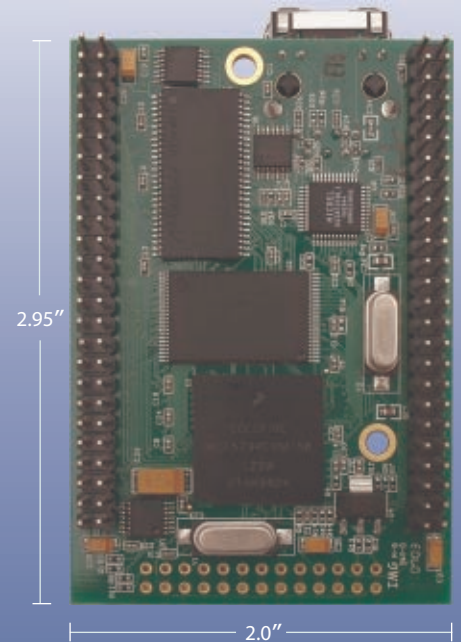
Customize with NetBurner's Royalty-free Software Suite

DEVELOPMENT SOFTWARE

| NB Eclipse IDE | Graphical Debugger | Deployment Tools | Examples

COMMUNICATION SOFTWARE

| TCP/IP Stack | HTTP Web Server | FTP | E-Mail | PPP | Flash File System



All hardware and software is included with the

NetBurner MOD5234 Development Kit for only \$299!

The Development Kit features NetBurner's Eclipse, an enterprise level professional IDE that provides editing, downloading and debugging in one environment.

Order the MOD5234 Development Kit: Product No. NNDK-MOD5234-KIT



Product No. | MOD5234-100IR
Information and Sales | sales@netburner.com
Web | www.netburner.com
Telephone | 1-800-695-6828



CapSense. Beauty is more than skin deep.

PSoC®-based Capacitive Touch Sensing

Maximize the design flexibility and integration of Cypress's PSoC solution to create a stylish, durable interface. CapSense replaces buttons, switches, sliders and other mechanical inputs in your product. CapSense enables:

- Fast changes to your design at any stage from concept through production. CapSense is not a fixed-function ASIC or module; you are in control of your design at all times.
- Single-chip implementation supporting multiple interfaces – buttons, sliders, touch screens, touchpads and proximity detectors – on a variety of conductive substrates.
- Unique integration of additional functions – LED control, battery monitoring, motor control, ambient light sensing, etc. – all with the same CapSense chip.
- Quick time-to-market with powerful, visual embedded design tools allowing customized, system-level design.

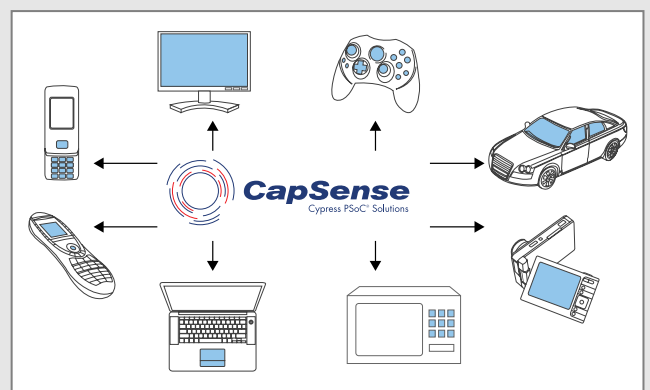
GET STARTED WITH CAPSENSE NOW

Order a discounted CapSense Development Kit:
www.cypress.com/capkit

Request free PSoC CapSense IC samples:
www.cypress.com/capchips

Download free PSoC Express™ visual embedded software:
www.cypress.com/capexpress

Register for a CapSense NetSeminar:
www.cypress.com/capseminar



Applications enabled by PSoC® CapSense.

Streamline your next design with CapSense:
www.cypress.com/gocapsense





Link Instruments

PC-Based Test Equipment

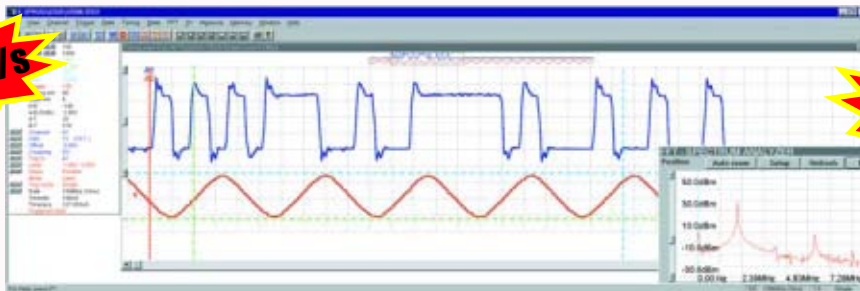
Digital Oscilloscopes



- 2 Channel Digital Oscilloscope
- **500 MSa/s** max single shot rate
- 1Mpt sample memory
250 MSa/S (Dual channel) 512 Kpts
500 MSa/S (Single channel) 1 Mpts
- Only 9 oz and 7" x 3.5" x 1.5"
- Portable and Battery powered
- USB 2.0
- FFT Spectrum Analyzer

DSO-8202 (200MSa,128K) **\$799**
 DSO-8502 (500MSa,1MPT) **\$950**

500MSa/s

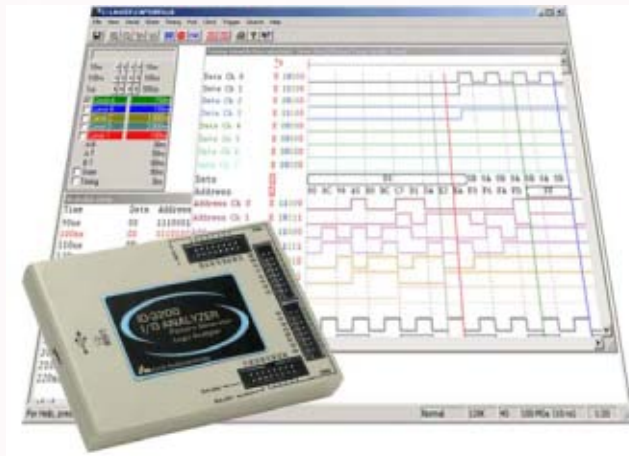


1Mpts

Logic Analyzer & Pattern Generator

NEW!

Portable
3.2" x 3" x 0.65"
USB 2.0 Powered



- Logic Analyzer (32 channels)
- Pattern Generator (up to 32 channels)
- up to 400 MSa/s
- Variable Threshold
- 2 External Clocks
- SPI output and disassembly
- I²C output and disassembly
- up to 2Msamples/ch

IO-3208A **\$750**
 IO-3232A **\$899**
 IO-3232B **\$1399**



Link Instruments (973) 808-8990

www.Linkins4.com



**FOR RELEASE
THIS MONTH**

DENSER CODE
better performance

PROPSOC[®]
for
with OMNISCIENT CODE GENERATION[™]

HI-TECH PRO for PSoC ANSI C COMPILER* - from the company that has, for over two decades, delivered the industry's most reliable embedded software development tools and compilers - comes with:

OMNISCIENT CODE GENERATION
Extracts information from multiple source files simultaneously, allowing more intelligent state-of-the-art code generation that:

- Optimizes the size of each pointer variable in your code based on its usage;
- Eliminates contention for PSoC index register;
- Eliminates the need for many non-standard C qualifiers and compiler options;
- Produces more efficient interrupt context switching code; and
- Customizes the functionality of the printf library function.

* Can run on multiple platforms including **Windows, Linux and Mac OS X.**

*Integrates into PSoC Designer[™] IDE and other PSoC development tools.

DENSER CODE, BETTER PERFORMANCE
Code compiled with HI-TECH PRO for PSoC can deliver over 40% denser code than its competitor.

FREE 45 DAY EVALUATION
A fully functional 45 day trial version of HI-TECH PRO for PSoC will be available, free of charge from HI-TECH's website. Order your demo at www.htsoft.com/portal/ccpsoc.

PRICE AND AVAILABILITY
HI-TECH PRO for PSoC includes, at no extra cost, HI-TECH PRIORITY ACCESS[™] - 12 months access to updates and technical support and HI-TECH SATISFACTION GUARANTEE[™] - a hassle free 30 day money back guarantee.



PSoC Designer[™] is a trademark of Cypress Semiconductor Corporation.

PSoC[®] is a registered trademark of Cypress Semiconductor Corporation.

HI-TECH
SOFTWARE
HI-TECH Software LLC

6600 Silacci Way Gilroy, CA 95020 USA
Ph: 800 735 5715 Web: <http://www.htsoft.com>

TASK MANAGER

Another Eclectic Mix

Here in the Northern Hemisphere, the warm months of June, July, and August mark our summer season. During this time, many engineers exercise their vacation days at work and take some time to travel, relax, and spend time with their families. It's also a time of contemplation. During the last two months, I've received dozens of e-mails from engineers who have been using their summer downtime to work on a wide range of projects. During the last few weeks in particular, I've read proposals and article submissions about a barometric altimeter, a ZigBee-based thermostat, and an "ultra-stable" precision frequency calibrator. These are the seeds of some great articles. We're looking forward to working on them.

If you've been brainstorming during the last few months, we'd love to hear from you. A simple e-mail describing your project will do. As you know, our submission guidelines and editorial calendar are available at www.circuitcellar.com/submissions.

This issue offers an eclectic mix of articles touching on a variety of interesting topics. As usual, there is something for everyone, from retired engineers interested in new hardware apps to young grad students with a penchant for video games.

On page 14, Jingxi Zhang and Yang Zhang describe how they built an award-winning multifunctional oscilloscope. The design placed first in the highly competitive Luminary Micro DesignStellaris2006 Contest. After you read the article, go to www.circuitcellar.com/design_stellaris2006/winners/winners.html to learn more about the project.

Ready for the next big thing in bicycling? On page 24, a team of designers from Camosun College describe an interesting electronic bicycle design that you can easily replicate on your workbench. It's the perfect combination of manpower and machine power.

Have you ever played the Pyramid 2000 text game? In "Text Adventure Gaming" (p. 40), Chris Cantrell explains how to give it a go with a Parallax Propeller chip and his cog coordination language (CCL).

On page 48, Keith Brown describes how to build a handy PHP components database. The useful program enables you to log and track the parts you keep in your design space.

The last feature article is the first part of Steve Hendrix's series entitled "1-Wire in the Real World" (p. 58). In this first installment, Steve, who is a great designer and an avid aviation enthusiast, explains the challenges associated with taking a 1-Wire deicing system for small aircraft from a laboratory prototype to a real-world application.

Finally, note that we're happy to welcome Robert Lacoste to the *Circuit Cellar* columnist team. From 1998 to 2004, Robert wrote 10 feature articles for us. His first article, "PIC's Spectrum: Audio Spectrum Analyzer," appeared in issue 98. This month, we're running Robert's first column, "Let's Play with EMI!" (p. 66). In the future, Robert's column—"The Darker Side," which is devoted to some of the lesser-known aspects of electronic design—will appear every other month.

cj@circuitcellar.com



CIRCUIT CELLAR®

THE MAGAZINE FOR COMPUTER APPLICATIONS

FOUNDER/EDITORIAL DIRECTOR
Steve Ciarcia

CHIEF FINANCIAL OFFICER
Jeannette Ciarcia

MANAGING EDITOR
C.J. Abate

MEDIA CONSULTANT
Dan Rodrigues

WEST COAST EDITOR
Tom Cantrell

CUSTOMER SERVICE
Debbie Lavoie

CONTRIBUTING EDITORS
Jeff Bachiochi
Ingo Cyliax
Robert Lacoste
George Martin
Ed Nisley

CONTROLLER
Jeff Yanco

ART DIRECTOR
KC Prescott

GRAPHIC DESIGNER
Mary (Turek) Sobuta

NEW PRODUCTS EDITOR
John Gorsky

STAFF ENGINEER
John Gorsky

PROJECT EDITORS
Steve Bedford
Ken Davidson
David Tweed

ASSOCIATE EDITOR
Jesse Smolin

ADVERTISING

860.875.2199 • Fax: 860.871.0411 • www.circuitcellar.com/advertise

PUBLISHER

Sean Donnelly
860.875.2199, Cell: 860.930.4326, E-mail: sean@circuitcellar.com

ADVERTISING REPRESENTATIVE

Shannon Barraclough
860.875.2199, E-mail: shannon@circuitcellar.com

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.circuitcellar.com/subscribe, E-mail: subscribe@circuitcellar.com

Subscribe: 800.269.6301, www.circuitcellar.com/subscribe, Circuit Cellar Subscriptions, P.O. Box 5650, Hanover, NH 03755-5650

Address Changes/Problems: E-mail: subscribe@circuitcellar.com

GENERAL INFORMATION

860.872.3064, 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 \$23.95, Canada/Mexico \$34.95, all other countries \$49.95. Two-year (24 issues) subscription rate USA and possessions \$43.95, Canada/Mexico \$59.95, all other countries \$85.** All subscription orders payable in U.S. funds only via Visa, MasterCard, international postal money order, or check drawn on U.S. bank. **Direct subscription orders and subscription-related questions to Circuit Cellar Subscriptions, P.O. Box 5650, Hanover, NH 03755-5650 or call 800.269.6301.**

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

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

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

Entire contents copyright © 2007 by Circuit Cellar, Incorporated. All rights reserved. Circuit Cellar is a registered trademark of Circuit Cellar, Inc. Reproduction of this publication in whole or in part without written consent from Circuit Cellar Inc. is prohibited.

OPA3875 High-Speed Video Multiplexer

Very wide band, 3-channel, 2:1 multiplexer in a small SSOP-16 package.



mouser.com/ti/a

AVR Microcontrollers

RISC core running single-cycle instructions with well-defined I/O structure limits for external components.



mouser.com/atmel/a

Faster Time to Market for YOUR New Designs!



- The ONLY New Catalog Every 90 Days
- NEWEST Products & Technologies
- Over 845,000 Products Online
- More Than 330 Manufacturers
- No Minimum Order
- Fast Delivery, Same-day Shipping

mouser.com (800) 346-6873



a tti company

*The Newest Products
For Your Newest Designs*

3-in-1 PWM Dual Buck and Linear DDR Power Controller



Complete power solution for MCH and DDR memory.



ON Semiconductor™

mouser.com/onsemiconductor/a

Fail-Safe Floating Electrode MLCC/FE-CAP



SMD MLCC utilizes floating internal electrode design, where electrodes form capacitors in single MLCC packages.

mouser.com/kemet/a

B3U Tactile Switch



Industry's smallest ultra-small tactile switch with high contact reliability.



OMRON
ELECTRONIC COMPONENTS

mouser.com/omron/a

August 2007: Embedded Development

FEATURES

- 14 **Handheld Multifunction Scope**
Jingxi Zhang & Yang Zhang
First Prize Luminary Micro DesignStellaris2006 Contest
- 24 **Electronic Bicycle Design**
Roxanne Ahmed, Michael Buchanan, Andrew Clarke, Chris Elder, & Warren Shrieves
- 40 **Text Adventure Gaming**
Chris Cantrell
- 48 **Build a PHP Components Database**
Keith Brown
- 58 **1-Wire in the Real World (Part 1)**
The Challenges
Steve Hendrix

Multifunction
Oscilloscope Design (p. 14)



Pedal Power (p. 24)

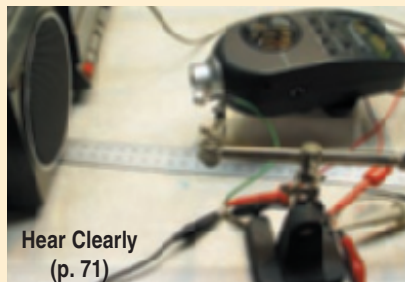


Gaming with Propeller &
CCL (p. 40)



COLUMNS

- 66 **THE DARKER SIDE**
Let's Play with EMI!
Robert Lacoste
- 71 **ABOVE THE GROUND PLANE**
Hearing Clearly
Ed Nisley
- 75 **FROM THE BENCH**
Graphical User Interfacing
Jeff Bachiochi
- 80 **SILICON UPDATE**
Silicon Secrets
Tom Cantrell



Hear Clearly
(p. 71)



CryptoMemory (p. 80)

DEPARTMENTS

- 4 **TASK MANAGER**
Another Eclectic Mix
C.J. Abate
- 8 **NEW PRODUCT NEWS**
edited by *John Gorsky*
- 93 **CROSSWORD**
- 94 **INDEX OF ADVERTISERS**
September Preview
- 96 **PRIORITY INTERRUPT**
Let's Count the Lumens
Steve Ciarcia



AT LAST!
AVR man's deepest secrets
are mine! MINE! And the birdbrain
is not to be seen!

It's all on this
hard disk! This technology
will make me the mightiest of
all! Even mightier than
that anabolic steroid
eater!

AVR man will never
catch me in The wastecave.
I'll be supersafe!

KLIK

Wastecave
GARAGE
KEEP OFF!

ROAD

BUT!!!!?
The battery is dead!
It doesn't work!
NOOOOO!!

**KLIK KLIK
KLIK KLIK
KLIK KLIK**

CRASH!!!

I'll take this back
since it's mine, Wasteman.
Anyhow, it's loaded with stuff
beyond your feeble
understanding.

**DUNK
DUNK
DUNK**

AVR

I'm giving you this
door opener for your own
safety. It'll last forever
and a day!

It's full of
AVR technology...
of course!!

FANTASI-FABRIKKEN 4/5 2007 - MYKLE - WOOD

Complete solution for **Remote Access Control**

Get more at: www.atmel.com/AVRman



© 2007 Atmel Corporation. All rights reserved. Atmel®, AVR® and logo are registered trademarks of Atmel Corporation or its subsidiaries. Other terms and product names may be trademarks of others. All Characters in this document are created by Mykle and Fantasi-Fabrikken AS 2007.

EVALUATION BOARD FOR LOW-COST IP CONTROLLER

The release of the **II EVB-630** evaluation board demonstrates how the iChip CO2128 IP controller provides Internet connectivity, encryption, and superior security in architecture and software for a host processor or device. The II EVB-630 supports the effortless integration of many upper-layer Internet protocols using either AT+i commands or the iChipConfig utility. They can also be used for evaluation and development of secure Wi-Fi, LAN, cellular, or dial-up networking, offering developers maximum flexibility in designing machine-to-machine applications.

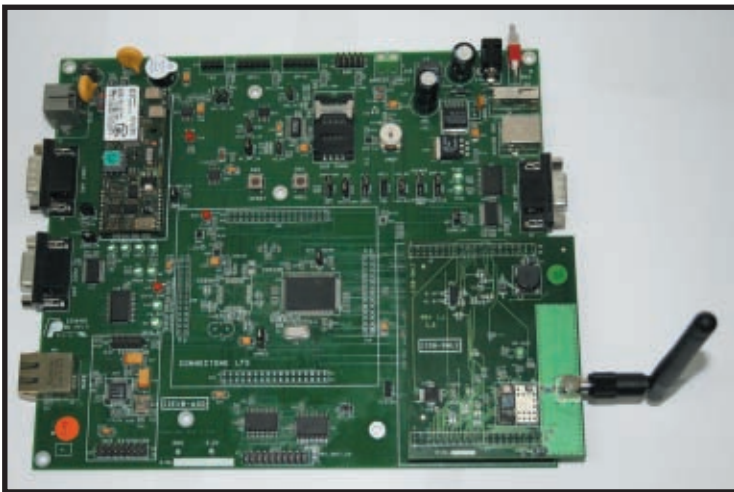
The II EVB-630 acts as a complete stand-alone platform that enables developers to fully test out, develop, and debug the func-

tionality of the CO2128's firmware with no modifications to the development environment. The chip features an open architectural design so developers can choose the Internet protocols required for their applications. The initial firmware, which is locally updatable firmware, supports up to 10 simultaneous TCP/UDP sockets or secure sockets (SSL3/TLS1), two listen sockets, HTTP/HTTPS, SMTP, FTP/FTPS and TELNET clients, SerialNET serial-to-IP bridging, an HTTP (web) server, MIME attachments, a POP3 client, and RAS. Future versions will support remotely updatable firmware.

The II EVB-630 comes loaded for connectivity. To complement the two iChipSec CO2128 chips included for prototyping, the board offers two USB host and device connectors, two male-female RS-232 DE-9 connectors, an RJ-45 connector, an RJ-11 connector, a SocketModem, a Wi-Fi daughter board with an antenna, a four-pin TWI connector, two six-pin SPI connectors, one 10-pin SSC connector, and an SIM holder with either a 110- or 220-V power supply. LEDs provide valuable feedback. Serial cables are included.

The II EVB-630 for secure Wi-Fi, LAN, cellular, or dial-up access costs \$1,725.

Connect One
www.connectone.com



DIGITAL LOAD SENSORS WITH USB OUTPUT

The **iLoad Series** of digital load/force sensors are based on patented capacitive sensing technology. The sensors combine the core load or force-sensing function with integrated computing and communications modules to offer plug-and-sense simplicity and unprecedented ease of use in a rugged, low-profile package.

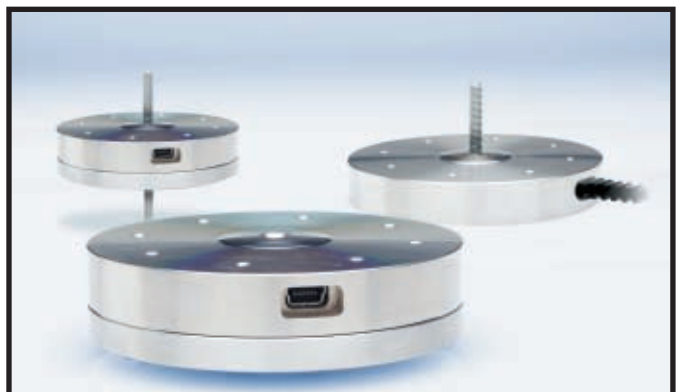
The iLoad digital USB sensor is the world's first capacitive load sensor with USB output. The load sensors are designed to plug directly into the USB port of a PC and enable users to measure loads, forces, or weights without any additional signal conditioning, data acquisition, or special software! The sensor appears as a virtual COM port on a PC, enabling a user to directly read data using simple ASCII commands with any terminal emulator. To make the process even simpler, Loadstar offers LoadVUE Lite software that takes care of all communication functions and displays loads in pounds, kilograms, or newtons.

For users who need an analog output, the iLoad analog sensor is built on the same digital platform as the digital USB version, but it is designed to interface directly to a PLC or DAQ system. The iLoad analog sensor offers a linear, 0.5- to 4.5-VDC output proportional to applied loads. No expensive amplification or signal conditioning is required because all electronics are built into the self-con-

tained integrated sensor.

The iLoad Series is available in 10-, 50-, 100-, 250-, and 500-lb capacities to measure both compressive and tensile loads and offer accuracies of 0.25% to 0.025% of full-scale loads. Prices for the sensors start at \$245.

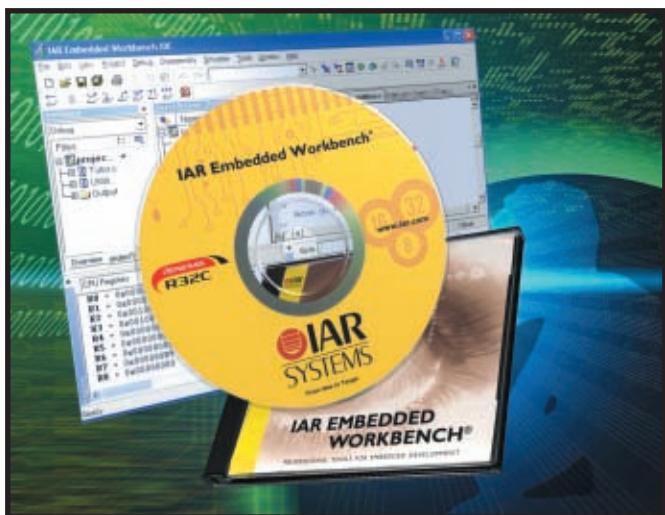
Loadstar Sensors, Inc.
www.loadstarsensors.com



NEW PRODUCT NEWS

IAR EMBEDDED WORKBENCH FOR RENESAS R32C

IAR Systems announces the launch of the **Embedded Workbench for the Renesas R32C series**. The new tools package is the result of a collaborative development and test program involving Renesas Technology and IAR Systems, and it is targeted at developers wishing to use a C/C++ compiler for R32C with advanced features particularly suited to automotive applications.



Because the Embedded Workbench for R32C supports all the devices in the R32C series with header files, device description files, etc., it is simple and quick to begin a new applications development project, saving time that would normally be spent on setting up a project. It also makes it easy to migrate from one MCU to another within the M16C family while still working within a familiar environment and with minimal changes to the source code. The ability to work with projects for different MCUs in the same workspace is especially valuable in the automotive industry where subsystems often use different MCUs, depending on the processing power required.

The Embedded Workbench for R32C includes a C-SPY driver for the Renesas E30 emulator, making it possible to use C-SPY hardware debugging in real time.

Built-in awareness for the OSEK run-time interface (ORTI) makes it possible to debug applications built around this interface standard, which is particularly prevalent within the automotive industry. A built-in MISRA C checker automatically checks C source code for compliance with the MISRA C rules. The Embedded Workbench is the only tool for R32C that integrates both ORTI debugging and a MISRA C checker. The Embedded Workbench for R32C costs \$2,695.

IAR Systems Software, Inc.
www.iar.com

Got Serial, Need Network?

Bluetooth Qty 1 \$175

Ethernet Qty 1 \$99

Wireless Qty 1 \$199

Volume Discounts Available

gridconnect
www.gridconnect.com
+1 800 975-4743

CIRCUIT BOARDS

Manufactured Domestically Since 1984

As low as...

\$9.95
each!

Two Boards
Two Layers
Two Masks
One Legend

Unmasked boards for even less!

e-pcb.com



NEW PRODUCT NEWS

MCUs OFFER NETWORKING WITH THE CORTEX-M3

The Stellaris LM3S6965 Ethernet evaluation kit provides engineers with an easy way to discover the LM3S6000 series of MCUs. The devices in this series are the first MCUs in the world to feature a fully-integrated 100-Mbps Ethernet solution with ARM architecture compatibility. LM3S6000 devices combine both the media access control (MAC) and physical (PHY) layers, marking the first time integrated connectivity is available with an ARM cortex-M3 MCU and the only integrated 10/100 Ethernet MAC

and PHY available in an ARM architecture MCU.

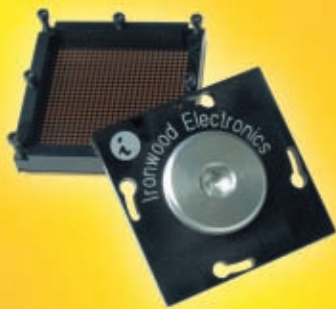
The Stellaris LM3S6965 Ethernet evaluation kit is exceptional for its compactness and versatility and includes a quickstart application that has an embedded web server using the open-source lwIP TCP/IP stack out of the box. Evaluation tools suite choices include an ARM RealView microcontroller development kit (MDK), IAR embedded workbench kickstart edition, and CodeSourcery Sourcery G++ GNU tools. In addition, ported demos of RTOSes include FreeRTOS.org, Micrium μ C/OS-II with μ C/Probe embedded system monitoring, Express Logic ThreadX, CMX Systems CMX-RTX, Keil RTX, IAR PowerPac, and SEGGER embOS. Stacks available include Express Logic NetX TCP/IP, InterNiche TCP/IP NicheStack, NicheLITE, and add-on modules, such as HTTP, SNMP, and security protocols, CMX Systems CMX-MicroNet, FreeRTOS.org-based open-source uIP stack, Micrium μ C/TCP-IP and μ C/Modbus, and RTA Automation EtherNET/IP. The kit also includes an embedded web server application based on the open-source FreeRTOS.org RTOS and open-source uIP Ethernet stack. The Stellaris LM3S6965 Ethernet evaluation kit is available today for \$79.



Luminary Micro, Inc.
www.luminarymicro.com

GHz Bandwidth Sockets for DSP's in BGA

Ironwood Electronics



Sockets for dozens of DSP chips from TI, Analog Devices, and other manufacturers.

- Pitch - 0.3mm to 1.27mm - BGA, QFN (MLF)
- Bandwidth to 10+ GHZ
- Smallest Socket Footprint in Industry
- Optional 500,000 insertions
- Heatsinking to 100 watts

We design custom sockets to your requirements including heat sink option.

Request our Free Catalog/CD today!

Tel: (800) 404-0204
Fax: (952) 229-8201
www.ironwoodelectronics.com

WIRELESS MADE SIMPLE

BRING YOUR PRODUCT QUICKLY AND LEGALLY TO MARKET

RF Modules

Low-Cost TX & RX Modules

Multi-Channel Modules

Long-Range Modules

Add **INSTANT** wireless analog / digital capability to your product.

OEM Products

Handheld TXs

Keyfob TXs

FCC PRE-CERTIFIED & ready to customize for your application.

Function Modules

Antennas

Specialty

GPS

Embedded Chips

Permanent Mount

From ceramic chips to gain Yagis, keyless entry to WiFi.

Whips

Magnetic Base

Gain Antennas

800-736-6677
159 Ort Lane · Merlin, OR 97532
www.linstechnologies.com

NEW PRODUCT NEWS

GlideSensor TOUCH SENSOR DEVELOPMENT KIT

The **GlideSensor development kit** provides a flexible tool set that allows OEMs to rapidly design and integrate custom touch input solutions into their latest consumer electronics products. GlideSensor touch sensors may take the form of thin films, flexible or rigid panels, irregular shapes, and optically clear indium tin oxide (ITO) panels.

GlideSensors can be placed behind and sense through most dielectric materials. They can also be quickly programmed to provide a variety of custom input functions, including text entry, buttons, linear and circular controls, scrollbars, and more. A complete GlideSensor solution combines a low-power Cirque mixed-mode ASIC controller and a touch input sensor. Interface to the host microcontroller is facilitated through either an SPI or I²C connection.

The GlideSensor development kit provides the tools and documentation necessary to introduce the capabilities of GlideSensor touch inputs and quickly evaluate, design, and integrate a touch solution. Developers have full support from Cirque to ensure the design process goes smoothly.

The kit costs \$500.

Cirque Corp.
www.cirque.com



ZigBee DEVELOPMENT TOOLS FOR MICROCHIP-BASED DEVICES

ZigBee Alliance members, Ember, and CCS announce the launch of the **Wireless-Ember ZigBee edition development kit** for wireless applications using Microchip PIC microprocessors. This low-cost development platform includes an Ember EM260 ZigBee network coprocessor and an EmberZnet ZigBee stack on CCS prototyping boards to be used with the CCS C compiler.

The CCS wireless-Ember ZigBee edition development kit enables the fast prototyping of ZigBee applications. The EM260 integrates an IEEE 802.15.4 radio, a network processor, and onboard memory to run a complete ZigBee network protocol stack. The EM260 reduces a product's cost, size, and power consumption with its inexpensive small package size, and twice the wireless range of competitors.

The development kit includes a sink board (with an EM260 module) with the PIC18LF4620, sensor boards (with EM260 modules) with the PIC16LF886, the CCS PCWH C compiler, an in-circuit debugger/programmer, an exercise book,


batteries and cables, and all the required royalty-free software source code.

Please note that the Ember USBLinks programmer for the EM260 will be sold separately at CCS.

The CCS development kit will be sold for a limited time to *Circuit Cellar* readers for \$599. After that, the standard price will be \$624.

CCS, Inc.
www.ccsinfo.com



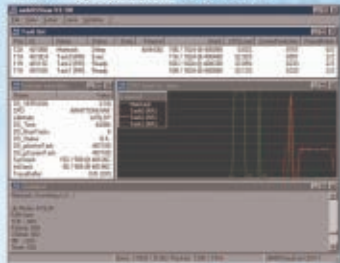


Eval Versions Available

RTOS (embOS®)

+++ 8/16/32 bits +++

- Object Or Source Code
- Small Footprint
- Preemptive Multitasking
- Zero Interrupt Latency
- Easy to Use Start Project
- Profiling Support Included




- RTOS profiling viewer

GUI (emWin®)

+++ 8/16/32 bits +++

- ANSI "C" Source Code
- 2D Graphic Library
- Window Manager/Widgets
- PC Simulation Included
- RTOS Independent
- Font Converter
- Image Converter



- car dashboard example

FILE SYSTEM (emFile)


+++ 8/16/32 bits +++

- ANSI "C" Source Code
- MS-DOS/MS-Windows Compatible
- FAT12, FAT16, FAT32 Support
- Non FAT File System Available
- RTOS Independent

JTAG (J-Link™)

+++ ARM7/9 Cortex M3 +++

Fast 720 kb/s Download
USB to JTAG



phone: 978-874-0299

www.segger.com

NEW PRODUCT NEWS

UP TO 27 TEMPERATURE SENSORS ON A SINGLE I²C BUS

The DS75LX is a digital temperature sensor that allows up to 27 addresses on one I²C communication bus. To enable 27 address combinations, the temperature sensor provides three dedicated address pins that can be placed in one of three states: tied to GND, tied to VDD, or left floating. Most competing devices allow only the three address pins to be tied to GND or VDD, resulting in a maximum of only eight addresses.

The DS75LX operates from a low 1.7- to 3.7-V supply voltage. It can be connected directly to low-voltage supply rails. The temperature sensor is factory calibrated to an accuracy of $\pm 2^{\circ}\text{C}$ (max)

over its entire supply-voltage range and a wide temperature range of -25° to 100°C . It is $\pm 3^{\circ}\text{C}$ (max) accurate over its full operating temperature range of -55° to 125°C . Temperature is reported as user-selectable 9-, 10-,

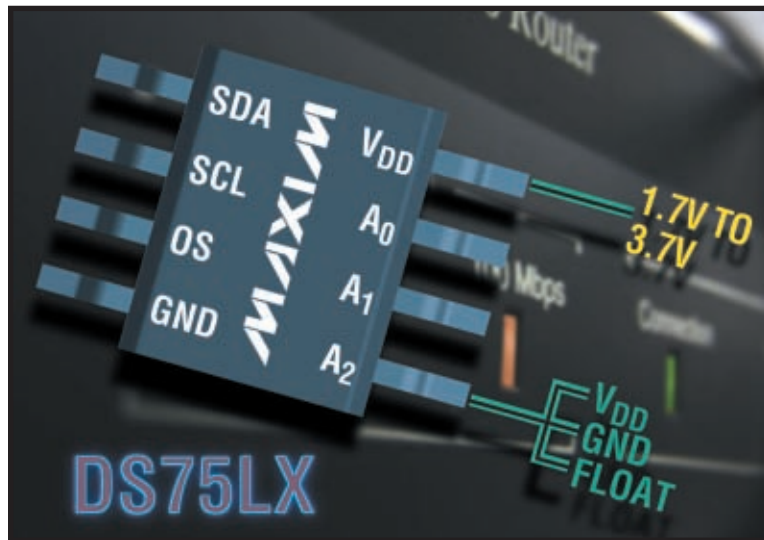
11-, or 12-bit digital temperature readings that translate to resolutions between 0.5° (9 bit) and 0.0625°C (12 bit). The DS75LX also has thermostat functionality with user-defined trip points that allow the user to

customize the threshold temperature and output hysteresis.

Additionally, this device includes an optional bus-timeout function that releases the data line if it is held longer than a pre-determined amount of time. This prevents any possible lockup problems that might occur on the two-wire bus.

Prices for the device start at \$0.75.

Maxim Integrated Products
www.maxim-ic.com



PCB-POOL[®]

SERVICING YOUR COMPLETE PROTOTYPE NEEDS

Price Example: 16 Sq-Inches
(double sided pth)

2 Days: \$ 90.00

8 Days: \$ 22.50

Standard PCB-Pool Service

SIMPLY SEND YOUR FILES AND ORDER ONLINE!

New Service:

WATCH "ur" PCB[®]

Save vital time on design errors in advance of receiving your Prototype. View high resolution photographic images of your PCB during each production stage. Be one step ahead, use our realtime PCB monitoring service.

WWW.PCB-POOL.COM

DOWNLOAD OUR FREE PCB SOFTWARE
www.free-pcb-software.com



Tollfree USA : 1877 390 8541
sales@beta-layout.com



NEW PRODUCT NEWS

Visit www.circuitcellar.com/npn for more New Product News.

UPGRADED FLOATING-POINT COPROCESSOR

The **uM-FPU V3.1** is a new version of the popular uM-FPU floating-point coprocessor chip. This version extends the powerful feature set of the original uM-FPU V3 chip to include serial I/O support, NMEA sentence parsing, block transfers, additional matrix operations and string support, and many other enhancements.

The new serial I/O capabilities with NMEA sentence parsing make it easy to add GPS data to embedded system designs. GPS data can be read and processed directly by the uM-FPU V3.1 chip, saving I/O pins, memory space, and execution time on the microcontroller, which can be used for the main application.

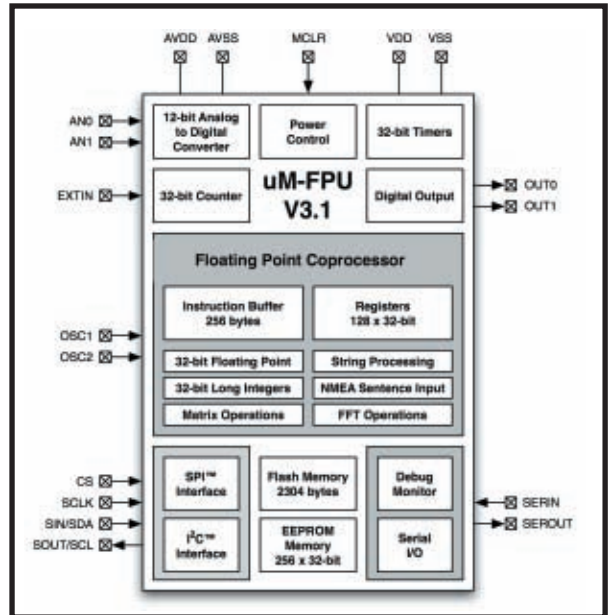
The chip interfaces to virtually any microcontroller using an SPI or I²C interface and supports 32-bit IEEE, 754-compatible floating-point, and 32-bit integer operations. The chip also provides two 12-bit A/D channels, two digital outputs, an external event counter, flash and EEPROM storage, and serial I/Os up to 115,200 bps.

The uM-FPU V3 IDE makes it easy to create, debug, and test floating-point code. The IDE code generator takes traditional math expressions and automatically produces uM-FPU V3.1 code targeted for any one of the many microcontrollers and compilers supported. The IDE also supports code debugging and programming user-defined functions.

The chip operates from a 2.7-, 3.3-, or a 5-V supply with power-saving modes available. It is available in an 18-pin DIP, SOIC-18, or QFN-44 package. The chip costs \$19.95 per unit

with volume discounts available.

Micromega Corp.
www.micromegacorp.com



WARNING

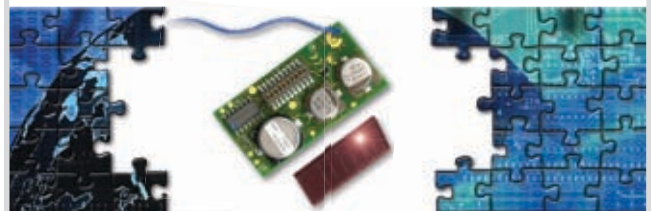
There are literally thousands of pages of innovative embedded systems applications crammed into one venue: www.circuitcellar.com. The expanded applications section we've created over the years includes source code, schematics, and design insight from your peers. It continues to grow by leaps and bounds, delivering an amazing online supplement to Circuit Cellar's print magazine.

It's all yours to view for free. Start with the latest project postings from the DesignStellaris2006 Contest. With so many applications from so many designers around the world, you're bound to find a ton of useful tips and techniques. Check back often for the latest additions.

Start at www.circuitcellar.com/contests/

Contents Under Pressure

Reliable, Easy-to-use Embedded RF Modules



and the **Wireless Expertise** to connect the pieces of your data system

| Featured RF Protocols | Range (power output) | RF Data Rate |
|---------------------------------|----------------------------|--------------------|
| Self-powered | up to 150' (5 mW) | 120,000 bps |
| ZigBee™ | up to 1 mile (100 mW EIRP) | 250,000 bps |
| Bluetooth® | up to 300' (100 mW) | 3 Mbps |
| WiFi® 802.11 | up to 1,000' (1 Watt) | 54 Mbps |
| 900 MHz ISM License-free | up to 40 miles (1 Watt) | 9600 - 115,200 bps |

✓ RoHS compliant

Development Kits In-stock!
Call & find which module best fits your application

AD HOC ELECTRONICS™



AdHoc101.com

Reliable RF Modules + Dedicated RF Expertise

(801) 225-2226



Handheld Multifunction Scope

If you need to monitor signals and inspect target devices, you can design your own LM3S811-based handheld multifunctional oscilloscope. This system supports several modes: Oscilloscope, Voltmeter, Ohmmeter, Capacitance Meter, Inductance Meter, Frequency Counter, Logic Probe, and Pulse Generator.

Many of our projects (e.g., robotic control applications) involve only audio or low-frequency analog signals, so we thought it would be handy to have a small pocket-sized oscilloscope for testing purposes. When we received an evaluation kit for the Luminary Micro DesignStellaris contest, we were happy to find that we could use the compact LM3S811 evaluation board to build a tiny multifunctional oscilloscope. The evaluation board had almost everything we needed. The only parts we had to add were the components for the analog signal conditioning circuit. In this article, we'll explain how we built our handheld system (see Photo 1).

The small LM3S811 microcontroller, at the center of the evaluation board, is a powerful 50-MHz 32-bit CPU, with 8 KB of RAM and 64 KB of flash memory. The microcontroller is fully loaded with on-chip peripherals that are suitable for building a variety of functions for mixed signal processing and control. The built-in peripherals include a 500-kps multichannel 10-bit ADC,

which can capture the input analog signal for oscilloscope display, a flexible analog comparator, three 32-bit (or dual 16-bit) general-purpose timers with input event capture, six channels of PWM output, a synchronous serial interface (SSI), which can work as a Freescale SPI, National Semiconductor MICROWIRE, or Texas Instruments synchronous serial interface protocols,

an I²C, two channels of UART, and plenty of general-purpose I/O ports (GPIO). There is a small OLED display on the evaluation board. This 1" bright yellowish panel is a fine-pitch 96 × 16 graphic display connected to the LM3S811 through the I²C interface. Unlike most LCDs, the OLED display has a very fast response time, which is ideal for displaying rapidly changing oscilloscope waveforms. A USB controller is also available on the board for JTAG control and serial port emulation. The USB serial port emulation connection can be used to deliver the captured waveform to a host PC and show the waveform on a larger, higher-resolution PC screen.

Finally, we developed a small battery-powered multifunction device with the following functions: A dual-channel oscilloscope supports 250 kps for each channel. Each channel has seven gain settings, which allow the full-scale display to be independently set to ±10 V, ±5 V, ±2 V, ±1 V, ±500 mV, ±200 mV, or ±100 mV. The trigger method includes Free-Running (Auto) mode,

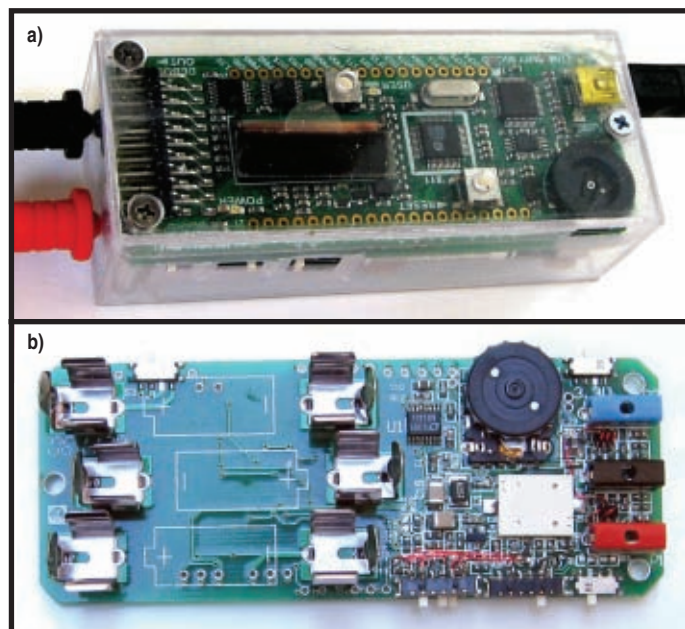


Photo 1a—The handheld multifunction oscilloscope is housed in a small plastic box. The front view of the device is Luminary Micro's LM3S811 evaluation board. The dual-channel input ports are at one end of the device, while the USB port is located at the other end. Slide switches for mode selection are visible at one side of the device. **b**—This is the PCB for the analog signal-conditioning circuit and batteries. The PCB and the LM3S811 evaluation board are stacked together back-to-back on their solder sides.

Channel 1 Edge-triggered (\pm Edge) mode, and Single-Shot mode. An AC/DC voltmeter can autorange from 100 mV to 10 V. The DC polarity and AC can be automatically detected and indicated in the display. An ohmmeter autoranges from 0 to 1.36 M Ω , with audible sound for connectivity. A capacitance meter is autoranged to measure the capacitances from 1 pF to 100 μ F. An inductance meter measures inductors in the millihenries to henries range. A frequency counter counts the frequency of low-level signals with a sensitivity of 150 mV. A logic probe senses logic high, logic low, open circuit (tri-state), and pulse. A pulse generator generates the pulse repeat frequencies from 5 Hz to 5 MHz and duty cycles from 1% to 99%. The generator also supports single-pulse outputs.

CIRCUIT DESCRIPTION

Figure 1 is a block diagram of the system. The LM3S811's rich on-chip peripherals and plentiful GPIO pins make it possible to build the multifunction tool with few external components. The analog components for the signal conditioning circuit and the batteries are mounted on a small PCB. The LM3S811 evaluation board and the analog PCB are stacked back-to-

back on their solder sides. Figure 2 and Figure 3 show the analog signal conditioning circuit and the connection circuit to the LM3S811 evaluation board, respectively.

Oscilloscope mode and Voltmeter mode share the same signal conditioning circuit (see Figure 4). In these modes, the LM3S811's GPIO PD3 pin is set to a low logic level. Thus, at the channel 1 input circuit the end of the 300-k Ω resistor R13 connection to the GPIO PD3 is virtually grounded. This makes the two 300-k Ω resistors parallel to form a 150-k Ω resistor in the attenuation network. With the 850-k Ω resistor in the input end, both channels' input impedance is 1 M Ω and the attenuation rate is 15%.

The input buffers U2 and U4 are the voltage followers that isolate the programmable gain amplifier (PGA) from the input end. (The PGA has a low-

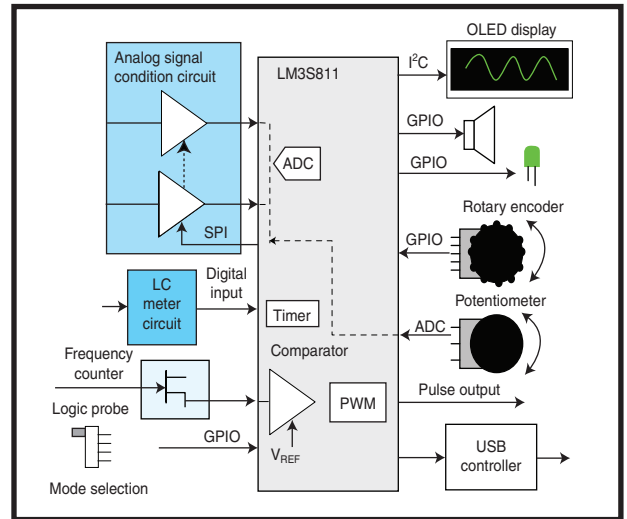


Figure 1—This is a block diagram of the multifunction oscilloscope. The LM3S811 evaluation board is in the center. The analog signal-conditioning circuit, LC meter circuit, frequency counter, and logic probe circuit were added to the system. The slide switches, the potentiometer, and the rotary encoder provide user control to the system. The visual output is the OLED display, which is connected to the microcontroller through the I²C port and the LED. The audio output is the tiny piezo speaker. The UART port provides a serial port link through the USB port to a host PC.

input impedance.) The diode pairs at each channel input (D3 and D4, D5 and D6) provide the protection circuit to prevent over-voltage damage to the amplifiers. Switches S7 and S8 are for the DC/AC input coupling selection. When the switch is open, a 0.1- μ F capacitor is used for the AC coupling (notice that the capacitor voltage tol-

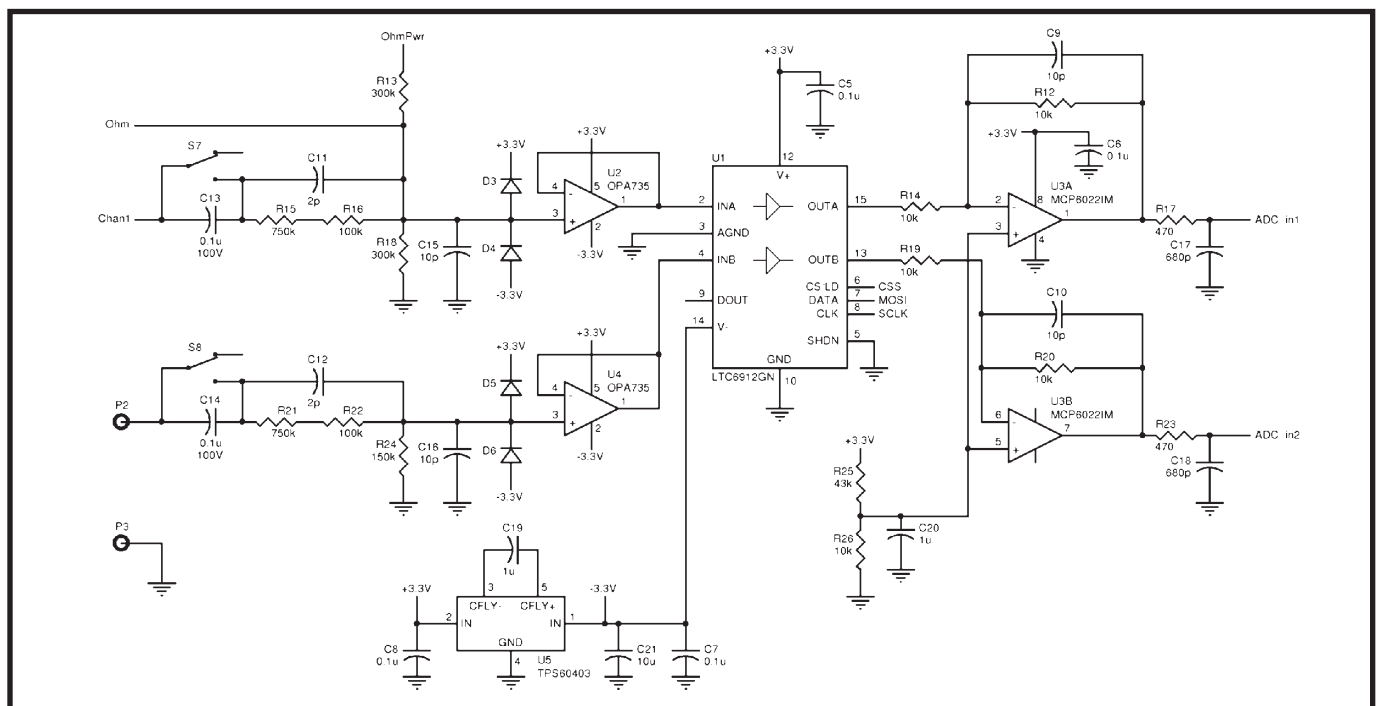


Figure 2—This is a schematic of the analog signal-conditioning circuit and the negative power supply.

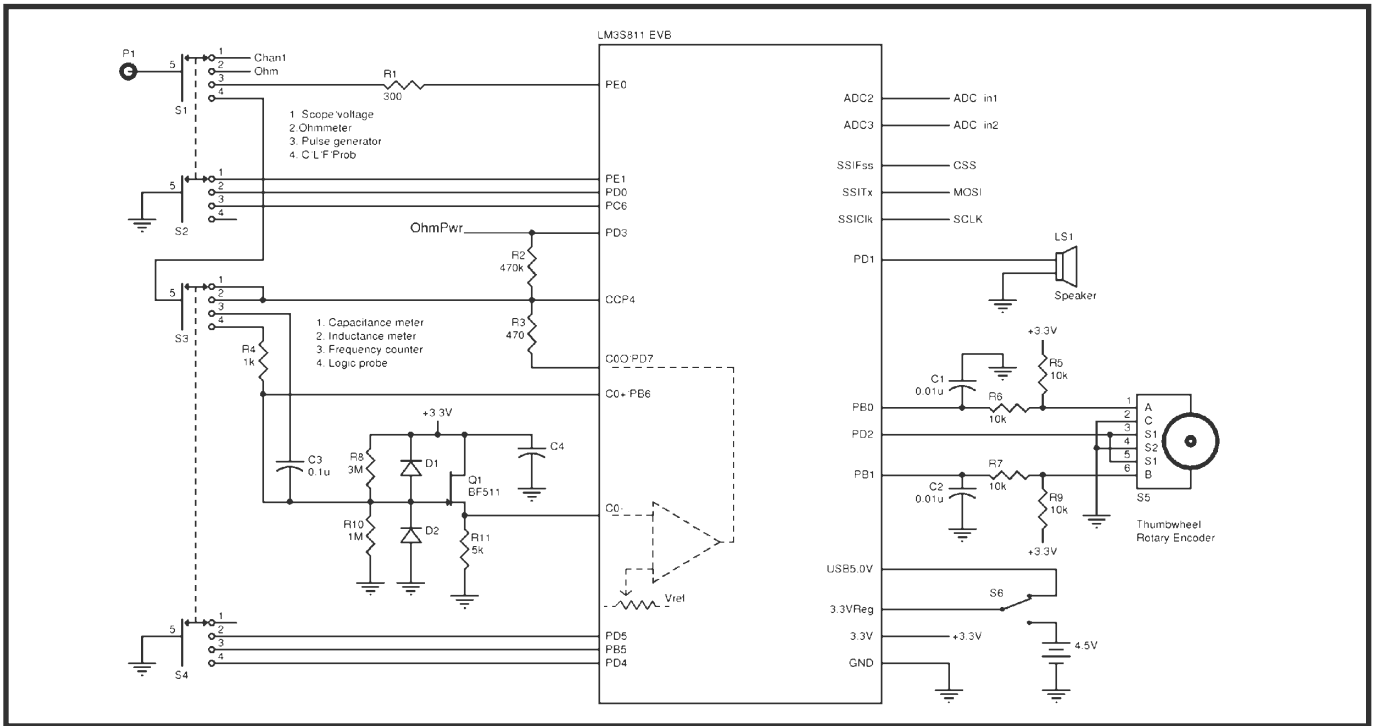


Figure 3—This is a schematic of the LM3S811 evaluation board's connections, mode selection switches, the rotary encoder and the circuit for the capacitance meter, the inductance meter, the frequency counter, and the logic probe.

erance is 100 V). The Linear Technology LT6912-1 is a dual-channel PGA where each amplifier gain can be independently set to $\times 1$, $\times 2$, $\times 5$, $\times 10$, $\times 20$, $\times 50$, and $\times 100$ via an SPI. The charge pump (U5, a Texas Instruments TPS60403) provides the -3 -V power supply for the input buffers and the PGA. The last stage in the signal conditioning circuit is a voltage shifter. Because the PGA and the input buffer are working on both positive and negative rails to accommodate the signal in negative range while the LM3S811's ADC is working only on the positive rail, a voltage shifter is needed to shift

the signal from ± 1.5 V to the 0- to 3-V range. The voltage level shifter has a unit gain for the negative input while its gain for the positive input is 2. The resistor divider made by R25 and R26 brings a static 0.75 V to the positive input. When the input is 0 V the resting output voltage to the ADC converter is shifted to 1.5 V, which is in the middle of ADC working range. The negative signal moves the output toward ground while the positive signal moves the output toward the positive rail. The low-pass filters, composed of R17, C17, R23, and C18, at the output of each voltage shifter are the antialiasing filters for filtering out the frequency components higher than the Nyquist frequency.

The fast on-chip ADC captures and digitizes the buffered dual-channel analog signal. The sampling rate is controlled by an on-chip timer. Thanks to the programmable sample conversion sequence units embedded in the LM3S811, sampled data is automatically saved in the sequence unit's FIFO. This prevents the data from being lost in case the interrupt service routine (ISR) cannot catch up with bursting interrupt events at fast sampling rates. The ADC also samples the voltage output of the potentiometer on the evaluation board. This volt value is used to control the oscilloscope trigger level as well as to adjust the frequency and the duty cycle of the pulse generator.

Voltmeter mode uses the same circuitry as Oscilloscope mode, except only channel 1 is used. The digitized signal at the ADC output is analyzed in order to discriminate between DC and AC voltages. If the signal contains both positive and negative components, the measurement is an AC voltage and the AC flag is indicated on the display. The AC voltage is rectified by software and the average value is used for the display.

Ohmmeter mode also shares the same basic signal conditioning circuit as the Oscilloscope and Voltmeter modes. However, GPIO pin PD3 in this mode is set to V_{CC} rail. Two 300-k Ω resistors form a voltage divider. The measuring resistor R_{TEST} is parallel with the low resistor of the voltage

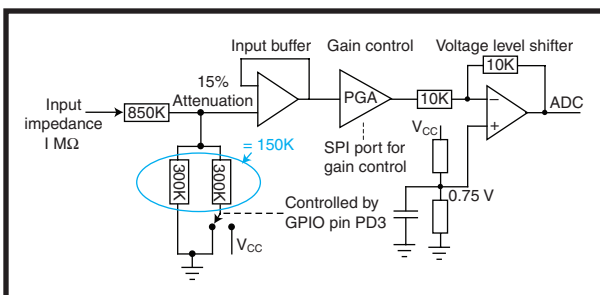


Figure 4—This is the analog signal-conditioning circuit for the oscilloscope and voltmeter. The input resistor divider attenuates the input signal to 15%. The input buffer provides an isolation to the input. The gain of the PGA is controlled by the microcontroller through the SPI port. The last stage is the voltage shifter to translate the voltage from the analog domain to the ADC input voltage range.

Put **YOUR** Creativity to the Test

Circuit Cellar magazine and Microchip are pleased to bring you the Microchip 16-bit Embedded Control Design Contest.



This is the perfect opportunity to learn about Microchip's 16-bit Microcontroller and Digital Signal Controller (DSC) families. These powerful workhorses offer a broad range of products while preserving the compatibility that helps save you time and money on subsequent designs. Put your creativity to the test. Your project could win you a share of \$15,000 in cash prizes and give you the international recognition you deserve.

Learn about unique contest sample opportunities, special offers from Microchip, and how you can be the next Circuit Cellar design contest success story.

For details, visit www.circuitcellar.com/microchip2007

CIRCUIT CELLAR®



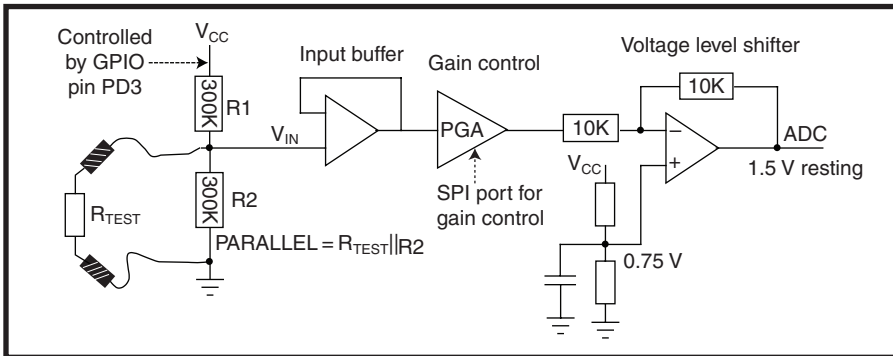


Figure 5—Ohmmeter mode shares the same analog circuit with the oscilloscope and the voltmeter, but with a different input resistor network. One of the 300-k Ω resistors is now connected to V_{CC} rail. The resistor to be tested (R_{TEST}) is parallel with another 300-k Ω resistor. When the input is open, the circuit outputs the highest voltage. The smaller the tested resistor, the lower the output voltage. By calculating the captured voltage, the resistor value can be easily determined.

divider (see Figure 5).

When the input end is open (the measured resistance equals ∞), the voltage at the voltage divider is about 1.65 V, which will saturate the voltage shifter output to the maximum. When R_{TEST} is connected to the input end, the voltage at the voltage divider drops as follows:

$$V_{IN} = V_{CC} \times \frac{R_{PARALLEL}}{(R_1 + R_{PARALLEL})} \quad [1]$$

Rearranging the terms, we get:

$$R_{PARALLEL} = \frac{V_{IN} R_1}{(V_{CC} - V_{IN})} \quad [2]$$

Because:

$$R_{PARALLEL} = \frac{R_{TEST} R_2}{(R_{TEST} + R_2)} \quad [3]$$

Rearrange the terms again to get:

$$R_{TEST} = \frac{V_{IN} R_1 R_2}{V_{CC} R_2 - V_{IN} (R_1 + R_2)} \quad [4]$$

Because the ADC sample is proportional to V_{IN} by a constant factor A , which relates to the gain of the PGA, we then get:

$$R_{TEST} = \frac{\text{Sample}_{ADC} A R_1 R_2}{V_{CC} R_2 - \text{Sample}_{ADC} A (R_1 + R_2)} \quad [5]$$

Therefore, by substituting in the actual values, we can calculate the R_{TEST} value by sampling the ADC input.

The PGA gain changes according to the resistor value being measured to make sure the resistor is measured at its highest resolution range.

Capacitance Meter mode uses a very simple RC circuit connected to the on-chip timer (timer 2A) input (see Figure 6). The C_{TEST} is charged and dis-

charged periodically through a resistor (the resistor value determines the testing range) by turning a driving voltage on and off. The voltage on the capacitor, V_C , cannot change instantly, but it is a function of time with time constant $\tau = RC$. During discharge:

$$V_C = V_0 e^{-\frac{t}{\tau}} \quad [6]$$

while during charge:

$$V_C = V_{DRIVE} \left(1 - e^{-\frac{t}{\tau}} \right) \quad [7]$$

where V_0 is the capacitor at the start of discharge and V_{DRIVE} is the driving voltage. At the time the LM3S811 changes the driving voltage, it also starts timer 2A. When the V_C reaches the threshold of V_{IL} (during discharge) or V_{IH} (during charge) of the LM3S811 GPIO input pin, the GPIO pin state is changed. Because this pin is the input of timer 2A, which is set to Input Edge Time Capture mode, the tick count of the time period is captured and stored in a register. Based on the captured tick count, the capacitance can be easily calculated.

There are two charge/discharge resistors. One resistor is smaller (470 Ω) and another is 1,000 times larger (470 k Ω). The large resistor is used for measuring small capacitors (from few pF to about 0.1 μ F). Because the larger resistor can compensate for small capacitance and make the RC time constant τ larger,

the timer is able to capture an adequate tick count during the charge/discharge period for calculating the capacitance value with sufficient resolution. When measuring large capacitors, a smaller resistor is used to prevent the large time constant from saturating the 16-bit timer or prolonging the measuring period. The resistors are automatically switched by the software. The system initially uses the large resistor to measure the capacitor. If the timer is not triggered within the threshold time window, the software switches to the small resistor to complete the measurement.

Inductance Meter mode uses the same principle as Capacitance Meter mode. However, the RL circuit time constant is $\tau = L/R$. Because R is the denominator, it requires a small resistor to measure a small inductor. The small resistor in the LR circuit causes a large current, which may exceed the maximal allowance of the GPIO pin. Therefore, the inductance meter is

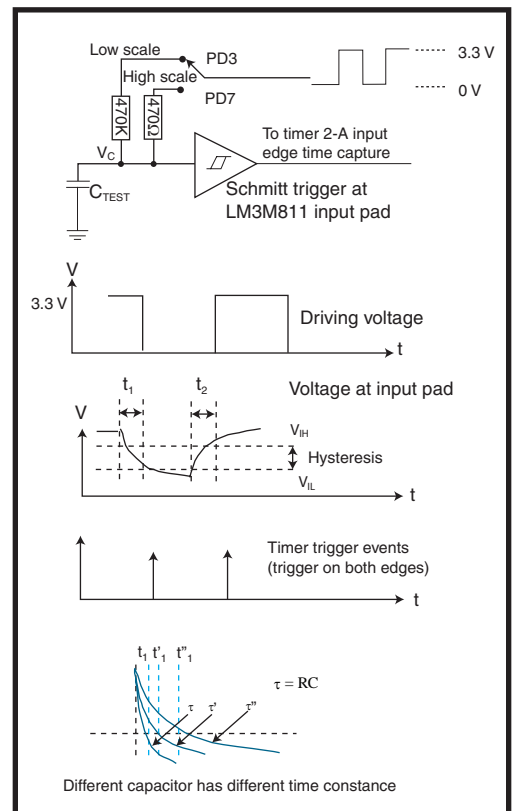


Figure 6—The capacitor to be tested (C_{TEST}) is charged or discharged through the 470- Ω or 470-k Ω resistor. When the voltage runs across the GPIO input Schmitt trigger threshold, the timer is triggered, and the tick count representing the charge/discharge period is captured. The capacitance can be deduced from the charge/discharge period, which is proportional to the RC time constant.

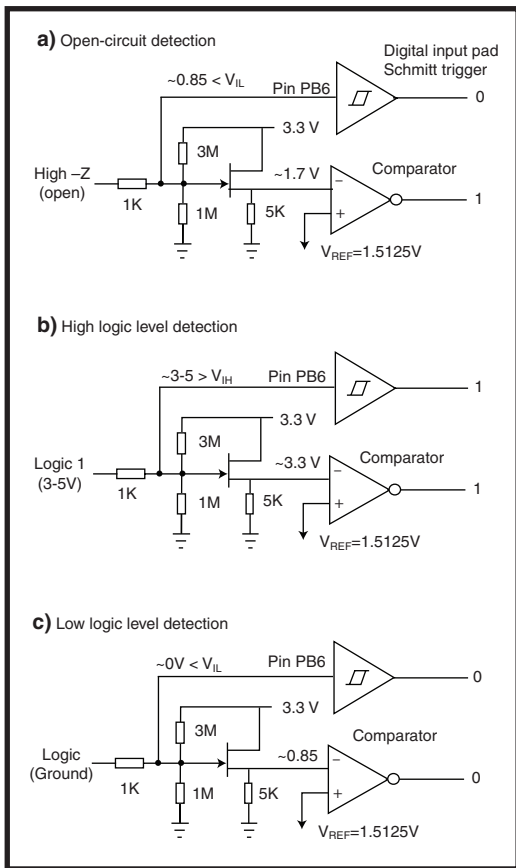


Figure 7—This figure shows the logic probe circuit detecting different input states. **a**—When the input is high-impedance or open-circuit, the comparator outputs 1 while the PB6 input is 0. **b**—When the input is in high logic level, both the comparator output and the PB6 input are 1. **c**—When the input is in low logic level or grounded, both the comparator output and the PB6 input are 0.

detect the open circuit, both the analog comparator output and the pin, PB6 events are observed.

Figure 7a shows high-impedance open circuit detection. Assume the JFET gate input impedance is large enough to ignore. The voltage at the PB6 digital input pad Schmitt trigger is determined solely by the voltage divider constructed by the 3-M Ω and 1-M Ω resistors (R8 and R10 in the schematics) when the input end is open. The voltage present at PB6 input is about:

$$3.3 \text{ V} \times \frac{1}{(1 + 3)} = 0.85 \text{ V} \quad [8]$$

It shares the JFET voltage follower circuit and the on-chip analog comparator with Frequency Counter mode, but with a different reference voltage for the comparator. In addition, the input is connected to another GPIO input pin (PB6). It is straightforward for logic low, logic high, and pulse detection. However, it is a little bit tricky to detect the high-impedance open-circuit condition (tristate). In order to

which is lower than the LM3811 input pin logic 0 threshold (maximal $H_{IL} = 1.3 \text{ V}$). Then, the microcontroller reads 0 from pin PB6. The output at the JFET source is about 1.7 V due to the gate-source voltage. The comparator reference voltage is set to 1.5 V, just below the JFET source voltage of 1.7 V. The comparator output is set to Inverse mode; therefore, the comparator outputs logic 1. Therefore, the software can determine the high imped-

limited to large inductance measurement.

Frequency Counter mode uses the on-chip analog comparator to convert signal waveforms to square waveforms. The capacitor C3 is for AC coupling of the input. A JFET voltage follower, Q1, is used to buffer the input signal. The resistors R8 and R10 are the bias resistors for the JFET gate. The source voltage of JFET is set to about 1.7 V. The positive input of the analog comparator is connected to the on-chip programmable band-gap voltage source and the reference voltage is set to 150 mV above the JFET source voltage. So, a small input signal (greater than 150 mV) can trigger the analog comparator, generating square wave outputs. The diodes D1 and D2 are for over-voltage protection. The comparator output is connected to the Timer 2A input through the small resistor R3. The timer is set to Input Edge Time Capture mode so the tick count between the rising edges of the square wave (representing the signal cycles period) are captured and then converted to frequency measurement by the software.

Logic Probe mode is shown in Figure 7.



Introduces iLoad

Capacitive Load Sensors with True USB connectivity



*Plug and Sense Simplicity
Digital Perfection*



- Integrated signal conditioning
- Rugged stainless steel construction
- Digital USB or Analog 0-5 V output
- Temperature compensated
- Accuracies - 0.25% to 0.025% of FS
- Easy mounting features built in

www.loadstarsensors.com
650.938.4282 | info@loadstarsensors.com

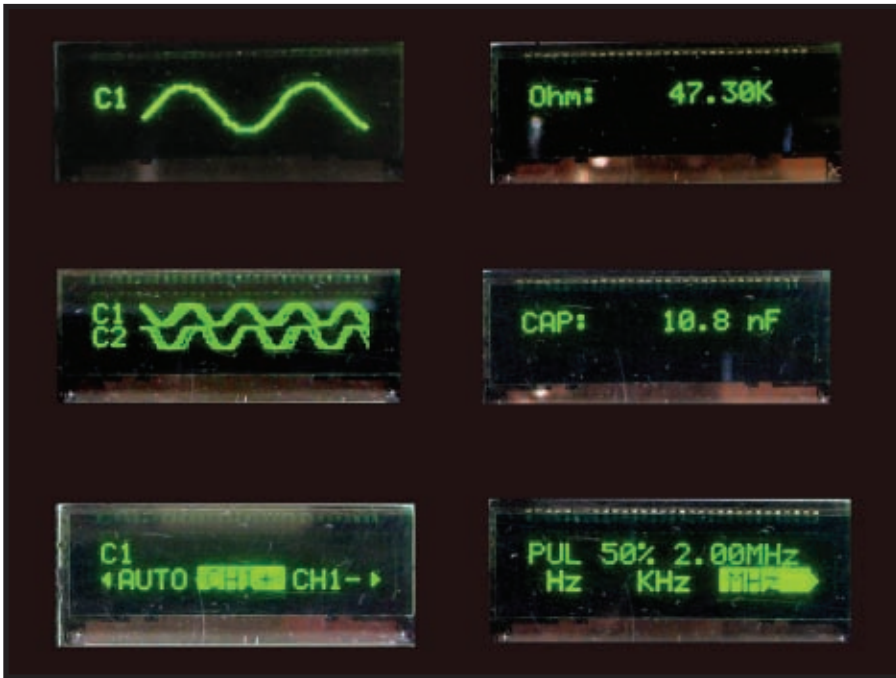


Photo 2—These are samples of the OLED display. The displays in the left column (from top to bottom) include a single-channel oscilloscope trace, a dual-channel oscilloscope trace, and a menu display showing the oscilloscope trigger selection. The displays on the right (from top to bottom) are an Ohmmeter mode display, a capacitance meter display, and a Pulse Generation mode display.

ance state if the PB6 input is 0 and the comparator output is 1.

When the input end is presented a logic-1 signal (3 to 5 V, the LM3S811 is 5-V tolerant), the voltage at the pin PB6 input is forced to rise to a high level. (R8 and R10 have a weak bias.) The JFET follower output also moves to a higher voltage level and the comparator output is maintained at a high level. The software can now determine the high-level logic signal if both the PB6 and comparator output are logic 1s (see Figure 7b).

If, on other hand, the logic 0 is present at input (or grounded), the voltage at pin PB6 input is forced to 0 (again due to the weak bias of R8 and R10). The JFET follower output drops to a lower voltage level (about 0.85 V). Because it is below the 1.7-V reference voltage, the analog comparator's inverted output is logic 0. That means the software can determine the low logic signal if both the PB6 and the comparator outputs are 0 (see Figure 7c).

The PB6 interrupt is enabled on both the rising and the falling edges. When a pulse is presented to the input, the GPIO interrupt service routine is called. Then the PB6 input

logic level and the analog comparator output are checked. If they are the same logic value, a pulse is detected, the on-board LED is turned on, and a beep tone is generated. The LED and the tone are not turned off until after a 500-ms delay. A software controlled time delay makes sure the visual and audio alerts can be easily noticed even for the short pulses.

Pulse Generator mode uses a PWM interface to output a stream of pulses. The PWM is set to Countdown mode. The PWM timer-load register determines the repeat frequency while the value in the timer comparator match register determines the duty cycle. The potentiometer is the user interface for setting the repeat frequency and the duty cycle. The voltage from the potentiometer is sampled by the on-chip ADC. The digitized value is used to set the PWM timer load register when the menu is in Frequency Setting mode and to set the PWM timer match register in Duty Cycle Setting mode.

Mode selection is made by two double pole 4 throw (DP4T) slide switches. We

could not find the miniature DP4T slide switch during hardware construction. Instead, we used two low-profile, miniature, single-pole 4-throw (SP4T) slide switches (Copal Electronics CUS-14B) stacked up to form a DP4T switch. Six GPIO pins on the LM3S811 are used for decoding the mode selection. The mode is selected so that when switches S1 and S2 are in positions one, two, and three, the device is in Oscilloscope/Voltmeter mode, Ohmmeter mode, and Pulse Generation mode, respectively. When switches S1 and S2 are placed in position four, the mode selection is determined by switches S3 and S4. The modes at positions one, two, three, and four of switches S3 and S4 correspond to Capacitance Meter mode, Inductance Meter mode, Frequency Counter mode, and Logic Probe mode, respectively. The edge-triggered interrupt and pull-up resistor on these GPIO pins' edges are enabled. Therefore, when the slide changes position, one of the GPIO pins is grounded and triggers the corresponding GPIO interrupt service routine. The software, after a debouncing delay (described later), terminates the previous mode and starts the new mode function.

The menu system consists of a thumbwheel rotary encoder (S5) and display menus. The rotary encoder is a quadrature encoder in which the outputs of two channels, channel A and channel B, are 90° out of phase. Between each rotating detent step, each channel outputs a complete pulse wave. Therefore, the combination of the two channels' outputs yields four

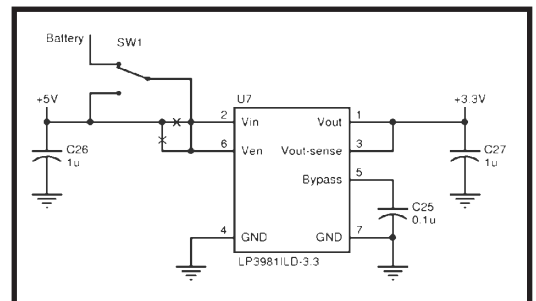





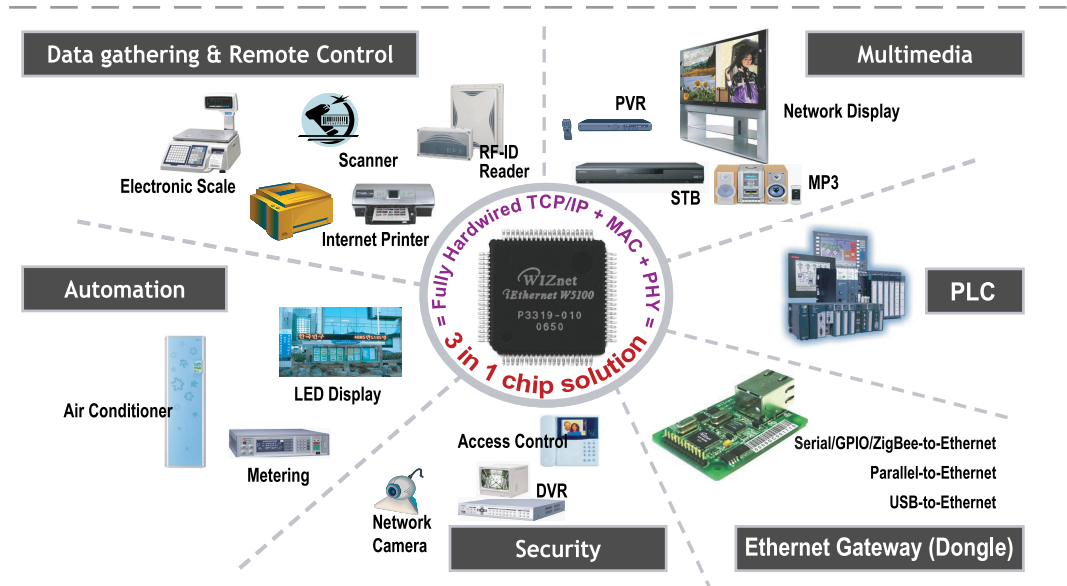
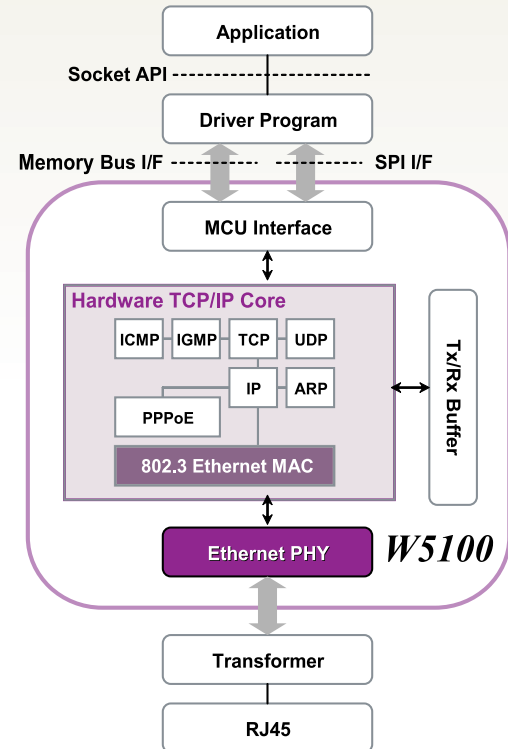
Figure 8—This is a modification of the power supply circuit for adding a battery source. You can see the power input (pin 2) and the enable input (pin 6) of the linear voltage regulator. U7 is disconnected from the 5-V source. These pins are reconnected to a switch, where either the battery or the USB 5-V source can be selected as the power source.

Born to *What a Simple* Internet!

WIZnet released new network modules with W5100

Anyone can implement Ethernet function by adding the WIZ module series without changing existing MCU of embedded system.

| | | |
|-----------------------------|--|--|
| <p>Serial-to-Ethernet</p> |  <p>WIZ 100SR</p> | <ul style="list-style-type: none"> - Support serial speed max. 230Kbps - 10/100Mbps Ethernet and auto-MDIX - Provide <i>*Configuration Tool</i> program to be installed in the PC (*With this program, users can configure all parameters of modules and control them through Ethernet.) - Plug-in type module having pin header (12pins) - Size : 50mmx30mmx12mm (W x L x H) |
| |  <p>WIZ 110SR</p> | <ul style="list-style-type: none"> - Spec is same with WIZ100SR - In addition, Include the DB9 and RJ45 I/F for user's convenience |
| <p>All-in-One Connector</p> |  <p>WIZ 800MJ (SMD Type)</p> | <ul style="list-style-type: none"> - One-stop solution for internet connectivity - Ethernet controller, Transformer, Mag Jack and TCP/IP stack are included - Support memory bus and SPI for MCU I/F |



different phases in sequence in between each detent position. These outputs are coded, instead of binary, in Gray code order where only one bit alters when the phase changes. For example, the 2-bit binary code sequence is 00, 01, 10, and 11. In Gray code, the sequence is 00, 01, 11, and 10. The next state of binary code 01 is 10, both bit 0 and bit 1 states change. In contrast, the next state of Gray code 01 is 11, only bit 1 state changes. Readers interested in the rotary encoder and Gray code may refer to “Using Rotary Encoders as Input Devices” (B. Millier, *Circuit Cellar* 152, 2003).

The outputs of the rotary encoder, with the mechanical switch debouncing RC networks (R6, C1, R7, and C2), are connected to two GPIO pins, PB0 and PB1, on the LM3S811. Edge-triggered interrupts on the GPIO pins are enabled. When the encoder rotates, the ISR of these GPIO pins is called. By checking the bit change in the Gray code, the software can easily detect the thumbnail rotary encoder rotating direction.

The menu items are rendered in the lower half of the OLED display screen. (See Photo 2 for the display example.) The menu display, when activated, can display three items, each with no more than four characters. If the menu list is longer than three, arrows are shown on the right and left ends of the menu display to indicate that more items are available. The currently selected item is highlighted with inverted background and foreground lighting. Rotating the encoder’s thumbwheel moves to the next or previous item, depending on the direction of rotation and the menu scrolls to the left or right to show the available items if necessary.

The menu is organized as a hierarchy. For example, if the thumbwheel’s rotary encoder push button or the push button at the front panel is pressed when the slide switch is in Oscilloscope/Voltmeter mode, the top-level menu is displayed where Oscillo-

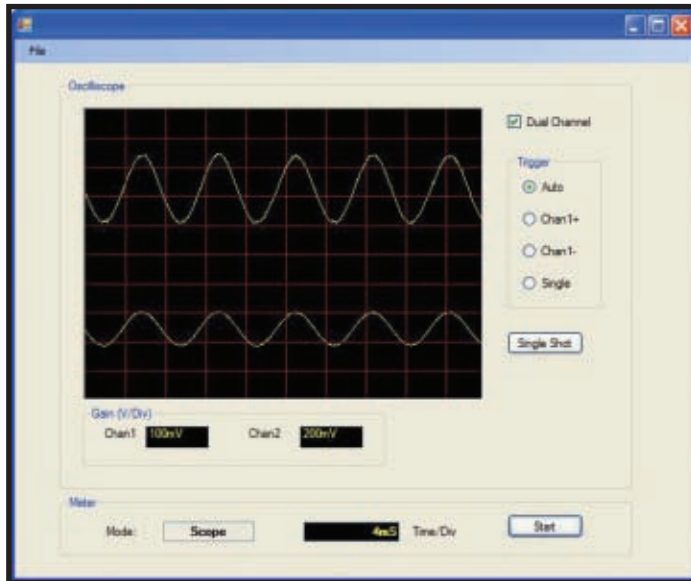


Photo 3—A screenshot of the oscilloscope trace is shown in the PC window. The data is delivered through the USB serial emulation port. All the parameters and settings, along with the oscilloscope on the PC screen, are updated automatically when the handheld multifunction device’s mode changes.

scope or Voltmeter mode can be selected. Highlighting Oscilloscope and pressing the thumbwheel push button brings up the Oscilloscope submenu, where single and dual-channel, gain setting, time scale, or trigger method can be selected. Selecting an item and pressing the thumbwheel will move deeper down the menu hierarchy, and so on, until a leaf item is selected.

Mechanical switch debouncing is implemented in software. There are many mechanical switches, including push buttons and slide switches, used in the device. To simplify the circuit, a software debouncing mechanism is used for all switches except rotary encoder quadrature switch output. The software uses the sysTick interrupt to handle the switch debounce delay. When a switch interrupt occurs, the ISR does not execute the command immediately. Instead, the ISR gets the current sysTick count and registers a time delay for the command. The sysTick interrupt occurs at 1-ms intervals. At each sysTick interrupt, the current tick count is compared with the registered delay tick count. If the registered delay is expired, the switch condition is reexamined. Only the delayed reexamined switch result is used to control the command execution.

Power supply modification for battery operation is made on the LM3S811 eval-

uation board (see Figure 8). It is powered by a USB 5-V supply. A 3.3-V linear regulator converts 5 V to 3.3 V for the LM3S811 (and the analog signal conditioning PCB). In order to allow the three AAA batteries to power the circuit when it is working in Standalone mode, the power supply circuit was modified. The PCB trace to the 3.3-V voltage regulator input is carefully cut. Note that the traces to voltage regulator pin 2 and pin 6 need to be cut separately because they are connected to the 5-V source at a different location. (We had to use a fine drill bit to carefully drill the via on the board to disconnect pin 2 from the back side of the 5-V source.) These two pins are resoldered together and connected to a power switch. The power switch is used to select either the USB 5-V source or the battery to serve the system.

SOFTWARE DEVELOPMENT

The software was developed using the Keil RealView microcontroller development kit. The Keil RealView is free for up to 16 KB of code. Unfortunately, the footprint of our project is about 18 KB, slightly larger than the free version’s limit.

The program is written in C and does not use any RTOS. It is an interrupt-driven system. A state machine is implemented in the main loop. The slide switch, push button, and thumbwheel rotary encoder as well as the on-chip peripheral interrupts, are triggered by the corresponding events. The ISR sets and resets the states of the state machine and the main loop executes consequent commands. Luminary Micro provided the Stellaris peripheral driver library, which is an abstract interface and hides the details of on-chip peripheral register manipulation. With the driver library, register file setting becomes easier and the source code is cleaner. Luminary Micro also includes the source code to the driver library. This enables you to look at the imple-

mentation or modify it if necessary.

In order to capture fast ADC samples, the system runs at its maximum rate of 50 MHz. The serial port data rate is set to 115.2 kbps. When the USB cable is connected to a host PC, the captured oscilloscope waveform can be sent to a PC through the USB serial connection emulation. A window display program written in C# on the .NET platform accepts and displays the waveform and the measurement data in other function modes on the PC display (see Photo 3).

IMPROVEMENTS

Luminary Micro's LM3S811 microcontroller is a nice device to work with. With its powerful ARM CPU core and rich peripherals, hardware design becomes easier. The Keil RealView is a solid integrated tool to develop and debug the embedded application. After a few months of enjoyable effort, we now have the multifunctional oscilloscope in our pocket. We are ready to probe target devices to find bugs.

But this is not the end of the story. The multifunctional oscilloscope could be

easily extended to include more functions, such as adding a function generator using the PWM output. The LM3S811 has room to accommodate such improvements. We encourage you to try building your own handheld device. ☒

Jingxi Zhang is the senior system architect at Jupiter Systems in Hayward, CA. He graduated from Zhongshang Medical University in Guangzhou, China. He has an M.S. in Electrical Engineering from the University of California, Irvine, and a Ph.D. in Neurosciences from the University of California, Los Angeles.

Yang Zhang is pursuing a Master's and Ph.D. in Computer Science at MIT. He has a B.S. in Computer Science from the University of California, Berkeley.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2007/205.

RESOURCES

Keil, "RealView Microcontroller Development Kit," www.keil.com/arm/rvmdkkit.asp.

Luminary Micro, Inc., "Stellaris Peripheral Driver Library," www.luminarymicro.com/documents/download/227/chk,666113193c85d62aefb475065abab521/.

SOURCES

LT6912-1 Dual-channel PGA
Linear Technology Corp.
www.linear.com

LM3S811 Microcontroller
Luminary Micro, Inc.
www.luminarymicro.com

CUS-14B Slide switches
Nidec Copal Electronics Corp.
www.copal-electronics.com

TPS60403 Charge pump voltage inverter
Texas Instruments, Inc.
www.ti.com

Wireless Control Application Kit

900 MHz and 2.4 GHz RF Control

- Low power LP3500 Single-Board Computer
- Two MaxStream® license-free RF Modules, FCC approved
- Software libraries and samples
- ModBus and PPP support for serving web pages

Complete Application Kit
\$599 2.4 GHz Version
\$499 900 MHz Version

RABBIT Semiconductor
Order Online At rabbitappkits.com

07070

ZigBee™ Certified Wireless Modules

XBee™ and XBee-PRO™

ZigBee™ Mesh Networks

- Range up to one mile
- Worldwide approved 2.4 GHz
- Modules start at \$19

Order your development kit online today!
from \$129

MaxStream
A Digi International® Brand

Toll-free 866-765-9885
www.zigbeemodule.com/cc

Electronic Bicycle Design

This electronic bicycle will work only as hard as you do. Motor power is controlled by the amount of torque you apply to the pedals: the harder you ride, the more torque is added to the rear wheel. Lithium-ion batteries and ultracapacitors power the system.

Have you ever had someone pass you on a bicycle while you were stuck in gridlock? If you have, you probably thought of how nice it would be to get more exercise, curse less, and help the environment. But, you may have assumed that you didn't have it in you to ride a bicycle all the way to work and back. After all, driving is more convenient. And that's not to mention the fact that you'd need a shower after each commute!

Electric vehicles are becoming a popular alternative to internal combustion engines because they are more efficient, require little maintenance, and don't produce pollution. The main disadvantage of electric motors is that battery technology needs to be improved to maintain the same range as combustion engines.

For our final project at Camosun College, we decided to design and build an electronic bicycle (see Photo 1). What is the point of a bicycle that powers itself? What exercise are you getting from that? What advantages are there over a regular bicycle?

Every motor needs some form of



Photo 2—This is the Crystalyte 408 brushless DC motor. The motor was ordered with the wheel and spokes already mounted. The axle was modified using a rotary tool in order to mount the four strain gauges.

throttle or torque control. In this project, motor power is controlled by the amount of torque you apply to the pedals. For every bit of torque you apply, the motor will add more torque to the rear wheel.

Think of it as having bionic legs. This bike makes you go really fast with the same effort. Hills will no longer be obstacles, and drivers will do double-takes while you ride past them with ease.

BRUSHLESS DC MOTOR CONTROL

We selected a Crystalyte 408 brushless DC three-phase hub motor for our design (see Photo 2). The 408 has eight poles and uses rare-earth alloy magnets made of neodymium, ferrite, and boron. According to Crystalyte's web site, the motor produces 473 W and 13.5 N-m of torque when drawing 15 A at 48 V. Because power is always directly proportional to fun, we applied a maximum of 20 A at 75 V.

Brushless motors have significant advantages over their brushed brethren: better efficiency, less maintenance, quieter operation, less emitted interference, and higher speeds. The major drawback of brushless motors is that their control systems are more complex and expensive.

Motor commutation is accomplished by energizing the internal coils in an appropriate sequence determined



Photo 1—This is our electronic bike. The ultracapacitor bank is mounted in a plexiglass case. The metal case in the middle of the bike holds the lithium-ion batteries. The red case on the handlebar holds the LCD and control buttons.

by the position of the rotor. Coils are energized two at a time to provide a current path through the motor. The position of the rotor is determined by three Hall-effect sensors embedded in the stator.

The Hall-effect sensors are spaced around the stator at 120° intervals. The sensors produce a high or a low for 180° of the electrical rotation, depending on the magnetic field of the rotor. As the rotor turns, the Hall-effect sensors will produce a 3-bit number from one to six, depending on the position of the magnets in the rotor. Using the position as an input, the software determines which coils to energize to make the motor turn.

POWER SOURCES

There are two power sources on the bike. The primary source is a bank of lithium-ion batteries. The secondary source is a bank of ultracapacitors.

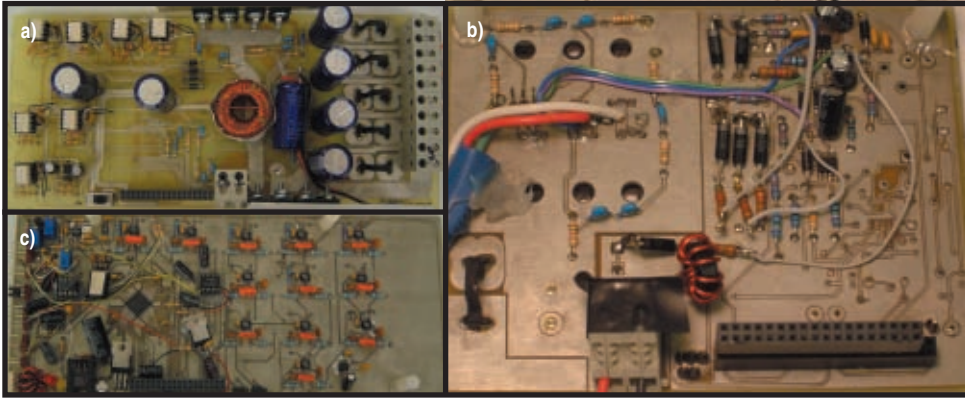


Photo 3a—A power source selection board was created for the MOSFETs that direct the power to and from the batteries and ultracapacitors. **b**—The drive board houses the H-bridge comprising the IR MOSFETs. Because of the high power supplied to the motor, external wire was used instead of PCB traces. **c**—The dsPIC 30F6015 and all A/D conditioning circuitry are located on the same board because of their low power characteristics and interference concerns.

There is, of course, an external power supply—the rider.

Lithium-ion batteries are ideal for this project because they have a high energy density, no memory effects, and minimal leakage current when they aren't being used. However, there are several inherent problems with lithium-ion batteries that must be dealt with for them

to function properly and safely.

In order to keep the bicycle and rider safe, the batteries should be mounted in an enclosure in case of a rupture. Also, protection circuits must be implemented so all currents and voltages stay within the specs. The battery packs also include internal overdraw, overcharge, and over-temperature shut-

downs.

In order to meet these specs and keep this project as simple as possible, we chose prepackaged Tenergy lithium-ion battery packs rated at 14.8 V and 2.2 Ah. We used 16 packs in a parallel arrangement with four series packs in each branch. This provided us with a nominal voltage of 59.2 V and 8.8 Ah, which is sufficient enough to power the bike for rides of over 1.5 h.

The ultracapacitors will initially power the bike when accelerating and absorb regeneration energy. They consist of 28 NessCap ESHR-0100-2R7 2.7-V, 100-F capacitors connected in series. This creates a 3.6-F, 75-V capacitor that can discharge 58.1 A over 1 s and 26 A over 5 s until they reach half of their maximum voltage.

HARDWARE

Three interconnected boards make up the bike's circuitry (see Photo 3).

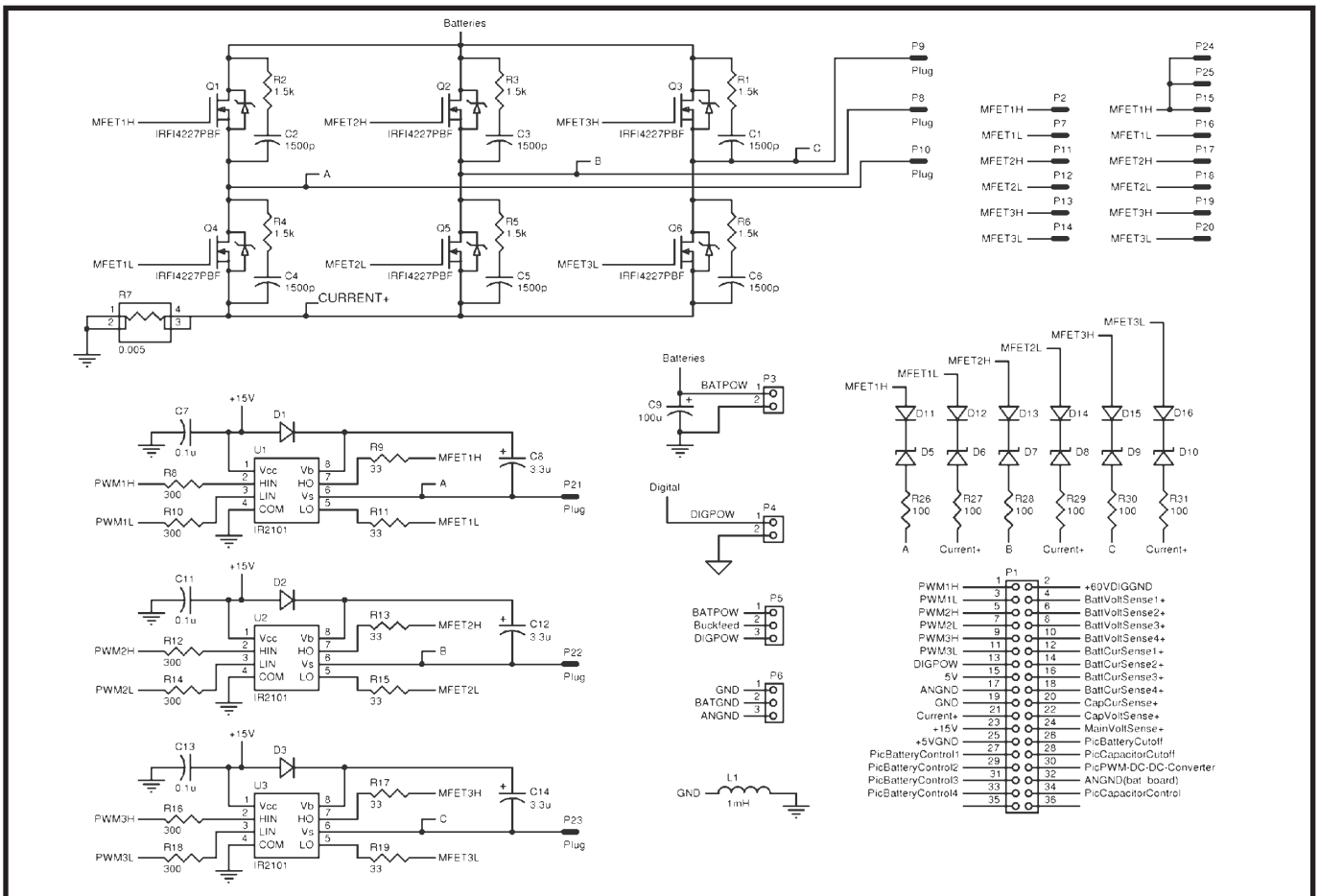


Figure 1—This is a three-phase H-bridge design using MOSFETs. Power to the motor is achieved by pulse-width modulating the low side of a single phase of the H-bridge.

Pololu

Robotics & Electronics



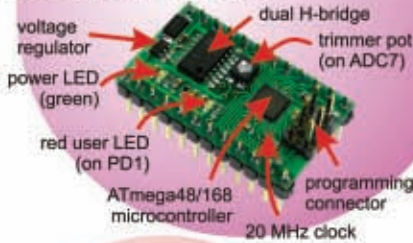
Robot Kits
Line followers
Robot arms
Hexapods
Chassis

Mechanical Components
Gearboxes, servos
Wheels, ball casters



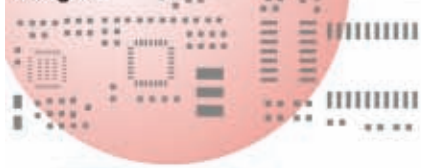
Motion Control
Servo controllers
Motor controllers

Robot Controllers



Solder Paste Stencils

Use our low-cost solder paste stencils to quickly assemble your surface-mount designs. **From \$25.**



Custom Laser Cutting

From \$35

Cut your own custom chassis, front panels, and more!

1-877-7-POLOLU
www.pololu.com

6000 S. Eastern Ave. 12D, Las Vegas, NV 89119

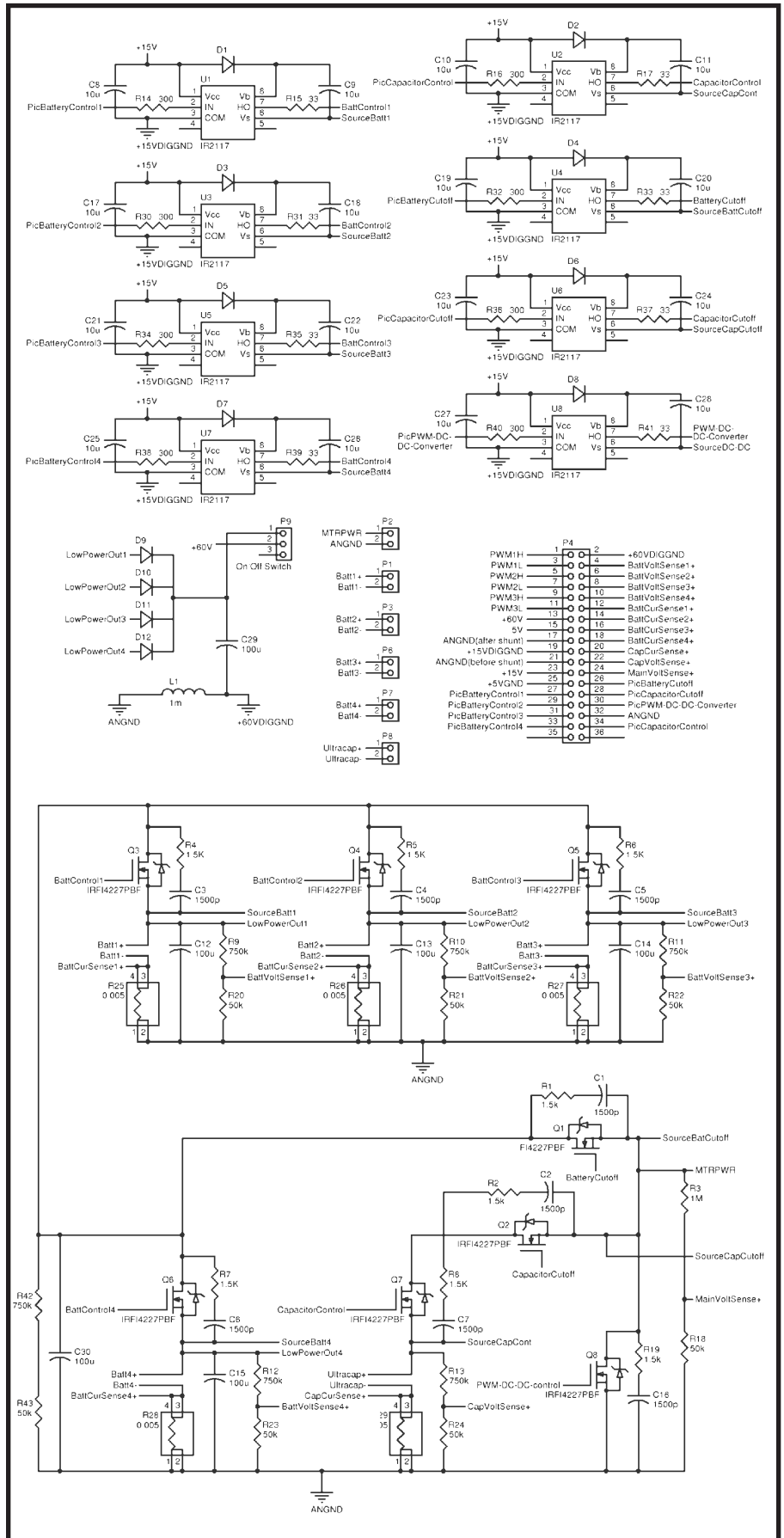


Figure 2—The power control board allows the bike to be driven from the batteries or the ultracapacitors. It implements the DC-DC converter using the internal inductance of the motor and controls the charging of the ultracapacitors.

Educators and Students:
Register Today

to Receive Exciting
Microcontroller Resources!

RENEASAS
UNIVERSITY
PROGRAM



Renesas — the #1 supplier of microcontrollers in the world — is launching Renesas University, an exciting educational program that gives educators a way to teach microcontroller (MCU) technology using a modern architecture and professional-grade tools. It also offers many valuable resources that help students learn about MCUs and how they can be applied in significant embedded system designs.

#1MCU
REACH
FURTHER

Renesas is a worldwide leader in:

- ▶ Microcontrollers
- ▶ Embedded flash microcontrollers
- ▶ MCUs in car navigation systems
- ▶ Power amplifiers for GSM phones
- ▶ LCD controllers for color mobile displays

The Renesas University program nurtures an online community where educators and students come together to share ideas, address technical issues and discuss microcontroller topics. It is characterized by:

Publish ▶ Renesas actively encourages academics and students to publish microcontroller-related papers. We provide assistance in publishing course material and microcontroller related books.

Toolchain ▶ The Renesas integrated development environment with toolchain is the commercial version of our development tools – with full C compiler, assembler, linker, and debugger. It is not a typical capability-reduced “educational” version. The only limitation is a 64KB code size after 60 days of use.

Modern ▶ Renesas microcontrollers utilize a modern architecture designed specifically for C and other high-level languages. Our devices handle the most demanding applications of today and tomorrow.

For more information on Renesas University and how to enroll, please visit www.renesasuniversity.com or email: University@rta.renesas.com



Complete Development Kits



Renesas Starter Kits provide a USB-powered, MCU-based system board with in-circuit debugger/flash memory programmer. They include a CD containing our integrated development environment with toolchain, plus documentation, example firmware and interesting projects.

▶ **Free for Educators:** Register at the Renesas University website to receive ten free Starter Kits per semester. In return, we request the submission of material that enriches Renesas University; i.e., code, student projects, technical papers, embedded control designs, etc.

▶ **Low cost for Students:** If actively enrolled in an educational institution, a Starter Kit can be purchased at a very low cost after registering at the Renesas University website.

The drive board controls the motor drive operation. The power control board controls the power source selection and regeneration. And, the processor board houses a microcontroller and ADC conditioning circuitry. All three boards are connected to each other in a stack using 36-pin headers.

Other hardware used on the project includes four strain gauges to measure pedal torque, a back-lit LCD with six control buttons, and a slide potentiometer for dynamic regenerative braking control.

DRIVE BOARD

One of the key components of our bike is the motor control board. It uses a simple H-bridge configuration with six International Rectifier IRFI4227PBF MOSFETs (see Figure 1). Using the three different channels of the H-bridge, we controlled the flow of

current through any coil of the motor.

Since we used MOSFETs, we needed to control the circuitry to turn on the appropriate channels at the proper time. This meant that drivers were needed to float the 5-V logic signal from the PIC up to 15 V above the source voltage. We chose International Rectifier IR2101SPBF high/low drivers because they require a very simple circuit with two external capacitors, one external diode, and current-limiting resistors on the inputs and outputs.

In real-world applications, protection is always necessary for unpredictable voltage spikes that occur when powering an inductive load. There are two different ways to fully protect the MOSFETs: gate-source voltage and drain-source voltage.

The IRFI4227PBF allows for a maximum gate-source voltage of 30 V. To protect against over-voltage conditions, we routed excessive current

away from the MOSFET. We did this by using 25-V Zener diodes and the appropriate current-limiting resistor, which connects the gate to the source. One issue that needs to be looked at when using the Zener diodes is Regeneration mode. If the motor is engaged in Regeneration mode under this setup, we would effectively destroy the high/low driver as well as the MOSFET. To prevent this, we placed a general-purpose diode in the opposite direction of the Zener diode. By placing the diode in the opposite direction, we blocked any current that might flow into the gate from the source during regeneration.

To protect against drain-source over-voltage, we needed a different kind of circuit than the gate-source protection. A simple snubber circuit was used to eliminate any possible transients. The snubber circuit consists of a series 1.5-k Ω resistor and a 1,500-pF

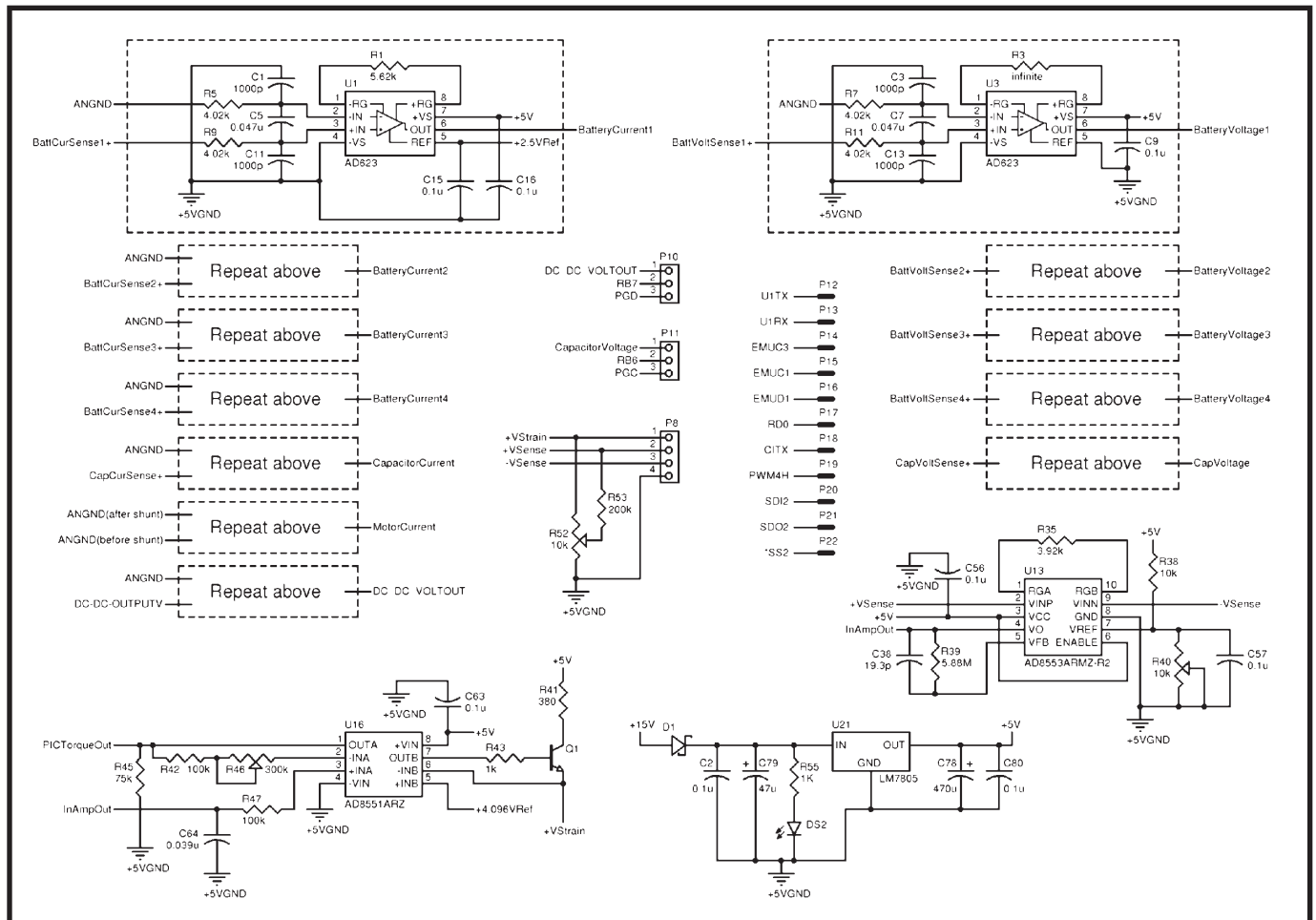
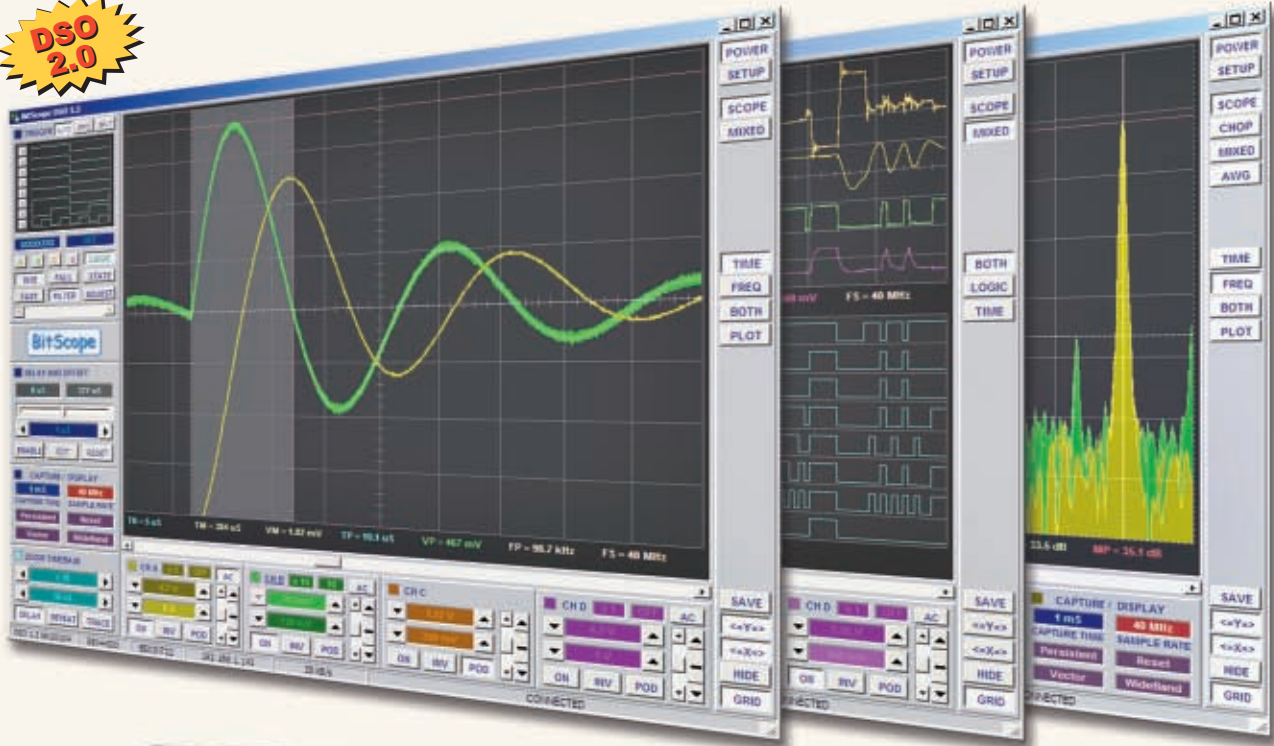


Figure 3—This is the ADC portion of the processor board. These are the conditioning circuits for quantizing the current and voltage levels of the power sources and the motor. The amplification for the strain gauges is also shown above.

BitScope PC Oscilloscopes & Analyzers

DSO Test Instrument Software for BitScope Mixed Signal Oscilloscopes

DSO 2.0



4 Channel BitScope



2 Channel BitScope



Pocket Analyzer

Digital Storage Oscilloscope

- ✓ Up to 4 analog channels using industry standard probes or POD connected analog inputs.

Mixed Signal Oscilloscope

- ✓ Capture and display up to 4 analog and 8 logic channels with sophisticated cross-triggers.

Spectrum Analyzer

- ✓ Integrated real-time spectrum analyzer for each analog channel with concurrent waveform display.

Logic Analyzer

- ✓ 8 logic, External Trigger and special purpose inputs to capture digital signals down to 25nS.

Data Recorder

- ✓ Record anything DSO can capture. Supports live data replay and display export.

Networking

- ✓ Flexible network connectivity supporting multi-scope operation, remote monitoring and data acquisition.

Data Export

- ✓ Export data with DSO using portable CSV files or use libraries to build custom BitScope solutions.

BitScope DSO Software for Windows and Linux

BitScope DSO is fast and intuitive multi-channel test and measurement software for your PC or notebook. Whether it's a digital scope, spectrum analyzer, mixed signal scope, logic analyzer, waveform generator or data recorder, BitScope DSO supports them all.

Capture deep buffer one-shots or display waveforms live just like an analog scope. Comprehensive test instrument integration means you can view the same data in different ways simultaneously at the click of a button.

DSO may even be used stand-alone to share data with colleagues, students or customers. Waveforms may be exported as portable image files or live captures replayed on other PCs as if a BitScope was locally connected.

BitScope DSO supports all current BitScope models, auto-configures when it connects and can manage multiple BitScopes concurrently. No manual setup is normally required. Data export is available for use with third party software tools and BitScope's networked data acquisition capabilities are fully supported.



www.bitscope.com

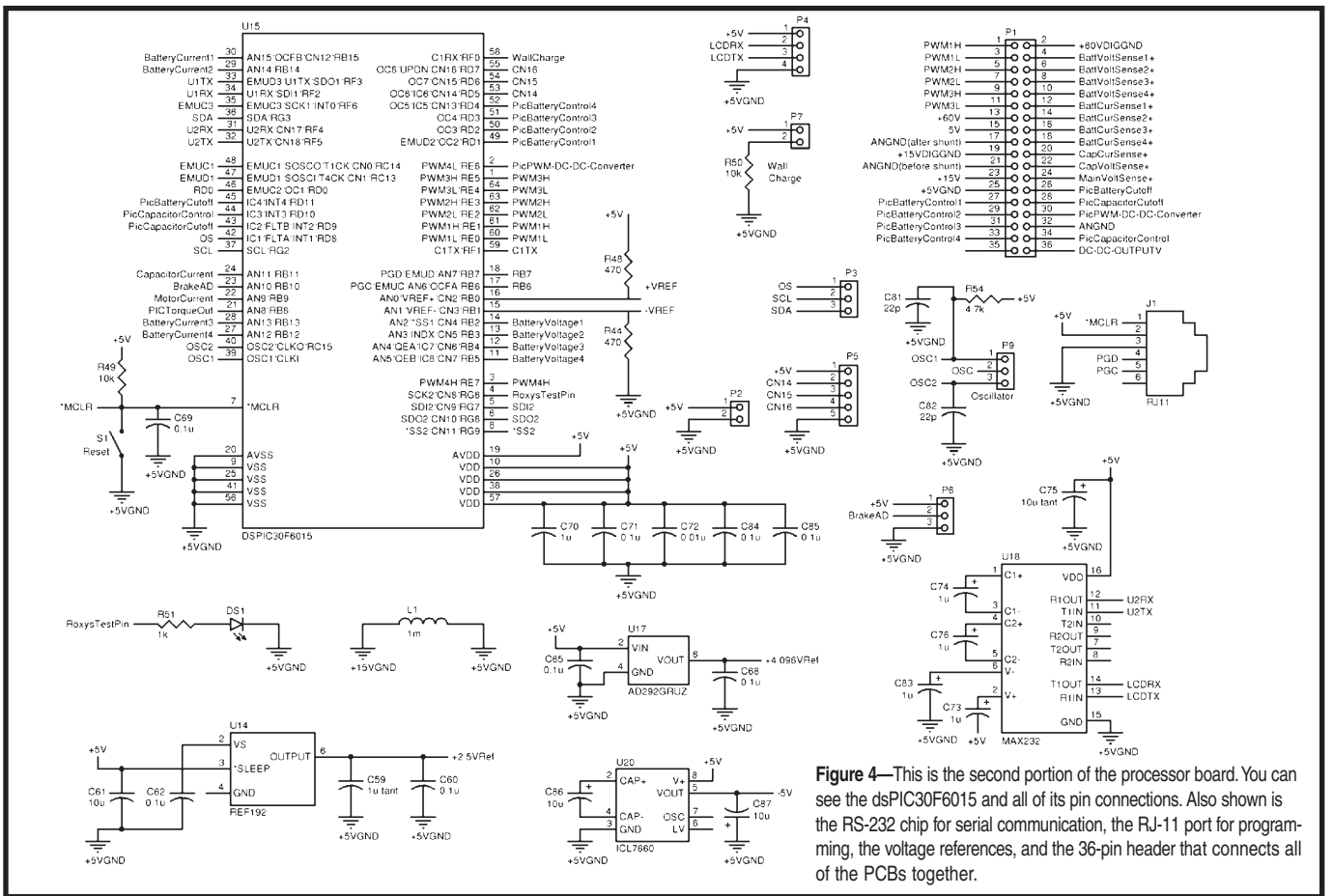


Figure 4—This is the second portion of the processor board. You can see the dsPIC30F6015 and all of its pin connections. Also shown is the RS-232 chip for serial communication, the RJ-11 port for programming, the voltage references, and the 36-pin header that connects all of the PCBs together.

Going Wireless is Easy!
 Wireless Mesh Networking

NEW! USB ZigBee™ Stick

Bluetooth with Dip-Switch options

Multi-Channel 900 MHz Wireless RF Modules

LEMOS INTERNATIONAL

Call Toll Free 1-866-345-3667 or email at sales@lemosint.com

www.lemosint.com

capacitor across the drain and source. Thermal protection is the final safeguard. To protect the MOSFETs from overheating, we placed a heatsink on the bottom of the PCB to dissipate any heat generated by the MOSFETs.

POWER CONTROL BOARD

The power control board has two main purposes. Synonymous with the name, the power control board houses the control hardware used for power source selection. The board also contains circuitry for regeneration, including a DC-DC converter that uses the internal inductance of the motor.

Switching between power sources is accomplished using IRFI4227PBF MOSFETs (see Figure 2). When using batteries, the SourceBattCutoff FET is on. When using ultracapacitors, the SourceCapCutoff is on. To prevent the power sources from shorting together, both control FETs can never be on at the same time.

A MOSFET on each parallel branch of batteries and ultracapacitors is used to direct current going into the power

supplies during regeneration. These can be seen as the battery and capacitor control FETs in Figure 2. Look at the direction of the internal diodes of the Cutoff FETs and Control FETs. This configuration enables us to direct current into and out of each branch of each power source.

The initial design of the system used the MOSFETs' internal diodes to prevent current flow from one parallel branch of the batteries to another. After further testing, additional diodes were needed in each parallel branch of batteries to prevent one battery pack from recharging another.

Because diodes were put in each battery branch, it limits the regenerative braking to just recharging the ultracapacitors. A future design will find a different means of applying the regeneration to the batteries.

In order to implement regenerative braking, we must increase the BEMF of the motor. This is necessary because the BEMF, even at high speeds, is still lower than the maximum voltages of the batteries and capacitors. To do this, we implemented a charge pump circuit, which works by pulse-width modulating a short across the motor. Regeneration will be discussed later in further detail.

To charge capacitors (and batteries in the future), we must be able to measure the voltage and the current at all times. Therefore, across every series battery pack and the ultracapacitor bank, there is a voltage divider circuit. For current measurements, we use a 5-m Ω shunt resistor in each battery branch and in the capacitor branch.

PROCESSOR BOARD

The processor board houses our Microchip Technology dsPIC30F6015 microcontroller. It has specific design features for motor-control applications. Features, such as 16 ADCs, 21 change notification pins, and four pairs of dedicated PWM pins simplified our design. The fact that we had already acquired a development platform using one of these chips was another nice incentive. The schematics for the processor board are shown in Figures 3 and 4.

The majority of the A/D circuitry is for monitoring voltages and currents.

The voltage of each of the four battery branches, the capacitor bank, and the line voltage in and out of the motor are all monitored for drive control, regeneration control, and shutdown control. The current into each of the four battery branches, the capacitor bank, and the motor are also monitored for safety and control.

For voltage and current monitoring, we used the Analog Devices AD623 instrumentation amplifier. The AD623 is powered by a single 5-V sup-

ply rail. For voltage monitoring, the AD623 is set to ground reference and unity gain. For current monitoring, the reference is set to 2.5 V and a gain of 18. This configuration allows us to monitor current during driving as well as regenerative braking.

In addition to voltage and current monitoring, the A/D conditioning circuits for the strain gauge force sensors are also mounted on the processor board. Because the differential output of the gauges is so small, a large gain is

\$51^{For 3} PCBs
FREE Layout Software!
FREE Schematic Software!

- 01 DOWNLOAD our free CAD software
- 02 DESIGN your two or four layer PC board
- 03 SEND us your design with just a click
- 04 RECEIVE top quality boards in just days

expresspcb.com

required to amplify the signal. Two stages of amplification are used: the first is through an Analog Devices AD8553 instrumentation amplifier, and the second is through an Analog Devices AD8552 op-amp. Both stages are powered with $\pm 5\text{-V}$ supplies. The op-amp uses a ground reference. The in-amp uses an adjustable reference down to -5 V .

The maximum change in voltage registered across the strain-gauge bridge is measured at 1.5 mV . If we have the maximum torque registering at 5 V into the

DSP, we will need a total gain of $3,333$. Because the CMRR of the in-amp is greater than the op-amp, the majority of the gain is left to the in-amp. This is done because when cascading amplifiers, the noise rejection of the first amplifier determines the total noise of the system. We set the gain of the AD8553 to $1,000$, and set the gain of the op-amp adjustable from one to four.

TORQUE SENSOR

Figure 5 shows the flow of forces

applied by the rider. Starting from the force the rider puts on the pedals, torque is transferred to the front sprocket and then to tension on the chain. The chain pulls on the rear gear cartridge that is threaded to the outer casing of the motor. The case of the motor has two ball-bearings that the rear axle is mounted through. The frame's vertical drop-outs anchor the ends of the axle. Overall, torque is applied to the rear axle, which is anchored at the ends and therefore unable to twist. Following the flow of forces, we see a very slight bending of the rear axle when a force is applied to the pedals.

In essence, we aren't measuring torque. We are measuring the deflection of the rear axle when a force is applied to the pedals. To measure this deflection, we used four thin-film strain gauges mounted on the rear axle (see Photo 4). Four gauges were used in a full-bridge configuration to magnify the voltage change and for temperature compensation. The tricky part was mounting the gauges on the axle. We used a rotary tool to grind and polish the vertical sides of the round axle flat. Thinning the axle also helps concentrate the deflection where the gauges are mounted.

FIRMWARE

So what happened behind the scenes? We wrote our firmware in C using Microchip's MPLAB with the student edition of its C30 compiler. To program the chip, we used a Microchip MPLAB ICD2. The handy device allowed us to tweak our program without removing the chip from its PCB.

The main task of the firmware is to measure the speed of travel (useful for calculating back electromotive force), handle user input (i.e., change the assist level), and generate PWMs for the motor and charging drivers. The firmware also monitors currents and voltages throughout the system to ensure everything stays within the regulated limits.

MAIN

To start things off, we implemented a `main()` function. Most importantly, our `main()` function reads

Fighting against your PCB-Design Software?

Here's something that will spare your time and your budget!

Boards designed under EAGLE are found in patient monitoring equipment, chip cards, electric razors, hearing aids, automobiles and industrial controllers. They are as small as a thumbnail or as large as a PC motherboard. They are developed in one-man businesses or in large industrial companies. EAGLE is being used in many of the top companies. The crucial reason for selecting EAGLE is not usually the very favorable price, but rather the ease of use. On top of that comes the outstanding level of support, which at CadSoft is always free of charge, and is available without restriction to every customer. These are the real cost killers!



EAGLE 4.1

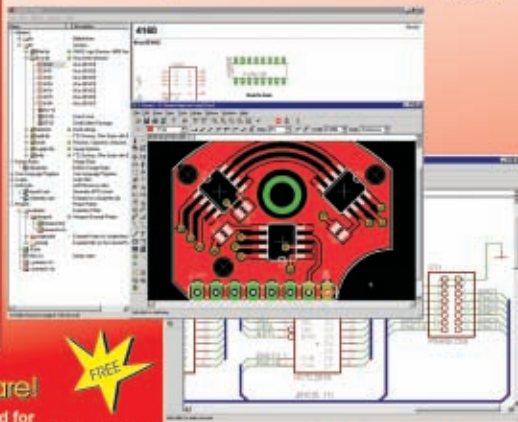
Schematic Capture • Board Layout
Autorouter

for Windows®
Linux®
Mac®

Now available for Mac!

Version 4.1 Highlights

- ▶ Powerful library management: e.g. move devices between libraries, base library for packages, generate package variants from other libraries.
- ▶ Dynamic ratsnest during routing process.
- ▶ Copy function in schematic.
- ▶ Rotate components in 0.1-degree steps.
- ▶ Blind & buried vias and pads with off-center drill.
- ▶ User-defined background color.
- ▶ Miter function for (rounded) tracks.
- ▶ Smash for groups.
- ▶ Measure distances between arbitrary points.
- ▶ Choose alternative raster on-the-fly with Alt-key.



FREE

EAGLE 4.1 Light is Freeware!

You can use EAGLE Light for testing and for non-commercial applications without charge. The Freeware Version is restricted to boards up to half Eurocard format, with a maximum of two signal layers and one schematic sheet. All other features correspond to those of the Professional Version. Download it from our Internet Site or order our free CD.

If you decide in favor of the Commercial Light Version, you also get the reference manual and a license for commercial applications. The Standard Version is suitable for boards in Eurocard format with up to 4 signal layers (max. 99 schematic sheets). The Professional Version has no such limitations.

<http://www.CadSoftUSA.com>
800-858-8355

| Prices | Light | Standard | Professional |
|---------------------------------|-------|----------|--------------|
| Layout | | 199\$ | 399\$ |
| Layout + Schematic | | 398\$ | 798\$ |
| Layout + Autorouter | | 398\$ | 798\$ |
| Layout + Schematic + Autorouter | 49\$ | 597\$ | 1197\$ |

Pay the difference for Upgrades

CadSoft Computer, Inc., 801 S. Federal Highway, Delray Beach, FL 33483
Hotline (561) 274-8355, Fax (561) 274-8218, E-Mail : info@cadsoftusa.com

© 2007 CadSoft Computer, Inc. All rights reserved. EAGLE is a registered trademark of CadSoft Computer, Inc. All other trademarks are the property of their respective owners.



R8C/Tiny Brings 16-bit Performance to 8-bit Applications

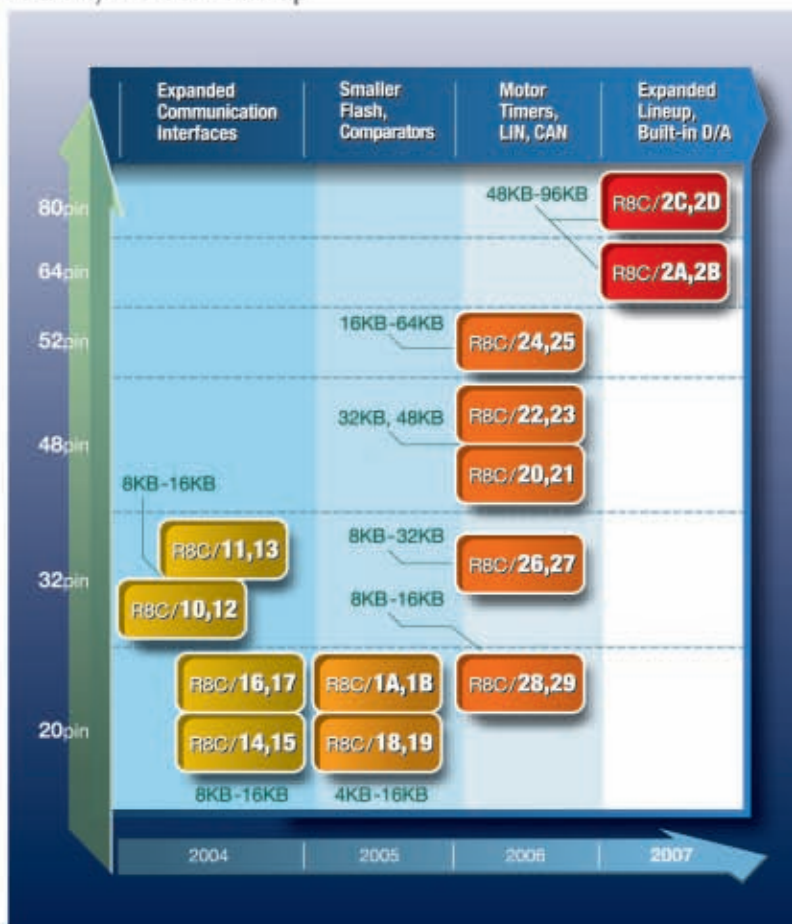
High-performance: 40MHz on-chip oscillator and single-cycle flash memory access

Renesas Technology

No. 1* supplier of microcontrollers in the world

introduces the R8C/Tiny Series of microcontrollers. Its powerful 16-bit CPU core running at 20MHz provides the performance never imagined in 8-bit MCUs. R8C/Tiny MCUs high level of integration and peripheral set will enhance your application's functionality while reducing the overall system cost.

R8C/Tiny Product Roadmap



HOT Products R8C/25

| | | |
|---|-------------------------------------|--------------------------------|
| R8C/Tiny CPU | Program Flash (16KB - 64KB) | RAM (1KB - 3KB) |
| Oscillation Circuit Main Clock (20MHz Max.) | Data Flash (2KB) | Power-On Reset Circuit |
| On-chip Oscillators (40MHz, 125KHz) | Low Voltage Detect Circuit | Protect Register |
| Oscillation Circuit Sub Clock (32KHz Clock) | Enhanced WDT | 16-bit motor control Timer (2) |
| RTC | External Oscillation Stop Detection | 8-bit Timer (3) |
| Serial I/O Clock Sync./UART (2ch) | SSU/12C | A-D Converter (10-bit x 12ch) |
| 44 GPIO | Hardware LIN | On-Chip Debug |

Package: 52pin LQFP (10mm x 10mm, 0.65mm pitch)

Top Reasons To Select R8C/Tiny

- High-Performance**
 - 16-bit CPU runs at 20MHz, executing instructions in as fast as 50nsec.
- Scalable**
 - Same core, same peripherals allows easy portability from 20 to 80pins
- High-Integration**
 - Includes 40MHz on-chip oscillator, data flash, power-on reset circuits, several 8- and 16-bit timers, up to 20 channels of A/D, D/A, LIN, CAN and more
- Reliable**
 - Oscillator stop detection circuits, access control of system registers
 - Watchdog timer with on-chip oscillator, programmable low-voltage detect circuit
- World-class Development Environment**
 - Complete software and hardware tools for short development cycle
 - Free 64KB software tool chain

*Source:Gartner Dataquest (April 2006) *2005 Worldwide Microcontroller Vendor Revenue* GJ06333



Get Started Today -

Go online and register to be eligible for a FREE Starter Kit

www.america.renesas.com/ReachR8C/d



Renesas Technology Corp.


**Add SD
in a FLASH!**
with our
Flash Card Library

- ✓ Easily exchange data between your product and a PC
- ✓ Create, Append and Delete PC compatible files
- ✓ Add GIGABYTES of onboard or removable storage to your product
- ✓ Compatible with over 20 card styles SD, MMC, MicroSD, TransFlash, ...
- ✓ No OS, RTOS or interrupts are needed
- ✓ Works with low cost Microcontrollers
- ✓ Support for FAT16 and FAT32 file systems
- ✓ Full C sources easily ports to many C compilers

Full directory support **NEW**


Flash Card Adapter Board

The flash card adapter board contains all the basic circuitry needed to add PC-compatible flash card memory to virtually any microcontroller application, using as few as four I/O pins. It also includes a 3.3V regulator, built-in level shifting, and sockets for both SD/MMC and MicroSD/TransFlash memory cards.



PIC Development Board

The development board includes a built-in demo application and all the necessary hardware to run each of the example programs included with the SFCLIB (sold separately). It includes sockets for SD/MMC and MicroSD/TransFlash memory cards. The board also has an MPLAB ICD 2 In-Circuit Debugger jack for easy programming and debugging, and a large prototyping area for adding your own circuitry.



**EFFICIENT
COMPUTER SYSTEMS**

Offering MEGA solutions for the embedded MICRO world.

For more information see our website www.ECS87.com/sfclib
603-776-3151 • 5 Emerson Drive Center Barnstead, NH 03225

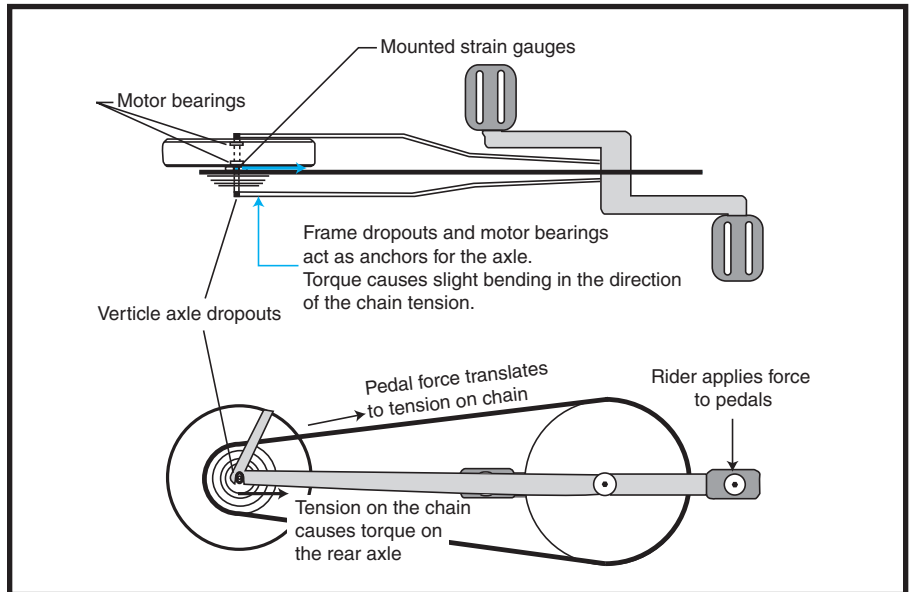


Figure 5—A small deflection of the rear axle occurs each time the rider applies force to the pedals. Resistance of the thin-film strain gauges changes when the axle bends. This allows the pedal force to be quantized into a throttle control.

power source voltages by calling the `Check_Voltages()` function. A flowchart for `main()` and `Check_Voltages()` is shown in Figure 6. `main()` also takes an initial torque reading to calibrate the motor assistance to the rider (this reading becomes the “zero” reference for subsequent torque readings) and handles input from the keypad on the LCD user interface.

One very important ADC channel to constantly poll is `m_Brake`, which is connected to the brake potentiometer. Whenever the brake is pressed, we want it to instantly kill the motor, whether or not regeneration is going to be used. We allowed a small brake depression margin. This ensures that braking routines are not called if the rider lightly taps the brake or travels over a surface bumpy enough to generate ADC readings. (A maximum depression ADC value is 1,023. Our margin is 50.)

`Check_Voltages()` decides which power source (batteries or capacitors) to use and whether to charge the capacitors when the brake is pressed. The ultracapacitors have priori-

ty over the batteries, so we check their ADC channel first. If the voltage measured at the ADC channel (`m_VCap`) exceeds 20 V, as well as the calculated BEMF of the motor, the ultracapacitors are selected as the power supply by setting a flag. If the capacitor voltage is too low, the flag is cleared.

If the capacitor flag is clear, `Check_Voltages()` will read the ADC channel of the batteries (`m_VBatt1`, `m_VBatt2`, `m_VBatt3`, `m_VBatt4`). Since the battery packs cease to operate safely below 48 V, we check for an average voltage above 50 V. If capacitors are not selected for use and the batteries have enough charge,



Photo 4—The four strain gauges were mounted on the rear axle. The axle was ground flat to allow more deflection and make it easier to mount the strain gauges.

a flag is set to use the batteries. If at any time the capacitors drop below 20 V and the batteries are below 50 V, the motor will disengage.

Note that a final operation of `Check_Voltages()` checks if the capacitors are low enough to handle regeneration. If the capacitors are low enough to accept a charge, a flag is set to confirm charging when the brake is pressed.

PWM MANAGEMENT

Most of the drive functionality is controlled by interrupts. The dsPIC includes change notification (CN) pins, which can be configured to generate an interrupt when a change in state is detected on one of the pins. Figure 7 is a flowchart of the CN interrupt. Using the interrupt, we can monitor the three Hall-effect sensor outputs from the motor to determine

which coils of the motor to energize. This is the essence of our forward motor control.

Writing a value to the dsPIC's OVDCON register states which PWM outputs should be driven high, and which should be controlled by its PWM generator. OVDCON is a 16-bit register. The upper byte dictates if the output on the PWM pin is driven by the dsPIC's PWM generator or by the active value (high or low) written to the corresponding pin in the lower byte of the value.

With six possible Hall values (three sensors, two states per sensor, no zero or "all on" state), how did we know what to write to the register to get the motor spinning? Each of the three sensors outputs a logic high for a certain 180° of electrical rotation, and a logic low for the other 180°. On each interrupt, we

read the new incoming Hall value on all three of the CN pins and combine these, using bit-shifting, into an integer to create a numerical value `HallValue` (1 to 6) for the three sensors.

Now that we have proper Hall values, we simply use an array whose contents are indexed by the incoming Hall value on each interrupt:

```
unsigned int StateLoTable[] =
{0x0000, 0x0120, 0x1008, 0x0108,
0x0402, 0x0420, 0x1002, 0x0000};
```

To write this drive sequence:

```
OVDCON = StateLoTable[HallValue];
```

The values of this array were determined by which drive FETs in the H-bridge should be energized to provide a clockwise rotation of the motor. We have shown you the array contents above, but a detailed explanation of its derivation would be too long for this article. The array was borrowed from Microchip application note AN957. Check out the application note for useful motor drive fundamentals.

ARE WE MOVING YET?

Now that we've written the array contents to the OVDCON register, we will use a state machine to track the incoming Hall values before turning on the motor. There are several reasons for this. We don't want the motor to provide assistance if the rider is idly pedaling backwards at a stop light, and we don't want the bicycle to take off if the rider is walking his/her bike under 5 km/h with one foot on a pedal, amongst other reasons.

The state machine advances only when the next proper Hall value is received (see Figure 7). By connecting the Hall-effect sensors of the motor to an oscilloscope, we verified that the incoming values were a repeating sequence of 3, 2, 6, 4, 5, 1. Since we used an eight-pole motor, this sequence is repeated eight times for one full rotation of the wheel. When the state machine detects a full rotation, it starts a timer (TMR2) and reads the count value every time a full rotation is encountered. Using a new timer value every rotation of the

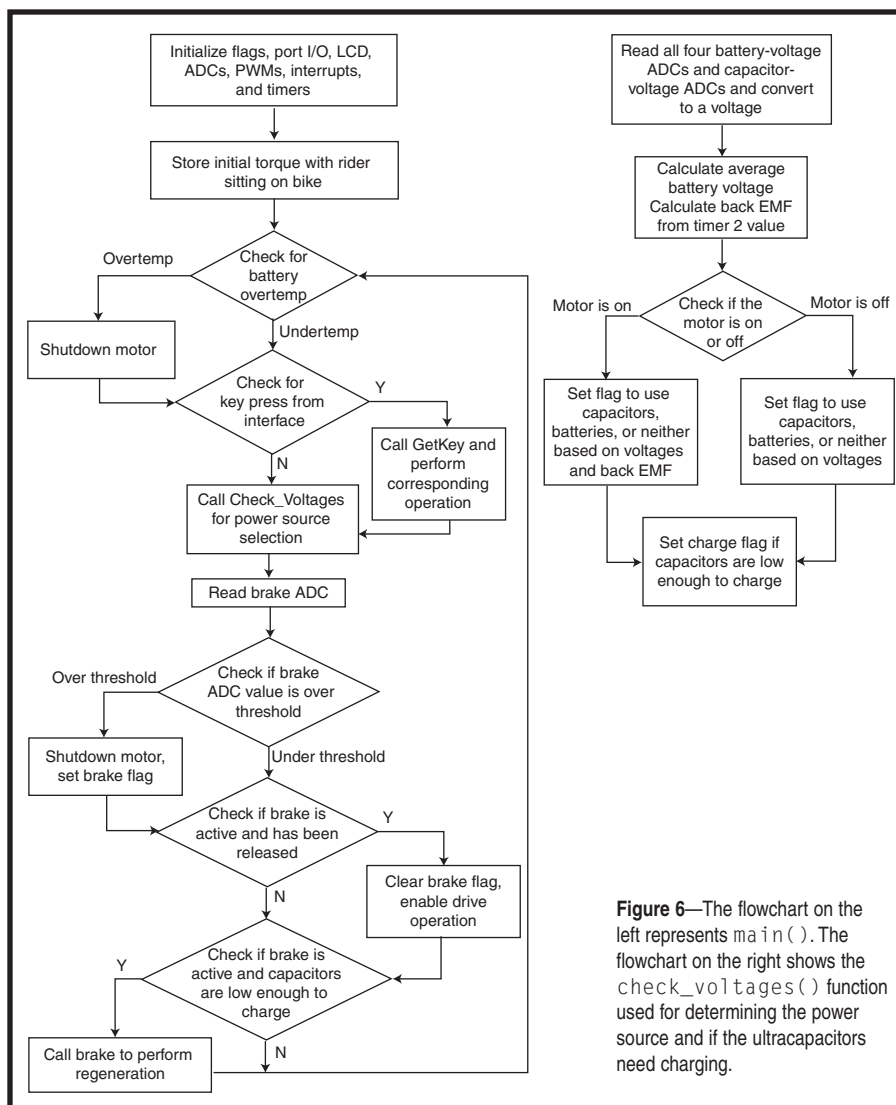


Figure 6—The flowchart on the left represents `main()`. The flowchart on the right shows the `check_voltages()` function used for determining the power source and if the ultracapacitors need charging.

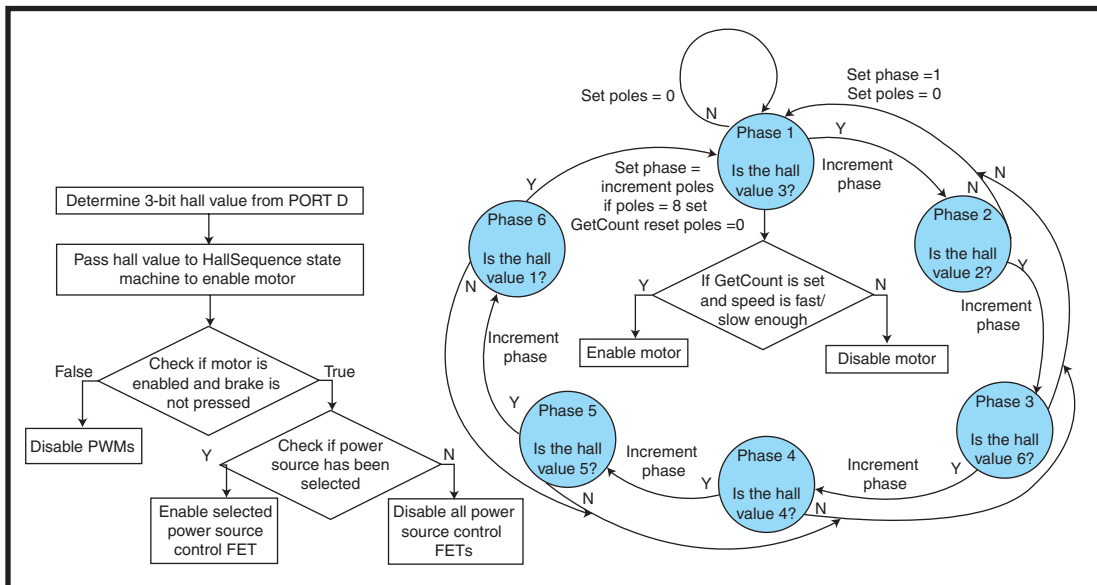


Figure 7—The flowchart represents the CN interrupt. The state machine shows the HallSequence() function that enables and disables the motor and determines the speed.

wheel, we accurately measure the speed of travel using the diameter of the bicycle wheel. If the timer's contents translate to greater than 5 km/h, it is safe to turn the motor by writing the appropriate value to the PWMCON register.

CALCULATING TORQUE

Since the force used in pedaling isn't uniform over a rotation of the pedals, we take 20 separate torque readings from the strain gauge ADC and store the values in an array. This is accomplished by triggering the Timer1 interrupt every 50 ms. After 20 interrupts (1 s), the dsPIC finds the largest torque value and uses it as an argument in the CalcDuty() function to calculate the duty cycle for PWMs. After another 20 interrupts, a new largest torque value is selected for CalcDuty(). Flowcharts for Timer1 and CalcDuty() are in Figure 8.

In CalcDuty(), another ADC is read to measure the actual motor current, which is then converted into a real current value. Next, the torque argument is multiplied by a constant to get a desired current value. The constant will change for each assist level:

$$k = \frac{I_{\text{ASSIST}}}{\text{max torque ADC}} \quad [1]$$

I_{ASSIST} can be determined by the maximum amount of current the

motor will draw at a selected assist level. The higher the assist level, the higher the maximum motor current. The maximum ADC value of torque (1,023) will correspond to the maximum motor current at the set assist level.

If the actual motor current is less than the desired current, we will increase the duty cycle by 5%. If the actual motor current is above the desired current, the duty cycle decreases by 5%. Because CalcDuty() is called every Timer1 interrupt, the duty cycle is updated every 50 ms.

DYNAMIC REGENERATIVE BRAKING

We designed the hardware and software to implement regenerative braking, but due to a lack of time, we were not able to test it. Let's focus on how it is supposed to work. The motor always produces a back electromotive force (BEMF) that varies with the motor speed. (Refer to the BEMF versus speed graph on the Circuit Cellar FTP site.) In order to use the voltage for regenerative braking, we must rectify it into a DC signal and amplify it to a voltage above that of the batteries or the ultracapacitors.

To rectify the signal, we used the internal protection diodes built into the MOSFETs on the H-bridge circuit. This provided us with a constant rectified signal, which has a voltage that

varies with the bike's speed.

In order to boost the voltage above that of the batteries and the ultracapacitors, we implemented a basic charge pump circuit (see Figure 2). The only difference is that instead of using an external inductor, we used the internal inductance of the motor, which we measured at approximately 1 mH.

When the PWM DC_DC_CONVERTER MOSFET is turned on, there is

effectively a short circuit across the motor. This charges the electromagnetic field within the motor/inductor. The field is still present when the MOSFET is turned off. This forces current through the diode, effectively increasing the voltage. The capacitor is used to smooth out the pulsed voltage for charging the batteries and ultracapacitors. By varying the duty cycle of a PWM signal going into the MOSFET, we are able to control the output voltage from the circuit.

The transfer function of this circuit is:

$$\frac{V_o}{V_{in}} = \frac{1}{1 - DC} \quad [2]$$

We are running it at 50 kHz.

BRAKING SOFTWARE

The software controlling the regenerative braking needs to be fast and accurate. Before operating the regeneration routines, it is very important to turn off all drive MOSFETs and PWMs to avoid shorting the power sources. If an error is made, short circuits, component damage, and fire may result (and did).

The brake routines need a high priority. Therefore, the brake ADC is polled in main() and the regeneration functions are called from main(). When the brake is pressed and the brake ADC inputs a value over a threshold of 50, the motor, interrupts,

and control MOSFETs are disabled and the brake flag is set.

Next (also in `main()`) we poll the brake flag. If the brake flag is set from the rider pressing the brake and the capacitors need charging, the `Brake()` function is called. `Brake()` controls the PWM duty cycle of the DC-DC converter MOSFET, as well as the capacitor charge control MOSFET.

With dynamic regeneration, continuous variable control of the braking force applied by the motor means controlling the amount of energy (current) drawn from the motor. We set the maximum current draw from the motor to 10 A. This means that a full pull of the brake lever will draw up to 10 A of braking current from the motor. The same formula used to calculate the assist-level constants for converting torque ADC values into desired current values can be used for braking.

Using the brake ADC value (up to 1,023), the desired braking current, in amps, is calculated:

```
Brake_Current = ReadADC(m_Brake)*0.01;
```

The actual motor current ADC is then read and converted into amperage. The constant used in this formula was

found using serial-port debugging using a known current:

```
Charge_Current = ReadADC(m_Current)*0.078;
```

Now the brake and charge currents are compared and the duty cycle of the DC-DC converter is increased or decreased as necessary until the actual motor current matches the desired braking current. By increasing or decreasing the duty cycle in 1% increments, the output voltage of the DC-DC converter will overcome the voltage stored in the capacitors and cause current to flow into the capacitors. When the potential difference between the capacitors and the DC-DC output is great enough, the desired charging current is established. In essence, this function is identical to the `CalcDuty()` function used for calculating the duty cycle when driving.

Finally, in `main()`, when the brake ADC is read to be under the threshold required for regeneration (when the rider releases the brake), the PWMs are reset to operate in Drive mode and the interrupts are reenabled.

TESTING PROCEDURES

When we started writing the software, we were still brainstorming

about what to do with the hardware. This meant that we had to start the software small and expand it to fit the needs of the entire system later.

We began by designing a basic motor controller that would load theoretical values into the `OVDCON` register based on which Hall values were active. We simulated the three Hall values manually using three switches, and monitored the PWM outputs on an LED bar.

Next, we used our PWMs to run a floppy drive motor. We set the PWM outputs to the corresponding pins for the floppy motor and started tapping away Hall values on the switches. To our surprise, the motor started spinning on the first try! This gave us confidence and allowed us to use the basic concepts to design the final motor controller.

To test the software as a whole, we set up 14 potentiometers on a breadboard to simulate various values, such as motor current, battery, and ultracapacitor voltages. Before we hooked hardware up to this thing, we needed to be sure that the software was going to respond predictably.

Once we were confident with the software's effect on LEDs, we began testing it with the hardware. We mounted the bike on a wind trainer that we got from the college gym, attached all the hardware to the bike, and started riding.

The main problem we found was that the values being read from our ADCs weren't ideal. We fixed this by writing the ADC register values to the `U2TXREG` and viewing them on a serial program. Since we now knew exactly what our PIC was seeing, we were able to tweak the software to deal with it.

RESULT

By the end of the three-month term, we had an electronic bicycle that could switch between lithium-ion batteries and ultracapacitors, adjust the assistance level relative to human effort, and reach speeds above 40 km/h in our testing environment. The top speed may not seem like much, but getting to that speed takes little effort and acceleration time. Furthermore, the junkyard bike we used for the

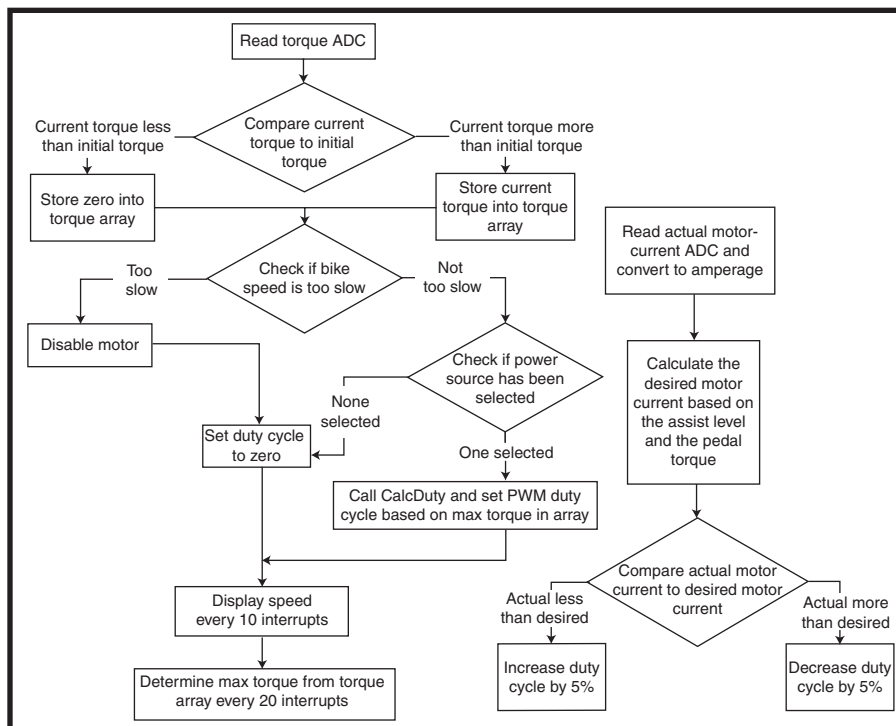


Figure 8—The flowchart on the left shows the timer1 interrupt. The flowchart on the right shows the `calc_duty()` function used to increase/decrease the PWM duty cycle based on motor current.

project could go much faster with a different set of gears. At top speed, it was impossible to go any faster because the top gear felt like a granny gear.

A few things come to mind for future designs and modifications. The first would be to discard our motto of more power equals more fun. This is still true in our minds, but a lot of design work and expense could be saved by dropping the voltage to 48 V. This would also give us a broader range of components to select from, including ICs that encompass most of the control and protection circuits onto one chip. Also, a lot of board space could be saved by eliminating a lot of the ADCs that could be deemed unnecessary. Obviously, further testing and operation of the regeneration is at the top of the list. ☒

Roxanne Ahmed recently received her Computer Engineering Technology diploma from Camosun College in Victoria, BC. She plans to attend university and finish her degree. Her professional interests include environmental technology, RF systems, firmware development, and ocean sciences. Her hobbies include cycling, rock climbing, hiking, and playing guitar. You can contact her at rahmed1@shaw.ca.

Michael Buchanan is interested in electric vehicles and other environmentally friendly alternatives, as well as electronic music, and pro audio technology. He recently graduated from the Computer Engineering Technology program at Camosun College and is now looking for employment. In his spare time, Michael enjoys hiking, playing the guitar, and learning about fascinating new things on the Internet. You can contact him at mike-buchanan@hotmail.com

Andrew Clarke recently received a diploma in Electronic Engineering Technology from Camosun College. He is interested in electric vehicles, renewable energy solutions, and home audio. In his spare time he enjoys fixing his car and snowboarding. You

can contact him at ajclarke@shaw.ca.

Chris Elder is interested in electric vehicles, environmental energy, home theater, and audio. He recently received a diploma in Electronic Engineering Technology and is currently looking for employment. In his spare time he snowboards, races dragon boats, and plays the ukulele. You can contact him at celder2003@hotmail.com.

Warren Shrieves recently received his diploma in Electronic Engineering Technology from Camosun College. He is interested in renewable energy production and finding more efficient ways to use energy in everyday life. In his spare time he enjoys hiking, boating, and being outside. You can contact him at wshrieves@hotmail.com.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2007/205.

RESOURCES

Analog Devices, Inc., "AD623: Single-Supply, Rail-to-Rail, Low Cost Instrumentation Amplifier," C3202c-0-9, 1999, www.analog.com/UploadedFiles/Data_Sheets/AD623.pdf.

———, "AD8551/AD8552/ AD8554: Zero-Drift, Single-Supply, Rail-to-Rail, Input/Output Operational Amplifiers," C01101-0, 2007, www.analog.com/UploadedFiles/Data_Sheets/AD8551_8552_8554.pdf.

———, "AD8553: 1.8 V to 5 V Auto-Zero, In-Amp with Shutdown," D05474-0, 2005, www.analog.com/UploadedFiles/Data_Sheets/AD8553.pdf.

W. Brown, "Brushless DC Motor Control Made Easy," Microchip Technology, Inc., 2002, ww1.microchip.com/downloads/en/AppNotes/00857a.pdf.

S. D'Souza, "Sensored BLDC Motor Control Using dsPIC30F2010", Microchip Technology, Inc., 2004, ww1.microchip.com/downloads/en/AppNotes/BLDC%20MC%2000957a.pdf.

International Rectifier, "IRFI4227PbF: PDP SWITCH," PD-97036A, 2005,

www.irf.com/product-info/datasheets/data/irfi4227pbf.pdf.

———, "IR2101(s)/IR2102(s)/(PbF): HIGH AND LOW SIDE DRIVER," PD60043, 2004, www.irf.com/product-info/datasheets/data/ir2101.pdf.

———, "Series PVI-N & PbF: Photovoltaic Isolator Single and Dual Channel 5-10 Volt Output," PD10054, 2006, www.irf.com/product-info/datasheets/data/pvin.pdf.

Microchip Technology, Inc., "Application Note AN857," ww1.microchip.com/downloads/en/AppNotes/BLDC%20MC%2000957a.pdf.

———, "Application Note AN957," ww1.microchip.com/downloads/en/AppNotes/BLDC%20MC%2000957a.pdf.

———, "dsPIC30F6015 Data Sheet," DS70150C, 2007, ww1.microchip.com/downloads/en/DeviceDoc/70150C.pdf.

P. Yedamale, "Brushless DC (BLDC) Motor Fundamentals," 2003, Microchip Technology, Inc., ww1.microchip.com/downloads/en/AppNotes/00885a.pdf.

SOURCES

AD623 Amplifier, AD8552 op-amp, and AD8553 in-amp
Analog Devices, Inc.
www.analog.com

408 Hub motor
Crystalyte
www.crystalyte.com

IR2101PbF MOSFET, IRFI4227PbF MOSFET, and PVI1050NPbF photovoltaic isolator
International Rectifier
www.irf.com

dsPIC30F6015 DSP, MPLAB C30 v3 Student Edition, and MPLAB ICD2
Microchip Technology, Inc.
www.microchip.com

ESHSR-010000-002RT Capacitor
Nesscap
www.nesscap.com

Li-Ion 18650 Rechargeable battery pack
Tenenergy Corp.
www.tenenergybattery.com

Only 4 Steps...

...are required to generate efficient, reliable applications with the μ Vision IDE and development tools from Keil.

Step 1. Select Microcontroller and Specify Target Hardware

Use the Keil Device Database (www.keil.com/dd) to find the optimum microcontroller for your application.

In μ Vision, select the microcontroller to pre-configure tools and obtain CPU startup code.

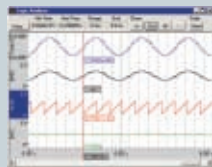
Step 2. Configure the Device and Create Application Code

The μ Vision Configuration Wizard helps you tailor startup code to match your target hardware and application requirements.

Extensive program examples and project templates help you jump-start your designs.

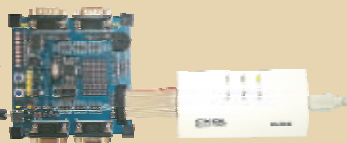
Step 3. Verify Program Execution with Device Simulation

High-speed simulation enables testing before hardware is available and helps you with features like instruction trace, code coverage, and logic analysis.

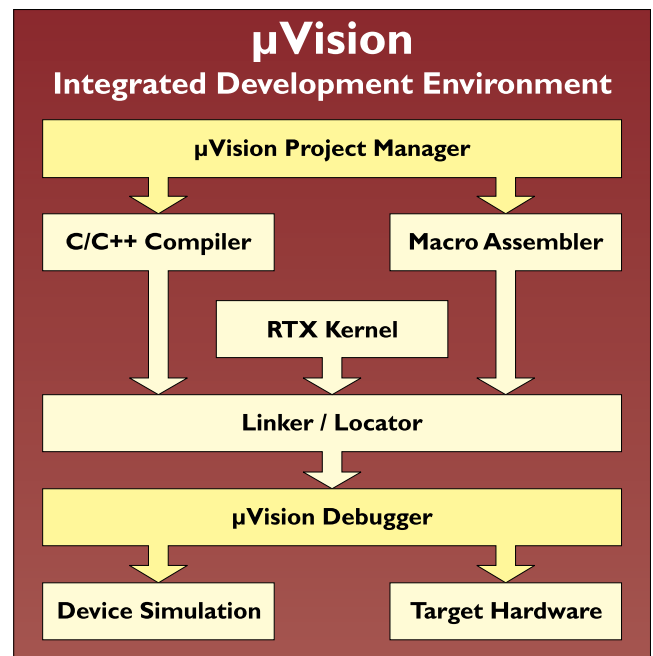


Step 4. Download to Flash and Test Application

Once your application is runs in simulation, use the Keil ULINK USB-JTAG Adapter for Flash programming and final application testing.



Keil Microcontroller Development Tools help you create embedded applications quickly and accurately. Keil tools are easy to learn and use, yet powerful enough for the most demanding microcontroller projects.



Components of Keil Microcontroller Development Kits

Keil makes C compilers, macro assemblers, real-time kernels, debuggers, simulators, evaluation boards, and emulators.

Over 1,200 MCU devices are supported for:

- **8-bit** - 8051 and extended 8051 variants
- **16-bit** - C166, XC166, and ST10
- **32-bit** - ARM7, ARM9, and Cortex-M3

Download an evaluation version from www.keil.com/demo

Text Adventure Gaming

Remember the old Pyramid 2000 text game? Chris shows you how to use a Parallax Propeller chip and the Cog Coordination language (CCL)—a simple interpretive language that he developed for the Propeller—to breathe new life into the old text adventure game.

You are crawling over pebbles in a low passage. There is a dim light at the east end of the passage, but a huge green, fierce serpent bars the way! What do you do now? Do you type EAST and circle back to look for another way around? Do you ATTACK SERPENT with your scepter? Maybe you can feed it something. You type INVENTORY to rummage around in your pack. Meanwhile, the serpent in your imagination wriggles around your feet while the blinking cursor on the computer screen waits for anything you can dream up.

As you can tell, I spent many long hours in front of the Pyramid 2000 text adventure game. The goal was to find the fabulous treasures while dodging the creepy mummy, but the trickiest part of the game was the 5-min. battle with the cassette tape drive trying to get the program to load.

I recently ran across a binary image of the old TRS-80 color computer game on the 'Net. I spent many more long hours on the game—this time digging into the mysterious 6809 disassembly. In this article, I'll describe how to breathe new life into the old text adventure game. I'll show you how I distributed the game tasks among the eight computers packed into the Parallax Propeller chip. The Pro-

peller demonstration board hooks directly to a TV and a PS/2 keyboard for user interaction, but I used an SD card instead of a tape drive (see Photo 1). I'll also show you how I interfaced an SD card to the Propeller board.

Finally, I'll introduce you to the Cog Coordination language (CCL), a simple interpretive language I created to run on the Propeller. With CCL, you can create your own text adventure games on your PC, drop them onto an SD card, and play them on the Propeller board. For simple applications, CCL is a quick and easy alternative to learning the propeller's programming languages and IDE.

THE PROPELLER

The Parallax Propeller chip is a collection of eight 32-bit RISC processors on a single die. Each processor, called a "cog," has 2 KB of RAM for code and

data. The cogs share access to 32 KB of global RAM and 32 KB of system ROM. The chip's hardware manages a "spinning" access window that gives each cog a chance to access the global memory one cog at a time. The cogs also share 32 general-purpose I/O lines, and there are no interrupts in the system. Instead, you launch driver cogs that bit bang outputs and poll inputs for the devices you connect.

Parallax has a wonderful development environment for programming the Propeller. The colorful GUI keeps all of your code files organized in tabs on the screen. You can download your applications from the GUI, through a USB cable to the shared RAM for testing, or you can download your application directly to a serial EEPROM connected to the Propeller. During start-up, the propeller automatically fills its 32 KB of global RAM from the EEPROM.

You can program the Propeller chip in either assembler or Parallax's new interpreted Spin language. Most applications will use a mixture of both styles, falling on assembly for the critical timing sections. Multicog operations are built directly into the processor's instruction set. The `coginit` instruction fills a target cog's 2 KB of RAM from global RAM and starts the cog

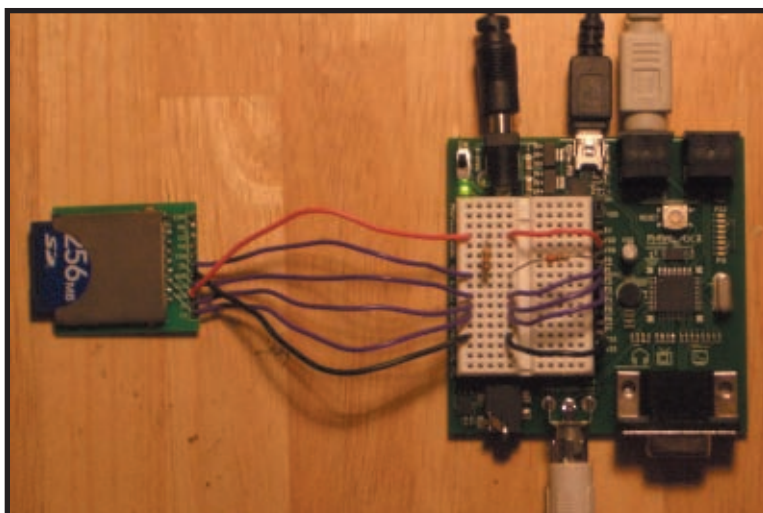


Photo 1—The SD card connector breakout board hooks to the Propeller demo board with two resistors and a few wires.

Listing 1—A Propeller program is a smooth mixture of languages: Spin and assembly. This program for the demo board launches keyboard and TV driver cogs along with two simple cogs that count up at different rates. The program waits for a key press and displays the current value of the counters.

```

CON
    _clkmode      = xtal1 + pll16x
    _xinfreq      = 5_000_000

OBJ
    term : "tv_terminal"
    kb   : "keyboard"

VAR
    long counterA
    long counterB

PUB start
    cognew(@r1, @counterA) ' Start a new COG
    cognew(@r2, @counterB) ' Start a new COG
    term.start(12) ' Start the TV and graphics COGs
    kb.start(26,27) ' Start the Keyboard driver COG

    repeat
        kb.getKey ' Wait for a key
        term.hex(counterA,8) ' Print counterA
        term.out(",") ' Spacer
        term.hex(counterB,8) ' Print counterB
        term.out(13) ' New-line

DAT
    org 0
    r1
    add    c1,#1 ' Add 1 to counter
    wrlong c1,par ' Write register 3 to main RAM
    jmp    #r1 ' Loop back
    c1
    long  0 ' Init counter to 0

    org 0
    r2
    add    c2,#2 ' Add 2 to counter
    wrlong c2,par ' Write register 3 to main RAM
    jmp    #r2 ' Loop back
    c2
    long  0 ' Init counter to 0

```

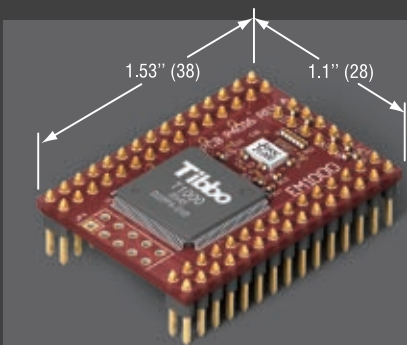
executing the newly loaded code. The assembly `rdlong` and `wrlong` instructions read and write 32-bit values from the shared RAM. The hardware automatically halts the CPU until its turn comes back around.

The hardware also includes eight individual mutex locks that you can check and set atomically with the `lockset` and `lockclr` assembly instructions. Locks are used as flags to tell other cogs when a resource (a section of memory or an I/O pin or whatever) is in use by a cog and should not be accessed. Implementation of a mutex lock is tricky because you have to make sure no other cog can change the lock between the time you check it and find it available and the time you set it to lock others out. The `lockset` instruction performs the check-and-set-if-available in one step, preventing the cogs from walking on each other.

Locks don't solve all of your inter-cog communications. You must still program your applications to avoid deadlocks. If you aren't careful, you might find cog A waiting on a lock held by cog B, while cog B is waiting

Tibbo
TECHNOLOGY

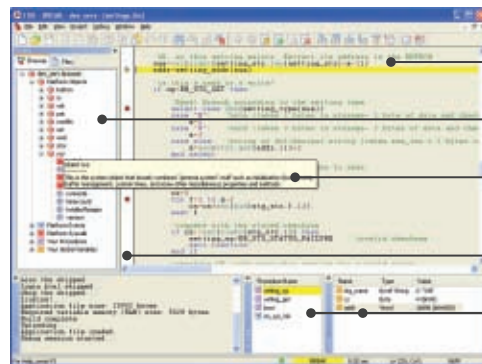
Build your next automation project around our EM1000 BASIC-programmable Embedded Module



- 50 MIPS CPU
- 100BaseT Ethernet port
- 512K flash disk
- 4x high-speed UARTs
- High-speed parallel slave port
- Real-time clock with backup power
- 49x general-purpose I/O lines
- Development kit available (EM1000-SK)

- Programmable – in BASIC!
- Optimized for real-time applications
- Rich object set
- Built-in webserver
- Event-driven operation
- Sophisticated development environment supports cross-debugging (no ICE needed)

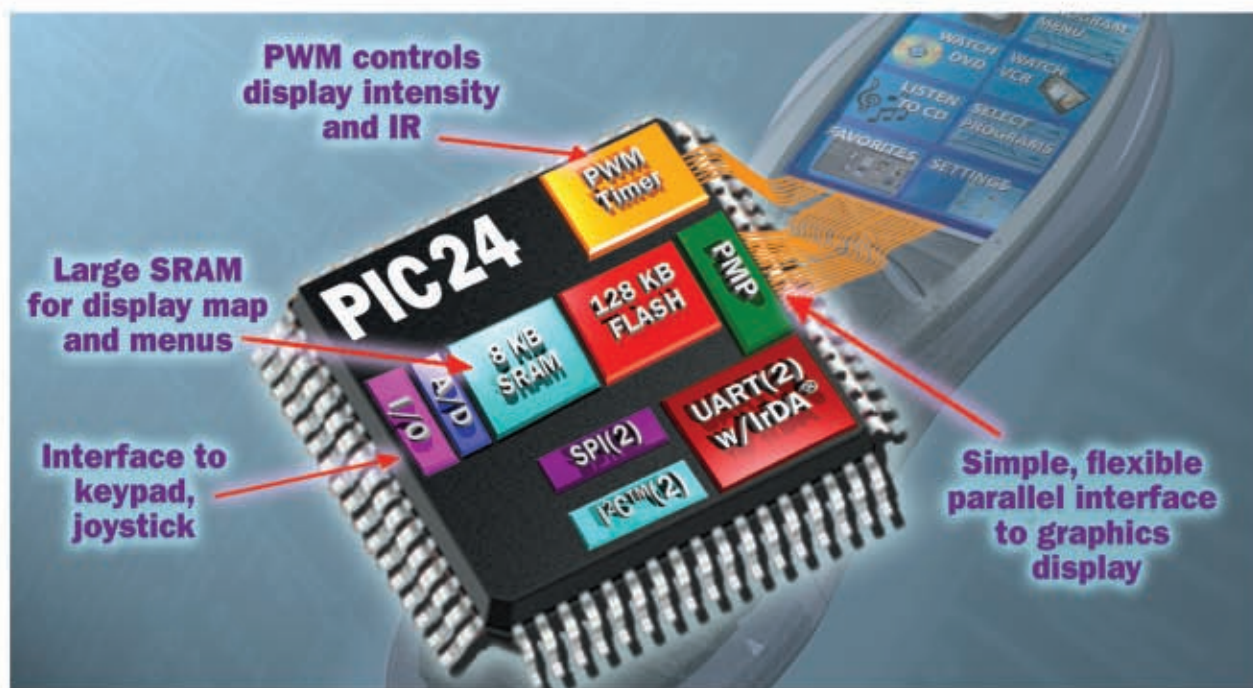
Code and debug your Tibbo BASIC application using Tibbo Integrated Development Environment (TIDE) software



- Write in familiar BASIC language
- Inspect objects, procedures, and variables
- Code faster with auto-completion and code hints
- Set breakpoints, execute step-by-step, etc.
- Monitor the state of variables and stack

web: www.tibbo.com email: sales@tibbo.com

Control Your Remote with Low-Cost 16-bit MCUs



Whether your design is a home theater remote or any other embedded application, the PIC24F delivers the peripherals, performance, development tools and software you need.

With 16 MIPS performance and an extensive peripheral set, Microchip's PIC24F microcontrollers are highly cost-effective solutions.

GET STARTED IN 3 EASY STEPS!

- 1. Learn all about our 16-bit products in only 20 minutes!** Our *FREE* 16-bit web seminars highlight our 16-bit architecture and comprehensive support suite that includes low-cost development tools, a variety of software libraries (many free), application notes, reference designs and more!
- 2. Take advantage of *FREE* resources for download!** Our MPLAB® IDE Integrated Development Environment and a full-featured trial of the MPLAB C30 C Compiler from our web site gets you up and running fast!
- 3. Get hands-on technical training for as low as \$49!**



Attend a half- or full-day 16-bit course at a Microchip **Regional Training Center** to take your designs to the next level! Attendees also receive deep discounts on related development tools!

Purchase and program your 16-bit PIC24F devices and related development tools at...

microchip
DIRECT
www.microchipdirect.com

| Device | Pins | Flash (KB) | PIC24F Family Features |
|-----------------|------|------------|---------------------------------------|
| PIC24FJ64GA006 | 64 | 64 | 8 KB RAM |
| PIC24FJ64GA008 | 80 | 64 | Parallel Master Port |
| PIC24FJ64GA010 | 100 | 64 | 5 - 16-bit Timers |
| PIC24FJ96GA006 | 64 | 96 | 5 - Output Compare/PWM |
| PIC24FJ96GA008 | 80 | 96 | 5 - Input Captures |
| PIC24FJ96GA010 | 100 | 96 | Real-Time Clock/Calendar |
| PIC24FJ128GA006 | 64 | 128 | 2 - UART with IrDA® and LIN Protocols |
| PIC24FJ128GA008 | 80 | 128 | 2 - SPI, 2 - I ² C™ |
| PIC24FJ128GA010 | 100 | 128 | 10-bit ADC, 16 Channels |
| | | | 2 Analog Comparators |

Visit our web site for more information about additional 16-bit devices with higher performance and added features like DSP and enter our 16-bit Embedded Design Contest today!



MICROCHIP

www.microchip.com/PIC24

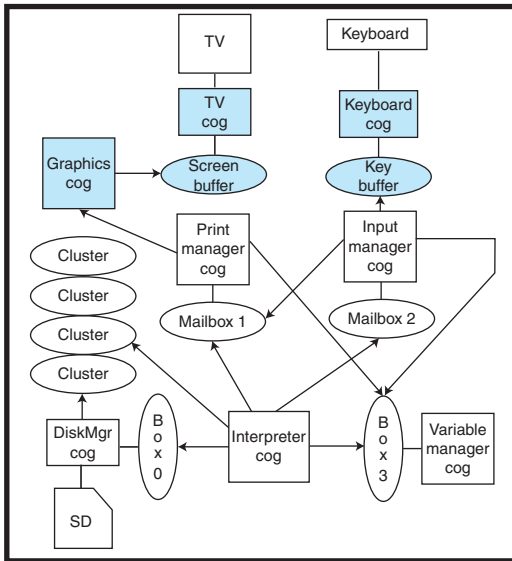


Figure 1—The text adventure game system uses all eight of the Propeller cogs. Three of the cogs (shown in blue) are library drivers for the TV and keyboard. The other five cogs communicate through an array of mailboxes in shared RAM.

on a lock held by cog A. The two cogs are then stuck in a face-off that you must break up with the Reset button.

You can buy Propeller chips individually. They require very few external parts to get up and going. In this project I used the Propeller demo board, which includes connectors for a TV, a VGA monitor, headphones, and a microphone. There is also a small breadboard for external development and quick connectors for the port pins. The Parallax web site has lots of information about the Propeller, including a 400-page user's guide and lots of free code from the community of developers. Tom Cantrell (no relation) gave a wonderful introduction to the Propeller chip last August in "Turning the Corner" (*Circuit Cellar* 193, 2006).

My text adventure game system interfaces to three physical devices: a TV, a keyboard, and an SD card. The Propeller library includes the necessary drivers for the TV and keyboard. By including the "tv_terminal" and "keyboard" modules in your code, you actually launch three cogs: one to bit bang the video-output signal to the TV, one to bit bang the PS/2 keyboard, and a third to manage the video graphics [lines, sprites, text, and such].

Listing 1 shows a complete example of starting cogs, printing to the screen, and reading from the keyboard. The flow looks straightforward, but there

are a lot of inter-cog communications under the covers. Take the print statement `term.out` from Listing 1. The Spin "out" function fills a section of RAM with a pointer to the message string and a command number. The graphics cog watches this shared section of RAM and processes the command. It twiddles the bits in the screen buffer's RAM to draw the text. The graphics cog reports back a success and the "out" function returns to the main code. Meanwhile, 60 times a second, the TV driver cog reads the screen buffer RAM and paints the TV screen.

That's a lot of cross communication that has to be perfectly synchronized. Needless to say, debugging multiprocessor programs in mixed Spin and assembly on the Propeller chip can be a challenge. As I will discuss later, CCL is a simpler alternative for simple applications.

INTERFACING TO AN SD CARD

Interfacing an SD card to the Propeller demo board is easy. You need a card socket to accept the SD card. The \$15 BOB-00204 SD breakout board from Spark Fun Electronics brings all the card signals to a nicely labeled row of solder holes. You can wire 3.3-V power and ground directly from the Propeller board. Three of the four standard SPI bus signals wire directly from the socket board to the Propeller chip's port pins: chip select (CS), data in (DI), and clock (CLK). The fourth signal, data out (DO), is pulled up to 3.3 V with a 10-k Ω resistor and then connected to a Propeller port pin through a 220- Ω resistor. Photo 1 shows the SD socket board connected to the Propeller board.

The Propeller chip does not include dedicated SPI hardware. Instead, you have to bit bang the communications yourself by toggling the clock line to read and write bytes one bit at a time. Commands sent to the SD card are 6 bytes long (a command ID, a 4-byte address, and a checksum byte). The card sends back a single-byte status response.

The SD card appears to your pro-

gram as an array of 512-byte sectors. Each sector has a unique address starting with sector 0. The larger the card, the more sectors you can address. The small 256-MB card I used in this project has over 500,000 sectors! You read a sector by sending the READ command, which includes the sector's address. When the card responds, you read 512 bytes, the contents of the sector stored on the disk. Similarly, you write a sector by sending the WRITE command followed by the 512 bytes to store in the sector. You always move 512 bytes at a time.

Having 500,000 sectors at your fingertips can be a bit overwhelming. Most processors have a hard time just holding on to a 3-byte quantity like "500,000," let alone keeping up with what's in each of its many sectors. That's where a file system comes in handy. File systems like FAT organize unwieldy sectors into groups (files). Files, in turn, are organized into a tree of directories and subdirectories. Instead of talking to the SD card directly, you talk to a FAT file system driver that talks to the SD card for you. If you want to use an SD card to share data with another computer (like your PC) you have to be FAT-aware.

The FAT file system is straightforward. There is plenty of code out on the 'Net to help you navigate a disk to find the data you want. The trouble is that the FAT code and data structures eat up a lot of precious RAM, and my text game application already stretches the Propeller's memory limit. If you don't have the resources in your project for a full-blown file system implementation, you can take advantage of a shortcut I used in my project.

On a freshly formatted file system, the first file you copy to the drive will always appear in contiguous sectors (not fragmented). The starting sector will always be the first data sector in the file system, which you can calculate from the information in the first sector of the drive (the boot sector). You need to read only the first sector of the card and do a little math to figure out where the first file starts.

Specifically, you must read the first sector (the master boot record) and parse the partition table to find the boot sector. Then load the boot sector

and calculate the offset to the data area as follows:

```
firstDataSector = numReserved + fatSize ×  
                numFatas + rootEntries  
                16
```

If the boot sector tells you that the drive is FAT32, you must add `numberOfSectorsPerCluster` to the final value to skip over the root directory that is also stored in the data area. The `diskmanager.spin` code (posted on the *Circuit Cellar* FTP site) implements the calculation.

Of course, you have to remember to reformat the SD card on your PC every time you change the shared file. This may seem like a kludge, but if you simply don't have the resources for a full-blown file system implementation, then this will allow you to read a single file written to the card from your PC. That is all the functionality I needed for this project.

CCL

The Spin and assembly languages in the Parallax IDE are great for advanced Propeller applications. But they are overkill for the simple logic, like "print this if the lamp is on," used in text adventure games. I decided to create my own simpler interpreted language to run on the Propeller. Write your CCL programs with your favorite text editor (avoiding the complex Propeller IDE). Then, compile and test your program on your PC with my Java tool. Finally, copy the CCL program to an SD card and play it on the Propeller. Thus, you can make simple adventure games without ever having to learn the Parallax IDE, assembler, or Spin languages.

Best of all, a CCL program is organized into many small modules that load from sectors on the SD card. With Spin and Propeller assembly, your applications are limited to 32 KB of code and data. But with CCL, you can fill an SD card with gigs of code/data modules that come and go from the card as needed.

The graphics, TV, and keyboard drivers take up three of the eight Propeller cogs. I used the code right out of the Propeller library that comes with the demo board. The Disk Manager

cog and CCL Interpreter cog make up the heart of the CCL runtime. The three remaining cogs implement functions needed by text adventure games: Text Output, Line Input, and a Variable Manager that tracks 128 numeric variables for game logic. Figure 1 shows the cogs and the communication paths.

The Disk Manager loads CCL code modules from the card into one of several cache areas in the global RAM. Sectors are always loaded in clusters of four (2 KB). Old clusters are automatically ejected from the cache when new clusters are requested, although your CCL program can mark/unmark important clusters, making them permanent in the cache. Video is very memory intensive. The pixel screen buffer eats up 13.5 KB of RAM. I managed to squeeze the interpreter code into 2.5 KB, leaving 16 KB left over for a cache of four clusters.

The CCL Interpreter reads instruc-

tions from the cached clusters. It only understands four commands directly: GOTO, CALL, IF, and RETURN. These are all 4-byte commands. The first three commands contain a destination address with both cluster and offset-within-the-cluster. Thus, your code can GOTO and CALL modules all over the card. The CALL command pushes the current cluster number and offset onto a small stack. The RETURN command reloads the needed cluster if it has been ejected.

All other commands besides these four are passed to a target cog for processing. The cog address is encoded into the command itself. These "cog" commands are all 8 bytes long and the interpreter passes an additional 2-byte offset to the currently loaded cluster. This allows the target cog to find the absolute address of any data it needs from the cluster. For instance, a PRINT command would reference a string of data within the same cluster.

Listing 2—The assembly at the top of the listing is from the VariableManager cog. The code waits for data and command values to be written into mailbox 3. The Spin code at the bottom of the listing (from the LineInputManager) writes command and data values into a mailbox and waits for the return value.

```
VariableManager  
  mov     mboxCommand,par           ‘ Base ptr passed at startup  
  add     mboxCommand,#3*3*4       ‘ We are mailbox #3  
  mov     mboxData,mboxCommand     ‘ Second word of box ...  
  add     mboxData,#4              ‘ ... is the data  
  
top      rdlong   temp,mboxCommand ‘ Get command word  
        mov     op,temp wc         ‘ Check the upper bit  
  
if_nc   jmp     #top                ‘ Not a command  
  
; Process the command here  
  
        wrlong  valueB,mboxData     ‘ Return value in data word  
        wrlong  CODE_01,mboxCommand ‘ Write 1 to the command word  
        jmp     #top                ‘ Wait for the next command  
  
-----  
PUB sendCommand(cogNum, com, data, ofs)  
  ‘ Wait for the box to free up and lock it  
  repeat while not lockset(cogNum)  
  ‘ Write offset, data, and command (command trigger last)  
  longmove(box+cogNum*4*3+8, @ofs,1)  
  longmove(box+cogNum*4*3+4, @data,1)  
  longmove(box+cogNum*4*3, @com,1)  
  ‘ Wait for the COG to respond  
  repeat  
    longmove(@com,box+cogNum*4*3,1)  
  while( (com&$80_00_00_00) <> 0)  
  ‘ Get return data  
  longmove(@data,box+cogNum*4*3+4,1)  
  ‘ Free the box for other cogs  
  lockclr(cogNum)  
  return data
```

Listing 3—CCL organizes code into separate clusters. Here the “first” cluster calls a print function in “another” cluster. Clusters come and go as needed from a cache in global RAM. The trick is to divide a large program into 2-KB chunks.

```
define MaxValue 10
variable temp
; -----
Cluster first
    temp = 0
here:
    CALL another:PrintIt
    temp = temp + 1
    IF(temp < MaxValue) {
        GOTO here
    }
done:
    GOTO done
; -----
Cluster another
PrintIt:
    PRINTVAR temp
    PRINT msg
    RETURN
msg:
    # “ Hello World!”,10,0
```

The CCL cogs communicate with one another through an array of mailboxes. Each mailbox has a command word (4 bytes) and a data word (4 bytes). One cog talks to another by acquiring the target box’s lock and writing data and command values into the box. The target cog processes the command and writes

return data and a response code to the mailbox. The original cog then releases the lock for another cog. Listing 2 shows the communication process between the Variable Manager (in Propeller assembly at the top) and the Print Manager (in Spin at the bottom).

The Disk Manager listens to mail-

box 0. The interpreter uses the Disk Manager to load code clusters from card. The Interpreter, which pushes commands to everyone else, has no inbox.

The Print Manager listens to mailbox 1. There are three basic print commands you can use in CCL: PRINTVAR N prints the numeric content of a variable N; PRINT MSG prints a text message; and PRINT MSG[N] prints a message from an array of messages indexed by variable N.

The Line Input Manager listens to mailbox 2. The INPUT command reads a line from the keyboard and parses it into individual numeric tokens. The parsed tokens are then stored in the system variables. You tell the cog what words to recognize and what token value is given to each with the INPUTTOKENS command.

The Variable Manager listens to mailbox 3. It is by far the most complicated cog in the system. The cog stores 128 variables in its local RAM. Complex expressions like “A = [B] + 78” are encoded into cog commands, and the cog returns the result of the expression for any IF command.

Thus, an IF command is really a variable-command and a GOTO address to take if the returned value is 0. The Java compiler tool automatically converts nested IF/ELSE structures into IF/GOTO sequences for you.

The Interpreter uses the Disk Manager cog at mailbox 0, but it has no knowledge of how many other cogs are in the system or what they are doing. The interpreter manages program flow with IF, GOTO, CALL, and RETURN, but all other commands are passed to the other cogs untouched. Thus, the interpreter acts as a coordinator, sending the right commands to the right cog at the right time.

COMPILER

The CCL compiler (written in Java), which converts CCL text files into a binary file of clusters for play on the Propeller board, is posted on the *Circuit Cellar* FTP site. The CCL text syntax is very simple. You will have no trouble using the CCL files as a basis for your own text adventure games. Listing 3 shows a simple CCL program ready to run through the compiler.

Listing 4—All 77 rooms in the Pyramid game have the same basic structure. The “Enter” pulls in a code fragment that initializes variables. The “Middle” inserts code to read a line from the keyboard. The “End” pulls in code to process commands not handled specifically by the room.

```
CLUSTER Room30
include RoomEnter.frag
    currentRoom = 30
    ambientLight = 0
include RoomMiddle.frag
    ; Process the room specifics
    if(word1==WD_NORTH) {
        GOTO Room32:Enter
    }
    if(word1==WD_EAST) {
        GOTO Room42:Enter
    }
    if(word1==WD_SOUTH) {
        GOTO Room43:Enter
    }
    ; ...
include RoomEnd.frag
PrintDesc:
    PRINT desc
    CALL Objects:PrintDescriptions
    RETURN
desc:
# “YOU ARE IN A MAZE OF TWISTY PASSAGES, ALL”,10
# “ALIKE.”,10,0
```



Photo 2—I translated the classic Pyramid 2000 game into CCL, where it lives again on my Propeller demo board.

The CCL syntax allows you to “define” constants substituted throughout the code, “reserve” variables, and give them string names. You can also “include” other CCL files, thus dividing your clusters into manageable units. A CCL IF/ELSE structure looks a lot like C/C++, which uses nested brackets to define code flow. The compiler converts the brackets into GOTOs among the code blocks as defined by your nested flow.

The compiler is also an emulator. You can use it to test your games on your PC. Once the compiler writes the binary file, it begins playing your game exactly as the Propeller would. You can build and play the pyramid game from this project file with the following command line:

```
java CCL pyramid.ccl pyramid.bin
```

PYRAMID 2000 IN CCL

The detailed FAQ on the Pyramid 2000 game is available on the ‘Net at www.figmentfly.com. Take a few minutes to read through the walkthrough, follow the link to the online color computer emulator, and play the real game.

There are 77 distinct “rooms” that you can move through in the game. You can pick up and move around 21 objects, 11 of which are treasures you have to get back to the entrance of the pyramid. Each room has its own command script for actions that are specific to the room. There is also a detailed

command script to process common commands, such as picking up and dropping objects. The common script processes all the little cute error messages you get when you try to do things like rubbing the lamp.

The game understands 110 words, which you can put together in simple phrases, such as “GET LAMP” or “WAVE SCEPTER.” In the course of the game you meet up with a fierce serpent, a talk-

ing plant, a treasure-stealing mummy, and a twisty maze. That’s a lot of action packed into 14 KB of game.

The original game in 6809 opcodes is 14,625 bytes long. In comparison, my translation of the game to CCL consumes 83 2-KB clusters (166 KB) on the SD card! I chose to use a separate cluster for each game room. Most of the rooms are small and there is a lot of empty space in the room clusters. I also duplicated some common logic processing into each separate room. The rooms.ccl file contains all 77 room clusters. The room clusters come and go in the cache as you move between rooms. Listing 4 shows the CCL code for one of the rooms in the twisty maze.

There are three “sticky” clusters that remain permanently cached in Propeller RAM. The objects.ccl cluster handles all the descriptions when you see the objects in a room or in your pack. The common1.ccl and common2.ccl clusters contain the long, common command script. This is the translation of the script for picking up objects and delivering the classic pyramid one-liner error responses. Even though I spread the script over two clusters, I still ran out of room and left out some of the funny messages. You’ll see them commented out in the code.

The inputs.ccl cluster contains the token data used to initialize the LineInputManager cog. If you get stuck in the game, have a look at this file to see what words you can use. The

mummy.ccl cluster contains the logic executed when the mummy steals your treasure. This only happens once in the game, so the logic and printed text come and go in a separate cluster.

Finally, the pyramid.ccl is the initialization cluster that places all of the objects in their rooms. This file includes all of the other CCL files. It also names and initializes all of the game variables.

FIGHTING THE SERPENT

With this project, I brought new life to the old Pyramid 2000 text game by translating the original 6809 game into CCL for play on the Propeller. Photo 2 shows the game running on my demo board. With a few wires and resistors (and the project file for this article), you can return to the mysteries of the ancient pyramid. Imagine the serpent wriggling around your feet, and suddenly you have an idea! You reach into your pack and “THROW STATUE.” As a result, the bird statue comes to life, attacks the serpent, and, in an astounding flurry, drives the serpent away! ☞

Chris Cantrell holds an M.S. in Electrical Engineering. He is a design engineer at ADTRAN in Huntsville, Alabama. Chris’s many hobbies include teaching scuba diving, writing code for the Game Boy Advance, and evangelizing Java. You can reach him at ccantrell@knology.net.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2007/205.

RESOURCE

“Pyramid 2000: An Interactive Fiction game for the TRS-80 and TRS-80 Color Computer,” www.figmentfly.com/pyramid2000/pyramid.html.

SOURCES

32100 Propeller demonstration board
Parallax, Inc.
www.parallax.com

BOB-00204 Breakout board
Spark Fun Electronics
www.sparkfun.com

At FTDI, you'll find more than just our acclaimed range of USB Interface Chips...

We also have a wide range of competitively priced USB interface modules and cables ideal for implementing one-off designs through to full mass production runs. Our on-line shop accepts payment by credit card - no minimum order restrictions apply

TTL-232R USB to TTL Serial I/F Cable

\$23.11



- Easy solution for attaching 5v / 3.3v MCU to USB
- Fully 5v compatible I/O signal levels
- 3.3v I/O version also available
- TTL UART interface
- FTDI VCP or D2XX Drivers available
- 1.8m long 6 way cable
- 6 pin SIL pin socket (0.1in pitch)
- FT232RQ embedded into USB type "A" plug
- Data transfer rates from 300 baud to 3M baud
- Optional RTS/CTS or X-ON/X-OFF handshaking

MM232R Miniature USB Module

\$20.82



- Micro Miniature USB Module
- FT232RQ USB UART
- 0.1in Pitch Pinout
- TXD, RXD, RTS#, CTS# UART Interface Pins
- Communication from 300 baud to 3M baud
- Four configurable GPIO Pins including clock output
- USB Self / Bus powered options
- 3.3v / 5v I/O signal level options

UM232R / UM245R DIL Modules

\$23.11



- 24 Pin DIL format USB Modules
- FT232RL USB UART (UM232R)
- FT245RL USB FIFO (UM245R)
- Turned pins fit standard 24 pin i.c. socket
- USB Self / Bus powered options
- 3.3v / 5v I/O signal level options.
- Full set of UART Interface Pins (UM232R)
- All multi-function CBUS GPIO Pins available (UM232R)
- Power Enable control available (UM245R)

VMUSIC2

\$43.41



- Not only can you easily add your USB flash disk to your product but you can also play back MP3 and other popular digital music from your USB
- Uses FTDI's USB host controller Vinculum VNC1L
 - Single 5V supply input
 - USB 'A' type socket
 - Stereo 3.5mm headphone jack socket for audio playback
 - Jumper selectable UART or SPI interfaces
 - VMUSIC2 is Pb-free and RoHS compliant

VDRIVE2

\$28.32



- An easy solution for adding a USB flash disk to an existing product
- Easy front panel mounting
- Uses FTDI's USB host controller Vinculum VNC1L
- Only 4 signal lines and 5V power supply plus ground are needed
- 2 mm (0.08") pitch 8 pin connector
- 8-way header interconnect cable provided
- Jumper selectable UART or SPI interfaces
- VDRIVE2 is Pb-free and RoHS compliant

US232R-10 Premium USB - RS232 Cable

\$26.03

10cm version



- Looks great and works great with MAC platforms as well as PC.**
- High tech white gloss enclosure
 - Blue side-lit LED TX and RX traffic indicators
 - Gold Plated USB and DB9 connectors
 - Supplied in retail packaging with driver CD
 - Communication rates from 300 baud to 1M baud
 - 10cm cable length (1m version available at \$24.98)

VDIP1

\$28.32



- The VDIP1 module is an MCU to embedded USB host controller VNC1L
- VDIP1 is supplied with pre-loaded Vinculum VDAC firmware
- USB 'A' type socket to interface with USB peripheral devices
- Single 5V supply input
- Jumper selectable UART, parallel FIFO or SPI MCU interfaces
- Auxiliary 3.3 V/ 200mA power output to external logic
- Designed to fit into a 24 pin DIP socket
- VDIP1 Pb-free and RoHS compliant

UC232R-10 Economy USB RS232 Cable

\$21.86

10cm version



- Matt finish, nicely sculpted white plastic enclosure
- Supplied loose packed in an anti-static bag
- Wide range of drivers downloadable from FTDI web site
- Communication rates from 300 baud to 230k baud
- 10cm cable length (1m available to special request only)

Future Technology Devices International Limited

7235 NW Evergreen Parkway Suite 600

Hillsboro, OR 97124-5803

USA

Tel: +1 (503) 547-0988

E-Mail : us.shop@ftdichip.com

Web : www.ftdichip.com



Volume Discounts Available

Over 50 different USB Interface products in stock at www.ftdichip.com/chipshop.htm

Build a PHP Components Database

Keith introduces you to the LAMP software suite and walks you through your first PHP project: a program for logging and tracking your parts. Now you can use your favorite browser to access your programs without having to generate a GUI for new applications.

Do you waste a lot of time looking for components that you purchased for a project some time ago but are needed now? This happens more than I care to admit, so I did something about it. A good starting point was my “log book” program for the books that I’ve read. (The data for that program was carried forward from a Word Perfect document, which I started years ago.) It did not take long to transmogrify that application to take over the functions of another database, written in Paradox, where I stored the data that described IC storage locations. In fact, I had the system useable in less than 6 hours. In doing so, the utility of the Paradox database was greatly surpassed and user-friendliness

was brought up at least a notch. Much of the gain can be attributed to the software tools that I used (the last three components from the LAMP suite of programs). Refer to the LAMP sidebar for more information.

In this article, I hope to show you how versatile the PHP language is. One key aspect to the language is that you can use your favorite browser to access your programs and you don’t have to generate a new GUI for every new application. Refer to the PHP Primer sidebar for more information.

SEARCHING

The initial screen contains a form that is used to enter search data. The

same page can also be used to enter data for a new part. This second function was added when it was realized that searching and entering new parts were the two most often used features.

A simple example is if the inventory of a drawer needs to be checked, then you simply enter that drawer’s name in the Location input box to get a list of all the expected components. If data is entered into more than one input area, then the conditions are ANDed together. This way, any combination of attributes, such as all 20-pin DIP packages, can be sought.

Once search criteria are entered and the request has been made to look for matching data by clicking on the

LAMP

There may still be people in the electronics world who don’t realize how powerful these four tools are. Many have worked with Linux, perhaps on a design workstation or using gcc to compile code for an embedded processor, but many of the same people have not been exposed to the others.

Apache is perhaps the most successful open-source project ever, even more than Linux. It is, simply put, an HTTP server. Whenever a web page is requested by a browser, the request is fulfilled by this type of program. Web-wide, the server is most likely to be a version of Apache.

PHP is a full-capability interpreted language, just as Python and Perl. (Sometimes the “P” in LAMP is said to be for Perl.) The closest language to it is Microsoft’s proprietary ASP. PHP was specifically developed to work hand in hand with HTML. A source file typically consists of sections of pure HTML, sections of PHP, and portions where PHP writes HTML code, which changes providing dynamic content.

MySQL is not an open-source database management

software system (DBMS), but for most intents and purposes, it might as well be. It is not the only DBMS that is supported in the Apache/PHP environment, but it does claim to be faster and easier to use, and it is probably the most popular. Interestingly, for non-Windows platforms, the MySQL server can be used freely, unless you sell software or services using it. For Windows, the server requires a license; however, older versions are GPL’d. I’m using version 3.23 of the server on an old Windows 98 SE system without worrying about doing anything illegal. (The current MySQL version is at least 5.0.)

Apache, PHP, and MySQL can be installed on most Windows versions, although my Vista experience so far indicates some subtle problems. Some versions of Windows support either Microsoft IIS or a simpler HTTP server, “Personal Web Server.” Either can host PHP pages, which in turn can access MySQL tables, just as is done under Apache. (I wonder: If Linux plus Apache plus MySQL plus PHP is called LAMP, then what should Windows plus IIS plus MySQL plus PHP be called?)



Go Wireless Go Rabbit®

Check out our new Wi-Fi and ZigBee® core modules — the latest additions to our pin-compatible family.

RCM4510W
ZigBee RabbitCore®

\$72 (qty. 100)

Wireless Development Kits

Only \$199 (reg. \$299)

RCM4400W
Wi-Fi RabbitCore

\$99 (qty. 100)



Applications

- Industrial Control
- Remote Terminal Unit (RTU)
- Building Automation
- Data Acquisition

RCM4500W Family

- ZigBee/802.15.4
- 49 GPIO
- 6 Serial Ports

RCM4400W Family

- Wi-Fi/802.11
- 35 GPIO
- 6 Serial Ports



2900 Spafford Street, Davis, CA 95618 Tel 530.757.8400

Order Your Kit Today

Development kits include everything to start your development, Dynamic C® software, wireless RabbitCore module, development board, programming cables and accessories.

www.rabbitkits-wireless.com

Register by July 24th for best conference rates.
www.embedded.com
Use Priority Code: U120

WITNESS THE POWER OF BRAINPOWER

WHEN BRAINS COLLIDE

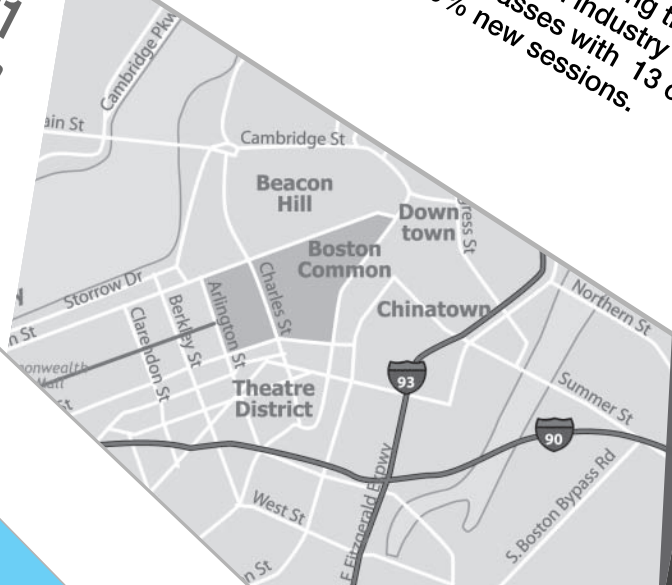
**Embedded
Systems CONFERENCE
BOSTON**

HYNES CONVENTION CENTER, BOSTON, MA
SEPTEMBER 18-21
www.embedded.com

Be there at ESC. Join the power of brain power in action. These, after all, are engineers. On their own, they tackle the toughest problems. But when these brains collide, any challenge confronting humankind can be taken down.

At ESC Boston, you can put your brains against the best in the world. Including 180 leading exhibitors showcasing the latest technologies. Learn from industry experts in 90 in-depth design classes with 13 class tracks and over 60% new sessions.

*Is this a must-attend event?
Let's just file that under no-brainer.*



CMP
United Business Media

Search button, three possibilities exist: no matches, one match, or many matches (see Photo 1). In the first case, the search form is presented again with a notification of what happened. If one match is found, then all the data that exists for that part number is filled into a form, which is laid out exactly like the search form. If multiple matches are found, a summary table is presented and filled in with the most important fields (see Photo 2). In all three cases, a human-readable version of the database query is also given, so the search criteria can be kept in mind.

The table produced in the third case has, in its last column, a link for each row. That enables the component to be selected for editing or viewing. When one of these is clicked, the completely filled out field set for the part

is produced, as if only one match had been found in the initial search.

The program also enables the user to enter a new part, list all of the parts, summarize the parts (total number of parts, total number of entries and summaries per location, and per classification), and edit and delete a record. A crude import—from a comma-separated value (CSV) format file—facility exists, but the PHP program has to be edited to define the relationship between the fields in the CSV file and the fields in the database. (Instructions for this are included in the comments.) So, aside from an export facility, pretty much anything that one would expect is here.

PROGRAM STRUCTURE

All of this functionality was created



Photo 1—This screenshot shows the start-up condition with one input field filled in. A search for all components manufactured by Texas Instruments will be started when the Look button is clicked. Options near the bottom of the window determine what happens if multiple part numbers are returned. In this case, the list will be sorted in part number order, and of the optional fields, only the “description” field will be displayed.

with little new code. The basis was already there: the overall structure of

PHP Primer

Every reader of this magazine has probably noticed that more and more web pages do not have the .html file suffix these days. Many now have .php instead. This is no trivial observation; it is an indication of just how useful PHP is.

The letters PHP don’t mean what they used to. In 1994, the letters stood for “personal home page.” Now they stand for “PHP: Hypertext preprocessor.”

New versions come out regularly. In the summer of 2004, a major revision to version 5 came out. However, many installations have not upgraded, and versions 4.3 and 4.4 are still prevalent. It is almost like there are two separate languages now since Version 4 has not been abandoned. Releases since 4.3.7 (up to 4.4.4 and counting as of August 2006) have come out since the time of the Version-5 fork.

PHP has all the usual elements of a modern language and borrows much of its syntax from them. All of the control structures exist (i.e., for(;;){}, while(){}, do{}while(), switch(){ case():{} case():{} }, as well as a new one (to me): foreach();).

Variables (signified by an initial “\$”) can be Boolean, integer, float, string, arrays, objects, or a couple of special types, including resource or NULL. Constants and functions are used as usual. Comments can be in either C style (/* ... */), C++ style (//), or Perl style (#). Statements are terminated by a semicolon and can be grouped in {} brackets.

PHP is a weakly typed language. Conversion from one type to another happens when possible. It is quite legal to add “3” to 3. This produces an integer with the value 6. Conversion even happens when one might not think it would. The calculation “2 cows” × 3 returns the same integer, 6!

When the HTTP server receives a request for a file with the PHP file extension, this kicks off the interpreter. If there is no actual PHP code in the file, all the contents of the file will be sent to the browser exactly as if it was an HTML file. This also means that anything that is not in a PHP section is similarly sent unmodified. A PHP section is identified to the interpreter by a “<?php some code ?>” structure. Anything that matches the “some code” portion of this syntax is expected to be PHP. It can be one statement, a whole program, or anything in between. If any output is produced by the PHP section, the string appears in the data that the browser receives in the exact place that the section appears in the original code.

Here is a simple example:

```
<input type='text' name='part_type' value='<?php echo $part_value;?>'>
```

This produces something like the following for the browser to interpret as a filled-in form field. (Note that the escape to PHP even works inside the quotes!)

```
<input type='text' name='part_type' value='74LS00'>
```

One important concept in PHP is that double quotes and single quotes do different things. A string that is enclosed with double quotes has the extra property of allowing variables within it. The values of these variables replace their name. So, another way of doing the same thing as above, from within a PHP code section, is:

```
echo "<input type='text' name='part_type' value='$part_value'>"
```

| PLD Designs Electronics Components Data Base. | | | | | | |
|---|-----|------|---------|----------|---------|-------------------------------|
| Searched for: manufacturer to match "TI" sorted by Part Number. | | | | | | |
| Part Number | Qty | Type | Made by | Bin # | Package | Description |
| LM111MTR | 2 | IC | TI | 2201-box | PM | adjustable regulator 0.5 A |
| MC1458 | 9 | IC | TI | LN4 | POP-8 | Dual gen purp op-amp * |
| OPA240UA | 9 | IC | TI | 8006-box | SO-8 | Dual nr op amp |
| OPA340UA | 2 | IC | TI | 8006-box | SO-8 | single nr op amp |
| REF104C-2.5 | 2 | IC | TI | 8006-box | SO-8 | 2.5V 0.4%, 20pinP reference |
| TL082CP | 1 | IC | TI | LN4 | POP-8 | Wide BW dual JFET Op Amp * |
| TL084 | 2 | IC | TI | LN4 | POP-14 | Quad wide BW op amp (LF347) * |
| TL497ACN | 1 | IC | TI | REG-1 | OP-14 | 5 A, +0.0% of switch reg |

Number of part numbers found: 8, total parts quantity: 28
 for parts, a new part, all parts, the data

Photo 2—This is the return from the request in Photo 1. It shows that I have 28 ICs of eight types in stock that have the manufacturer field filled in with the letters "TI." It shows where I can find each type, for instance the last remaining TL082CP is the "LIN-1" bin.

the program, the code that formatted the two versions of the form (filled in and blank), and the code that created the summary tables. Sure, most code had to be modified to work with the different field names, but that did not take much time.

Besides the common start-up and completion code, the program consists of one big switch statement, which decodes the possible values of the \$mode variable (see Listing 1). The variable must be "sent in" to the program every time it is run. How this is done tells a lot about how PHP (and any other languages like ASP) runs. There are four ways that a variable can be passed into such a program: GET, POST, Cookies, and Sessions. The first two normally originate in an HTML form, which the user has filled out. The method, either get or post, must be specified in the <form> tag's action parameter. From a practical point of view, there is not much difference between these two, except get variables are tacked onto the end of the URL as in EComponents.php?mode=list.

Posted variables are handled differently. (They are a bit more secure in that they cannot be casually edited by the user, but that won't stop somebody who knows what he is doing from changing them.) Reloading the variables will seem strange if you are used to writing application programs that run until they are told to exit, waiting for user input between bursts of doing something. In the world of web programming, however, you have to get used to the fact that your server-based program is run afresh each and every time an action is taken. By the time the page has finished loading

into the browser the program has stopped, so the programmer has no choice but to start everything over to carry out a new action. What is run to produce the next page might be the same program operating differently, as I do here, or a totally different program designed to perform the specific task. This is specified to the HTTP server as either an action='...' value for an HTML form or an href='...' value for a link.

Some of the facilities need two sections (i.e., two values for the \$mode variable) to implement. An example is Edit. The code in the Edit section retrieves all the data corresponding to the record that will be edited and fills out a form with that data. Any changed values are processed by Submit mode, which is entered when the

Submit button is clicked.

Chunks of code are included using the PHP include() built-in functions. This technique is used to make the program easier to read. Most "commands," in other words, the actions for the different values of \$mode, are implemented this way. The names of these files all start with do_ (e.g., do_edit.php). The part after the underscore matches the value of \$mode. This naming convention allows the actions to be easily identified and maintained.

It may not be obvious what one little function and a Boolean variant of it do. That's the var_set(\$string) function. In some earlier versions of PHP, this would not have been needed; however, a security consideration has changed how most PHP installations work now. Previously, it was possible to use a variable's name and expect

Listing 1—The program's structure is one big case (switch) statement. Here are the first four matched conditions and the default, which would catch any erroneous values of the \$mode variable. The first two share some code: the difference is determined within the do_search.php code, which adds the extra capability to enter a new part for the initial \$mode="" case.

```
// The start of the action-specific code: //
switch ($mode):
case "":
case "Search":
    //present a form to fill out: (page display is slightly different depending on value of $mode)
    include('do_search.php'); //the blank form, mostly menu();
    break;
case "Enter": //accept data to define a new component
    echo "
<h3>New: fill in the form with data for a new component...</h3>
<form action='index.php' method='post'>;
    disp_fields(false);
    echo "
<input style='font-size:20px;font-weight:bold;color:red' type
    ='submit' name='mode' value='File' />
    (Put the data in the data base)
    ";
    echo "
</form>";
    menu();
    break;
case "File": //put the data away.
    include('do_file.php');
    menu();
    break;
... (rest of the modes) ...
default:
    echo "
<h3 style='color:red;'>Invalid mode! ($mode)</h3>
<span style='color:red;margin-left:10em;font-weight:bold;'>&gt;&gt;&gt;&gt;&gt;&gt;
    <a href='\$PHP_SELF' style='color:red;text-decoration:none;background-color:white;'>Re-start</a>
</span>";
    menu();
endswitch;
```

that if it had been sent into the program (by GET, POST, Cookie, or Session means) and it would be defined. This accepting mannerism allowed attackers to guess which variables might be used and override them. Now, variables have to be explicitly specified to be brought in.

MySQL CALLS AND QUERIES

The interface between PHP and MySQL consists, not surprisingly, of a set of function calls. The program uses the following: `mysql_connect()`, `mysql_select_db()`, `mysql_query()`, `mysql_error()`, `mysql_fetch_row()`, `mysql_fetch_array()`, `mysql_num_fields()`, `mysql_num_rows()`, `mysql_errno()`, and `mysql_field_name()`. This represents only 10 of the 50 available functions. Some of the remaining are

involved with more advanced features and more are simply variations of others.

The first two, `mysql_connect()` and `mysql_select_db()`, are responsible for establishing where the database is and logging onto it. They are run each and every time the program is run (via my own little function, `auth_sql()`). If the database can be opened, a check for the table specified by the `$table1` variable is made. If the table does not exist, an attempt is made to create it. This is mostly needed just after installation; however, it also allows the same program to handle multiple tables as long as they share the same structure.

The next function in the list, `mysql_query()`, returns some data from the database or stores some new data into it. (Or, if there is trouble, it

indicates an error. If that is the case, `mysql_error()` can be used to fetch the error message.) A query is a string that has been preassembled to be a valid SQL command. It specifies a table within the database, what data in what fields to look for, and what to do with them. Much of the work of this program is assembling these strings (see Listing 2). A query can be written to do most anything that is possible to do with the data or the table itself.

Data returned by `mysql_query()` is not in a form that can be used immediately. That is where the rest of the functions in my short list above are used. They all take what is returned by `mysql_query()` as an argument. The classification of what is returned by `mysql_query()` in PHP is called a resource. Resources are also seen, for example, when a file is opened. In other words, an open file's handle is also a resource.

As an example of a query, consider the following string:

```
select * from Parts where (pins
LIKE '14' AND pkg LIKE 'so' AND
class LIKE 'I' AND location LIKE
'cmos-1').
```

A more English-like translation is displayed by the program:

```
pin count to match "14" AND
package to match "so" AND clas-
sification to match "I" AND bin
to match "cmos-1"
```

Both of these strings would be produced by the code in Listing 2 if 14 was entered in the pin count input box, "so" was entered in the package input box, "IC" was selected in the classification drop-down box, and "cmos-1" was entered in the location input box. (Note that the raw queries, as well as a lot of other debug information, can be seen if the `$dbg` value is changed to `true` from `false`. This variable can be found right near the start of `index.php`.)

SOME FINER POINTS

As you can see in Listing 3, a good example of how PHP writes HTML can

Listing 2—This is how you build up a complex database query. The `foreach` loop traverses through the `$_REQUEST` array, which is provided by PHP. This array contains all variables, which have been passed into the program, whether through `get`, `post`, `cookie` or `session` means. Here you are only looking for ones that exist in the `$titleArray` variable, which is used to identify names of fields in the database. The two strings built up are `$qual` and `$summary`. The first is used within the query to specify what fields need to match what values, and the second is a more English-like version to show to the user. The loop also looks for superfluous table headings. Flags such as `$loc_no_wild` are set to `true` if the associated heading and column are removed by the software that builds the table.

```
$qual = '('; $add = ' '; $summary = '';
$class_no_wild = false; $qty_no_wild = false; $loc_no_wild = false;
foreach ($_REQUEST as $key => $value){
    $_REQUEST[$key] = trim(strip_tags($_REQUEST[$key]));
    if ($value != ""){ //skip any blank categories
        if (array_key_exists($key,$titleArray)) { //skip any that are not fields
            //in database
            if (!(($key=='class' AND $value=='X')){ //look up the translated
                //classification names
                $qual.="add $key LIKE '$value'";
                $summary .= "add {$titleArray[$key][3]} to match \"$value\" ";
                $add = " AND";
            }
            // The next two ifs set flags to delete columns if no wild
            // card has been specified (they do the job partially so far).
            // It is rather useless to search for only transistors and
            // then be told, in each row, that the item is a transistor!
            if ($key=='class' AND $value!='X'){
                $class_no_wild = true;
            }
            // These next two are only handling '%' wild cards right
            // now. Ideally want any non-escaped '_' or '%' to be caught!
            // Would likely have to use preg_match() to do this fully...
            if ($key=='qty' AND !$value==" AND (false==strpos($value,'%')) ){
                $qty_no_wild = true;
            }
            if ($key=='location' AND !$value==" AND (false==strpos( $value,'%')) ){
                $loc_no_wild = true;
            }
        }
    }
}
$qual .= ')';
$query="select * from $table1 where $qual";
if ($dbg) echo "Query (look 1):". $query."<br />";
echo "<h3>Searched for: $summary";
$result = mysql_query($query)
```

be seen in the code that creates a drop-down list, which allows you to select the component classification (IC, resistor, capacitor, etc.). The application uses the technique of cascading style sheets (CSS) to define the presentation. CSS is a relatively new methodology. As well as encouraging separation of content from how that content is displayed, it allows a style to be defined, which can be used throughout a page or set of pages.

When it comes to the creation and changing of the multiple files used in an application like this, it helps if a PHP-aware editor is used. There are many others, but I find that PSpad does the job. When set up to interpret what it calls "XHTML multihighlighter" type files, it differentiates between the PHP and the HTML sections, highlighting each one's keywords appropriately.

INSTALLATION

If you have a web server up and running, and if you have access to MySQL on that server or another one, then you have to decide where you want the code to reside under the HTML root directory and copy it there. Plunk my files down there, keeping the Include directory as a subdirectory. Then, decide where the database will live and change the values of \$user, \$host, \$password, and \$database (it can be called anything, I call it parts) in the index.php file so that it is pointed to properly and can be logged into.

If you don't have an HTTP server available, you have many choices. If you have a LAN-connected PC, you should be able to install one of the packages that exist, such as EasyPHP. This sort of package bundles Apache, PHP, and MySQL together into one installer. I got my start with the CD that came with a certain "for dummies" book, and that has been fine. (I have since upgraded the version of PHP.) If you want to do it yourself by downloading the modules from each supplier's site, the install order is Apache, PHP (this step includes editing Apache's httpd.conf and PHP's php.ini files), then MySQL, testing each component as you go. Search

Listing 3—This short piece of code has a lot of power. These few lines traverse through the array \$classArray, which is a set of key => value pairs (e.g., "Q"=>"transistor"), and creates a drop-down list on the HTML page. This allows the user to select the category of the component being looked at or entered.

```
echo "<select name='class'>";
foreach ($classArray as $key => $value){
    $sel = ($class == $key)? " selected='selected'":"";
    echo "\n    <option value='$key'{$sel}>$value</option>";
}
echo '</select><br />';
```

Listing 4—The required procedure to create an empty database in MySQL is quite simple. You are just required to be in the right directory. The writing between the square brackets are comments, and they must not be typed.

```
cd C:\mysql\bin [assuming that mysql was installed in root]
mysql -h hostname -u root -p [the password will be prompted for]
mysql> CREATE DATABASE parts; [note the uppercase and the semicolon]
mysql> exit
```

the Internet, there is a lot of help available.

Once all the tools are available, the database has to be created. If you have not used MySQL before, this will be the hardest step. There are several ways to do it. The easiest way is to tell it to use the MySQL command, which should exist as c:\mysql\bin\mysql.exe (Windows) or /usr/local/mysql/bin (Linux). In Windows (DOS), this would look like the code in Listing 4. Once the database is created, the components database program can be run to create the necessary table.

THE PRICE YOU PAY

Most of us are rightly suspicious when something is free. The phrase to remember with free software, however, is "free as in freedom, and not free as in beer." Totally free software (as in beer) would come preinstalled, work properly from the outset, and have a zero-time learning curve (and would run on a \$1 pocket calculator, I suppose). Instead, free software takes time to install, requires hardware resources to run on, and takes some time to learn how to use it and even more time to use it effectively—but this can still be better than the alternative. (In this comparison, the only difference between what you pay for and the free stuff is that you pay for the other!)

I have attempted to show how useful the LAMP suite of software is by producing a practical application. And

the "L" isn't even required! I may make a much-improved version of the program available as shareware, or even buyware, in the near future. In the meantime, keep an eye on my site (www.island.net/~kdbrown) for updates beyond the version posted on the *Circuit Cellar* FTP site. ☐

Keith Brown, P.Eng., lives on Vancouver Island, B.C. He has been operating PLD Designs, consulting in the selection of PLDs and designing circuit boards for customers (with and without PLDs) since leaving Spar Aerospace in 1992. During the last few years, he has been adding web programming techniques to his toolbox.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2007/205.

RESOURCES

E. Faivre, "Easy PHP," 2003, www.easyphp.org/?lang=en.

J. Fiala, "PSPad," 2007, www.pspad.com/en/.

G. Papayiannis, "How to: Install Apache, PHP, Pear, MySQL, and PHP-MyAdmin for Windows XP," 2006, www.sematopia.com/?p=28.

The PHP Group, 2007, www.php.net.



We search the world for time-saving, easy-to-use state-of-the-art products!

Testing

Color LCD Scope



Owon - 25MHz or 60MHz 2-ch + trigger standalone USB bench scope with 7.8" color LCD. Also battery models!
PDS5022S (25MHz) \$349
PDS6062T (60MHz) \$599

Handheld Scope



Owon - 20MHz or 60MHz 2-ch standalone USB scope with 3.8" color LCD. Built-in meter - great for your tool kit.
HDS1022M (20MHz) \$529
HDS2062M (60MHz) \$799

Mixed-Signal PC Oscilloscope



100 MHz Scope and Logic Analyzer lets you do complex triggering to find hard-to-get-at glitches, spikes, etc. Huge 4 or 8 MS buffer for deep data drilling and zooming. Optional built-in swept signal generator. 2 Analog channels x 10, 12, 14 bit with more than 60 dB dynamic range. 8 digital inputs for mixed signal display/trigger.
CS328A-4 (4MS Buffer) \$1149
CS328A-8 (8MS Buffer) \$1642
CS700A (signal generator) \$249

Pen Scope



PS2105 - 25MHz USB powered scope-in-a-probe! Up to 100MS/s, 24kS buffer C/VB/ Delphi/LabView/VEE drivers.
PS2105 (25MHz, 100MS/s) \$366
PS2104 (10MHz, 50MS/s) \$229

Logic Analyzer



ZeroPlus - Intuitive full-featured 16-channel with 4mb memory. Sophisticated triggering. I²C, SPI, and UART monitoring. Includes clipleads, USB cable and software.
LAP-16128U \$399

I2C

USB to I2C



Devasys - Provides a simple "drop-in" solution for connecting your PC to 90kHz I²C + 20 I/O lines. Free software. Use multiple boards for more I²C/I/O.
USB I2C/I/O \$79

Thousands of
products with more
arriving every day!

Please visit
www.saelig.com
to see the latest
additions.

Ethernet to I2C



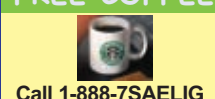
Fiveco - TCP/IP server with I2C, 19 x digital I/O, 5 x 10-bit A/D inputs, and UART serial interface. Connect I2C devices/sensors to an Ethernet network!
FMod-TCP from \$250

I2C for PCs



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

FREE COFFEE



Call 1-888-7SAELIG
Mention Offer# SBW
to get a free
Starbucks Card
with your order!
While supplies last -
not available with any other offers

Ethernet to I2C



NM7010A-LF REV 2 - Complete, hardwired TCP/IP-I2C module. Offloads stack for hi-speed Internet-enabling microprocessor systems. Drop-in W3100A-based board for instant network speeds to 8Mbps. \$22

I/O

Stepper Motor



Easy-Step 3000 - Advanced stepper motor drive and control system for unipolar stepper motors up to 35V/3A - ideal for robots/industrial control - turns you into an instant motor control expert! from \$84

Touchpanel Controller



CUTOUCH - 320 x 240 Blue Graphic LCD Touchpanel/controller. CUBLOC Program in BASIC and/or LadderLogic. 82 I/Os and 2 RS232 MODBUS support.
CT1720 \$359

USB to 24 x I/O



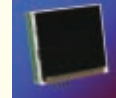
Elaxol - 24 latched I/O lines self-powered from USB port. Ubicom SX52 allows individual pin I/O programming. I/O control via PCs USB port.
USB/O24R \$69

FATfile Storage



Micro ALFAT-SD - Easily add SD storage to your product. Serial data in - FATfile data out.
uALFAT-SD \$40
uALFAT IC \$14

Easy LCD



Micro LCD - Easy way to add 2" sq 128 x 128 color LCD to your project. Add text/graphics/animations for any micro with serial or USB.
Micro LCD MK II \$81
Micro LCD-OLED \$99

Touch-Sensing ICs



Quantum ICs - World's best capacitive sensor ICs for switching & control. Patented Adj. Key Suppression, Spread-spectrum. In use worldwide by the world's largest consumer/appliance mfrs from \$1

Ethernet

Serial-Ethernet Cable



Network your serial product easily without a PC using the 28" cable. Provides an instant IP address. RS-232 port compatible with most 10/100Mbps Ethernet hubs.
eCOV-110-P \$99

Ethernet to Serial



WIZnet - Ready-to-go serial-to-Ethernet Gateway. Simple set-up for 10/100Mbps Ethernet - 230kpbs serial. Ethernet/Internet-enable any serial port.
EG-SR-7150MJ \$34

Ether-I/O



Ether-I/O - UDP/IP-controlled 24 X digital I/O board. 3 x 8-bit TTL ports each independently programmable. Connects to any TCP/IP Ethernet network.
Ether-I/O 24 \$109

Mini Web Server



PicoServ™ Miniature Web Server allows remote I/O and servicing. Collect analog & digital data over a web connection. Remote sensing/control, SCADA, building control.
PP1001-V00 \$197

WIZnet TCP/IP IC



10/100 Ethernet controller on a chip! Fully hardwired TCP/IP core. No OS required. TCP offload improves system performance.
W5100 (integrated PHY) \$6.95
NM7010B+ Eval \$22

Motion Control



PIC-Servo - Motion control boards with integrated power amplifiers for DC-servo/brushless-servo/stepper-motors. Easy connection to standard serial ports for powerful multi-axis, mixed-motor control. from \$160

RF

Bluetooth



Emmys PearlBlue - 100m-Bluetooth module (UART master/slave bridge). Easy connection for micro's to PCs, PDAs, phones, etc. **microIceBlue** - PIC16F877 based emb. dev. system for shortest Bluetooth development. \$99

3ALogics



RFID Reader Chip - Reads FIVE International Standards ISO14443 A/B, Tag-IT, Felica and ISO15693. Single-chip solution.
TRH031M \$6.95
RSK-100 Eval Kit \$350

Transceiver Modules



LPRS easyRADIO™ makes wireless data transmission simple for US and Europe! Add wireless capability to your project today!
ER900TRS from \$38 (100)

Wireless Controllers



GSM/GPRS/GPS/Bluetooth/Ethernet wireless controllers for M2M - for autos, machines, etc. RS232/485/CAN/ A/D-I/O.
Owasys from \$441

TinyOne Zigbee™



Complete RF solution for fast & reliable data transmission. Miniaturized OEM RF module ensures fast and easy integration into wireless applications.
TinyOne Zigbee \$55
B915FH-tiny/40-SMD-WA \$61

Data Modem



USB-powered radio transceiver boards add wireless RS232 to your device! Get remote control for your product from PC USB port!
RTM-USA \$89

USB

USB in one IC



FT232RL - FTDI USB1.1-serial IC with new "security-dongle" feature. Each unique ID and Internal EEPROM,
FT232RL \$2.45 (1K)
FT232BL \$2.93 (1K)
 Check out the whole FTDI line!

FT232R Module



UM232R - FT232RL 24-pin DIP module - USB-TTL serial converter for your project. Reference circuit for FT232R. Plug-in module adds USB1.1 instantly!
UM232R \$19.99

USBwiz



USBwiz - Easily add USB peripherals to your system. Includes FATfile system for USB thumb & external USB hard drives. No USB knowledge needed. Plug-n-play!
IC \$29 Demoboard \$59

Easy USB



microUSB - Tiny ready-made USB Interface to 1 Mbps 0.7" x 0.5" based on SiLabs CP2101. Gnd/RX/TX Suspend, Reset, & 3.3V 100mA source.
microUSB \$25

GPS Logger



TrackStick II - Pocket-sized logger tracks position data for vehicles, personnel, etc. Records date, time, location, route and speed. Shows data on Google maps!
Super TrackStick II \$199
TrackStick II \$239

USB-Serial Adapter



FTDI - based serial port laptops. Instantly adapts older RS232 products to USB! Included drivers provide virtual COM port.
USB-COM-S (45") \$24
USB-COM (13") \$29

www.saelig.com

1-888-772-3544

1-888-7SAELIG

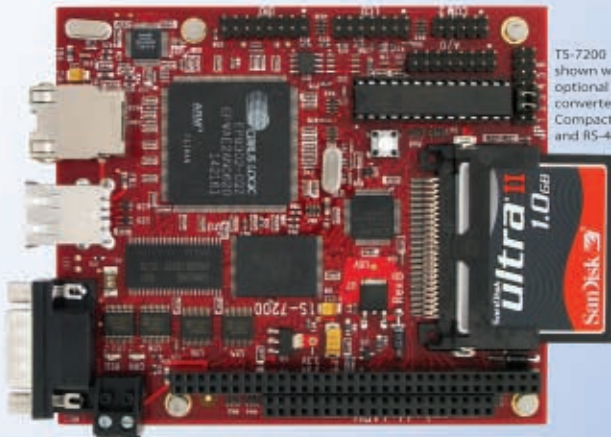


info@saelig.com

585-385-1768 fax

PC/104 Single Board Computers

Low Price, Low Power, High Reliability
using Linux development tools



TS-7200 shown with optional A/D converter, Compact Flash and RS-485

- options include:
onboard temperature sensor, A/D Converter 8 channel 12 bit, Extended Temperature, Battery Backed Real Time Clock, USB Flash 256 M (with ARM Tool Chain), USB WiFi

200 MHz ARM9
Power as low as 1/4 Watt

- 5 boards, over 2000 configurations as low as
\$99
qty 100
- Fanless, no heat sink **\$129**
qty 1
- SDRAM - up to 128MB
- Flash - up to 128MB onboard
- 10/100 Ethernet - up to 2
- DIO lines - up to 55
- 2 USB ports
- COM ports- up to 10
- Programmable FPGAs
- Linux, Real Time extension, NetBSD
- SD card option
- VGA video

Off-the-Shelf Solutions ready to design into
your project using DOS development tools



TS-5600 Shown with optional flash modules, A/D, RS-485 and Merlin cellular modem

- options include:
RS-485 Half and Full Duplex, A/D Converter up to 8 Channels at 12 bits, DAC up to 2 Channels at 12 bits, Extended Temperature

133 MHz 586

- 5 boards in series as low as
\$229
qty 100
- Power as low as 800mA **\$259**
qty 1
- Fanless, no heat sink
- SDRAM - up to 64MB
- COM Ports - up to 4 ports
- Ethernet Ports
- DIO Channels - up to 40
- PCMCIA II adaptor
- Compact Flash adaptor
- USB Ports (Except on TS-5300)

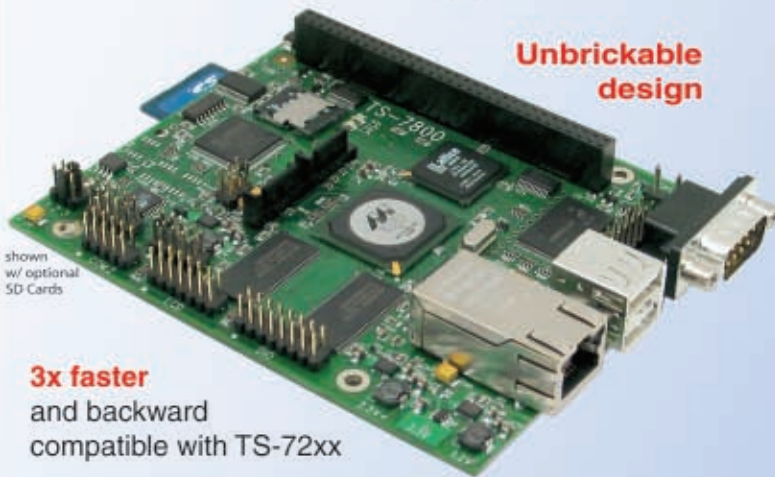
- Over 20 years in business
- Open Source Vision
- Never discontinued a product
- Engineers on Tech Support
- Custom configurations and designs w/ excellent pricing and turn-around time
- Most products stocked and available for next day shipping

Design your solution with one of our engineers (480) 837-5200

New Products and PC/104 Peripherals

NEW!

High-End Performance
with Embedded Ruggedness



shown
w/ optional
SD Cards

**Unbrickable
design**

3x faster
and backward
compatible with TS-72xx

TS-7800 500 MHz ARM9

- Low power - 4W@5V **\$229**
qty 100
- 128MB DDR RAM
- 512MB high-speed (17MB/sec) onboard Flash **\$269**
qty 1
- 12,000 LUT user-programmable FPGA
- Internal PCI Bus, PC/104 connector
- 2 USB 2.0 480 Mbps
- Gigabit ethernet
- 2 SD sockets
- 10 serial ports
- 110 GPIO
- 5 10-bit ADC
- 2 SATA ports
- Sleep mode uses 200 microamps
- Boots Linux in < 2 seconds
- Linux 2.6 and Debian by default

NEW!

TS-POE100 Power-Over-Ethernet

- 2.4 Amp 5V isolated power (12W)
- PC/104 peripheral w/ 10/100 Ethernet
- Sleep Mode & Wake-on-LAN with PoE



\$99
qty 1

board shown
with optional
16 bit PC/104
connectr and
2nd RS-485

NEW!

TS-PLC Programmable Logic Controller

- Use standalone or cluster via wireless or RS-485
- 7 analog/digital I/Os
- 4.7-30VDC or USB power input
- opt. 500 meter wireless
- Program in ladder logic or C
- Rugged aluminum enclosure 3.6" x 3.1" x 1.2"
- Web interface

\$89
qty 1



see our website for more boards and option details



Technologic
SYSTEMS

We use our stuff.

Visit our TS-7200 powered website at
www.embeddedARM.com

1-Wire in the Real World (Part 1)

The Challenges

Steve explains the challenges of taking a 1-Wire deicing system for small aircraft from a laboratory prototype to a real-world application. The ThermaWing Deice System relies on a proprietary algorithm requiring tight control of surface temperatures to enable the system to melt ice.

As a general aviation enthusiast, certified flight instructor, and former U.S. Air Force flight instructor, I'm always happy to work on a project that combines my two favorite fields. Many years ago, I was in an aircraft accident when the small aircraft I was flying was brought down by in-flight structural icing. Now I work as an independent electrical engineering consultant. Thus, I jumped at the request from what is now Kelly Aerospace Thermal Systems to take a deicing system for small aircraft from a laboratory prototype into the real world. Photo 1 shows a sneak peek at the ThermaWing Deice System. A clean wing is in the protected area, while there is significant ice accumulation on the unprotected landing light lens.

It quickly became clear that the electric deicing system required some form of distributed switching to spread out the thermal load of the switches. The heat resulting from rapidly switching 7,500 W among six to 12 heaters or more is quite substantial. The laboratory prototype attempted to do all of the switching in one box, but a brief test simulating a real-world operation ended with the heatsinks hotter than a Texas tarmac in the summertime.

The power switches were big, insulated-gate bipolar transistors (IGBT) because mechanical relays lacked ruggedness when they were switched rapidly. The system requirements mandate that the heaters repeatedly switch at intervals ranging from a few tens

of milliseconds to a few seconds. Relays would clearly be a bad substitute for the IGBTs. But, if we could move each individual IGBT out next to the heater controls, we could distribute the heat. That would also reduce the amount of heavy power wiring, and weight is expensive in any aircraft installation. More importantly, we could dissipate the heat into one of the best heatsinks imaginable—an aircraft skin with ice frozen to the other side and below-freezing air blowing past at a couple hundred knots!

WORLD OF BUREAUCRACY

The Federal Aviation Administration (FAA) requires an extensive certification process for every piece of software used. I was already planning a software certification effort for the controller, as spelled

out in "RTCA/DO-178: Software Considerations in Airborne Systems and Equipment Certification," but I had neither the plans nor budget for a parallel effort for another piece of software to run in a remote switch module. Thus, I needed a way to remotely control the switches and monitor the temperature of the heater surface and the switch itself (for diagnostics) via a remote communications link, all without requiring a local processor and the attendant software.

I have considerable experience with Dallas Semiconductor 1-Wire bus and chips, both on a board level in embedded systems and on a longer bus of tens of meters. The 1-Wire protocol and some of the early chips were described in great detail by Jeff Bachiochi in "Dallas 1-Wire Devices" (*Circuit Cellar* 105 and 106, 1999). I have the original Dallas Semiconductor promotional 1-Wire weather station, and I have expanded it to include a number of extra temperature sensors, a barometer, a rain gauge, humidity sensors, and more. Based on my experience with the reliability of 1-Wire in that application, it was with some trepidation that I introduced the idea of using 1-Wire in this application.

APPLICATION

Reliability is clearly required. Turning a 7,500-W heater on when it should be off could do nasty things to an airframe. Less obviously perhaps,



Photo 1—The protected area covered by the gray heater is clean, while the unprotected landing light lens shows the severity of the icing where the test flight took place. (Popsicles fly poorly!) Also note the "nose caps" of ice on the wheel fairings, and the chunks of ice that are falling on the ramp as the unprotected areas of the airframe warm up after landing.

turning it off with high current flowing from a highly inductive source (the alternator) could cause voltage spikes in hundreds of volts. Such voltage spikes could do considerable damage to the electronics or even cause electrical arcing. After switching to my pilot hat, it was clear that reliability would be a very high priority.

Accurately sensing the heater's surface temperature is almost as important as the switching. The system relies on a proprietary algorithm requiring tight control of surface temperatures. The algorithm makes the system shed ice from the aircraft skin without melting a significant quantity of ice, which would become liquid and run back from the leading edges where ice forms, to refreeze in unprotected areas.

Taking a laboratory demonstration out into the real world also requires some serious considerations of protection against voltage spikes and other electrical problems. Although the system specification doesn't require the electronics to survive a direct hit from a lightning bolt, the voltage spike tests in "RTCA/DO-160E: Environmental Conditions and Test Procedures for Airborne Equipment" anticipate a near-miss lightning bolt that induces substantial voltage and current for a brief interval.

TEMP-SENSING SWITCH

A number of 1-Wire chips include some form of internal temperature sensing. I selected a Maxim Integrated Products DS2751 multichemistry battery fuel gauge for the temperature-sensing section of the remote-switch architecture.^[1] The chip includes an internal temperature sensor with 0.125°C resolution, a single-ended voltage-sense input with 4.88-mV resolution from 0 to 4.75 V, and a differential current-sense input intended for use with an external current-sensing resistor over a range of ±64 mV with a resolution of 15.625 μV. The chip also includes 32 bytes of lockable EEPROM that could be used to store configuration information.

Our original intent was to laminate this or a similar 1-Wire chip directly into each heater to allow us to read the heater temperature directly without external analog electronics, and to couple the EEPROM contents with one specific

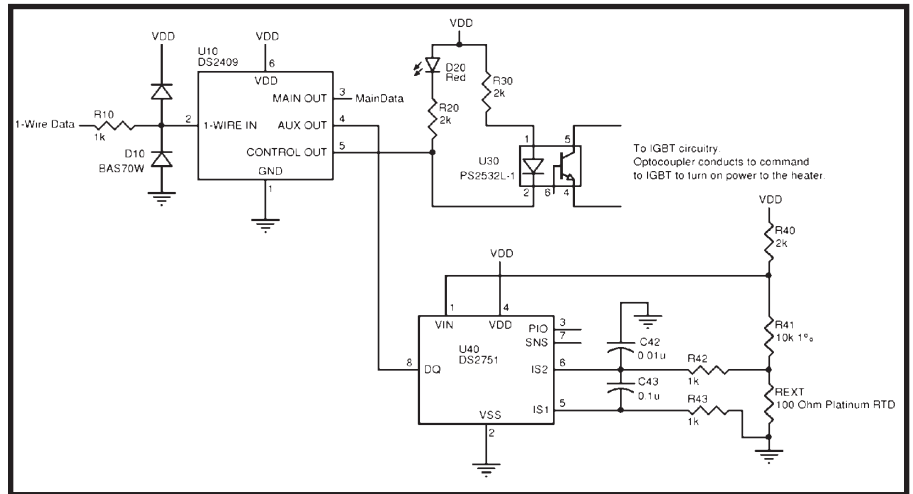


Figure 1—LED D20 provides the visual indication of heater switching for troubleshooting and R10/D10 provide protection from incoming transients. Temperature is sensed via REXT, a 100-Ω platinum RTD. The DS2751 voltage sense input confirms and corrects for current variations, while the current-sense input measures the actual voltage developed across the RTD. Ohm's law yields the RTD resistance and firmware converts the measured resistance to heater temperature.

heater. This tight coupling would also allow us to store the heater's serial number and other characteristics (i.e., total area and nominal resistance) to help adjust applied power and diagnostics. Although this approach was elegant, it exposed the chip and interconnections directly to the hazards of impact damage on the leading edges of airfoil surfaces.

Therefore, I chose to sense temperature using a rugged 100-Ω platinum resistance temperature device (RTD) laminated into the heater, with the chip covering safely inside the potting in the switch enclosure mounted inside the wing. Turning the chip's logic upside down, I chose to use the voltage-sense input to confirm current through the RTD and use the current-sense input to measure the resulting voltage. The V_{DD} of 5 V nominal excites the series string of R40, R41, and the RTD (Rext) (see Figure 1). The RTD resistance can range from 84 to 133 Ω over the temperature range of interest. Thus, the voltage measured at the top of R41 is applied to a total resistance ranging from 10,084 to 10,133 Ω. If the firmware blithely assumes 10,100 Ω and divides the sensed voltage at V_{IN} by this resistance, the computed current will be in error by no more than 0.3% due to the change in RTD resistance. If we need more accuracy, a bit of algebra would correct this error, but the error is so small relative to the other errors

(notably the use of 1% resistors) that it is negligible. However, we can use the V_{IN} voltage to correct for any reasonable variation in V_{DD} . We also could have used a four-wire sensing arrangement to correct for resistance in the wires to the RTD, but the wires are short and the resistances are small and predictable, so we simply allow for that resistance and apply a correction in firmware.

By applying a nominal 5 V to the 12,100-Ω string (2 kΩ plus 10 kΩ plus the 100-Ω RTD), we achieve a nominal current of 413 μA. At the coldest operating point of -40°C, the RTD resistance of 84 Ω produces a sense voltage of 34.7 mV. At the hottest operating point of 85°C, the RTD produces 133 Ω and 55 mV. The 15.625-μV resolution yields 1,300 measurable steps between the endpoints for a resolution of just over 10 steps per degree centigrade. Linearity is good, but not perfect. The system spec, however, requires good accuracy at only a single point (freezing) and allows degraded accuracy as the temperature moves away from freezing. The software provides for a single-point offset calibration to achieve the required accuracy at freezing, with the offset constant stored in the DS2751E's EEPROM. The calibration data is thus tightly coupled to a specific heater and RTD. Note the combination of surge protection and antialias filtering provided by the combination of R42/C42/R43/C43.

To allow for the rapid and repeated reading of a single temperature while

one heater is in operation, I placed a Maxim Integrated Products DS2409 MicroLAN U10 between the main 1-Wire bus and the DS2751E U40.^[2] With all the temperature-sensing sections similarly behind such a coupler, the software can select a single sensor on the bus and repeatedly read the temperature without going through the complete addressing protocol for each reading. This is a useful technique in any 1-Wire design that requires bursts of readings from a specific sensor. More importantly, the DS2409 also provides a single bit of digital output, which the system uses to enable the IGBT driver and turn on the heater. The digital output is an open-drain output that can sink current to ground and drive an optocoupler (see Figure 1). The balance of the circuit beyond the optocoupler is proprietary and includes transient protection and the IGBT driver. Isn't it amazing how this heater power switching, the primary function of the circuit, appears as almost an afterthought to all the ancillary functions, such as temperature sensing?

1-WIRE BUS PROTECTION

Maxim Integrated Products touts the fact that 1-Wire chips can run from "parasite power." That is, each chip includes a diode and enough capacitance to maintain a workable internal supply voltage by stealing power from the data line during its logic-one state. When the data line is high, the current flows through the diode to power the chip and charge the capacitor. When the data line is low, the reserve charge in the capacitor is sufficient to run the chip. The protocol guarantees that the data line should never go low for longer than the "ride-through" of the capacitor, except in the case of a deliberate reset signal. This scheme works well when only a few chips use parasite power and the data line is completely contained within a circuit board or other well-controlled enclosure.

Protection is required to prevent transients from damaging either the master or the slaves where the data line travels through hostile territory. Protection components limit the amount of energy that can transfer on

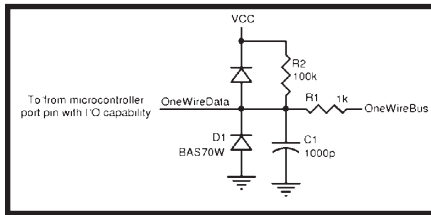


Figure 2—A simple and robust microcontroller to 1-Wire bus interface. R1 limits the available fault current into or out of the microcontroller pin, while dual-diode D1 clamps the pin voltage between the supply rails. R2 provides a very weak bus pull-up when the microcontroller is not providing an active pull-up, such as during microcontroller reset and firmware start-up.

the data line. With multiple slaves attempting to share the data line, reliability suffers. Because this application required regulated power within each enclosure for other purposes, I chose to ignore the parasite-power capability and provided separate power directly to each chip on its dedicated V_{DD} pin.

Having eliminated the requirement to provide parasite power, we can afford more resistance in the data line. The data line just has to supply enough current to charge the cable capacitance plus the chip-input capacitance of 50 to 60 pF per chip during transitions and enough steady current to feed the 1 to 5 μ A of leakage at each chip's input. A simple 1-k Ω resistor R1, where the 1-Wire data line departs each enclosure, limits the amount of fault current. Schottky clamp diodes D1 to the supply rails then limit the maximum voltage that can be applied to the sensitive inputs of the 1-Wire chips and the microcontroller. Figure 2 shows the complete 1-Wire connection between the microcontroller port pin and the external bus. Although it is simple, the protection allows the hardware to survive applications of full aircraft bus voltage (even if it is reversed) for an extended time and short spikes of a few hundred volts with no hardware damage.

Also, note the 1,000-pF capacitor C1 from the data line to ground. This capacitor aids in slew-rate limiting and helps absorb fast transients until the clamp diodes have time to switch, but it looks a little odd between the port pin and the resistor. You might expect to see such a filter arrangement reversed with the resistor between the port pin and the capacitor. But we

must be aware of what we're trying to filter. We are primarily concerned with keeping transients induced on the line from getting to the electronics. Reducing the slew rate of the signal we place onto the line is a secondary consideration and helps reduce radiated emissions. Here, the limited current available from the microcontroller port (approximately 20 mA) limits the slew rate to about 20 V per microsecond, preventing significant radiated energy above a handful of megahertz while still keeping signal-transition times under 1 μ s.

But wait. R2 provides the 1-Wire bus passive pull-up, and it's shown as 100 k Ω . Maxim Integrated Products recommends a pull-up of 5 k Ω . With several chips on the bus, we could have leakage current in the tens of microamps, and our pull-up would have a volt or more of drop, so the positive swing of the data line would be severely limited. Even worse, the rise time of the data line with this weak passive pull-up will run into the hundreds of microseconds—and we need to be able to distinguish between pulses of approximately 1, 60, and 120 μ s. Just how can we make this work reliably?

I'm gl-a-a-a-a-d you asked that question. By making the hardware and the software sing together, we can do magic. Here's how.

BITS AND PIECES

First, bury the archaic idea that a microcontroller output pin must be either high or low, or even that it must be either an output or an input. Virtually all modern microcontrollers have the capability to float an output pin. I have implemented 1-Wire masters using both the Freescale Semiconductor 68HC family and the Microchip Technology PIC family, both of which have this capability and have made very nice 1-Wire masters for me. Usually, the documentation describes this as changing the direction of the pin from Output to Input mode, but that mode of thinking is for the digital-only guys. We need to put on our analog hats and see an output port as a complementary pair of MOSFETs that can be controlled independently. After looking at the implemen-

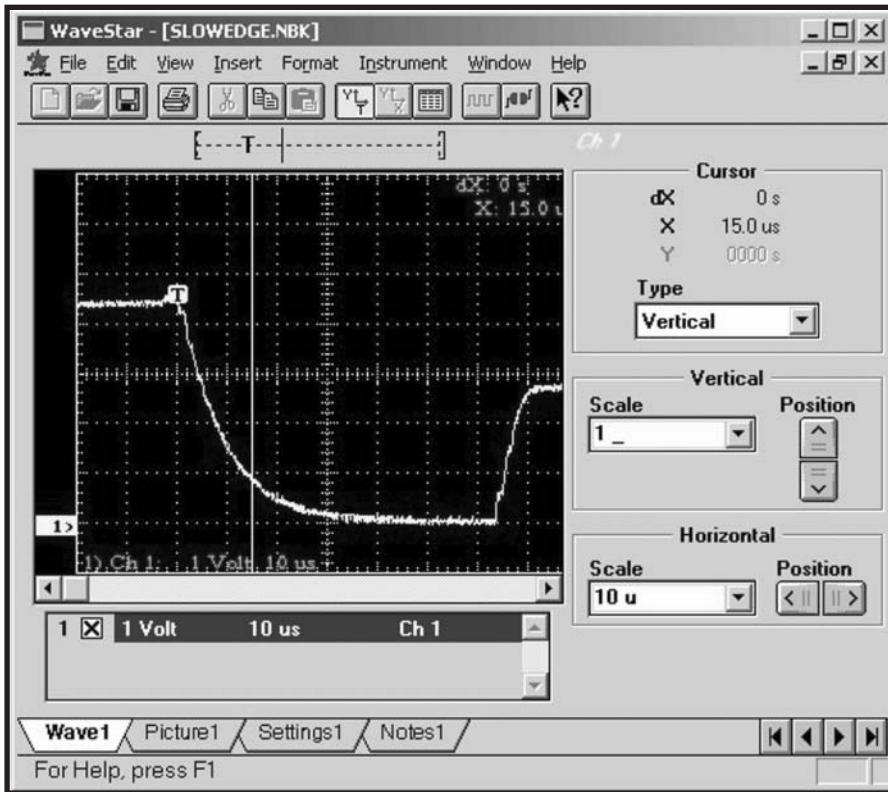


Photo 2—The driver's 1-k Ω impedance interacts with cable capacitance to produce a very slow R-C waveform. The slow edge rates can combine with noise to cause false edges. The long delay to reach valid logic levels causes serious timing problems with the 1-Wire protocol.

tation of a general-purpose port pin (watch out for special cases, such as port A on the PIC family), we see that the pin is always an input pin and the controller can read the state of the pin at any time. We configure a pin as an output by storing a zero into a TRISx register on a PIC or a one into a DDRx on a Freescale Semiconductor microcontroller. Making the pin an output does not change the input function. It simply guarantees that one or the other drive transistors will be turned on depending on whether you have written a one or a zero to that port pin's data register. Thus, we can very rapidly change a port pin between driving low, floating, and driving high.

The 1-Wire protocol is defined so each unit, master, or slave asserts the bus low to actively place data on the bus and allows the bus to float high via a passive pull-up when it is not actively driving the bus. The master is also allowed to actively drive the bus high at certain times in order to augment the passive pull-up and provide additional parasite power.

Dallas Semiconductor almost got this

COMPACT EMBEDDED SERVER

- x86 200MHz CPU
- 128MB SDRAM On Board
- 512MB CompactFlash™
- 10/100 Base-T Ethernet
- Reliable (No CPU Fan or Disk Drive)
- Two RS-232 & Two USB 1.1 Ports
- Optional Wireless LAN & Hard Drive
- Optional On Board Audio
- Dimensions: 4.5 x 4.5 x 1.375" (115 x 115 x 35mm)

Microsoft Windows CE 2.6 KERNEL

- EMAC Linux 2.6 Kernel
- Menu Drive Configuration Utility
- Eclipse Development Environment
- HTTP and FTP Servers
- PPP Dial In/Out Server & Client
- Telnet Server

Compact SIB (Server-In-a-Box) Starting at \$170.00 Quantity 1.

Since 1985 OVER 22 YEARS OF SINGLE BOARD SOLUTIONS

EMAC, inc.

EQUIPMENT MONITOR AND CONTROL

Phone: (618) 529-4525 • Fax: (618) 457-0110 • www.emacinc.com

Save Up To 60% On Electronic Components

Exciting New Mini-Boards

Wide Range Including,

- Real Time Clocks (DS1307)
- Memory Cards
- Power Supplies
- DC Motor, Stepper Motor and More

ONLY \$6.90

Powerful New ATmega Controller

- Includes ATmega128 Microcontroller
- High-Speed Operation
- Heaps of I/O
- In-Circuit Programming
- Ideal Embedded Controller

ONLY \$28.90

Save Heaps on Components

We carry a wide range of Integrated Circuits, Microcontrollers, Capacitors, LED's and LCD's.

ONLY \$4.90

All at very competitive prices.

We are your one-stop shop for Microcontroller Boards, PCB Manufacture and Electronic Components.

www.futurlec.com

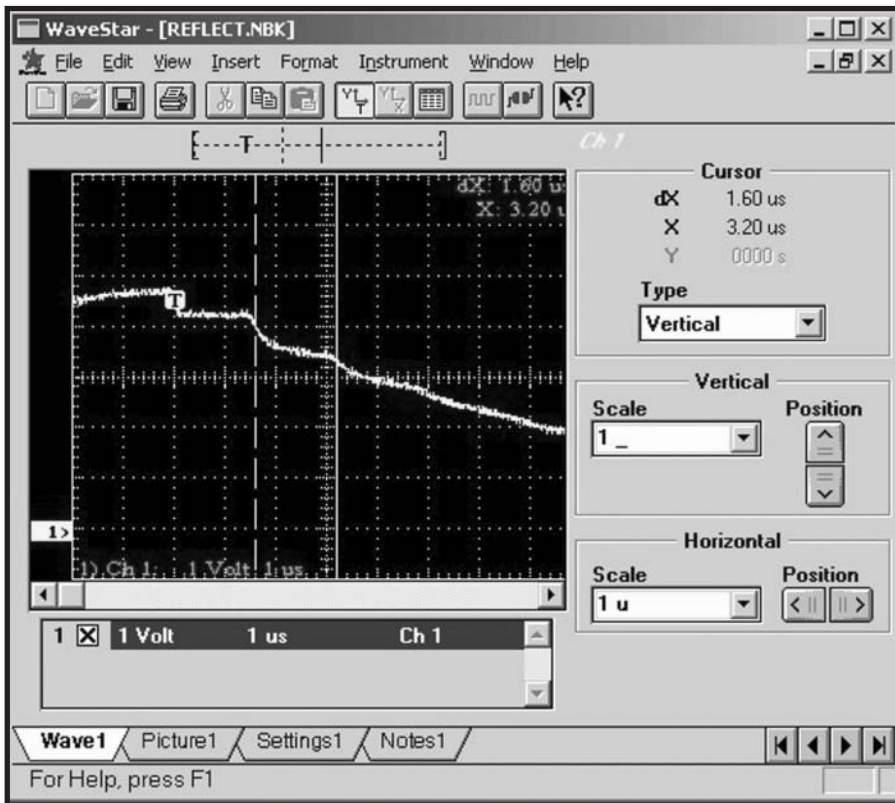


Photo 3—The master attempts to drive the line low at T. The reflection on the 500' cable returns at 1.6 μ s, as marked by the first cursor, and returns a second time after another 1.6 μ s, at the second cursor.

right. A number of e-mail interest groups have sprung up around the 1-Wire bus and their promotional “1-Wire Weather Station” in particular. A number of extended discussions on these groups have focused on the reliability problems with extended cable lengths, as well as the ramifications of branches in the line. One common thread in most of these discussions is a general susceptibility to reflections developed in long cables.

Take a look at the actual reflections in real cables in a controlled environment. Photo 2 shows the 1-Wire waveform with our protected driver attempting to access a bus containing a single device with a 500' tail consisting of a box of Cat5E cable. The dominant feature is the extremely slow R-C time constant as the 1-k Ω driver attempts to discharge the cable capacitance. The waveform takes approximately 15 μ s to fall past the guaranteed-zero threshold of 0.8 V. The 1-Wire slave must see this valid zero before it begins the 15- μ s timing to the sampling point. That delay is going to wreak havoc with the synchronization between master and slave. Reflections? What reflections?

In my college days, the late Capt.

Grace Hopper gave a talk on the history of computers, the development of COBOL, and other topics. One thing that has always stuck with me came when she promised to pass out nanoseconds to each student since none of us had ever seen a nanosecond. She passed out pieces of AWG #30 wire-wrap wire cut to exactly 11.8", the distance light or any electromagnetic wave travels in a nanosecond. Although the illustration was graphic (and lasting), there's a slight inaccuracy: propagation at the speed of light applies only in a vacuum. In real wires or on real circuit boards, speed is slowed by a factor that depends on the surrounding medium—the insulation on a wire, the core material on a PCB. If we were to assume a factor of about 0.66 the round trip on a 500' wire (1,000' total) should take about 1.5 μ s. See those tiny steps right at the beginning of the falling waveform?

Photo 3 expands the tiny steps for better visibility. The first step lasts 1.6 μ s. When the reflection returns, the voltage steps down again as the attempt to drive the line low is reinforced by the reflection. After another 1.6- μ s passes, this reflected step has made another round

trip, and we see another (less pronounced) step in the waveform. By the third round trip, the reflections are just barely visible and they essentially die out completely after that.

A 1-Wire master with sufficient drive to provide parasite power to the slaves will, of course, provide much faster edges as well—if the slaves do not further burden the line. But even in this extreme case, with 500' of cable, the reflections are completely finished within 3 μ s, well before there should be any sampling going on. With old-time, full 5-V logic, we should have plenty of noise margin against any noise picked up on that long line. So why do long lines cause such a problem?

As noted in the documentation of the DS2480 used in 1-Wire masters, the Maxim Integrated Products master drives the bus low to produce a 1- μ s pulse, then releases the bus and watches to see if a slave replies with a zero or a one.^[3] A slave replying with a zero will hold the bus low for at least 60 μ s, whereas a slave replying with a one allows the passive pull-up to pull the bus high within 15 μ s. Because the recommended 5-k Ω pull-up provides a relatively slow rise when asked to charge up the capacitance of a significant amount of cable, their master watches the line to see if it's rising. If the line rises above a defined threshold voltage of approximately 1.2 V, the master assumes the slave is returning a “one” and switches to an active pull-up to “help” the line return to a high (logic one) state. All was well and good until users started deploying 1-Wire weather stations with a couple hundred feet of cable.

With a significant cable length, echoes occur. But more importantly, the cable capacitance smears out the signal edges and the cable itself picks up noise. Because of the reduced-threshold voltage of 1.2 V, these unwanted signals can fool the active pull-up into driving the line high while a slave attempts to drive the line low. Such contention continues through the sample point of the master, yielding an undefined voltage when the bit is sampled. Thus, very frequent errors occur, to the extent that communication essentially ceases.

With the bus protection in place, there is a 1-k Ω resistor between each active

A Design Conference with some Real Muscle.

ARM
Developers'
Conference
and Design
Pavilion



October 2-4
2007
Santa Clara
Convention
Center

Strengthen your skills and speed your time-to-market

Only the ARM Developers' Conference offers:

- Over 90 track sessions providing a complete end to end design tutorial for leveraging ARM IP in advanced embedded applications
- Combined tutorials with Portable Design Conference
- Design Centers and exhibitions from leading ARM licensees and Connected Community members offer a full complement of workshops and presentations
- Forums and analyst presentations on industry trends

Plus: through the combination of track sessions, presentations and company design centers you will be fully immersed in leading strategies and methodologies for building complex designs with the ARM architecture.

And Finally-- Conference Delegates who pre-register before September 10 receive the conference proceedings on a 4 GB iPod Nano to take home!




Register early and save!

www.arm.com/developersconference

Co-Located
with the
Portable Design
Conference & Exhibition

device and the bus. In the worst-case contention situation, the master drives high while the slave is driving low. Both ends then see a fairly reasonable 2-k Ω load resistance and the bus goes to a well-defined voltage at one-half the supply voltage. However, even though the slave is driving the line to ground to present a logic zero, the master is driving the line high with the active pull-up on the same side of the protection resistors as the sampling point. The master reads a logic one even though the slave is presenting a logic zero. All we need to provoke the situation is 1.2 V of noise, minus the residual voltage on the still-settling cable at the sample point. Thus, our noise margin is reduced to much less than the 1.2-V threshold of the "helper" detector, rather than the full 3.5-V (70% of V_{DD}) logic-one threshold we might expect from other digital inputs.

Now, we have the information needed to build a truly robust master that can drive any reasonable length of cable. I'm out of space for this month, so I'll give you time to think about how this all fits together. Next month, we'll see

how a more detailed view of an output pin enables us to write a firmware 1-Wire master that avoids some of the noise-susceptibility problems that can make the Maxim Integrated Products 1-Wire masters appear to be less than robust. We'll then build on that foundation to see how neatly 1-Wire fits into the application. 

Steve Hendrix, P.E., CFIAI, lives with his wife Kathy and their five children ages three through 21 next to Cuyahoga Valley National Park in Ohio. When he isn't busy hiking with them in the Park, leading Boy Scouts, foster parenting, or ringing handbells, he runs Hx Engineering, LLC. Steve has designed software and hardware for projects including Space Station Freedom, vibration analysis, IEEE-488 bus hardware, biomedical sensing (including wireless EEG, EKG, and EMG recorders), industrial instrumentation, and phototherapy equipment.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2007/205.

circuitcellar.com/pub/Circuit_Cellar/2007/205.

REFERENCES

- [1] Maxim Integrated Products, "DS2751 Multichemistry Battery Fuel Gauge Datasheet," 0201005, <http://datasheets.maxim-ic.com/en/ds/DS2751.pdf>.
- [2] Maxim Integrated Products, "DS2409 MicroLAN Coupler Datasheet," 111901, <http://datasheets.maxim-ic.com/en/ds/DS2409.pdf>.
- [3] Maxim Integrated Products, "Serial 1-Wire Line Driver with Load Sensor," DS2480B, <http://datasheets.maxim-ic.com/en/ds/DS2480B.pdf>.

SOURCE

DS2409 MicroLAN coupler, DS2480B serial 1-Wire line driver, and DS2751E gauge

Maxim Integrated Products, Inc.
www.maxim-ic.com

Easy Embedded Linux

\$169
Qty 1



16MB FLASH / 32MB RAM

200Mhz Arm9 CPU

16 Digital I/O

Watchdog

10/100 Ethernet

Battery backed Clock/Calendar

Audio In/Out
2 USB
2 Serial Ports

We brought you the world's easiest to use DOS controllers and now we've done it again with Linux. The **OmniFlash** controller comes preloaded with Linux and our development kit includes all the tools you need to get your project up and running fast.

Out-of-the-box kernel support for USB mass storage and 802.11b wireless, along with a fully integrated Clock/Calendar puts the **OmniFlash** ahead of the competition.

Call **530-297-6073** Email sales@jkmicro.com
On the web at www.jkmicro.com

JK microsystems



EzPCB

45

Professional PCB Supplier

High Quality
Competitive Price

URL: www.EzPCB.com

Email: sales@ezpcb.com Tel: +86 139 1002 1704
HDI Up To 50 Layers, 2.5mil T/C, 0.1mm Hole Size
Other Products: Stencils, Keypads, Frontpanels,
Flex PCB, Enclosures, Turnkey Services

USBee DX

Oscilloscope/Logic Analyzer

2 Analog and 16 Digital Signals
Up to 24Mps, 100+ Million Samples
Dual 8-bit ADC, +/-10V inputs

Bus Decoding

Click and Drag Instant Decode of Bus Transactions
I2C, SPI, ASYNC, CAN, USB Low and Full Speed
I2S, SM Bus, 1-Wire, PS/2

Digital Voltmeter

2-Channel, +/-10V, 8-bit ADC, Logging

Data Logger

2 Analog and 16 Digital Signals Plus Timestamp

Digital Signal Generator

16 Digital Outputs, Up to 24Mps
Playback of Logic Analyzer Traces

Pulse Width Modulator

16 User Controlled PWM Channels

Frequency Counter

16 Channel Counter to 24MHz

Frequency Generator

Generates Sets of Common Frequencies

I2C Controller

Control I2C Devices Using Simple Text Scripts

Remote Controller

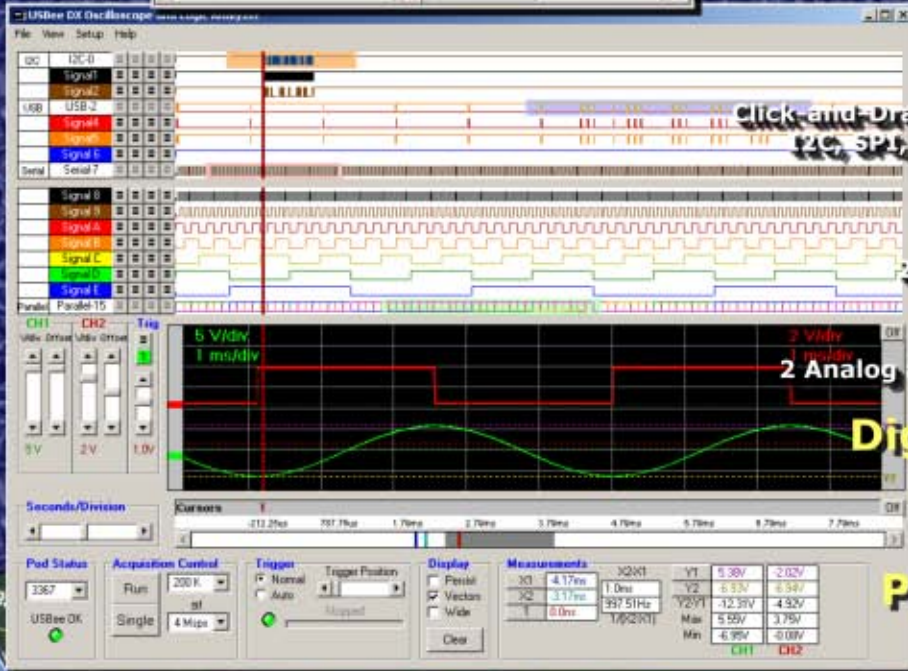
Easily Control Your Hardware Using Your PC

Pulse Counter

16 Channel Pulse Counter With Gating Control

plus the USBee Toolbuilder Source code and Library

Create Your Own Applications to Control the USBee DX



Actual Size

Data Extractors

Optional Software Modules for the USBee AX-Pro and USBee DX

Continuous Real-Time Embedded Bus Data Streaming

Store Bus Data to Disk or Send to Your Custom Application

Capture and Process Entire Test Sequences

Parallel, Serial, I2C, USB, ASYNC, CAN, SMBus, SPI, I2S, 1-Wire

USBee.com (951) 693-3065
support@usbee.com





"The Darker Side" is a new column devoted to some of the lesser-known aspects of electronic design. It will appear every other month.

Let's Play with EMI!

Robert sheds light on the "dark" subject of electromagnetic interference. He covers everything from electromagnetic compatibility (EMC) issues to helpful design techniques. With a few modifications to your next design, you will be able to reduce the amount of unwanted radiation emission.

Welcome to The Darker Side! I am really excited to present this new column devoted to some of the lesser-known, more obscure aspects of electronic design. Scary, isn't it? If you find the title strange for a magazine as serious as *Circuit Cellar*, note that I chose it after a brainstorming session with the magazine's great editorial staff—so the responsibility is shared.

The purpose of this column is to highlight concepts too often perceived as expert-only subjects. If you read "analog" between these words, you will probably be right 90% of the time. I will do my best to help you understand what's really going on while focusing on application-oriented explanations and pragmatic tips rather than painful theory and math (although some equations may show up from time to time). Lastly, because dark subjects may sometimes be understood differently from other points of view, and because I will probably be wrong or imprecise from time to time, I will be more than happy to receive your feedback. Just drop me an e-mail and I will be in touch!

I want to start with one of the darkest possible subjects. It is something that can give headaches to engineering managers evaluating the risks on their projects. It is something that is often not learned until it is

too late. It is electromagnetic interference, or EMI for short.

EMC BASICS

Electromagnetic compatibility (EMC) is about being reasonably sure that a given piece of equipment will work peacefully with other equipment in its neighborhood. "Peacefully" means the equipment should not be perturbed by others (EMI immunity), and it should not perturb them (EMI emission limitation).

Let's look at the details. In the EMC/EMI world, equipment is usually classified into two categories: voluntary versus unintentional transmitters (or receivers). For example, a voluntary transmitter could be your remote control transmitter, while an unintentional transmitter could be a badly designed power supply generating plen-

ty of EMI noise. Similarly, a voluntary receiver could be your FM radio, while an unintentional receiver could be your MP3 player picking up the RF noise transmitted by your cell phone.

EMI immunity relates to voluntary transmitters (generating strong RF fields, because they are designed to transmit something) and unintentional receivers (low-sensitivity receivers) that could be agitated by these strong RF fields. Reciprocally, the EMI emission limitation is a concern between unintentional transmitters (low-level but not expected) and voluntary receivers (high-sensitivity because they are designed to receive). Inspired by Tim Williams's very good book, *EMC for Product Designers*, Figure 1 illustrates these two fundamental EMC gaps.

Because nobody knows which EMI environment a given product will be used in, legislation defined the absolute maximum admitted emission power and minimum immunity requirements. Legislation on EMC was initially limited to radio communications (i.e., voluntary transmitters and receivers), but it was extended to unintentional transmitters about 20 years ago, particularly in Europe. The first EC EMC directive, 89/336/EEC, was ratified in the early 1990s and was made mandatory in 1996. It covers both immunity and emissions limitations. In

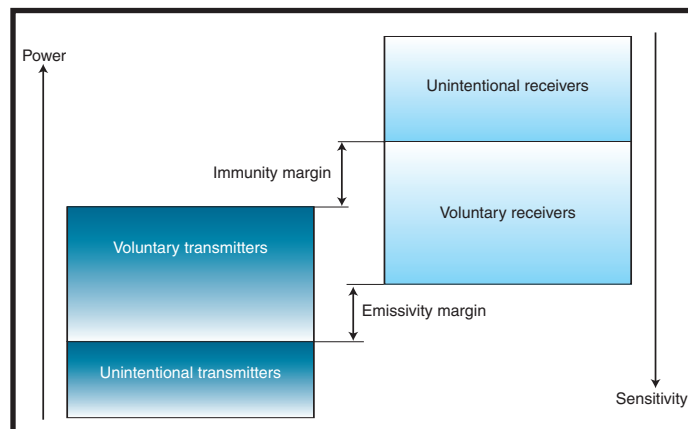


Figure 1—This figure illustrates the two main EMC requirements: EMI immunity, related to the interaction between voluntary transmitters and unintentional receivers, and EMI emission limitations, related to the interaction between unintentional transmitters and voluntary receivers.

the United States, FCC Part 15 limits emissions of unintentional transmitters (spurious emissions, subpart B), as well as low-power, unlicensed transmitters (subparts C to G). But as far as I know, emission compliance with RF immunity standards is not mandatory for most products under the FCC's guidelines, even if additional standards are usually requested by the marketplace. In all other countries, the international standards IEC61000-xxx from the IEC's TC77 are usually translated into national standards, providing a similar framework.

Finally, electromagnetic signals can be either radiated (through the air) or conducted (through wires). Standards usually deal with these two aspects separately, even if they are often closely linked, as you will see.

You probably think that I was lying in my introduction when I promised to focus on application-oriented explanations and pragmatic tips rather than on long and painful theory. Agreed, I am going to the bench. Because the EMC subject could fill up thousands of pages, I will concentrate on only one topic: unintentional emissions. To demonstrate the basics of this nasty phenomenon I will not describe a rocket science project. I'll focus on one of the simplest unintentional transmitters: a small TTL oscillator built around a 74HC00 chip and the 3.5795-MHz crystal I had on my desk, with a 1-k Ω load resistor simulating a clocked device on the board (see Figure 2).

Before I demonstrate the phenomenon, I need to introduce the EMC guy's "pocket knife."

THE SPECTRUM ANALYZER

You will need a way to see and measure RF because EMI is mainly about RF signals covering a large frequency range (from nearly DC up to several gigahertz) and a large signal

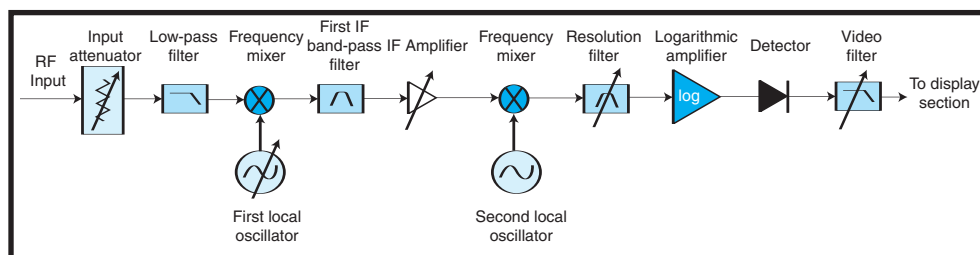


Figure 3—Here is the overall, but largely simplified, architecture of a super-heterodyne low-band RF spectrum analyzer.

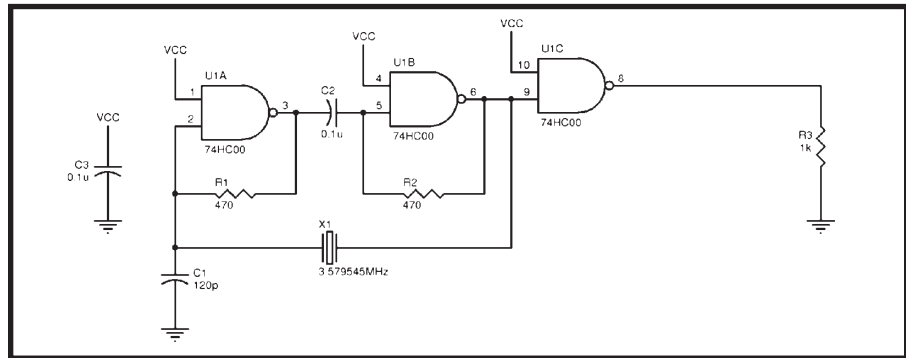


Figure 2—My experiments will be conducted with a very simple project, a crystal oscillator built using a 74HC00 NAND gates chip and a 1-k Ω load resistor.

amplitude range (from watts down to nanowatts or lower). A spectrum analyzer, an expensive but invaluable tool, must be used. Oscilloscopes are currently available far above the gigahertz region (even up to 18 GHz for those with really deep pockets), and high-end digital models provide a spectrum analyzer using a fast Fourier transform of the time-acquired signal. Don't be confused. This is often a useful feature, but by no means equivalent to an RF spectrum analyzer. The reason is the dynamic range. The average high-speed oscilloscope uses an 8-bit ADC in the giga-samples-per-second range and may be a 10-bit ADC for higher-range scopes. This provides, at best, seven or eight effective bits. From MSB to LSB, this provides, at best, a dynamic range of 1 to 256 (i.e., 2^8). This is in terms of voltage. Because power is proportional to voltage squared, this translates to a dynamic range of 1 to 65,536 (i.e., 256^2) in terms of signal power. This is not bad, but it represents only 48 dB (i.e., $10 \log(65,536)$). The worst real spectrum analyzer provides at least 80 or even 100 dB of dynamic range.

A spectrum analyzer is nothing more than a tuned, swept, very selective receiver. The architecture of a simple RF super-heterodyne spectrum analyzer can be seen in Figure 3. The

measured signal, coming from a wire or an antenna, first goes through a step attenuator to avoid saturation. A low-pass filter gives it a clean signal from kilohertz up to the maximum frequency of the equipment. Assume the maximum is 3 GHz. The signal is then mixed with a clean high-frequency swept local oscillator covering a 3-GHz range (from 3.5 to 6.5 GHz). It is band-pass filtered around a frequency just higher than the maximum usable frequency (3.5 GHz). This architecture allows it to limit spurious receptions. For example, when the local oscillator is set at 4.223 GHz, the only input frequency that will provide a signal going through the 3.5-GHz filter will be at 723 MHz (i.e., $4.223 \text{ GHz} - 3.5 \text{ GHz}$).

The 3.5-GHz intermediate frequency signal is amplified and down-converted through several mixing stages to a low intermediate frequency. Then, one of the most important parts of the spectrum analyzer comes in: the resolution filter. This is a very selective, calibrated, and variable band-pass filter that allows it to set the measurement bandwidth, which has a strong influence on sensitivity, frequency discrimination, and measurement speed. The signal is sent through a logarithmic amplifier, allowing the analyzer to display a power level in a logarithmic scale over several decades. Finally, its amplitude is detected, filtered again to remove noise by averaging, and displayed.

This presentation is simplified because digital techniques are heavily used from the latest intermediate frequency blocks down to detection through digital resolution filters and FFT. Moreover, the microwave

equipment that has a frequency range far above a few gigahertz uses a different conversion scheme for its input section. They use exotic Yig-based preselection band-pass filters and harmonic mixers. Anyway, this simplified architecture fairly matches the architecture of 20-year-old spectrum analyzers, such as my Hewlett-Packard HP8569. If you want to know more about spectrum analyzers, the bible is "Spectrum Analysis Basics" (application note AN150).

Now that you have a spectrum analyzer, connect a wideband antenna, or better yet, a preamplified electric field probe to its input. Now you have a real-time view of the surrounding RF environment (see Photo 1). Such a test setup is not adequate for EMC formal assessments requiring calibrated antennas and shielded anechoic chambers, but it is a very useful relative measurement tool, as you will see.

RADIATED EMI

Photo 1 was taken with a "clean"

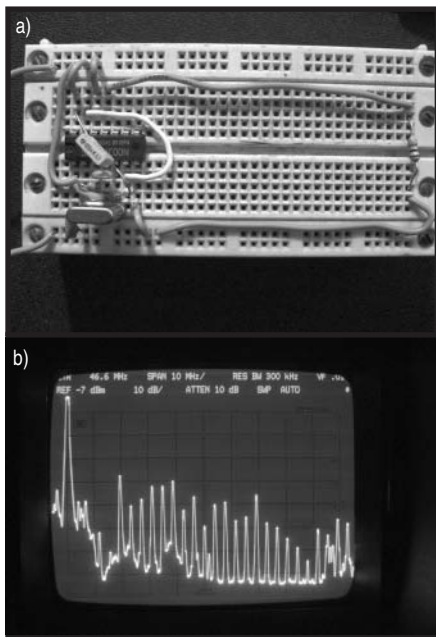


Photo 2a—This is the test crystal oscillator in its first nonoptimized version. **b**—This is the spectrum received 2 m away from the test oscillator in Photo 2a. The spurious emissions are nearly 1,000 times more powerful than the FM broadcast signals.

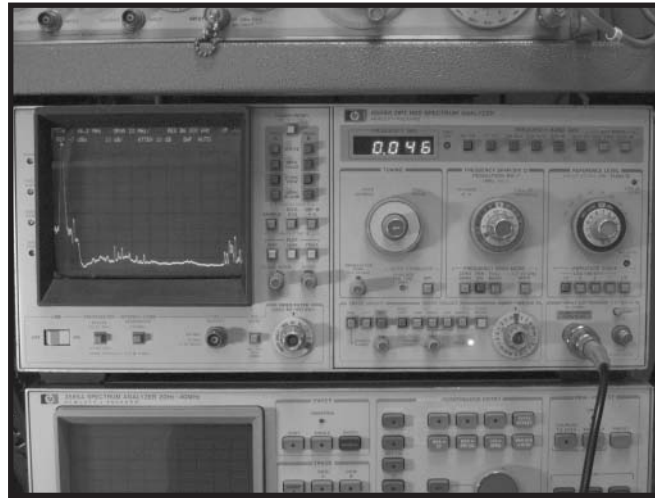


Photo 1—The plot shows the background RF field in my lab from nearly DC up to 100 MHz with a 10-MHz/div horizontal scale. The left-most peak at 0 Hz is an artifact of the spectrum analyzer. Significant emissions are visible from 20 to 50 MHz with a peak at 40 MHz (short waves, police, and so on), as well as from 90 to 100 MHz (FM broadcast). The vertical scale is logarithmic.

environment, but what happens when you switch on the test crystal oscillator? I built it on a small prototyping board, even if you don't agree that it is the nicest idea for a good RF design (see Photo 2a). I then dropped it on a desk 2 m away from the receiving antenna (that's around 6 ϕ or 7 ϕ for those living on the wrong side of the ocean), and powered it from a 5-V bench supply. Photo 2b shows what was displayed on the spectrum analyzer's screen. I am not joking! The separation between peaks is close to 3.5795 MHz, which is not a surprise. You can see all the harmonics of your crystal oscillator from 20 MHz up to a few gigahertz. That's a good illustration of unintentional emissions, isn't it? Note the power of the maximum peak on the first test as a reference. The top of the screen is -7 dBm. The vertical scale is 10 dBm/div. I found the maximum peak at about -34 dBm (including a gain of the preamplified antenna, air path loss through the 2 m between the test oscillator and the antenna, and so on).

Go back to the reference spectrum in Photo 1. In the same setup, the FM broadcast peaks were measured at -63 dBm. That means the EMI field transmitted by the small crystal oscillator is 29 dB (i.e., 63 - 34) more powerful than the FM broadcast signals, even if the crystal oscillator is 2 m away from the receiving antenna. And, if you are still

reluctant to use decibels, this means that this interference is no less than 794 times (i.e., $10^{29/10}$) more powerful than the radio station you want to listen to!

What happened? Any AC current going through a conducting loop generates an electromagnetic field that will propagate in the space with an amplitude roughly proportional to the current, to the surface of the loop, and to the square of the frequency. Because your crystal oscillator is more or less using square waves, all harmonics of the clock signal are transmitted and unfortunately well received, up to the maxi-

imum frequency of the HC00 gates.

What should you do? First, try to reduce the area of the current loops. Photos 3a and 3b show what happened when I routed the wires differently, connecting the 1-k Ω resistor differently. The spectrum looks the same, but the maximum peak is now -39 dBm. That seems like a small improvement, but remember that you are using a logarithmic wide dynamic scale.

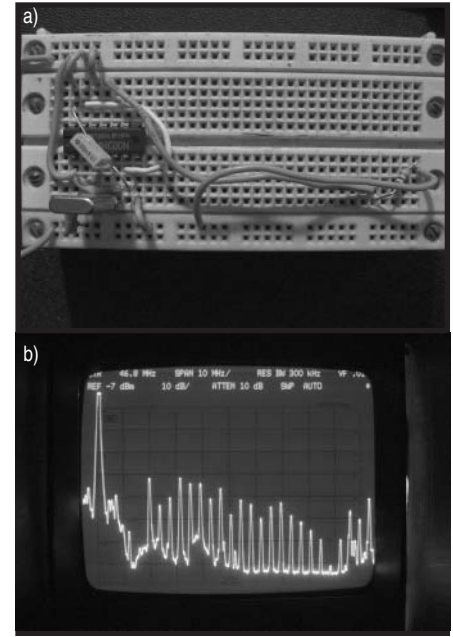


Photo 3a—The first improvement to the test oscillator was to move the resistor load wire to reduce the loop area. **b**—When compared to Photo 2b, the improvement in this photo provides a reduction of 5 dB of the unintentional emission. Not bad.

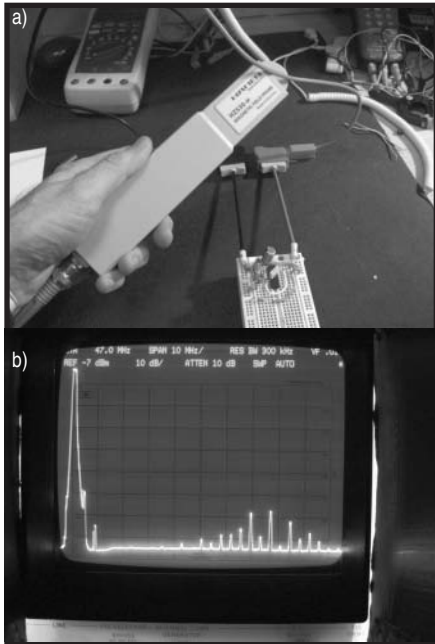


Photo 4a—Using an H-field probe allows it to pinpoint a key contributor to the radiated emissions, namely the 5-V and GND power wires. **b**—The H-field probe shows a high level of radiated signals from the power lines.

That's a 5-dB improvement (from -34 down to -39 dBm). It means the power of the received signal is $10^{5/10}$ lower, so it was reduced by a factor of 3.2 just by moving a wire. Not so bad. So, the golden rule is to limit all loops carrying high-frequency signals or low-frequency signals with abrupt slopes like logic signals. Because the return-current path is usually the ground, this recommendation translates into the most important EMC tip: build your circuits with a full ground plane on one layer of your PCB. That will ensure that there is always a return path just below every wire. Some will argue that split ground planes could be good, but I'm sure that 99.9% of the designs will work better with a full ground plane rather than with more complex schemes (except for some audio designs well packed in a shielded enclosure and not really affected by radiated EMI). For example, if you design a board using two ground planes joined only by a small bridge, then there will be current loops, except if all other wires going from one area to the other are passing exactly over the bridge, which is not easy to do.

How do you go further with minimal changes to the design? The level of harmonics generated by our test project is

as high as it is because the signals are steep square signals, but is this steepness actually needed? Why use a 74HC00 with a propagation delay of 7 ns, high output current, and more importantly heavy current peaks at each transition due to its CMOS technology to build a simple 3.5-MHz clock? I dug into my shelves and swapped the HC00 with an old LS00. The maximum peak was immediately reduced to -45 dBm, which is a 6-dB incremental improvement. Compared with the initial situation, these two small changes reduced the transmitted noise by 11 dB, or more than a factor of 10, just by moving a wire and swapping a chip with an older one! Let me summarize the lesson learned by this second experience. Always use the slowest possible technology for a given project, with the slowest possible output currents and the slowest possible voltages, particularly in digital signals. That's obvious, but it's too late when you remember it after the design is complete.

CONDUCTED EMI

I didn't want to talk about conducted EMI in this column, but I have to. Assume that the noise level is still higher than acceptable. You can continue to improve the design every-

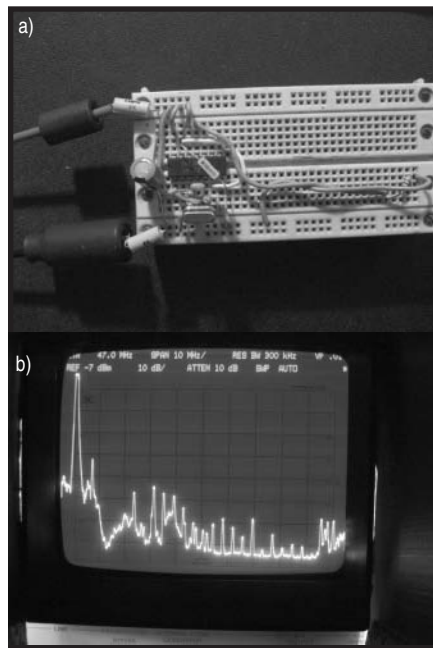


Photo 5a—This is the next improvement: coils and ferrite filters on the power lines and a slower LS00 chip. **b**—With the improved design in Photo 5a, the unintentional emissions are drastically reduced.

ANDRE LAMOTHE'S
XGAMESTATION
DO-IT-YOURSELF VIDEO GAME SYSTEM

Inspired by the Atari 2600, Apple II & Commodore 64!

SX52 @ 80 MIPS

INCLUDES:

- Assembled XGS Micro Edition Unit!
- Complete Development Kit!
- Tools, Demos & Utilities!
- eBook on Designing the XGS Console!
- Cables and Power Supply Included!

www.XGAMESTATION.COM
(925) 736-2098 SUPPORT@NURVE.NET

USB/ETHERNET DAQ

LabJack UE9

Available now for only ...
\$429 qty 1 USB/Ethernet Data Acquisition & Control

- * USB 2.0/1.1 and Ethernet
- * 14 analog inputs (12- to 16-bit)
- * Stream input data up to 50 kHz
- * Use with C, VB, LabVIEW, etc.
- * Includes DAQFactory Express
- * Operates from -40 to $+85$ deg C
- * 2 analog outputs (12-bit)
- * 23 digital I/O
- * Up to 2 counters (32-bit)
- * Up to 6 timers
- * Approx. 3" x 7" x 1"

LabJack Corporation, Colorado, USA
info@labjack.com, (303) 942-0228

www.labjack.com

ADD MASS STORAGE TO YOUR DESIGNS

USB MASS STORAGE

Designers that share information with PC and other USB hosts
Need real-time data in flash memory and other media
Check out the new USB MASS STORAGE

DESIGNING AND PROGRAMMING DEVICES AND EMBEDDED HOSTS
Jan Axelson author of USB COMPLETE

USB Mass Storage Designing and Programming Devices and Embedded Hosts
Jan Axelson

ISBN 1-931448-04-3 \$29.95
Lakeview Research LLC www.Lvr.com

From the author of USB Complete

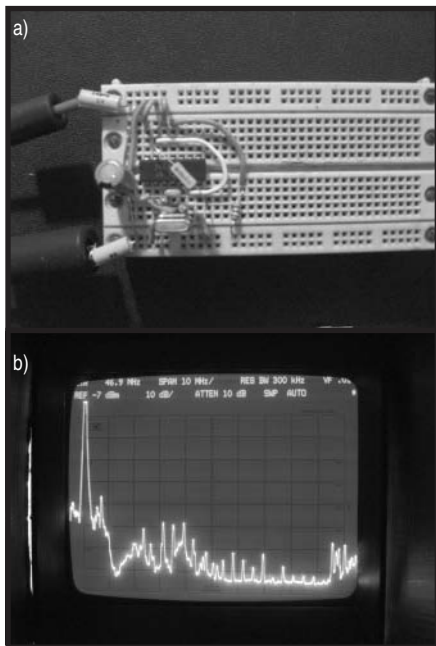


Photo 6a—This is the last improvement. The load resistor is moved close to the oscillator. **b**—All of these improvements give an accumulated reduction of the unintentional emissions by 21 dB as compared to Photo 2a. That's more than a factor of 100!

where you can and see the result on the transmitted noise 2 m away, but that could be quite time consuming. It would be easier if you could pinpoint the precise location of the transmitted noise source. Luckily, this is also possible with a spectrum analyzer. The only additional tool needed is a magnetic field probe (H-field probe). The reason is that an H-field is decreasing as the cube of the distance, while the E-field is decreasing as its square. So, using an H-field probe is far more effective to locate a transmitter at very short ranges because the H-field is decreasing very quickly when you leave the immediate surroundings of the emitter. You can buy good, low-cost, preamplified H-field probes like the Hameg HZ530 probe set that I'm using, but after buying the spectrum analyzer your pockets will be empty. Fortunately, a simple H-field probe can be easily built using a small 50-w shielded wire. Just strip the cable a few millimeters from the end, turn it to make a small loop, and solder the inner connector back to the shield braid. The end of the braid should not be connected to itself in order to provide a gap in the shield. That's it. Now you have a good magnetic field probe.

Assuming that you have an H-field probe connected to the spectrum analyzer, move it around the design, and in particular, around the wires connecting it to the 5-V power supply (see Photos 4a and 4b). Bingo! You can see that the power cables are radiating a strong RF field in particular in the 50- to 100-MHz range. The current peaks drawn by the logic chip show up as conducted current peaks on the power lines because of insufficient decoupling, even though the ubiquitous 100-nF capacitor is in place on the power rails. The conducted EMI is then radiated due to the long unshielded power wires. That's why conducted and radiated EMI are often linked.

You must reduce the conducted EMI through the power lines. I've modified the test oscillator with the addition of a 1-mH coil plus a ferrite bead both on the GND and 5-V lines, as well as the addition of a 10- μ F low-ESR capacitor in parallel to the 100-nF one (see Photo 5a). One caution regarding filtering coils: in high frequencies, a higher-value coil is not always better, because parasitic capacitances can quickly make it useless. Technology does matter, and that's why ferrite coils are usually far more efficient for EMI reduction than standard coils of the same value. This subject will need a column by itself.

Back to the spectrum analyzer. The maximum received peak is now at only -51 dBm if you consider only the HF contributors above 10 MHz, another 6-dB improvement (see Photo 5b)! As a last improvement, I tried moving the 1-kW resistor closer to the chip (see Photo 6a), and the received EMI level is down to -55 dBm, 4 dB better (see Photo 6b).

IS IT ALL?

Using one of the most simplistic designs (a crystal oscillator) built on the most simplistic platform (a proto-board), I found a couple of very simple design modifications that allowed me to reduce its unwanted emissions by 21 dB (i.e., 55 - 34), effectively reducing the spurious noise power by 99.2% ($10^{21/10} = 125$, $1/125 = 0.8\%$)! With these changes, the FM broadcast sig-

nals are now clearly visible on the spectrum analyzer, which means that you will be able to hear your favorite songs. And more importantly, these design changes were really minor, including rerouting some wires, an HC00 replaced by an LS00, coils, and ferrites on the power supply. Impressive, isn't it?

In this short article, I have highlighted some key points of one aspect of EMC, unwanted radiated emissions, and some design tips that will make your life easier. The EMC-minded designer will also need to take care of immunity, conducted EMI, common mode noise, ESD, and the regulation side of the EMC. Anyway, I hope that I have successfully demonstrated that the EMC is not black magic, even if it is occasionally on the darker side. 📡

Robert Lacoste lives near Paris, France. He has 18 years of experience working on embedded systems, analog designs, and wireless telecommunications. He has won prizes in more than 15 international design contests. In 2003, Robert started a consulting company, ALCIOM, to share his passion for innovative mixed-signal designs. You can reach him at rlacoste@alciom.com. Don't forget to write "Darker Side" in the subject line to bypass his spam filters.

RESOURCES

Agilent Technologies, "Spectrum Analysis Basics," application note AN150, www.metrictest.com/resource_center/pdfs/agl_spec_analyzer_basics.pdf.

T. Williams, *EMC for Product Designers*, Newnes, 2007.

SOURCES

HP8569 Spectrum analyzer
Agilent Technologies, Inc.
www.agilent.com

HZ530 Probe set
Hameg Instruments
www.hameg.com



Hearing Clearly

After experiencing some hearing loss, Ed decided to perform several measurements on ears, earphones, and microphones. In this article, he introduces a circuit that compensates for his poor hearing.

Several years ago, my hearing went from pretty good to pretty bad over the course of a few days. After extensive testing, I received a diagnosis of “Stuff happens” and some disconcerting pictures of my head’s innards. My hearing loss wasn’t severe enough to warrant hearing aids, but it punched the treble right out of my life.

Maxim recently introduced their MAX5406 Audio Processor with Push-button Interface, which provides volume, balance, and tone control for portable devices. As soon as I read the datasheet, I thought it might be useful as a “treble boost” for my ears.

After I made a few measurements and doodled some initial designs, I discovered that simple filters can accomplish much of what I need. I learned a bit about sound and hearing, though, which you may find interesting, too.

This month I’ll present my measurements on ears, earphones, and microphones, as well as the SPICE simulation of a filter that compensates for the largest part of my poor hearing. My next column will describe a hardware implementation, along with a switching power supply, and take a look at some real-world problems.

PROBING THE EARS

Audiologists performed several hearing tests on me, eventually determining that I had “no aidable loss.” That simply means that I could hear well enough to get by, not that I could hear well, and I didn’t need hearing aids. One test showed my hearing dropped off by 10 dB at 2 kHz and was 25 dB down at the test’s upper limit of 8 kHz.

Telephones and amateur radio use a 3-kHz bandwidth, so I have “communications grade” hearing. In essence, everything I hear sounds like it’s coming through an old landline telephone and, no, I don’t like on-hold music any more than you do.

When I’m traveling by train or airplane, I use a pair of Shure E2C sound-isolating earphones with my MP3 player. Although they resemble nothing so much as a pair of large beetles jammed in my ears, their foam sleeves eliminate much of the background noise and the transducers have good fidelity even for normal ears. I bought these before my hearing tanked and they’re obviously overqualified for my present needs.

I measured the lower threshold of my hearing using Digital Recordings’s Audio-CD hearing test disc, played on

a laptop PC through a Creative Technology Sound Blaster USB MP3+ external “sound card” and the Shure E2C ear beetles. The laptop’s built-in sound hardware would also work, but the USB box had a nice analog volume dial.

A threshold test requires setting the volume of a calibration tone so that it’s not quite audible, then recording the level at which each test frequency becomes audible. The results depend on the ambient noise level, audio hardware imperfections, listener attention span, and many other factors, but my results remain consistent within a few decibels over several tests.

The blue trace in Figure 1 represents my unassisted hearing, which is most sensitive around 1 kHz, drops off by 20 dB between 2 and 3 kHz to a plateau out to 6 kHz, and is essential-

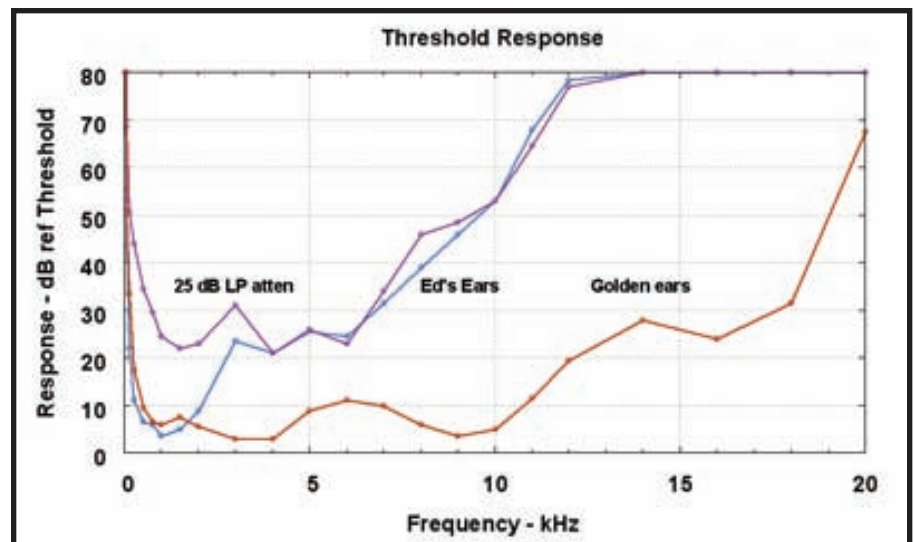


Figure 1—These traces show the sound level required to distinguish a pure sine-wave tone. The lower trace shows that my daughter hasn’t burned out her ears, while the blue trace indicates my “communications-grade” hearing. The filter circuitry in Figure 2 improves my overall response, as shown by the purple trace.

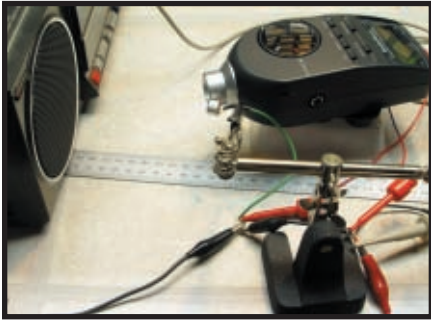


Photo 1—Accurately measuring microphone response requires better equipment than this, but even a simple measurement helps separate the good from the bad. The Radio Shack sound-level meter helped set a constant volume for each test tone.

ly dead by 12 kHz. The plot shows the additional volume above each listener's threshold, so this is not an absolute pressure measurement. Lower numbers are better: 80 dB over the threshold is *really loud*.

The brown trace near the bottom of Figure 1 shows my 14-year-old daughter's hearing. While it's not perfect, it does show that the hardware is actually making sounds that I cannot hear and the calibration is reasonably flat. However, even though the test CD tones begin at 20 Hz, it seems the hardware isn't much good below 100 Hz, where my low-frequency response matches hers.

The first step in correcting my hearing will be to fill in that bowl around 1 kHz. There's not much point in correcting anything beyond 7 kHz, for reasons that will become clear when we look at dynamic range.

MIC CHECK, ONE TWO

I plan to use this circuit to correct the output of digital music players,

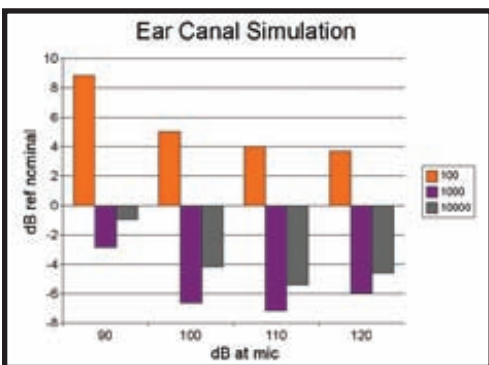


Figure 2—The response of an E2C earphone in a simulated ear canal shows that it requires more power at low frequencies and less at high frequencies than its 105-dB SPL at 1 mW spec would indicate. Some of the variations certainly come from my technique.

rather than as a true hearing aid, if only because I cannot stuff the circuitry into my ears. Despite that, I want to see how it works at concerts, which means I need microphone inputs and some degree of portability.

My parts collection includes a wide variety of electret microphones, so I spot-checked their frequency response at 100 Hz, 1 kHz, and 10 kHz. Photo 1 shows the test setup: a boom box driven by a sine-wave signal generator, with the microphone and a sound level meter from Radio Shack on-axis at 10 cm. I set the test frequency and adjusted the signal generator amplitude and boom box volume control to produce 100 dB SPL at the meter, then recorded the microphone's output voltage from the RMS value computed by my digital oscilloscope.

I set the sound level meter to show fast-response C-weighted levels, which provides a reasonably flat response across the entire audio spectrum. Because I was using a single test tone in a quiet environment, the meter's integration of all sounds at all frequencies wasn't relevant. Moving the meter slightly didn't produce any large changes, so I believe the readings are a fair estimate of the actual levels, despite the layout's acoustic imperfections.

Electret microphones include a preamplifier that modulates the current drawn through a bias resistor to produce a voltage output corresponding to the sound input. I attached a 2.2-k Ω resistor and a 4.5-V power supply using clip leads, which worked well enough for these quick measurements.

I don't have room for the table here, but some mics were flat within a few decibels and others had dramatic differences at either the high or low frequencies. You're welcome to test your own mics and generate far more data points!

I picked a mic with a reasonably constant response at the three test frequencies for further use. I think it was intended for a mobile phone, but it's a surplus find lacking pedigree and exact specifications.

The output signal at 100 dB SPL was 14.6 mV_{rms} at 1 kHz. It increased by 2.2 dB at 100 Hz and dropped by 1.6 dB at 10 kHz, so



Photo 2—A 7 x 22 mm brass tube resembles an ear canal closely enough to measure my Shure E2C earphone response. The foam plugs seal tightly in a real ear canal to exclude sounds that don't come from the electronics.

the overall range is about 3.8 dB. This is for a bare mic held in mid-air, which means I must remeasure it when I figure out what sort of case I'll use.

Remember that a decibel value is a power ratio. Because power varies as the square of the voltage, the decibel value for a voltage ratio is:

$$\text{dB} = 20 \times \log_{10} \left(\frac{V_{\text{TEST}}}{V_{\text{REF}}} \right) \quad [1]$$

For example, I measured 18.9 mV at 100 Hz, so the decibel value is:

$$2.2 \text{ dB} = 20 \times \log_{10} \left(\frac{18.9 \text{ mV}}{14.6 \text{ mV}} \right) \quad [2]$$

I then measured the mic's output for sound levels of 90, 100, 110, and 120 dB at 1 kHz. Referred to the 90-dB level, the voltage increased by only 7, 9, and 8.5 dB for successive 10-dB input steps, so the microphone does induce some compression.

I wore liquid-filled shooting-sports ear protectors during these tests to prevent further injury to my ears. If you must ever endure a 1-kHz tone played at over 100 dB SPL, you'll understand my motivation. Permanent hearing damage starts at levels over 85 dB: protect your ears!

ARTIFICIAL EAR CANAL

The Shure E2C spec sheet shows the earphones produce 105 dB SPL with 1 mW input and that they have a 16- Ω impedance. Delivering 1 mW to a 16- Ω load requires a voltage of:

$$0.126 \text{ V} = \sqrt{1 \times 10^{-3} \text{ W} \times 16 \Omega} \quad [3]$$

Permanent hearing damage occurs

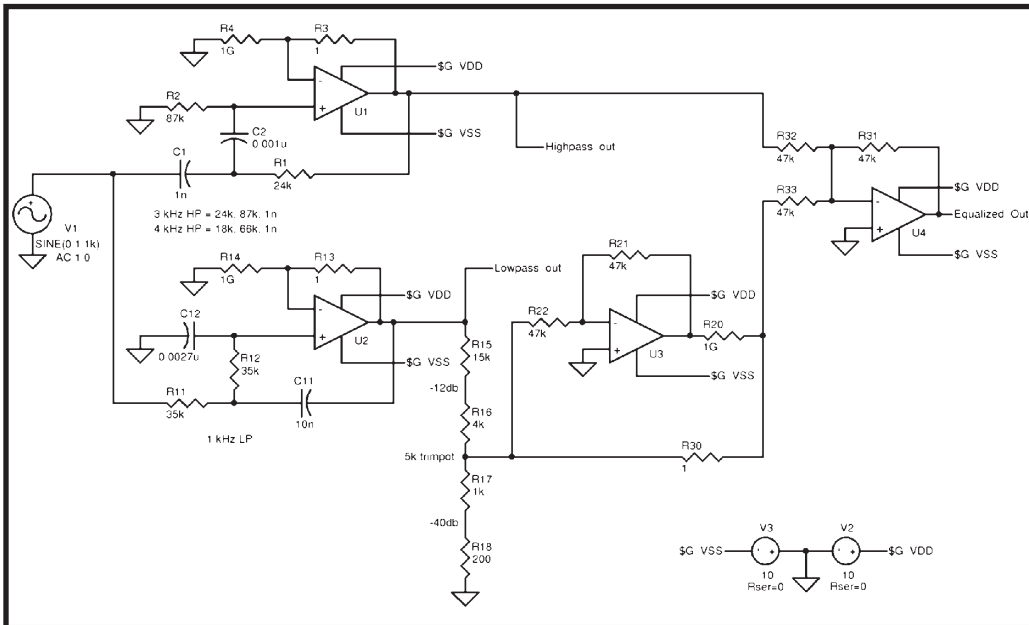


Figure 3—Combining the 0-dB output of a 3-kHz high-pass filter with the 25-dB attenuated output of a 1-kHz low-pass filter compensates for the 25-dB high-pass attenuation in my ears. U3 inverts the low-pass phase; you can see its effect by fiddling with R20 and R30. The 1-Ω and 1-GΩ resistors are modeling conveniences that act as switches to pass or block the corresponding signals.

instantaneously at levels above 120 dB, which explains why many young people have degraded hearing. Assuming the earphones behave linearly, producing 120 dB (15 dB above 105 dB) requires a voltage of:

$$0.71 \text{ V} = 0.126 \text{ V} \times 10^{\frac{15}{20}} \quad [4]$$

Unlike microphones, you must measure an earphone's output in its intended environment, an ear canal, rather than in the open space more suited to a speaker. A typical human ear canal seems to be about 0.25" in diameter and 0.75" long, so I found a 7 × 22 mm brass tube in my heap, mounted an E2C ear beetle at one end and the electret mic at the other, and sealed them in place with modeling clay.

Photo 2 shows the test setup on my workbench. The mic circuitry is the same clip-lead kludge as before and the earphone connects to the boom box's speaker/headphone output jack. My digital oscilloscope displayed both of these voltages and calculated their RMS values.

I adjusted the level of a 1-kHz sine wave to produce 90, 100, and 110 dB SPL at the microphone, for the same three test frequencies as before, by matching the mic's output voltage I measured earlier for each SPL in open air.

Figure 2 plots the power required for each level, relative to the spec sheet's nominal 1 mW. The midrange 1-kHz

signal requires 3 to 7 dB less power than expected, while the 100-Hz bass requires much more. The 10-kHz treble requires slightly more power than the 1-kHz signals.

Although I didn't wear the shooting muffs for this part of the project, the sounds coming from that poor ear beetle were just about as loud as those I've heard out in the real world. Yes, people really do listen to music at 120 dB and, yes, they are destroying their ears in the process.

COMPENSATING FILTERS

The MAX5406 has a maximum bass and treble adjustment range of about ±10 dB relative to the level at 1 kHz, but, as Figure 1 shows, a 20 dB range would be just barely adequate. I'd also have to shift the filter shoulders from 1 kHz to the 3 kHz required by my ears.

However, I found that if I amplify everything above 3 kHz or attenuate everything below 3 kHz by about 25 dB, the overall response should be more or less flat from 100 Hz to 7 kHz. Well, even if it's not flat, it'll be much better than it is now!

The microphone produces about 130 mV at 120 dB SPL, the loudest sound it should ever hear (at least near me!). Amplifying that level by 25 dB corresponds to a voltage ratio of:

$$17.8 = 10^{\frac{25}{20}} \quad [5]$$

That produces a 2.3-V_{rms} signal, a peak sine-wave voltage of 3.2 V.

That may not sound like much, but circuits powered from a 5-V supply have a maximum voltage swing of about ±2.3 V around the midpoint. Unlike SPICE simulations, real circuits put a solid upper limit on the maximum signal swing and, therefore, limit the maximum allowable

input signal amplification. The chips you choose determine how much gain you can apply: your block diagrams must include maximum signal levels along with decibels of gain.

It should be obvious that I do not intend to jam 120-dB SPL sounds into my ears, but both analog and digital signal processors have strict limitations on the dynamic range they can handle. Limitations on the high end come from analog signal swing limits or digital word sizes, while noise typically limits the low end. As always, trade-offs abound!

As a result, rather than amplify the high frequencies, I chose to attenuate the signals around 1 kHz by 25 dB. I

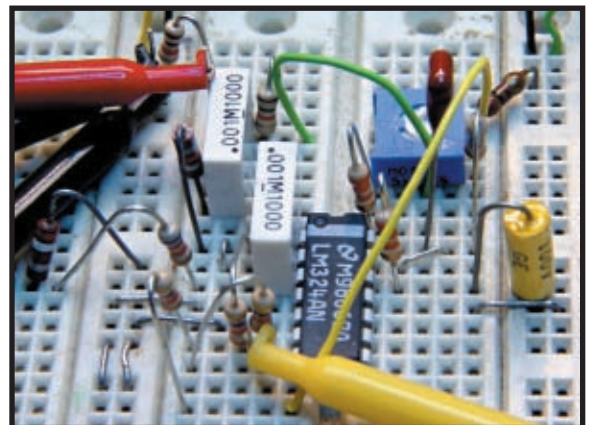


Photo 3—Although I'm no fan of solderless breadboards, they work well enough for a low-frequency application like this.

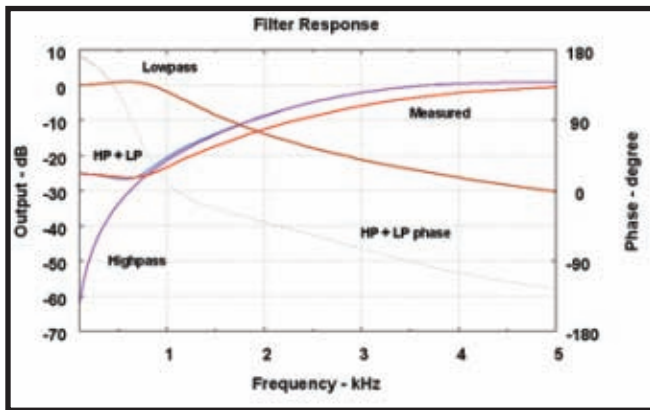


Figure 4—The “Measured” filter response tracks the “HP+LP” simulation reasonably well, given the accuracy of the parts. My ears are the weak link in this project, so a few decibels one way or the other really don’t make any difference.

picked two-pole, 1-dB Chebyshev filters to get a fairly steep drop-off at the corner frequency. In this application, the usual Chebyshev band-pass ripple doesn’t matter. Analog Devices’s online filter synthesizer made short work of the design, Linear Technology’s SwitcherCAD III verified the results, and the breadboard shown in Photo 3 worked the first time.

The high-pass filter built with U1 in Figure 3 passes everything above 3 kHz at 0-dB attenuation. The low-pass filter around U2 passes everything below 1 kHz at 0 dB to the 25-dB attenuator formed by R15 through R18, which reduces the low-pass level to match my ear’s response. The sum of those two signals in U4 has the response shown by the simulated “HP+LP” trace in Figure 4 and the actual “Measured” results match it quite closely. At this scale, the 1-dB Chebyshev ripples aren’t visible.

Putting the low-pass cutoff at 1 kHz, rather than 3 kHz, flattened the overall response. A filter’s cutoff marks the point where its response is down 3 dB from its peak, but the bump in my hearing around 3 kHz called for a bit more attenuation between 2 and 3 kHz.

I measured the filters with RealAudio’s Real-Time Analyzer PC-based audio spectrum analyzer. The test signal from the PC’s Line-Out goes to the filter, which then drives Line-In.

After collecting the numbers, I connected the filter output to Line-In on the boom box, plugged the E2C earphones into its Speaker-Out, and played music from the PC through the filter to my ears. It sounded awful: no

bass, hot treble!

That was, in fact, pretty much what I expected. After all, I’ve been listening to audio with severe high-pass attenuation for the last few years, so any treble at all should sound unfamiliar. This will require fine tuning, but I think I’m on the right track.

CONTACT RELEASE

I’ll put these pieces together into a complete system in the next column. Meanwhile, you can examine my measurements and schematic on the *Circuit Cellar* FTP site.

Keep in mind that I am not an audiologist. If you’re having hearing problems, this is definitely not the place to get advice or hearing aids. What you’ve seen here is just one geek’s way of experimenting with his own ears; you’re welcome to do the same at your own risk. ☠

Ed Nisley’s an EE and author in Poughkeepsie, NY. Contact him at ed.nisley@ieee.org with “Circuit Cellar” in the subject to avoid spam filters.

PROJECT FILES

To download additional files, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2007/205.

RESOURCES

American Academy of Otolaryngology—Head and Neck Surgery, “Noise & Hearing Protection,” www.entnet.org/healthinfo/hearing/noise_hearing.cfm.

Analog Devices, Inc., Active Filter Synthesis, www.analog.com/Analog_Root/static/techSupport/designTools/interactiveTools/filter/filter.html.

J. Beale, “Microphone Comparison

Test,” www.bealecorner.com/trv900/micctest/micctest.html.

Digital Recordings, “Hearing and tinnitus tests,” <http://digital-recordings.com>.

E. Nisley, “Audio Transformation,” *Circuit Cellar* 147, 2002.

———, “HTML Front Panels” *Circuit Cellar* 149, 2002.

———, “Learning from the Past: A Look at a Chrome Lollipop,” *Circuit Cellar* 135, 2001.

———, “Mic Check: A Communication System for Cyclists,” *Circuit Cellar* 133, 2001.

———, “PC Audio Bits,” *Circuit Cellar* 145, 2002.

S. Somers, “The Mysterious Loudness Control: What Does It Do?,” Extron Electronics, www.extron.com/company/article.aspx?id=loudnesscontrol_ts.

J. Wolfe, “Equal Loudness Contours and Audiometry: Test Your Own Hearing,” The University of New South Wales, www.phys.unsw.edu.au/jw/hearing.html.

SOURCES

USB MP3+ Sound card

Creative Technology
www.creative.com

SwitcherCAD III

Linear Technology
www.linear.com

MAX5406 Audio processor

Maxim Integrated Products, Inc.
www.maxim-ic.com

33-2055 Sound-level meter

Radio Shack Corp.
www.radioshack.com

E2C Earphones

Shure, Inc.
www.shure.com

TrueRTA Real-time audio spectrum analyzer

True Audio
<http://trueaudio.com>

Audio spectrum analysis: spectrogram version 14

Visualization Software
www.visualizationsoftware.com



Graphical User Interfacing

Jeff explains how he used NS Basic to connect the Palm operating system to a microcontroller through its hard-wired serial connection. Now he can create applications for handheld devices and harness his PDA's full potential.

Veteran readers of *Circuit Cellar* may remember the Amulet Technologies GUI that I presented in my January 2003 article "GUI Interfacing: A Straightforward, Simple Solution" (*Circuit Cellar* 150). The tools available for the LCD and the touchscreen required little programming to use the graphical and touch user interface. Although the interface is a bit expensive for my current (electric bike) project, it would make sense for a true electric vehicle graphical user interface (EVGUI). The Toyota Prius hybrid and others are now using the GUI for many automotive functions (see Photo 1). Dashboards, like vehicles, are undergoing a radical change.

Many people have rejected having their car talk to them. I don't need my car telling me to "buckle up" or "get gas now"! The gentle reminder of an idiot light is sufficient, thank you very much. The industry is beginning to understand what we, as drivers, are willing to accept. I foresee a time

when a youth will enter the family car and ask it to be whisked off to, say, soccer practice; but for now, we are still required to remain in control.

LCDs can offer more than just an electronic representation of the mechanical gauges we are so content with. They can offer a personal reconfiguration of the display, such as the skins we have become accustomed to in our PCs, cell phones, MP3 players, and so on. That small touch of personal choice helps us accept and feel more comfortable with the technology.

POWER IN YOUR HAND

Whether you are a PDA user or someone who has refused to deposit your life into its depths, you must agree that they can be amazing and useful gadgets. Today's cell phones are beginning to lean toward PDA functionality, while some PDAs have integrated phones. The lines are becoming blurred. The use of an LCD as a GUI



Photo 1—The Toyota Prius hybrid features a GUI that brings all vehicle functions to a central location on the dashboard. The real-time display of energy consumption and the dynamic interplay between internal combustion and the electric motor is fascinatingly educational.

has made this possible. One of the most exciting aspects of a PDA is its operating system, which opens the door for third-party applications. While you've seen that an open system also invites rogue applications, it would be unfortunate if you allowed the possibility of evil to limit free access and its potential good.

Years ago, I bought a Handspring Visor to play with its potential, rather than its PDA functionality. To access its potential, I investigated NS Basic's (NSB) Palm operating system (see Tables 1 and 2). As the name implies, the version of BASIC can create applications for handheld devices with the Palm OS. This column will describe how you can use NSB to connect your Palm device to a microcontroller through its hardwired serial connection.

Like any application generator, NSB is only as good as the tools that make it easy to use. Just having a language that's easy to use is not enough. Now most development systems come with an IDE to make the process as painless and productive as possible. The

| Statements | Descriptions |
|-----------------|---|
| Structure | END FUNCTION, END SUB, EXIT FUNCTION, EXIT SUB, FUNCTION, SUB |
| Flow of control | CALL, CASE, CASE ELSE, CHAIN, DO, DO UNTIL, DO WHILE, END SELECT, ENDF, EXIT DO, EXIT FOR, FOR, GOSUB, GOTO, IF, LABEL, NEXT, NEXTSCREEN, RETURN, SELECT CASE, STOP |
| Menus | MENUDRAW, MENUERASE, MENURESET |
| Variables | DEFINE, DIM, DISPLAY, END TYPE, GLOBAL, TYPE |
| Drawing | CREATEWINDOW, DRAWBITMAP, DRAWCHARS, DRAWLINE, DRAWRECTANGLE, ERASEWINDOW, FILLRECTANGLE, REDRAW, DESTROYWINDOW |
| Other | BEEP, CURSOR, DELAY, LET, MSGBOX, SETCURRENTWINDOW, SETEVENTHANDLED, SOUND |
| Variable types | INTEGER, FLOAT, DATE, TIME, STRING, BYTE, SINGLE, DOUBLE, SHORT, DATABASE, USERTYPE |

Table 1—Most code contains statements that perform arithmetic operations and manipulate variables and objects.

NSB IDE for a Windows PC is no exception. Don't have a Palm device? No problem. The Palm OS Emulator (POSE) enables your application to run on your PC.

INTERACTIVE DE

Windows displays a look and feel that makes most applications feel friendly, even if you have never used them before. The NSB IDE also displays the familiar top row of shortcut-enabled pull-down menus for supported functions. The rest of the screen is dedicated to the work area, which includes other tools that are made visible using the View menu item. They include the Project Explorer, Properties, and the Toolbox, as well as a Toolbar and a Status Bar (see Photo 2).

Applications written with the NSB IDE become a project file (.prj) that incorporates all forms, program code, menu definitions, and more for a specific application. All of that gets boiled down into the application file (.prc). An exception to this is a database file (.pdb). It is the .prc (and possibly the .pdb) that actually gets downloaded to the PDA with HotSync, which is an automatic exchange of information, between a PDA and a PC, to update any application (or data) that has changed.

Your application or project will contain one or more forms (or screens of information). A form is a collection of

| Functions | Descriptions |
|---------------|---|
| Math | ABS, ASC, CEILING, CHR, FLOOR, INT, MOD, RAND, REM, ROUND, SIGN |
| Trigonometric | ACOS, ASIN, ASINH, ATAN, ATAN2, ATANH, COS, COSH, DEGTORADIANS, EXP, LOG, LOG10, POW, POWER10, SIN, SINH, SQRT, TAN, TANH |
| String | FORMAT, INSTR, LCASE, LEFT, PROPER, RIGHT, RIGHTPAD, RTRIM, STR, TESTNUM, TRIM, TRUNC, UCASE, VAL, LEFTPAD, LEN, LTRIM, MID, NOOCCURS |
| Date | ADDDAYS, ADDMONTHS, ADDYEARS, DATEMMDDYY, DATEVAL, DAY, DAYOFWEEK, DAYOFYEAR, FIRSTOFMONTH, LASTOFMONTH, MMDDYYTODATE, MONTH, NOW, POPUPDATE, SUBTRACTDAYS, SUBTRACTMONTHS, SUBTRACTYEARS, TODATE, TODAY, YEAR, YEARMONTH |
| Time | HOUR, HOURMIN, HOURMINAMP, MINUTE, POPUPTIME, SECOND, TIMEVAL, TOTIME |
| Miscellaneous | ALERT, GETEVENTTYPE, GETKEY, SYSEVENTAVAILABLE |
| Database | DBCLOSE, DBCREATE, DBDELETE, DBERASE, DBFIND, DBGET, DBGETNORECS, DBINSERT, DBOPEN, DBPOSITION, DBPUT, DBREAD, DBREADNEXT, DBREADPREV, DBRESET, DBUPDATE |
| Serial | SERIALCLOSE, SERIALDIAL, SERIALHANGUP, SERIALOPEN, SERIALRECEIVE, SERIALSEND, SERIALSET |

Table 2—Functions allow programmers to use modular code. These might be written by the programmer, provided as calls to built-in routines, or accessed via libraries.

objects. Objects are buttons, fields, lists, labels, bitmaps, or other items used to display or collect information. NSB is an object-oriented programming language that is much different from the top-down, linear programming of many BASIC languages (i.e., the PC's early GW BASIC). This means that execution is based around the objects.

When you begin a new project, the NSB IDE starts by placing a single form in it. The project has two areas associated with it that can contain code. Startup code is executed before any forms are created and termination

code executes after the form is displayed, and event code is executed when the form detects an event.

Most of the standard Palm OS objects are supported by NS Basic. Refer to Table 3 for a list of objects found in the toolbox. I chose a number of different objects to help illustrate the usefulness of the language in an application that lets you monitor a microchip's registers via a Palm device. The application can view and change internal registers via a three-wire serial TTL connection made between the PDA and the microcontroller.

The objects I placed on the application's form are in Photo 2. The "Choose a Register" popup at the top of the form will provide the user with a list of the available registers' names. The chosen name will replace the "Choose a Register" label. Two button objects, whose actions will be based on the chosen register, are just below the object. The "Read this Register's Value" button will send a request out the serial port for data and look for a reply. The second button, "Write this Value to the Register," will use the data entered into any of the following three formats, send an update request out of the serial port, and look for a reply.

One byte of data is displayed and collected in all formats: binary, decimal, and hexadecimal. I used individual buttons for each digit of each for-



Photo 2—The NS Basic IDE is similar to many development tools used in the Windows environment. From here you can design screens, write code, compile, and download to either hot-sync the application to your handheld or download into the Palm OS Emulator (POSE).

mat. I did this not because it is the most efficient form of data entry, but merely for its interesting display. A tap on any digit increases that digit by one rolling over from "1 to 0" for binary, "9 to 0" for decimal, and "F to 0" for hexadecimal. All of the displays are updated each time a digit is changed.

Two field objects are at the bottom of the form. The left-hand field displays the transmitted data string. The right-hand field shows the received data string (or received error). I chose to use an ASCII format so the data could be viewed easily. The request-for-data format is seven characters: three hexadecimals (register), two hexadecimals (checksum), <cr>, and <lf>. A PIC can have multiple banks, more than 256 registers, so three digits are used. The checksum is a single-byte representation of the total of all bytes up to the checksum. The update request contains three additional characters for a total of 10 characters: three hexadecimals (register), "=", two hexadecimals (data), two hexadecimals (checksum), <cr>, and <lf>. Again, the checksum includes all of the characters up to the checksum.

The replies for both requests have the same format. In fact, the reply for the update request is nothing more than the same 10 characters returned. While both replies have data, only the request for a data string carries new information. The returned value is displayed on the form via each of the data formats. I wrote simple ASC2Integer and Integer2ASC routines for this application, but these functions and many more are available via shared libraries. For instance, MathLib contains functions: log, sqrt, sin, cos, tan, asin, acos, and atan. The BitsNBytes shared library contains all of the format conversions and bit manipulations. Some shared libraries are

specific to a particular flavor of Palm handheld. These give access to additional hardware for playing MP3s or color LCDs.

PHYSICAL FEATURES

My Handspring Visor's cradle connector has eight pins and a direct-connect cable (see Photo 3). Both USB and TTL serial connections are available on this connector; however, only USB is used by the cradle. While USB requires a host to connect, the serial port can be easily connected to other devices (see Photo 3). Note most handhelds run on 3 V, so interfacing with 5-V circuitry requires level shifters (or at least current limiting). I modified my cradle with a stereo phone jack for making connections easily. However, you can purchase a plug-in connector to use the PDA without the cradle. You can see the application running from the cradle in Photo 4.

On the microcontroller side, the three-wire connection is made to a Microchip Technology PIC16F687.

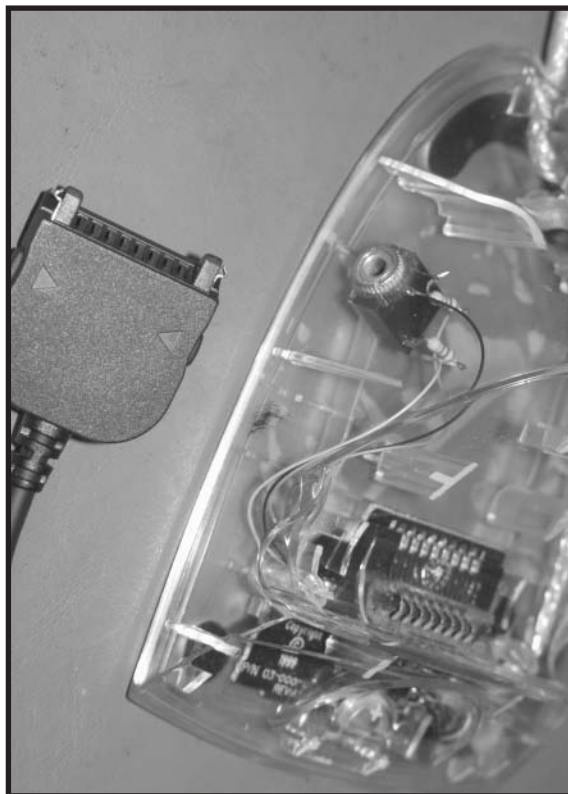


Photo 3—The eight-pin cradle connector has both USB and TTL serial signals available. I added a stereo jack to my cradle for easy access. Pin one equals RX, two equals KbdDetect, three equals HotsyncInterrupt, four equals GND, five equals USB+, six equals USB-, seven equals VUSB, and eight equals TX.

PIC-SERVO

MOTION CONTROL

MOTION CONTROLLERS FOR BRUSH, BRUSHLESS AND STEPPER MOTORS.

- controller chips
- controller boards

www.picservo.com
JEFFREY KERR, LLC


bob


basic overlay board

Introducing:
Vector graphics
Custom Font Capability
SPI or RS-232 Interface

Meet bob-4
Video display generator

www.decadenet.com

 **DECADE ENGINEERING**
503-743-3194 Turner, OR, USA



CIRCUIT CELLAR®

back issues available as

Searchable Archives on CD-ROM

NOW SHIPPING:

- CD-ROM #11 2006 Issues 186-197
- CD-ROM #10 2005 Issues 174-185
- CD-ROM #9 2004 Issues 162-173

Order Online:
www.circuitcellar.com
or call 860.875.2199

| NS BASIC object | What is it? | What can it do? |
|-----------------|---|--|
| Bitmap | A fixed area for the display of a bitmap | An event is triggered when it is tapped (with no visual cue) |
| Button | A framed or frameless area containing text | An event is triggered when it is tapped (with a visual cue) |
| Checkbox | A selection box that can be grouped with others such that the selection of one clears others in group | An event is triggered when it is tapped (visual cue alternates checked and unchecked condition) |
| Field | An area for data entry and display | An event is triggered once the focus of this field is changed. Gather data input to the field or be used strictly for displaying data. |
| Gadget | A frameless (invisible) area | An event is triggered when it is tapped (with no visual cue). The pen position of the tap can be determined. |
| Label | An area for data display | An event is triggered when it is tapped (with no visual cue) |
| List | An area for a list of data | An event is triggered once a selection is made. |
| Popup | A button holding a list's selection | An event is triggered when it is tapped. A list is presented for the user and disappears once an item is selected. |
| Push button | A selection button that can be grouped with others. The selection of one clears others in the group. | An event is triggered when it is tapped (visual cue alternates between the normal and the selected condition) |
| Repeater | A framed or frameless area containing text | An event is retriggered as long as the stylus remains in contact (with a visual cue) |
| Scrollbar | A horizontal or vertical indicator | An event is triggered when it is tapped. Position can be determined (visual cue position) |
| Selector | An area for data entry and display with a dotted border | An event is triggered when it is tapped (with a visual cue) |

Table 3—This table of standard Palm objects forms the pallet from which you can create dazzling interactive applications with little support code.

Since the serial communication is TTL (no need for level shifters for true RS-232), the idle state is a logic “1.” The start bit is a falling edge. The only requirement for the microcontroller is that it has a falling-edge external interrupt. Sure, a UART would be handy, but what if it was already being used for the microcontroller’s application? Instead, I wrote a bit banging interface for this application. (I certainly wouldn’t go through the exercise if I had access to a UART.) This project takes into consideration that the microcontroller will be (eventually) using its LIN interface for LIN bus monitoring and that operation requires the hardware EUART. So, even though that has not been implemented, I want to reserve it for that purpose.

The bit banging must have as low an impact on the microcontroller as possible, while still taking care of business. (The flowchart outlining the code strategy is available on the *Circuit Cellar* FTP site.) The main program loop is shown in Figure 1. Only two checks need to be added to whatever application has been written for the microcontroller. The checks look for flag status, indicating actions that need to be performed by the monitoring program. FLAG.LF is set whenever a line-feed character has been received from the PDA. This signals that a complete string has been received and the buffer holding the

string can be processed. The call to `Process_Buffer` first determines whether the string is requesting to store or retrieve register data. This is based on the string’s length. The string’s checksum is verified to assure the data has not been corrupted. With a little data conversion, the ASCII characters are converted to a register pointer (when retrieving data) and data (when storing data). Only the “retrieval of data” path requires adding characters to the buffer so both responses use the same format

of the “request to store data.” The routine finally returns to the main loop after clearing FLAG.LF and setting FLAG.SEND. Any interrupt can take control from the `Process_Buffer` routine (and any code in the main loop):

```
“ppp=ddcc<cr><l f>”
```

where p is an ASCII HEX character (register location), d is an ASCII HEX character (data), and c is an ASCII HEX character (checksum). Thus:

```
093=003A<cr><l f>
SSPADD=0x00 (checksum=0x30+0x39+
0x33+0x3E+0x30+0x30=0x3A)
```

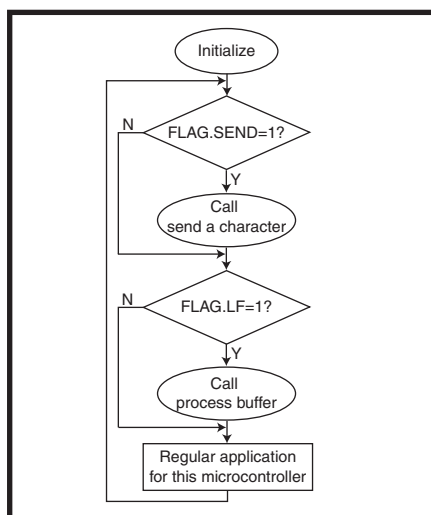


Figure 1—Three routines need to be added to whatever application is running in your microcontroller. A falling-edge_interrupt routine handles received data. The `Process_Buffer` routine processes the received data in the buffer. The `Send_a_Character` routine transmits a serial response. The last two routines are positioned within the main application loop, as shown in the flowchart.

The other check in the main loop will be continually called as long as the FLAG.SEND is still set. The `Send_a_Character` routine handles sending a single character from the buffer. To prevent any interruptions to this time-critical code, system interrupts are disabled within the routine from the clearing of the PORT.TX (beginning of start bit) until the raising of PORT.TX (beginning of the stop bit). This requires 468 μs (i.e., approximately 9 × 52) of exclusive execution time. The routine doesn’t return until it has timed all of the stop bits. However, since interrupts are enabled right after the beginning of the stop bit, any interrupt can take control from the `Send_a_Character` routine after that point.

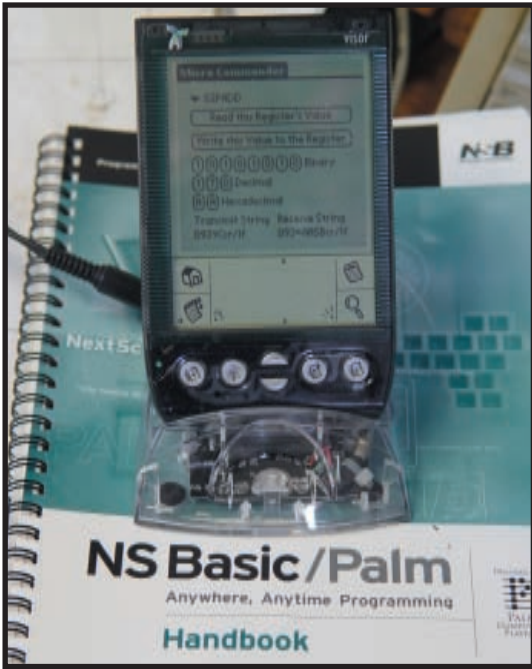


Photo 4—Here is my Visor running the Micro Commander Application. A three-wire serial interface to your microcontroller (with a little support code in the microcontroller) enables you to visually retrieve and store register data while your microcontroller executes its application.

The last routine necessary for this monitoring example is the falling-edge interrupt on PORT.RX. The interrupt must take precedence over any other. A sample of the incoming datastream must take place 1.5 bit times from the start bit (halfway between the beginning and the end of the first data bit). After that, additional bits are sampled every 1-bit time (halfway through each additional bit). A little time can be saved if you don't bother looking for the stop bit.

AN LCD WITH BUTTONS

I don't complicate things without a reason. Most information can be displayed on a standard character LCD. These can even include graphic-like bar graphs for visual interest. And yes, a couple of mechanical push buttons aren't that hard to add for user feedback. The use of a graphic LCD usually requires considerably more programming. And the addition of a touchscreen for user input also adds complexity. And that's if you can find the pieces to construct something. I don't think you will find any prepackaged terminal-type graphic/touchscreen solution for less than a few hundred dollars. Inexpensive Palm devices are still available for less than \$100.

NS Basic enables you take advantage of the handheld screen and touch interface that make PDAs so popular. I love the idea of using a piece of equipment you may already own or were itching to get around to playing with. If you put off getting a PDA because you weren't interested in the kinds of applications that come installed, NSB offers you a way to look at a PDA in a whole new backlight.

Perhaps you are a power user and you own a Treo Smartphone. This particular phone/PDA comes with either a Palm or a Microsoft CE OS. You can purchase NS BASIC support for either OS for \$129. Visit www.nsbasic.com for a list of other high-powered PDAs that are supported by NS Basic. ☒

Jeff Bachiochi (pronounced BAH-key-AH-key) has been writing for Circuit

Cellar since 1988. His background includes product design and manufacturing. He may be reached at jeff.bachiochi@circuitcellar.com.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2007/205.

RESOURCE

M. Verive, *NS Basic Programming for Palm OS*, The Eastwood Group, 2002.

SOURCES

CCA00-000-0007 Handspring visor connector

Cypress Industries
www.cypressindustries.com

PIC16F687 Microcontroller
Microchip Technology, Inc.
www.microchip.com

NS Basic
NS Basic Corp.
www.nsbasic.com

We Listen. Think. And Create.

Distributed I/O

Serial I/O

Digital I/O

Industrial Computing

HMI

SeaDAC USB modules are the fastest, most reliable way to connect I/O to any computer.

SeaDAC USB Modules Offer:

- Optically Isolated Inputs, Reed Relay Outputs, Form C Relay Outputs, Digital/Analog Combo
- Status Indicator LEDs for Communication, Fault, & Status
- Field Removable Terminal Block Connectors
- High-Retention USB Type B Connector
- Rugged Plastic Tabletop Enclosure
- Extended Temperature Option Available

FOCUS
On Success
Call Today!

SEALEVEL
sealevel.com > sales@sealevel.com > 864.843.4343



Silicon Secrets

Tom assuages your concerns about identity theft with information about some new chips that can protect your embedded designs. Atmel's AT88SC CryptoMemory lineup features passwords, encryption, and other security features that will guard your system from hack attacks.

The good news is that modern technology, such as the PC and the Internet, gives us the ability to handle our affairs from the comfort of home and even when we're on the road. The bad news is that such power can easily be turned against us if we're not careful. You need go no further than the headlines to see the growing concern over identity theft, fake IDs, etc.

I've personally been lucky, but some of my acquaintances haven't. Imagine you show up at the bank one day and step up to the counter to find that all your money has disappeared? That will make your day, eh? That's just what happened to one of my friends. It turns out thieves had simply printed copies of his checks (and presumably fake IDs to go with them). The bank ended up eating the loss, which actually means the honest customers of the bank ended up eating it in the form of higher fees, interest rates, etc.

What goes around sometimes comes around. Another acquaintance told me of the time \$80K suddenly appeared out of nowhere in his friend's checking account. That person informed the bank of the problem, but the bank insisted the money must be his! At that point, it must have been tempting to head to Vegas with the "lucky" \$80K, but this honest fellow simply sat on it. About six months later, the \$80K simply disappeared with nary a peep from the bank.

Fake financial transactions are only the tip of the iceberg, with fake products now posing a threat that goes well beyond knock-off DVDs and Rolexes. Another story I heard starts

with a company that urgently needed some MCU chips, which were in short supply at the time. Desperate for parts, they made a deal with some shady overseas "Midnight Chip Supply" broker. When the parts arrived, they looked completely legitimate (i.e., package mark), but they didn't seem to work. It should come as no surprise that upon popping the hood it was discovered that the motor was missing (i.e., no die in the package).

Losing money is one thing, but losing your life is another. Did you know that even the FDA admits that up to 10% of the drugs sold in the global pharmaceutical market are counterfeits? And in parts of Asia and Africa, the percentage of fake drugs is estimated to be as high as 50%. These estimates are found in Robert Cockburn's article "The Global Threat of Counterfeit Drugs" in *PLoS Medicine*, with other instances outlined that make for disturbing reading indeed. For instance, an estimated 192,000 Chinese people were killed by counterfeit drugs in a single year (2001), during which Chinese authorities

shut down 1,300 illegal factories (maybe that should be "fake-tories"). In another case, more than 500 patients, mostly children, were killed by fake analgesic syrup that contained poisonous diethylene glycol. During a 1995 Meningitis epidemic in Niger, up to 66,000 people were inoculated with what turned out to be a fake vaccine. The list of horror stories goes on.^[1] Hmm, you might want to think twice before you score that cheap Viagra over the Internet.

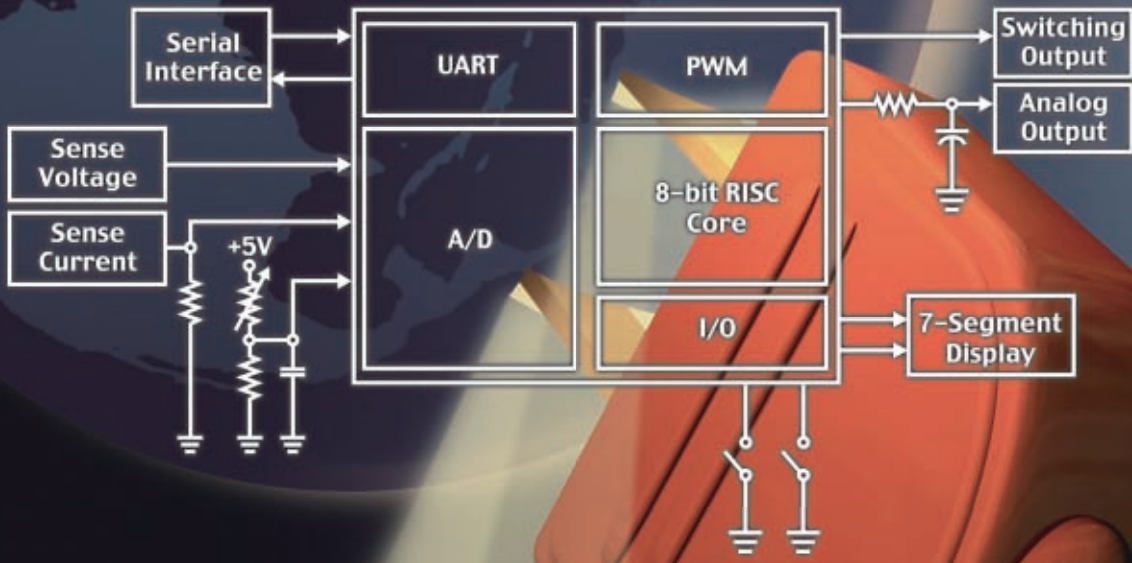
TAG, YOU'RE IT

Malaysia Car Thieves Steal Finger

Police in Malaysia are hunting for members of a violent gang who chopped off a car owner's finger to get



Photo 1—Dogs and cats are already getting "chipped." Can people be far behind? The world may be braver and newer than you think—it's already happening. (Source: www.rfidtoys.net/author.asp)



Superior MCU Measurement and Control Solutions

The embedded A/D and PWM functions within the HT46R2X and HT46R5X series of microcontrollers, provide a comprehensive range of control application MCU solutions.

By reducing the need for external A/D and PWM components, significant cost reductions are made possible in application areas such as switching power supplies, intelligent household appliances, multi-function battery chargers as well as a host of other application possibilities.

- ◆ Wide operating voltage: 2.2V~5.5V
- ◆ Industrial specification: -40°C~+85°C
- ◆ Internal UART function

- ◆ Internal PFD function
- ◆ High noise immunity and excellent ESD protection
- ◆ Fully compatible Mask and OTP devices

| Part No. | Program | Data | I/O | Timer | I ² C | A/D | PWM | UART | Package |
|------------|---------|-------|-----|-------------------|------------------|----------|---------|------|------------------|
| HT46R22 | 2Kx14 | 64x8 | 19 | 8-bitx1 | v | 9-bitx8 | 8-bitx1 | -- | 24SKDIP/SOP |
| HT46RU22 | | | | | | | | v | 24SKDIP/SOP/SSOP |
| HT46R23 | 4Kx15 | 192x8 | 23 | 16-bitx1 | v | 10-bitx8 | 8-bitx2 | -- | 24,28SKDIP/SOP |
| HT46R232 | 4Kx16 | 192x8 | 40 | 16-bitx2 | v | 10-bitx8 | 8-bitx4 | -- | 28SKDIP/SOP, |
| HT46RU232* | | | | 8-bitx1, 16-bitx2 | | | | v | 48SSOP |
| HT46R24 | 8Kx16 | 384x8 | 40 | 16-bitx2 | v | 10-bitx8 | 8-bitx4 | -- | 28SKDIP/SOP, |
| | | | | | | | | | 48SSOP |
| HT46RU25 | 16Kx16 | 576x8 | 48 | 8-bitx1, 16-bitx2 | v | 12-bitx8 | 8-bitx4 | v | 48,56SSOP |

| Part No. | Program | Data | I/O | Timer | A/D | PWM | Package |
|----------|---------|-------|-----|---------|----------|---------|-------------------|
| HT46R51A | 1Kx15 | 96x8 | 14 | 8-bitx1 | 12-bitx5 | 8-bitx1 | 18DIP, 20SOP/SSOP |
| HT46R52A | 2Kx15 | 128x8 | 14 | 8-bitx1 | 12-bitx5 | 8-bitx1 | 18DIP, 20SOP/SSOP |
| HT46R53A | 2Kx15 | 192x8 | 22 | 8-bitx1 | 12-bitx8 | 8-bitx1 | 28SKDIP/SOP |
| HT46R54A | 4Kx15 | 208x8 | 22 | 8-bitx1 | 12-bitx8 | 8-bitx1 | 28SKDIP/SOP |

*Available soon.

Advertising Calendar

October Issue #207
Signal Processing
Space Close: Aug. 10

BONUS DISTRIBUTIONS:
ARM Devcon
PCB EAST

November Issue #208
Analog Techniques
Space Close: Sept. 12

Call: Shannon Barraclough
(860) 872-3064
Shannon@circuitcellar.com

Big things come in small packages...

- Highlight key product developments
- Introduce new products
- Recruit the best engineers
- Create valuable brand awareness

Reach the right audience, right now with Circuit Cellar's Idea Boxes!



Give the Customer More!

If your company has a physical storefront, you may qualify to become a *Circuit Cellar* distributor.

Circuit Cellar will:

- Entice customers to stay in your store longer
- Attract repeat walk-in business
- Serve as a useful product demonstration tool

Contact: Sean Donnelly
sean@circuitcellar.com

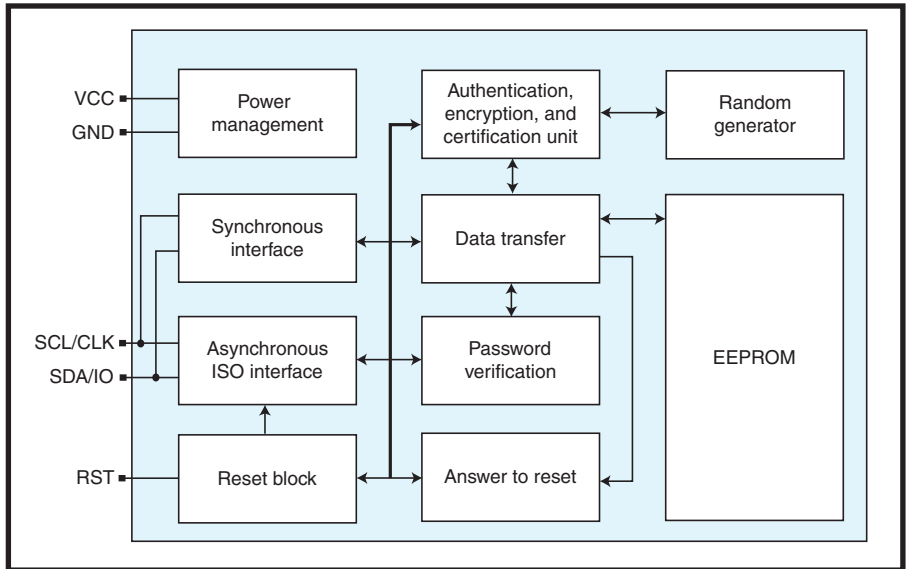


Figure 1—Atmel offers serial EEPROMs with a difference, namely built-in security features, such as passwords, authentication, and encryption. The AT88SC CryptoMemory lineup ranges from 1 Kb to 256 Kb, and is offered in DIP, surface-mount, and Smartcard packaging.

round the vehicle's hi-tech security system. The car, a Mercedes S-class, was protected by a fingerprint recognition system.

By Jonathan Kent
BBC News, Kuala Lumpur
<http://news.bbc.co.uk/2/hi/asia-pacific/4396831.stm>

Constantly losing your keys? Prone to forgetting your passwords? Fear someone will fake your ID? Consider getting "chipped" (see Photo 1).

Implanted RFID tags are already at work in livestock and pets, so isn't it just a matter of time before humans get tagged?

Actually, it's already happening. One company, VeriChip, claims to have thousands of "installations" using their FDA-approved implantable RFID tags. For now the company is mainly targeting relatively niche healthcare applications, namely storing ID and medical information for patients who are likely to experience loss of consciousness (e.g., diabetic



Photo 2—Pick a card, any card. The AT88SC demonstration kit (AT88SC25616C-DK) packages the memory chips in Smartcard format and comes with a USB-based reader for connecting to your PC.

coma) or are otherwise unable to identify themselves (e.g., stroke victims, the mentally disabled, and infants) in the event of a medical emergency.

More provocative is the case of Amal Graafstra, a fellow who had himself “chipped” and then hacked his car, motorcycle, door locks, and PC to automatically open sesame at the wave of his hand.^[2]

Many barrels of Op-Ed page ink will be spilled before a “chipped” future comes to pass.^[3] At the same time, there’s no doubt that confidentiality and authenticity concerns are no longer confined to traditional “security applications” (e.g., banks, alarm systems, and the military).

Indeed, if there’s a message in this month’s column, it’s that every embedded designer should take a moment to consider the situation, no matter how innocuous or traditionally “trustworthy” your application. What is the downside if your product is stolen, copied, adulterated, or otherwise perverted in a malicious manner? For instance, if fake drugs are already a problem, can the entire food chain be far behind? Considering such examples as the recent E. Coli outbreaks and contaminated pet food. I think the answer is clear.

HIDDEN MEMORIES

Usually, when I write about a chip, I’ve got a few preconceived ideas of how it might be useful in particular applications. But, when it comes to the general subject of “more security” for embedded applications, I must admit the variety of issues and use cases has me a bit stymied. I’ve got a strong feeling that designers need to think about this stuff, but it’s hard to get a handle on specific tactics for specific applications.

With that said, let’s take a look at the AT88SC “CryptoMemory” line-up from Atmel (see Figure 1). As the name implies, it’s a family of EEPROMs (capac-

ities from 1 to 256 Kb) that includes passwords, encryption, and other security features that harden it against hack attacks. The part is very versatile, which means it may well offer an answer to your security concerns, at least once you figure out exactly what those concerns are and how to deal with them.

The AT88SC is like the chips you’d typically find in a “Smartcard” used, for example, in a building’s security and access control system. However, two features set it apart in ways that broaden the spectrum of potential applications.

First, in addition to the “Smartcard” format, Atmel is offering the line-up in a variety of standard IC packages, including SOIC and even DIP. That means you can incorporate the chips in designs that don’t need or can’t accommodate the packaging and reliability issues associated with a bulky “Smartcard” connector.

Smartcards typically use a specialized asynchronous serial interface (ISO 7816), and the Smartcard versions of the AT88SC naturally support that option. However, the IC-packaged versions use a synchronous clock serial interface. This so-called “Two-Wire Interface” (TWI) is similar to I²C, but not exactly the same. Where I²C uses

| AT88SCxxxx Device | Zones | Capacity | Multibyte transfer |
|-------------------|-------|----------|--------------------|
| 0104C | 4 | 1 Kb | 16 |
| 0204C | 4 | 2 Kb | 16 |
| 0404C | 4 | 4 Kb | 16 |
| 0808C | 8 | 8 Kb | 16 |
| 1616C | 16 | 16 Kb | 16 |
| 3216C | 16 | 32 Kb | 128 |
| 6416C | 16 | 64 Kb | 128 |
| 12816C | 16 | 128 Kb | 128 |
| 25616C | 16 | 256 Kb | 128 |

Table 1—The various parts in the AT88SC family partition their EEPROM memory into four, eight, or 16 zones, depending on capacity. Each zone can be configured individually with its own security setting (passwords, encryption, etc.). Zone(s) can also be configured with security disabled, allowing that portion of the memory to mimic a standard (i.e., no security) EEPROM.

the least significant bit of an address to differentiate between reads and writes, TWI uses separate read and write commands. In any case, it appears to be a fairly simple matter to interface one or more (up to 15) AT88SC chips to any MCU with a couple of spare pins for bit-banging.

The AT88SC chips run on any voltage from 2.7 to 5.5 V and the clock serial interface runs from DC to 1 MHz, so there’s no critical timing. Since the open collector data line (SDA) relies on a pull-up resistor, do note that designs using higher frequencies

(and/or more bus loading/devices) require a stronger pull-up (i.e., weaker resistor), which increases power consumption.

Otherwise, the chip itself draws about 5 mA in operation and 100 µA in standby (5.5 V). The standby consumption seems a little high, but for many applications a viable option is simply to power down the chip between accesses. That works for situations where the EEPROM is idle much of the time and a bit of power-up delay (up to 70 µs) can be tolerated. The active power consumption is low enough that you can likely just power the chip with an MCU I/O pin for easy on/off control.

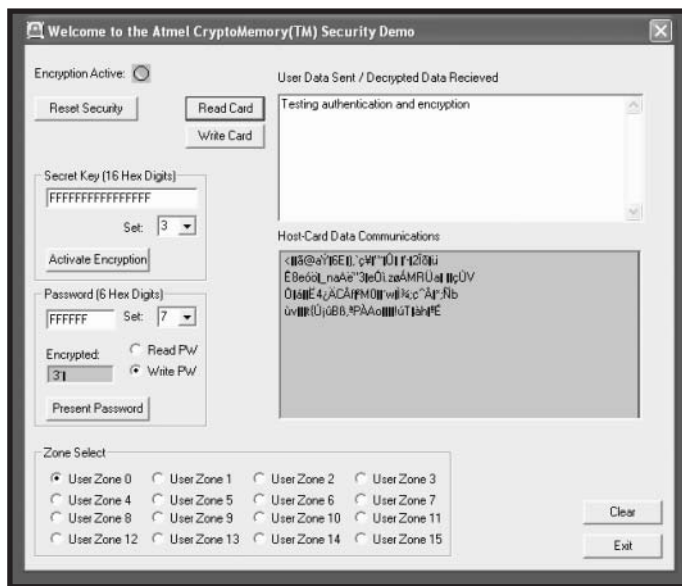


Photo 3—Here you can see encryption in action. The top window shows the cleartext stored in the EEPROM, while the lower window shows how the data is encrypted once it leaves the chip to stymie eavesdroppers. Note that the encryption algorithm is dynamic (i.e., the encrypted version is recalculated and will change every time the EEPROM is read).

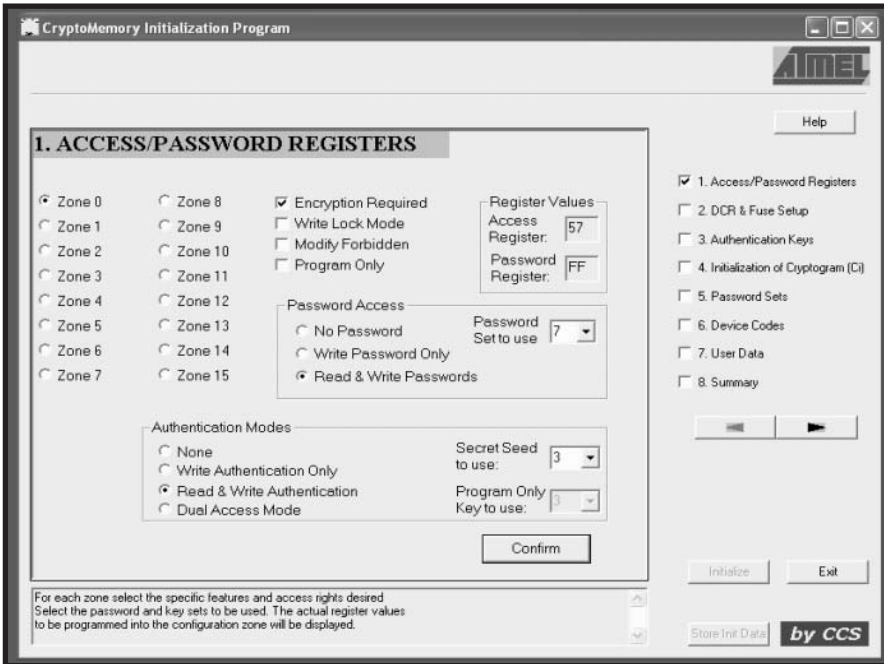


Photo 4—Unlike a regular memory chip, the AT88SC requires significant set up, such as setting zone-by-zone security options, passwords, etc. Get a bit wrong, such as inadvertently blowing a fuse or garbling a password, and you'll be sorry. The Initialization utility that comes with the demonstration kit makes it quick and easy to get the AT88SC configured correctly.

Putting aside the security features for a moment, AT88SC operation is similar to that of a typical serial EEPROM. Basic read and write commands require 4 bytes: one for the command (e.g., read or write), two that specify the byte address, and one that specifies the number of bytes of data to transfer. Smaller capacity parts can transfer up to 16 bytes at a time, while the larger units handle bursts of up to 128 bytes.

Keep in mind all the security features in the world won't help a memory chip that can't hold its bits. Notably, the concern for an EEPROM is what happens if power fails during the long (multi-millisecond) write cycles.

The AT88SC solves the problem with a nifty double-buffered write option called "anti-tearing." When enabled, writes are directed to an intermediate EEPROM buffer and then transferred to the ultimate EEPROM destination in a second step. If the first (buffer) transfer is interrupted, the main array is left untouched. On the other hand, if the second (main array) transfer is interrupted, the chip will redo it at the next power-up.

There's no free lunch, so enabling the "anti-tearing" option does involve some compromises. Multibyte writes are limited to 8 bytes rather than the larger transfers (e.g., 16 or 128 bytes) otherwise allowed. Furthermore, the scheme essentially doubles the time (and the energy consumed) for write cycles since you have to wait for the second (main array) write to complete successfully before proceeding. Similarly, you must be prepared for the possibility of a much longer (milliseconds versus microseconds) power-up delay if a prior failed transfer needs to be redone.

TRUST BUT VERIFY

So far, we've seen you wire up and talk to the AT88SC pretty much like any other serial EEPROM. Now let's get into the security features that set it apart from its loose-lipped cousins.

Protecting the family bits starts with a separate area of "configuration" EEPROM that defines the various security settings. Nothing prevents an application from accessing the configuration memory per se, but most scenarios

would have the AT88SC configuration initialized in production and only rarely (e.g., change passwords) modified by the application.

Key to the versatility of the chip is that the main memory array is broken into zones, which feature independent security settings (see Table 1). A typical application might find the chip configured with a combination of "public" information accessible to all and one or more "private" areas, each with its own access rights and limitations. The chip holds up to eight password sets comprising separate 3-byte read and write passwords, and multiple zones can be assigned a single password to accommodate larger (i.e., bigger than a single zone) data items.

The 3-byte passwords provide a basic level of protection. Of course, it wouldn't take long to crack them just by brute force. Let's see, trying all 2^{24} combinations at roughly 2^{13} combinations per second (i.e., assuming something on the order of 100 μ s per trial) means it would only take about 2^{11} seconds, or a little more than half an hour, to break in. To counter that threat, the chip has options to restrict failed password attempts to four or eight trials, after which access to the zone is permanently blocked.

Of course, the failed password

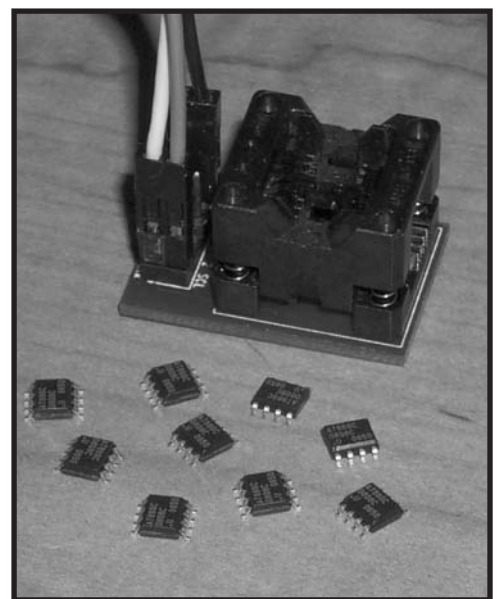


Photo 5—Bring a magnifying glass and tweezers. The AT88SC development kit (AT88SC-DK1) comes with a ZIF socket adapter for experimenting with the oh-so-tiny surface-mount chips.

Listing 1—Here’s an example of the Atmel CryptoMemory library and API at work. It makes accessing the AT88SC easy and only consumes about 4 KB of code space. Just keep in mind that there are strings attached if you want to use it with anything other than an AVR.

```
#include "cm_Lib.h"
#include "ll_port.h"
// These 2 files are required to allow access to the library.

// How many unique CryptoMemory devices are on the bus.
#define NUM_DEVICES 1

// Device Address Table. One entry per CM device on the bus
const uchar device_addresses[NUM_DEVICES+1] PROGMEM = {0xb, 0};

// Registers to store the state of each CM device address on the bus.
CM_DEVICE_REGISTERS device_registers[NUM_DEVICES];

// Device Index table used by cm_FindDeviceIndex()
const uchar cm_DeviceIndexes[16] PROGMEM =
{CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR,
CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR,
CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR, 0,
CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR, CM_NUM_DEV_ADDR};

int main(void) {
    RETURN_CODE ucReturn;

/* This is a character array to hold some data for the example. */
    unsigned char ucData[8] = {0,1,2,3,4,5,6,7} ;

/* This is a character array to hold a secret seed for the example. */
    unsigned char ucSecretSeed[8] =
    {0xa5,0x16,0xd2,0x39,0xa4,0x55,0xf6,0xc7} ;

/* This is a character array to hold a password for the example. */
    unsigned char ucPasswd[3] = {0xdd,0x42,0x97} ;

    ll_PowerOn() ; // Calls the low level function to start up the CM bus.
    if (cm_Init() != SUCCESS)
        return 0 ;

/* cm_Init() initializes the CryptoMemory Devices (defined in
device_addresses) and the allocated memory for them. */

    ucReturn = cm_SetUserZone(DEFAULT_ADDRESS, 3, FALSE);
/* Tells the device we will be working with User Zone 3.
if (ucReturn != SUCCESS) return ucReturn ;

    ucReturn = cm_VerifyCrypto(DEFAULT_ADDRESS, 1, ucSecretSeed, NULL, FALSE);
/* Activates the authentication mode using key set 1.
NULL passed as the next to last parameter means create random number.
FALSE passed as the last parameter means no encryption. */
if (ucReturn != SUCCESS) return ucReturn ;

    ucReturn = cm_VerifyPassword(DEFAULT_ADDRESS,ucPasswd,7,0);
/* Presents the Write Password, Password set 7, to the device for authentication.
Passwords are automatically encrypted when in Authentication Mode.
The second parameter is the password tested. */
if (ucReturn != SUCCESS) return ucReturn ;

    ucReturn = cm_WriteUserZone(DEFAULT_ADDRESS, 0, ucData, 16);
/* Writes the data defined in ucData to the Device
(to previously specified User Zone 3). */
if (ucReturn != SUCCESS) return ucReturn ;

    ucReturn = cm_ReadUserZone(DEFAULT_ADDRESS, 0, ucData, 16);
/* Reads back the data (from User Zone 3).*/
if (ucReturn != SUCCESS) return ucReturn ;

    ucReturn = cm_DeactiveSecurity(DEFAULT_ADDRESS);
/* If we want, cm_DeactiveSecurity() deactivates the Authentication mode.*/
if (ucReturn != SUCCESS) return ucReturn ;

    return SUCCESS ;
}
```

count accumulates across power cycles, as it must to prevent a brute-force attack. However, that probably makes it too finicky for any situation where typo-prone humans are involved in the password entry, so there’s also an “unlimited trials” option. In any case, the ultimate rigor of the password scheme is compromised if the passwords travel as “cleartext” between the host and the AT88SC and are thus subject to eavesdropping.

Upping the security ante is “authentication,” which relies on an 8-byte key as the means for both the host and the AT88SC to verify that the chips on the other end of the line are who they say they are. Authentication uses a challenge-and-response scheme that has both the host and the AT88SC able to determine that the other knows a secret without passing the secrets themselves. That approach should stymie eavesdroppers, but in addition, as with passwords, authentication can be configured to lock-up after four or eight failures. In any case, a brute-force attack isn’t likely to succeed since a full search would take 2⁵¹ seconds (i.e., 2⁶⁴ trials at 2¹³ per second), which is...a long time.

Note that even with password and authentication protection, the system is still subject to real-time “man-in-the-middle” attacks. Yes, it would take an authorized user to “unlock” the AT88SC, but after that, cleartext data transfers could be monitored. The AT88SC counters the threat of eavesdropping with an option to encrypt data transfers across the channel.

Besides passwords, authentication, and encryption, there are additional zone-by-zone restrictions you can choose from. Once initial data is loaded into a zone, it can be set to read-only to prevent further modification. One variation breaks the zone into 8-byte pages, the read-only or read-write status for each of those 8 bytes is controlled by a bit in a ninth byte. Another “Countdown” mode allows bits to be programmed, but only from a “1” to a “0.” Besides setting the security for each zone, the configuration memory also handles device-wide settings, such as the aforementioned

four-, eight-, or unlimited password trials option. There are also fields you can program with card-specific IDs, such as a 128-bit card number.

Once everything is configured properly, a "Permanent" (PER) fuse can be blown to lock in the configuration. Once the PER fuse is blown, the only items in the configuration memory that can be changed are passwords (presuming you've got the password to do that). The advice here is to check your configuration carefully, because once the PER fuse is blown, there's no going back.

A car thief can steal your car by stealing your keys (or your finger), but that's not the only way. Like cars, security chips must also be able to stymie someone who gets under the hood and tries to "hot-wire" it. To that end, the AT88SC incorporates various hardware measures to counter sophisticated break-in techniques, such as physical examination of the die itself (i.e., pop the lid) or stressing the device (e.g., voltage, clock frequency) to probe for weakness.

If some of the details seem nebulous, it's not entirely my fault. It turns out there's one more spy-versus-spy trick up Atmel's sleeve, namely you've got to sign a nondisclosure agreement (NDA) to get a copy of the "real" datasheet with all the security details, such as the particulars of the encryption algorithm.

PLAYING WITH A FULL DECK

Atmel offers two different CryptoMemory evaluation kits. The "demonstration" kit (AT88SC25616C-DK) uses AT88SCs packaged in Smartcard format, while the "development" kit (AT88SC-DK1) supports chip versions, in this case the tiny 4 × 5 mm SOIC.

The demonstration kit comes with five 16-Kb AT88SC-based Smartcards and a USB card reader that connects to your PC (see Photo 2). Various PC utility programs are provided to handle programming, verification, and testing of the Smartcards. For example, a "Security" utility let's you try out AT88SC features, such as passwords and encryption (see Photo 3).

Of these, the "Initialization" utility is super helpful (see Photo 4). Fact is,

initializing the AT88SC configuration involves setting lots of bits in lots of configuration memory locations. It's tedious and error prone to figure out manually. Furthermore, the stakes are a bit higher than with most chips, since getting a single bit wrong may lock something up permanently.

The "Initialization" utility software handles all of the bit-twiddling details of AT88SC configuration for you, leading you step-by-step through the process of choosing device options (e.g., password trial count) and the security settings for each zone. It's easy to jump back and forth between the steps, playing around until you get things just right. Once you're satisfied, just click on "Initialize" and the Smartcard is configured and ready for application duty. There's also the option to save a text file with a complete dump of the configuration. That's highly recommended during evaluation ("...darn, what did I set that 64-bit authentication to?"), as well as using labels to keep track of which card is which.

The AT88SC-DK1 "development" kit takes a bit of a different tack. Besides the fact that it targets the chip version with a ZIF-socket-jumper-cable lash-up, the software is quite different as well (see Photo 5). This kit is all about a CryptoMemory library and application program interface (API) Atmel has developed for embedded applications (see Listing 1).

The library version that comes with the DK1 kit targets an AVR-based STK500 evaluation board. According to Atmel, the library can theoretically be ported to other little Endian MCUs (e.g., ARM) that have about 4 KB to spare. However, getting access to the source code for the library requires signing the aforementioned NDA, as well as convincing Atmel that you have a justifiable need to know (e.g., medical applications). In short, this kit is really only useful if you plan on hooking the AT88SC to an AVR. If not, another easy (and cheaper) prototyping solution is to just score some AT88SC's in the eight-pin DIP package.

CENTS AND CENTS-ABILITY

Whether your application needs security upgrades is something all

embedded designers should think about. For those who conclude such are in order, the AT88SC represents a very cost-effective alternative. Note that besides Smartcard and IC packages, Atmel offers an RF version of the AT88SC for contactless applications.

Volume pricing is generally under a buck, maybe way under depending on capacity, volume, and packaging. The system cost adder shrinks further if you're already using a serial EEPROM in your design. All you have to do is get a larger AT88SC, configuring a portion to work just like your existing EEPROM (i.e., no security) and using the extra capacity for new security features.

Many vehicle thefts occur when the keys are left in the ignition. Don't make the same mistake with your embedded design. ☒

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] R. Cockburn, "The Global Threat of Counterfeit Drugs: Why Industry and Governments Must Communicate the Dangers," *PLoS Medicine*, Issue 4, 2005, www.fip.org/files/fip/counterfeit/The%20Global%20Threat%20of%20Counterfeit%20Drugs.pdf.
- [2] A. Graafstra, "Hands On: How Radio-Frequency Identification and I Got Personal," *IEEE Spectrum*, March 2007.
- [3] K. Foster, "RFID Inside: The Murky Ethics of Implanted Chips," *IEEE Spectrum*, March 2007.

SOURCE

AT88SC25616C-DK Demonstration kit and AT88SC-DK1 development kit
Atmel Corp.
www.atmel.com

IDEA BOX

THE DIRECTORY OF PRODUCTS AND SERVICES

AD FORMAT: Advertisers must furnish digital submission sheet and digital files that meet the specifications on the digital submission sheet. **ALL TEXT AND OTHER ELEMENTS MUST FIT WITHIN A 2" x 3" FORMAT.** Call for current rate and deadline information. Send your disk and digital submission sheet to: IDEA BOX, Circuit Cellar, 4 Park Street, Vernon, CT 06066 or e-mail adc copy@circuitchellar.com. For more information call Shannon Barraclough at (860) 875-2199.

The Suppliers Directory at www.circuitchellar.com/suppliers_dir/ is your guide to a variety of engineering products and services.

Start designing with Freescale's advanced S12X family!

Technological Arts



freescale
Alliance Member

Toll-free: 1-877-963-8996
www.technologicalarts.com

USB

Add USB to your next project—it's easier than you might think!

- USB-FIFO up to 8 mbps
- USB-UART up to 3 mbps
- USB/Microcontroller boards pre-programmed with firmware
- 2.4GHz ZigBee™ & 802.15.4
- RFID Reader/Writer

Absolutely NO driver software development required!

www.dlpdesign.com



PHYTEC

phyCORE® OEMable Single Board Computers

XScale: PXA270, PXA255

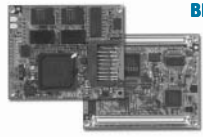
ARM: LPC3180 (ARM9); LPC22xx, LPC229x, AT91 (ARM7)

PowerPC: MPC5554, MPC5200B, MPC565, MPC555

ColdFire: MCF5485 **x86:** Elan SC520

C166/XC16x/ST10/8051 **CAN**

Blackfin: BF537



Faster-to-Market: Save time by integrating a PHYTEC Single Board Computer Module into your target circuitry.

Make-or-Buy: Why make your own when you can buy PHYTEC off-the-shelf solutions, cost-effective to 1000s units/year?

Integrated Support Services: Let PHYTEC assist you in the design of your end product; from tools and RTOSes to production. Our hardware is bundled with leading compilers (Keil, IAR, CodeWarrior), RTOSes (WinCE, Linux) and debuggers.

Immediate Support: Talk to PHYTEC technical staff with every call. No waiting for answers.

Your OEM solution: With 20 years design, production, and integration experience, PHYTEC is your OEM partner.

PHYTEC America, LLC ■ 203 Parfitt Way SW, G100 ■ Bainbridge Island, WA 98110 USA
www.phytec.com ■ (800) 278-9913 ■ www.phycore.com

The \$69 PLC

Work as Stand-Alone Ladder Logic PLC.
Or as Smart Remote I/Os of PC/PLCs.
RS485 allows 256 units to be networked.



E10-Relay+ RS485

Opto232

RS232
RS485

E10-npn+

Incredibly Easy to Program!
Our software is used by many colleges for teaching PLCs!

Get Free Ladder Logic Simulator:
www.tri-plc.com/cci.htm

TRI
Triangle Research

Tel: 1-877-874-7527 - PLC specialist since 1993

PRICING Alert!!!

BUY YOUR CCS PIC® MCU C COMPILER BEFORE PRICES CHANGE!

Current pricing until **31-AUG-07**

GET \$10 off YOUR PURCHASE WHEN YOU USE DISCOUNT CODE BPC*.

FOR MORE INFO AND A COMPLETE PRICE LIST, VISIT:

www.ccsinfo.com/BPC

(*Not valid with any other offer. Shipping not included.)

262.522.6500 x35



PIC® is a registered trademark of Microsoft Technology Inc. in the U.S. and other countries.

Low Cost PGM/DEMO Kits



USB-DONGLE

The **USB-DONGLE** and **Derivative Boards** allows quick and easy ICP/ISP programming of many popular microcontroller families such as the LPC9xx, ARM7 LPC2xx and 89V52X2 from NXP Semiconductors. The kit includes a Virtual COM Port Driver that allows hex files to be downloaded and programmed using Flash Magic or other common utilities. The unit also provides a low cost platform for testing or prototyping of simple microcontroller based designs (blink an LED, measure I/O, test a Timer or PCA output). Low cost **Derivative Boards** are available for many different microcontrollers from NXP. Please consult our website for details.

Derivative Boards

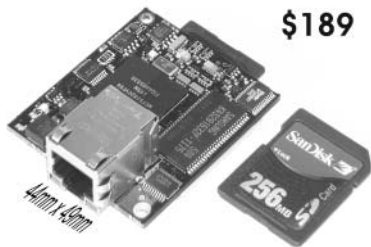


LPC9103 HVSON10 ARM7 LPC2103 LQFP48

XA Development Kits, I2C, MDIO & SPI tools in stock, visit www.teamfdi.com for details.

FDI Future Designs, Inc.
2702 Triana Blvd
Huntsville, AL 35805
(256) 883-1240
Fax (256) 883-1241

www.teamfdi.com
VISA/MC/Amex



\$189

Tiny Linux controller

16MB fast SDRAM
64MHz Coldfire MCU
4.5 MB flash memory
SD card socket (to 2GB)

10/100 Ethernet
3 serial ports (RS232)
1 CAN port
LCD/KPD port

Eclipse/CDT dev. env. Free serial debugger

55 I/O pins & capabilities to burn...

www.steroidmicros.com



Embedded DSP Hands-On Workshops

Avoid confusing datasheets.

Rapidly learn to program, design and debug your application.

Reinforce course concepts with included development tools.

Let us help you accelerate your design process. Visit:
www.dsp-workshops.com

USB Data Acquisition

ADU208 -USB Relay I/O Interface



Complete SDK Online at: www.ontrak.net

FEATURES

- 8, 5-AMP relay outputs
- 8, ISOLATED digital inputs
- Port Powered, Aux 5VDC output \$189.00 QTY 1

Other Models:

ADU200- 4 Channel Version with RS232 \$139.00
ADR218- Solid-State Version 8 Channel \$225.00
ADU100- 3 CH, 16-Bit ISOLATED Analog Inputs, PGA, 4 digital I/O, RS232 and 5 AMP Relay Output \$199.00

ONTRAK CONTROL SYSTEMS INC.
PH: (705) 671-2652 Fax: (705) 671-6127

www.ontrak.net

ValueCAN

The High Value Tool For Controller Area Network

- USB to CAN
- Simple software analyzer included
- DLL with examples for custom applications
- PC isolated from CAN
- 100% bandwidth at 500Kb

Only

\$295



www.intrepidcs.com

High-speed USB 2.0 + Xilinx FPGA Complete Software API



XEM3010-1500P:

- 1,500,000-gate FPGA
- Programmable PLL
- Over 110 I/Os
- 32 MB SDRAM
- Configuration PROM
- 0.8-mm expansion
- \$399.95 / Qty 1
- \$349.95 / Qty 10

XEM3001:

- 400,000-gate FPGA
- Programmable PLL
- Over 80 I/Os
- 0.1" expansion headers
- Business-card size
- \$199.95 / Qty 1
- \$174.95 / Qty 10

FrontPanel Software API:

- Windows XP, Mac OS X, Linux
- C/C++, Python, Java
- Configure FPGA and communicate with your design
- The easiest way to integrate USB into your product
- Use for image capture, control, test equipment, etc.
- Up to 38 MB/s transfer rate!

5% off with Coupon Code: CKTCLR73*

* Valid for first order only. Void 30-days after issue publication.

Opal Kelly Visit us online at: www.opalkelly.com

WIRELESS RS-232

WCSC (Willies Computer Software Co)



- Extremely easy to install and use
- Distances up to 100 meters (329 feet)
- Bluetooth Serial Port Profile
- Communicates with Bluetooth enabled PDAs, Laptops, Smartphones, & Computers
- No programming or special software needed

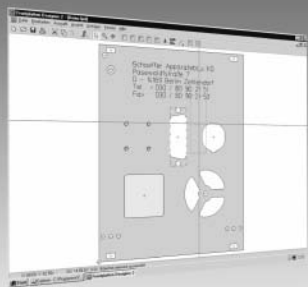
Other Products

- Professional serial communication libraries & development tools.
- PCI, PCMCIA, USB, Universal PCI, & ISA multiport RS232, Rs422, & Rs485 cards, Bluetooth/USB Dongles

<http://CircuitCellar.wcscnet.com>
sales@wcscnet.com (281)360-4232

Front Panels?

Download the free Front Panel Designer to design your front panels in minutes



Order your front panels online and receive them just in time

www.frontpanelexpress.com

Unrivaled in price and quality for small orders

USB Device Stack

- Compatible with Windows USB 2.0 Host Class Drivers:
 - Mass Storage
 - Mouse
 - Ethernet over USB
 - Serial
- Composite Devices
- ARM, ColdFire, Philips ISP 1161, 1181, 1362, 158x, 1761
- Low Cost, Royalty-Free
- Full Source Code
- Standalone or RTOS
- Host and OTG Available

Micro Digital Inc
RTOS Innovators

www.smxrtos.com/usb

CHINA PCB SUPPLIER

- DIRECT SALE FROM CHINA
- PROTOTYPE TO PRODUCTION

instant online quote
shopping cart ordering system
China competitive prices
free Electrically test



web: <http://www.pcbcart.com>
email: sales@pcbcart.com
Tel: +86-571-87012819
Addr: No. 2 Huxiang Road,
Hangzhou, P.R. China

WWW.PCBCART.COM



...the key to your success!

- KEYPAD
 - IMD
 - P+R
 - M+R
 - SILICONE
- CONNECTOR
 - FMC
 - ELASTOMERIC
 - HSC
 - FFC
- MEMBRANE SWITCH
- GASKETS
- PCB
- FPC

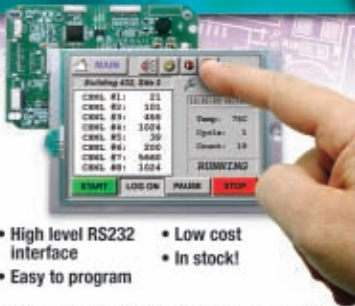
2009 108th St Suite 901
GRAND PRAIRIE, TX 75060
800-654-3662
972-606-1114 FAX
WWW.ARCUSAINC.COM



Multiple solutions for your electronic needs

©2007 ARC-USA, INC.

Add a color touch interface
to your embedded product!



- High level RS232 interface
- Low cost interface
- Easy to program
- In stock!

Add color graphics to any 8/16 bit embedded system.
Easy, fast and flexible. Up and running in hours!

REACH
TECHNOLOGY INC.

www.reachtech.com • 510-770-1417

842 Boggs Avenue • Fremont, CA 94539

Instant LCD



- Versatile Programmable Module
 - Serial/PC/USB/Parallel Interface
 - AVR/BASIC Stamp/VB Compatible
 - Onboard Flash Bitmap Memory
 - Downloadable TTF Fonts
- 2.7" or 5.6" TOUCH Color TFT LCD
 - 240x160 or 320x240 resolutions
 - Transflective w/ LED Frontlight (2.7")
 - Transmissive w/ 350 nit backlight (5.6")
 - 512 colors or 65,535 colors

EARTHLCD.COM We Make LCDs Work.™

MSP430
CPU12
ARM
AVR

Easy to Use!
Low Cost!
State of the Art!

IMAGECRAFT compilers

starting at
\$199!

high powered,
development,
garage
powered
prices!



Fully Functional
Demos!

Plus:
AVR, ARM & HCS12 Hardware Kits

www.imagecraft.com

info@imagecraft.com

(650) 493-9326 • FAX (650) 493-9329

MYLYDIA, INC.

Layout Gerber,
Prototype Making

QUICK TURN

PCB & Turnkey
@
The Best Prices

Tel: 800-695-9342

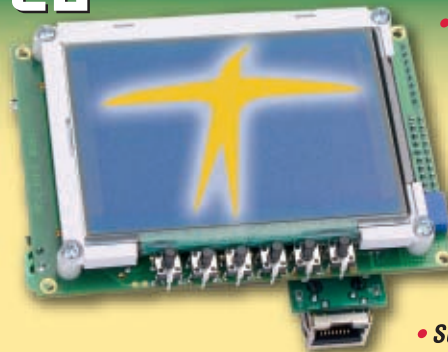
Sales@mylydia.com

WWW.MYLYDIA.COM

EL™

Networked, x86 Graphic LCD controller with Ethernet,
CF FAT16, 24-bit ADC, DAC, solenoid drivers, and relay.

Qty 100
\$99



- 4.2 x 3", Program in C/C++
- 192x128 pixel (71x54 mm) monochrome graphic LCD, 6 Keys
- 100M Base-T Ethernet, 2 RS232, I/O, RTC
- 16 ch 24-bit ADC and 4 ch 16-bit ADC
- 8 ch. 16-bit DAC and 2 ch 12-bit DAC
- CompactFlash card with FAT file system support
- Solenoid drivers and Reed Relay

50+ Low Cost Controllers with ADC, DAC, solenoid drivers, relays, PC-104, CF, LCD, DSP motion control, 10 UARTs, 300 I/Os. Custom board design. Save time and money.

TERN
INC.

1724 Picasso Ave., Suite A, Davis, CA 95616 USA

Tel: 530-758-0180 • Fax: 530-758-0181

www.tern.com

• sales@tern.com

Net Modules



programmable
revolutionary

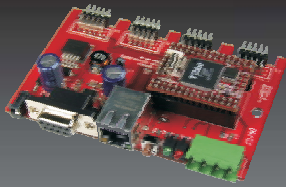
SAVE 5% enter coupon code CCI at checkout

ABACOMdirect.com



tel: +1(416) 236 3858
fax: +1(416) 236 8866

32io SERVER



control & monitor
anything anywhere
over the internet

SAVE 5% enter coupon code CCI at checkout

ABACOMdirect.com



tel: +1(416) 236 3858
fax: +1(416) 236 8866

Wireless I/O

extend your digital I/O's
over 100's of feet up to
tens of miles - in both
directions - with the
16IO-SSRT RF transceiver
modules



SAVE 5% enter coupon code CCI at checkout

ABACOMdirect.com



tel: +1(416) 236 3858
fax: +1(416) 236 8866

Solve complex signal acquisition problems...

- positioning & control
- environmental
- acceleration
- transients
- pressure
- vibration
- sonar
- GPS



- Linux Driver
- Guaranteed in stock
- Customization available
- 16-bit analog inputs and outputs
- Million sample FIFO eliminates interrupts
- Wide analog input and output ranges
- -40°C to +85°C Standard

www.stx104.com
Apex Embedded Systems

sales@stx104.com • 608-256-0767 x24

SpectraPLUS 5.0 Audio Spectrum Analysis

Features

Sound Card based I/O
FFT sizes to 1048576pts, 1/96 Octave
Up to 24 bit, 200kHz sampling rates
3-D Surface and Spectrogram
Digital Filtering, Signal Generation
THD, IMD, SNR, Transfer Functions
DDE, Macros, Data Logging,
Vibration Analysis, Acoustic Tools

FREE 30 day trial!
www.spectraplus.com

PHS Pioneer Hill Software
360 697-3472 voice
pioneer@telebyte.com

BG MICRO The Electronics Supplier!



Serving the Electronic industry for over
25 years with New and Surplus
Electronic Parts, Kits, LCD's, LED's,
Audio Parts, Meters, Switches, Power
Supplies, Fans, Relays, Tools, etc....
Come On Down and Visit Our Internet
Store at www.bgmicro.com, or you can
call our toll free number for a live
person.

We'll be Expecting you!

WWW.BGMICRO.COM

1-800-276-2206



Hands-on
Training
for Rabbit Developers

Learn More at
Rabbit-U.com

07114

www.can232.com

Only
\$108
€89

CAN232 Features:

- Free sample programs
- 8-15VDC supply via CAN
- Timestamp in mS
- Small size 2.7" by 1.2"
- 150% 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 independent
- 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

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

**Serial • Ethernet
Web Server**

**Simple, DTE, DCE
No SW Changes**

**\$99.⁹⁵
Qty 1**



+1 630-245-1445
Naperville, Illinois USA
www.gridconnect.com

gridconnect.



ANYONE
Can Now Easily
Hand Solder Surface-
Mount Components!
**Even A 10
Year Old!**

www.schmartboard.com




**Weather
Instruments**
for PCs

AAG
electrónica

www.aagelectronica.com

**PRESTO In-System USB
Programmer**



Very fast and flexible in-system programmer for Microchip & Atmel MCUs, serial EEPROMs & Flash PROMs, devices with JTAG (CPLDs, MCUs, FPGAs, FPGA PROMs, ...)

SIGMA - unique logic analyzer
USB modules - parallel & serial

ASIX ASIX, Prague, Czech Rep.
info@asix-tools.com

www.asix-tools.com

**Embedded Ethernet
ONLY \$98**

- 10Base-T Ethernet
- 186 Processor @ 40 MHz
- DOS w/ Flash File System
- 16 Digital I/O • 5V DC
- Console / Debug Port
- Hardware Clock/Calendar
- Socket for DiskOnChip
- 512K DRAM & 512K Flash • 3.75" x 2.50"

NEW! picoFlash

\$129
Development
System

Development Kit Includes:

- picoFlash Controller
- Borland C/C++ 4.52
- TCP/IP, PPP & Web Server
- Serial Driver Library
- AC Adapter and Cables

Call 530-297-6073 Email sales@jkmicro.com
On the web at www.jkmicro.com

JK microsystems

**ALL
ELECTRONICS**
CORPORATION

Electronic and Electro-mechanical Devices, Parts and Supplies.
Wall Transformers, Alarms, Fuses, Relays, Opto Electronics, Knobs, Video Accessories, Sirens, Solder Accessories, Motors, Heat Sinks, Terminal Strips, L.E.D.S., Displays, Fans, Solar Cells, Buzzers, Batteries, Magnets, Cameras, Panel Meters, Switches, Speakers, Peltier Devices, and much more....

www.allelectronics.com
Free 96 page catalog
1-800-826-5432

CAN-4-USB FX

USB to CAN Interface
USB 2.0 Hi-Speed 480Mbps!

Other companies may claim USB 2.0 but if it isn't Hi-Speed it is only 12Mbps.

2007 marks our 11th year of selling CAN interfaces in over 30 countries!

\$249
USD.



Zanthic Technologies Inc.
403-878-2202
www.zanthic.com

PRINTED CIRCUIT BOARDS

QUALITY PRODUCT
FAST DELIVERY
COMPETITIVE PRICING


PROTOTYPE
through
PRODUCTION

- Aluminum Backed PCB
- Single & Double sided
- SMOBC/RoHS
- LPI mask
- Through hole or SMT
- Nickel & Gold Plating
- Routing or scoring
- Electrical Testing
- Artwork or CAD data
- Fast Quotes
- Flex Circuits

SPECIAL OFFER:
10 pcs (3days) 1 or 2 layers \$249
10 pcs (5days) 4 layers \$695
(up to 30sq. in. ea.) includes tooling, artwork, L.P.I. mask & legend

PULSAR, INC

9901 W. Pacific Ave. Franklin Park, IL 60131 • Phone 847.233.0012
Fax 847.233.0013 • www.pulsar-inc.com • sales@pulsar-inc.com



Full Speed It writes your USB Code!

NEW! HIDmaker FS for Full Speed FLASH PIC18F4550

Creates complete PC and Peripheral programs that talk to each other over USB. Ready to compile and run!

- Large data Reports
- 64,000 bytes/sec per Interface
- Easily creates devices with multiple Interfaces, even multiple Identities!
- Automatically does MULTITASKING
- Makes standard or special USB HID devices

NEW! "Developers Guide for USB HID Peripherals" shows you how to make devices for special requirements.



Both PC and Peripheral programs understand your data items (even odd sized ones), and give you convenient variables to handle them.

PIC18F Compilers: PICBASIC Pro, MPASM, C18, Hi-Tech C.

PIC16C Compilers: PICBASIC Pro, MPASM, Hi-Tech C, CCS C.

PC Compilers: Delphi, C++ Builder, Visual Basic 6.

HIDmaker FS Combo: Only \$599.95

DOWNLOAD the HIDmaker FS Test Drive today!

www.TraceSystemsInc.com

301-262-0300

Low Cost EDA Software For Windows 98, XP, NT, 2000

- Circuit design package with schematic entry, circuit board layout and simulation (SPICE and digital) for only \$599
- Buy modules for as low as \$119
- Purchase and download products instantly from our website!
- Libraries include latest micros, memories, analog, and others

Contact **Mental Automation** at

(253) 858-8104

<http://www.mentala.com>

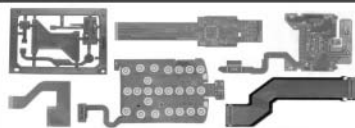
NAND FLASH FILE SYSTEM

- Wear Leveling, Garbage Collection
- Power-Fail Safe
- ECC and Bad Block Handling
- File System with Std C Lib API
- FAT File Structure
- Page Caching to Maximize Performance and Minimize RAM
- 20KB Code Size
- RAM Required < 1% Flash Size

www.smxrto.com/ffsc.htm

Micro Digital Inc
RTOS Innovators

800.366.2491 sales@smxinfo.com



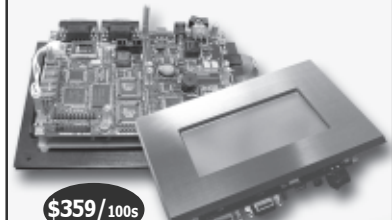
1-6 Layers Flex PCB
High Quality
1-2 Weeks Delivery
Start at Only \$299



Custom Membrane Keyboards
PCB/FPC Backed, Built-in LED
Start at \$245

EZPCB www.ezpcb.com
sales@ezpcb.com

QScreen™ - Low Cost Versatile Instrument Controller



\$359/100s

- Touchscreen Operated GUI
- 128x240 CCFL- Backlit Graphic Display
- Programmable in C or Forth
- Hundreds of Screens, Buttons, & Menus
- A/D and Two RS232/485 ports
- 8 Timer-Controlled Digital I/O Lines
- Up to 1 MB Flash, 512KB RAM
- Wide Selection of Plug-in I/O



Mosaic Industries Inc.
tel: 510-790-1255 fax: 510-790-0925
www.mosaic-industries.com

Order online at:
www.melabs.com

microEngineering Labs, Inc.

Development Tools for PIC® Microcontrollers

Phone: (719) 520-5323

Fax: (719) 520-1867

Box 60039

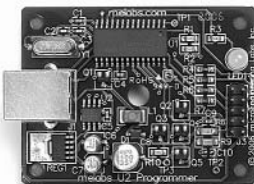
Colorado Springs, CO 80960

USB Programmer \$89.95 for PIC® MCUs

RoHS
Compliant

Programs
PIC MCUs
including
low-voltage
(3.3V) devices

Includes
Windows
98, Me, NT,
2K, and XP
Software



With Accessories for \$119.95:
Includes Programmer, Software, USB Cable,
and Programming Adapter for 8 to 40-pin DIP.



EPIC™

Parallel Port Programmer
starting at \$59.95

Serial Port Programmer
starting at \$79.95

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

PICPROTO™ Prototyping Boards



Double-Sided with Plate-Thru Holes
Circuitry for Power Supply and Clock
Large Prototype Area
Boards Available for Most PIC® MCUs
Documentation and Schematic

Pricing from \$8.95 to \$19.95

BASIC Compilers for 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

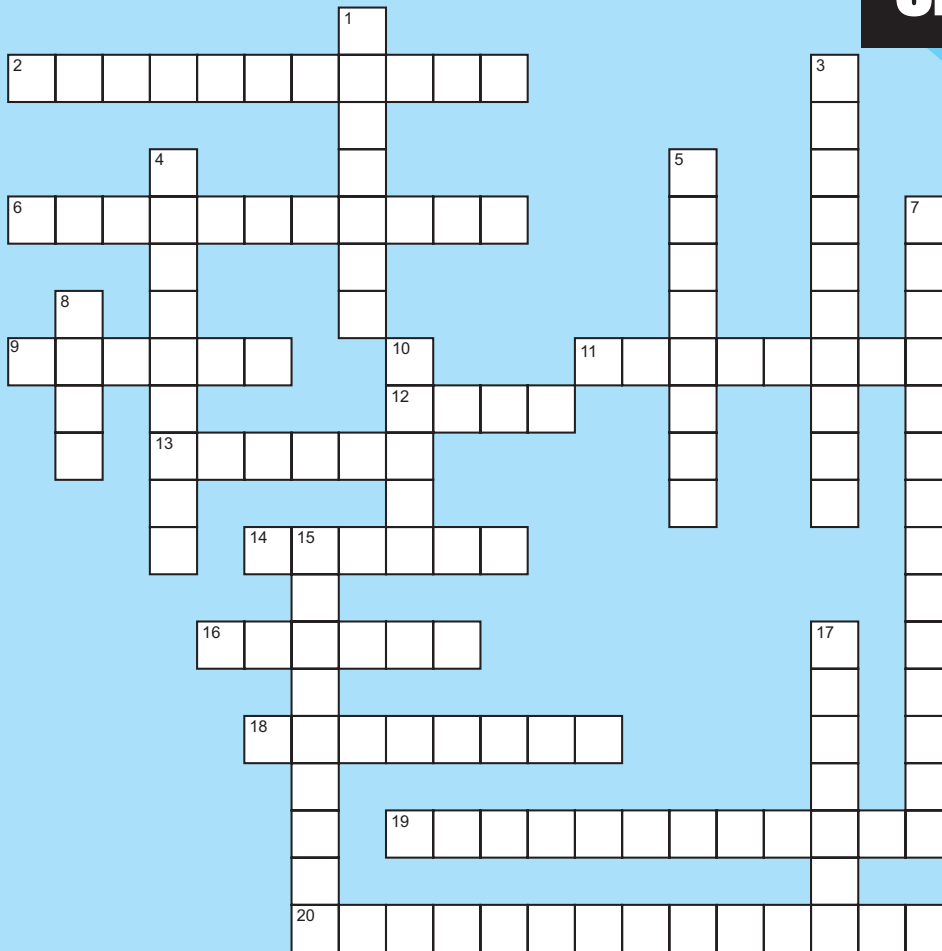
PICBASIC PRO™ Compiler \$249.95

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

See our full range of products, including
books, accessories, and components at:

www.melabs.com

CROSSWORD



Across

- The decrease in amplitude and intensity of a signal
- A numeric system, in base-16, that represents binary code in a format that is easy to read
- A small piece of wire used to connect two pins on a circuit board
- Hardware or software that enables a CPU to run programs designed for a different system
- A protocol used to retrieve e-mail on a remote server
- Software that displays advertisements on your computer
- A software application that surveys the web and collects and analyzes documents for indexing
- A force that causes a change in rotation
- A graphic representation, on a computer screen, of an audio signal generated by an oscillator
- An amplifier that strengthens weak electronic signals and transmits them for further amplification
- A material with zero resistance to the flow of electric current and no interior magnetic field

Down

- Using an acid to remove the unprotected parts of a metal surface to create a design
- A chemical process during which a substance combines with oxygen to produce heat and light
- A circuit board with sockets where other electronic devices on circuit boards can be plugged in
- A circuit that removes a ripple signal after a push button is pressed
- Common in automotive applications, this electrochemical capacitor has a higher energy density than most other capacitors.
- Create a CD or DVD
- A small piece of data from a larger group
- The point in an orbit closest to a planet
- The Swedish engineer (1889–1976) who studied data transmission. His theory states that the number of independent pulses that could be put through a telegraph channel is limited to twice the bandwidth of the channel.

The answers are available at
www.circuitcellar.com/crossword.

INDEX OF ADVERTISERS

The Index of Advertisers with links to their web sites is located at www.circuitcellar.com under the current issue.

| Page | | Page | | Page | | Page | |
|------|--------------------------------|--------|----------------------------|--------|--|--------|--------------------------------------|
| 91 | AAG Electronica, LLC | 9 | e-PCB | 30 | Lemos International | 90 | Pioneer Hill Software |
| 89 | ARC-USA, Inc. | 31 | ExpressPCB | 2 | Link Instruments | 26 | Pololu Corp. |
| 63 | ARM Developers' Conf. | 64, 92 | ezPCB | 10 | Linx Technologies | 91 | Pulsar, Inc. |
| 91 | ASIX s.r.o. | 87 | FDI-Future Designs, Inc. | 19 | Loadstar Sensors, Inc. | 23, 49 | Rabbit Semiconductor |
| 90 | Abacom Technologies | 47 | FTDIChip | 90 | MCC (Micro Computer Control) | 90 | Rabbit Semiconductor |
| 13 | Ad Hoc Electronics | 88 | Front Panel Express, LLC | 23 | Maxstream | 89 | Reach Technology, Inc. |
| 91 | All Electronics Corp. | 61 | Futurelec | 92 | Mental Automation | 27, 33 | Renesas Technology |
| 90 | Apex Embedded Systems | 89 | General Circuits, Inc. | 88, 92 | Micro Digital, Inc. | 55 | Saelig Company |
| 7 | Atmel | 9, 91 | Grid Connect | 42 | Microchip | 91 | Schmartboard |
| 90 | BG Micro | 3 | HI-TECH Software LLC | 17 | Microchip 16-bit Embedded Control Design Contest | 79 | Sealevel Systems |
| 29 | Bitscope Designs | 81 | Holtek Semiconductor, Inc. | 92 | microEngineering Labs, Inc. | 11 | SEgger Microcontroller Systems LLC |
| 65 | CWAV | 89 | IMAGEcraft | 92 | Mosaic Industries, Inc. | C3 | Tech Tools |
| 32 | CadSoft Computer, Inc. | 88 | Intec Automation, Inc. | 92 | Mouser Electronics | 56, 57 | Technologic Systems |
| 87 | Custom Computer Services, Inc. | 88 | Intrepid Control Systems | 5 | Mylydia, Inc. | 87 | Technological Arts |
| 1 | Cypress | 10 | Ironwood Electronics | 89 | NetBurner | 89 | Tern, Inc. |
| 87 | DLP Design | 64, 91 | JK microsystems, Inc. | C2 | Nurve Networks LLC | 41 | Tibbo Technology, Inc. |
| 88 | DSP Workshops | 95 | Jameco | 69 | Ontrak Control Systems | 92 | Trace Systems, Inc. |
| 77 | Decade Engineering | 77 | Jeffrey Kerr, LLC | 88 | Opal Kelly Inc. | 87 | Triangle Research Int'l, Inc. |
| 61 | EMAC, Inc. | 39 | Keil Software | 88 | PCB-Pool | 88 | WCSC (Willies Computer Software Co.) |
| 50 | ESC Boston | 69 | LabJack Corp. | 12 | Parallax, Inc. | 21 | Wiznet |
| 89 | Earth Computer Technologies | 69 | Lakeview Research | C4 | Phytec America LLC | 91 | Zanthic Technologies, Inc. |
| 34 | Efficient Computer Systems | 90 | Lawicel AB | | | | |

Preview of September Issue 206 Theme: Data Acquisition

Smart Power: An Intelligent Power Supply for Embedded Systems

PIC an LCD: A Character-Based Serial LCD Controller

Pulse Generation: Encoder Interfacing to Microcontrollers

Resilience in Embedded Designs (Part 1): Power Supply, Inputs, and Ground

1-Wire in the Real World (Part 2): The Solutions

LESSONS FROM THE TRENCHES String Theory

FROM THE BENCH I-Zip Dashboard

SILICON UPDATE Game On

ATTENTION ADVERTISERS

October Issue 207 Deadlines

Space Close: Aug. 10
Material Close: Aug. 21

Theme: Signal Processing

BONUS DISTRIBUTION:
ARM Devcon
PCB East

Call Shannon Barraclough
now to reserve your space!

860.875.2199

e-mail: shannon@circuitcellar.com

- Every week, Jameco is giving extra discounts on major brands like these:
- AlcoSwitch • AMP/Tyco
 - Dallas Semiconductor
 - Intersil • Maxim
 - Microchip • Molex
 - Panasonic • AVX
 - Fairchild Semiconductor
 - Tyco • Aromat
 - Aavid Thermalloy
 - Atmel • ST Micro

WIN BIG



**TAKE AN
EXTRA
15%
OFF
ON 2 NEW
MAJOR
BRANDS
EVERY WEEK!**

And that's in addition to the industry-best pricing you already get at Jameco every day!

You already know about Jameco's best-in-the-business low pricing... Now we're out to get a *new* message across: *Major Brands—at Jameco pricing!*

To dramatize the point, we're giving

you an *additional* 15% discount on two different major brands each week. From now through the end of May!

**SEE
WHO'S UP
THIS WEEK
GO TO
THE WEBSITE**

And that's on *top* of our everyday discounted pricing!

We wanted to make this so good a deal that you just *couldn't* pass it up.

So... Want to know which two major brands you can save extra-big on *this* week? Just head to the website...

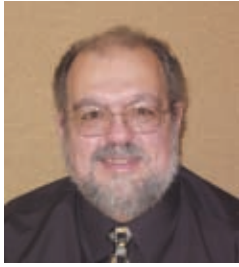
www.Jameco.com/CCW

- T.I. Semiconductor
- Augat/Tyco • Bourns
 - CTS • Cypress
 - C&K Switches
 - Comair Rotron
- Condor • Grayhill
 - Intel • Micron
- Philips • Power-One
 - Raychem/Tyco
- Renesas • SanDisk
 - Toshiba
- Vishay Siliconix

EVERY WEEK — AT — JAMECO

Great Products.
Awesome Prices.

JAMECO®



PRIORITY INTERRUPT

by Steve Ciarcia, Founder and Editorial Director

Let's Count the Lumens

The primary way most of us have been lowering power consumption is by swapping out incandescent lights with compact fluorescent lights (CFLs). Putting in a 15-W CFL in place of a 60-W incandescent amounts to a 75% average power saving—provided it works.

Like PV, I was one of the first people on the CFL bandwagon, too. About five years ago, the local power company offered CFLs at a real bargain and I bought over 100 lights. Don't laugh, I have more lights around here than the average landing strip and they add up. (Heck, I'm sitting here on an 11' x 16' enclosed "porch" writing this editorial and I count 11 individual lights just out here.) Before we all got energy conscious and a kilowatt hour was \$0.06 instead of \$0.20, designing rooms with 20 to 30 track lights and banks of dimmer switches was the norm. The one saving grace in all of this incandescent lighting overkill is that much of my house uses "room occupancy sensing" via my home control system (HCS), and they are not just needlessly burning power all the time.

The unfortunate problem I ran into with CFLs was that you can't just substitute them for every incandescent. It depends on the control circuit (manual switch versus something electronic). They absolutely didn't like dimmer switches of any kind and they wouldn't work properly with my HCS X-10 controllers either. Consequently, five years ago, I was able to swap out only about 20% of my incandescents with CFLs.

The encouraging responses from everyone cheering my PV installation along prompted me to take a new look at my lighting issues. I hoped that CFL ballast electronics had improved in five years. While I didn't expect these new lights to all be dimmable, perhaps the new electronics might tolerate the full-on TRIAC of a dimmer switch and work with my X-10s. After buying another five dozen CFLs of various brands, I can categorically state that a game of roulette has better odds than making every kind of CFL automatically controlled. It's too much to discuss here, but I'll go into it more in my PV installation articles.

After not having much additional luck with newer CFLs, I then went "from the frying pan into the fire." I decided to see if using LED lights would provide an alternative solution. After all, they offer the same energy reduction as CFLs if you read the advertising hype. Were they the best lighting solution since the whale oil lamp or just a new version with the same light level and no fishy smoke?

The first problem was separating the hype from the real facts when it came to the people selling LED bulbs. It's easy to come away with the impression that LED lights, like CFLs, are an instant swap for incandescents. As an LED lighting neophyte, I eventually put my trust in Best Home LED Lighting (www.besthomeledlighting.com) and sampled or purchased a dozen different LED bulbs to experiment with. Again, I'll cover this in greater detail in later articles, but let me give you the punchline while I have your attention. Basically, if the lumen level is acceptable, LED bulbs work well with simple AC line toggle switches, but anything else and it's back to Vegas. Ironically, we all think of LEDs as being much simpler electronics than a CFL and hence easier to control. My reality was that I can't even tell you that two units of the dozen LEDs I tried have the same electronic configuration. Hence, any automatic control devices are an adventure in customization. Here is the short list of my experience:

1. LED lights have a wide range of efficiencies and configurations. Current units seem to be about 50 lm/W, while new high-output bulbs are approaching 100 lm/W.
2. The number of LEDs is not the determining factor of bulb brightness. Different types of LEDs vary greatly in brightness. A better indicator of luminous intensity is the quoted bulb wattage. Current generation white LEDs are about 30 lm/W, while the newest power LEDs from CREE seem to be about 65 lm/W.
3. Currently, there are no LED bulbs that can match the light output of a regular 60-W incandescent bulb. The brightest LED bulbs are directional floodlights and spotlights with the brightest being comparable to a 45- to 50-W incandescent. The brightest nondirectional LED bulbs are more like a 25-W incandescent.
4. In my experience, LED bulbs don't like dimmer controls because most 120-V AC dimmers need a minimum amount of resistive load in order to function properly. Because most of these bulbs are under 10 W and many have reactive-capacitive loads, TRIACs go nuts.
5. In my experience LED bulbs don't like traditional X-10 modules either. While some LEDs may turn on successfully, many will not turn off. Instead, the trickle current used to sense manual lamp or appliance control causes the LED to flash.
6. No one wants to admit that the biggest problem with using LEDs is heat—they self-destruct. Currently, most manufacturers tell you not to install LED bulbs in enclosed ceiling fixtures for that reason. Ironically, the multi-LED bulbs have a larger surface area to dissipate heat even if they are less efficient. The new high-intensity CREE bulbs are more efficient, but they are constructed on a large aluminum heatsink for a reason.

Don't get me wrong. Even with all my criticism, I think LED bulbs are the real future of lighting. In my opinion, right now they are at the same evolutionary point that CFLs were five to eight years ago. I'm encouraged that in just the few weeks I've been playing with them, I've seen a doubling in lighting efficiency and power. It won't be long before there really is a 7-W 800-lm LED that doesn't double as a space heater. I don't mind being early on every bandwagon when I see progress like this. Stay tuned. The fun is just starting.

steve.ciarcia@circuitcellar.com

Stellaris[®] Means...



Luminary Micro Stellaris microcontrollers are designed for serious embedded applications:

- The industry's first ARM[®] Cortex[™]-M3 implementation.
- Always single-cycle flash for maximum performance.
- Real MCU GPIOs — always interruptible.
- 32-bit performance at 8-bit & 16-bit prices.
- No functional pin muxing; easy to select the device you need.
- No peripheral sharing — all peripherals operate concurrently.

Discover what Stellaris can mean for your next design at:
www.LuminaryMicro.com/StellarisMeans



LUMINARY MICRO[™]

Compact. Versatile. Cost-effective.

From evaluation to development to prototyping,
Stellaris® Family Evaluation Kits
contain everything needed to develop and run applications
for Stellaris microcontrollers out-of-the-box.



An embedded Web
server out-of-the-box.

Stellaris LM3S6965 Evaluation Kit

featuring integrated Ethernet.

\$79



Stellaris LM3S811 Evaluation Kit

Discover the exciting ways
this evaluation kit was used
in DesignStellaris2006 at:
[www.LuminaryMicro.com/
DesignStellaris2006](http://www.LuminaryMicro.com/DesignStellaris2006).

\$49



A CAN network out-of-the-box.

Stellaris LM3S2965 Evaluation Kit

featuring integrated CAN.

\$89

Discover what Stellaris can mean for you:
www.LuminaryMicro.com/StellarisMeans



LUMINARY MICRO™



LUMINARY MICRO is the creator of the award-winning Stellaris family of microcontrollers. For more information about the entire Stellaris family of products, please visit our website at LuminaryMicro.com!



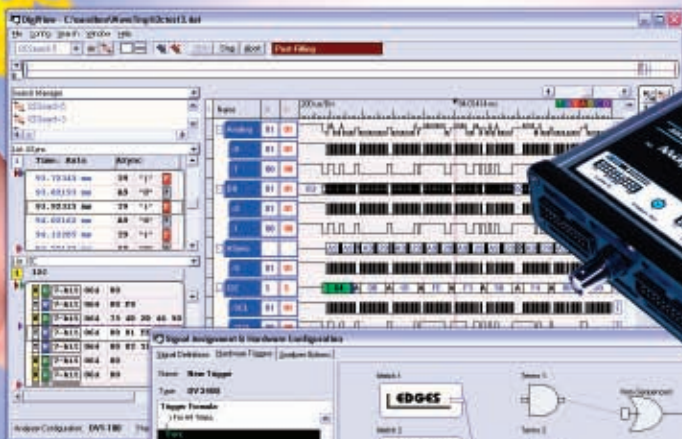
Cortex™
M3/M4 Processors by ARM

PC Based Logic Analyzers

from \$499.00

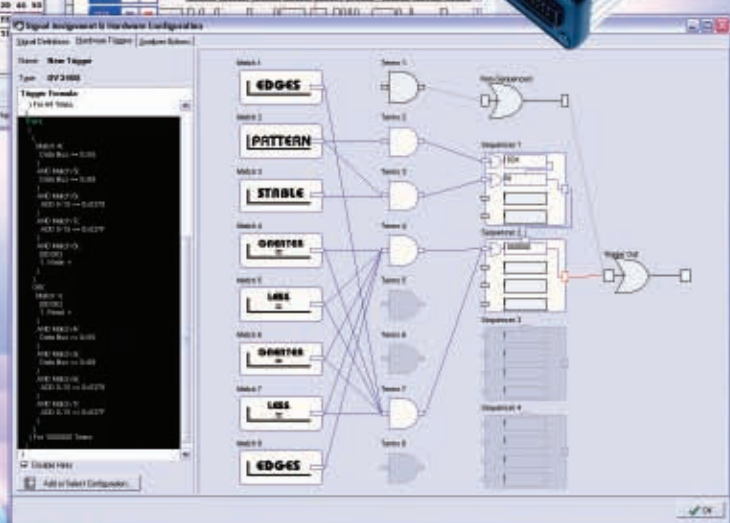
NEW Model DV3400

- Faster Sampling (200/400 Msps)
- More Channels (36)
- More Memory (4x)
- Advanced Match circuits (8)
- $\pm 6V$ Adjustable Threshold (x2)
- Cascadable Sequencers (4)
- Enhanced Compression



Professional Hardware Capture + Software Analysis

- Automatic Real-time Hardware Compression eliminates the need to reduce resolution. Our newest version makes "dead time" insignificant.
- Edge and Pattern Triggers on all models. Advanced Model also includes Range (>, <, =, !=, >=, <=) and Stable Matches with Duration, and 4 flexible cascadable sequencers with pass-counters.
- Pattern Searches with Match & Duration.
- Specialized Sequential Searches for Serial and State Mode Signals.
- Display I²C, Synchronous (SPI), Asynchronous (RS-232), State, Boolean, Bus and Analog Data.
- Single or Dual Waveform Views.
- Resolution Zoom in Wave Views.
- Time based Link Groups for all views.
- Specialized Exports from Data Tables and List Views.
- Multi-Signal Data Tables.
- Drag & Snap Markers.
- "Click to Center" function.
- "Snap Previous/Next" function.
- Print or Save Images with comments.
- USB 2.0 (480 Mbps).
- USB 1.1 compatible (12 Mbps).



Very flexible, "easy to configure" Advanced triggering.

PICmicro[®] MCU Programmer

Multi-Function, In-Circuit & Gang Operation

only \$199.00



See our Memory Emulators

from \$179.00



TechTools

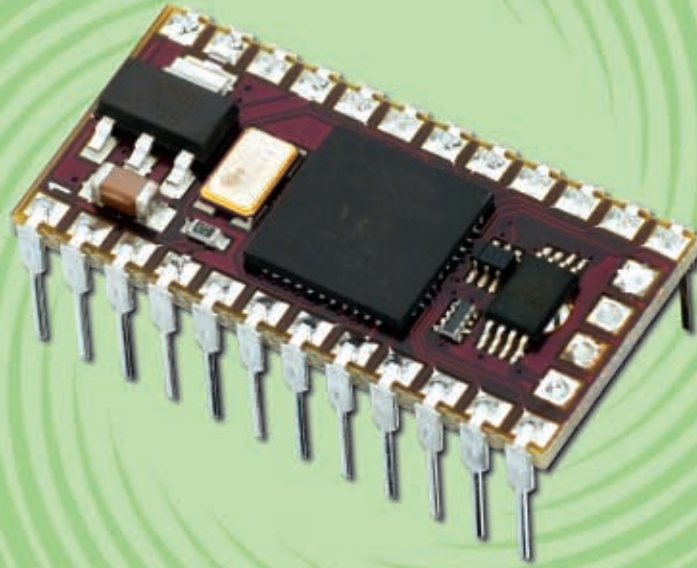
www.tech-tools.com

(972) 272-9392 • sales@tech-tools.com

Copyright © 2007 TechTools • DigiView, FlexROM, EconoROM and QuickWriter are trademarks of TechTools • PICmicro is a registered trademark of Microchip Technology Inc.

QuickWriter™

Spin Stamp



The Spin Stamp is a 24-pin module similar in form and size to the popular BASIC Stamp 2 module. This incredible new microcontroller was designed to give engineers and other users true multi-processing capabilities in a familiar 24-pin module design.

The Propeller is a multi-processing chip with eight 32-bit processors (cogs) and shared memory with 32 KB of RAM and 32 KB of ROM holding a font, math tables, and Spin interpreter. The Spin Stamp includes a Propeller, 32 KB EEPROM, 3.3 V regulator, and 10 MHz crystal oscillator.

The Spin Stamp module is not intended to be a direct drop-in replacement for any BASIC Stamp module.

- A Prop Clip (#32200; \$29.95) is required to program the Spin Stamp. The Prop Clip plugs into the four-pad connector on the bottom edge of the Spin Stamp.
- The Spin Stamp can be programmed in Propeller Spin (high-level) or Assembly (low-level) Propeller Assembly language. *Note: the Spin Stamp cannot be programmed in PBASIC.*
- The Spin Stamp module's I/O voltage is 3.3 V.



Order the **Spin Stamp module** (#SSI-IC; \$59.95) at www.parallax.com or call our Sales Department toll-free at 888-512-1024 (Mon-Fri, 7am-5pm, PDT).

Propeller, Spin, Parallax, and the Parallax logo are trademarks of Parallax Inc.

PARALLAX 
www.parallax.com