Fast Digital Filtering 101 (p. 40) • An RTOS for a Powerful MCU (p. 56) • Next-Gen Embedded Options (p. 76)

www.circuitcellar.com

CIRCUIT CELLAR

THE MAGAZINE FOR COMPUTER APPLICATIONS

#218 September 2008

DATA ACQUISITION

Acquire and Stream Internet-Based Content

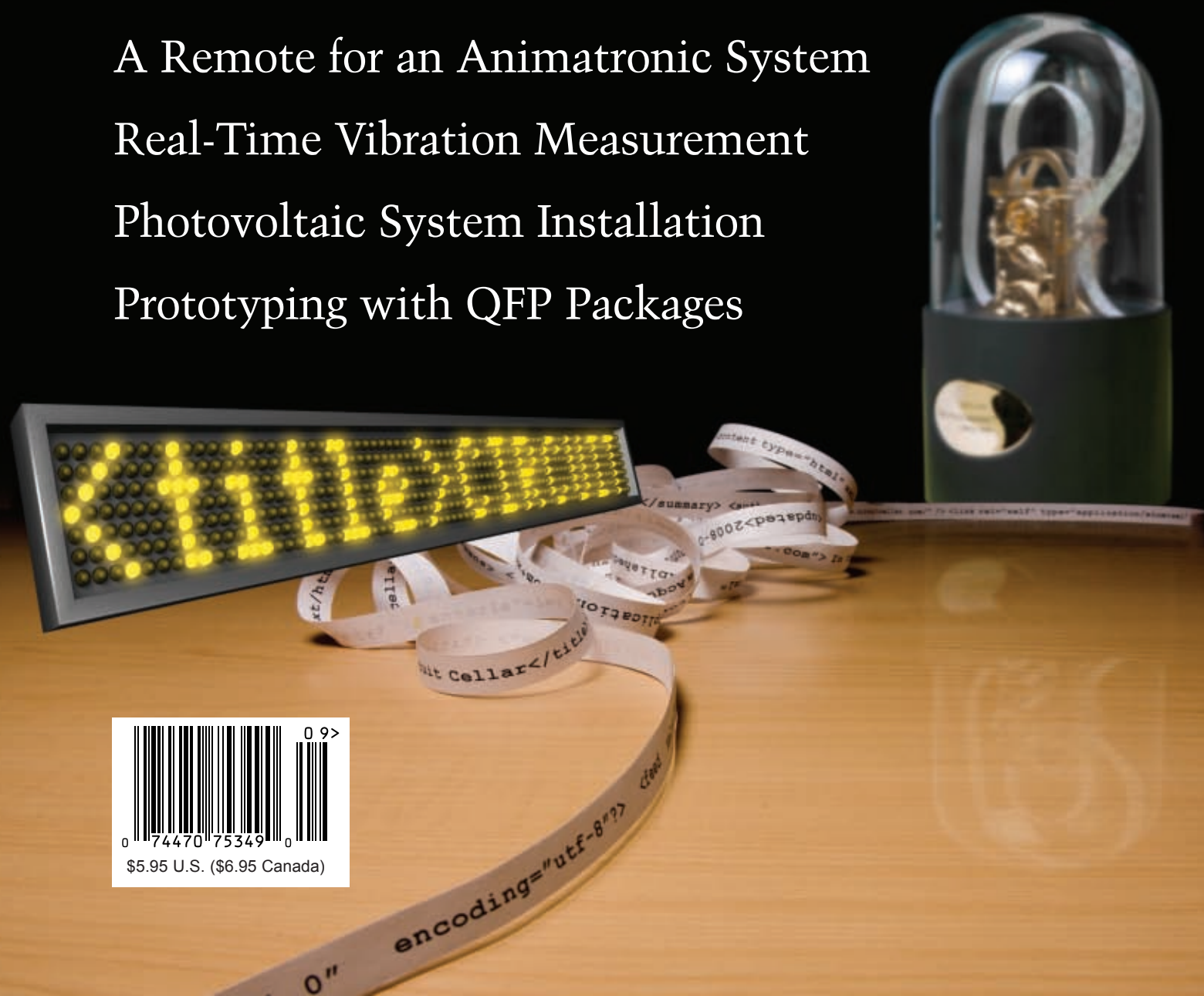
Biometric Data Acquisition and Processing

A Remote for an Animatronic System

Real-Time Vibration Measurement

Photovoltaic System Installation

Prototyping with QFP Packages



0 74470 75349 0
$5.95 U.S. ($6.95 Canada)

NetBurner Customer Feedback: Why are you using NetBurner?

The excellent out-of-the-box experience

I would like to comment on the quality of a product that I purchased - the MOD5270. I've purchased products in the past from other companies that claim "plug and play" functionality. Usually this means plug it in and try to figure out why it doesn't work. With the MOD5270 I literally plugged in the power, connected it to my router, and ran the factory demo.

I am also impressed with the number of example applications that are included and the fact that all the development tools are included. It was nice to be able to create a new project, import one of the examples, and run it within 5 minutes.

Keith Gilman, Sr. Design Engineer

The complete hardware and software solution

We did a previous project using different companies for the RTOS, network stack, and compiler. Yeah, three vendors and all the associated finger pointing. We had to buy the flash file system as an extra/add-on. We put an 8M flash device on our processor bus, and a guy had to write a driver for the device (data register, command register, poll the ready bit, blah, blah), it was a 3+ month job. I thought I was in for the same thing. Thanks to you, we will drop the soldered-on Flash and put on the SD connector which will be ready to roll with your file system that runs 'right out of the box'. I was sweating about all the Flash work, and now its gone.

John Ramsy, Project Engineer

The active forum filled with NB Gurus

NetBurner offers a highly versatile and affordable solution set for embedded design, remote control, and product development. From the very active user forum (with Netburner gurus in attendance), to the great IDE and tool chain. NetBurner is the first and only solution for TCP/IP-Internet-Web product automation in my book.

Chris Ruff, Software Engineer

The quality of the technical support

I've been really impressed with the quality of the tech support. The guys answering the questions are right on and super helpful (probably has to do with your policy of having the engineers handle the tickets).

Nicolae P. Costescu, Project Engineer

The full-featured tool set, examples, and documentation

The device and the NetBurner tool-set far exceeded our expectations. I spent a week familiarizing myself with the NetBurner tools, and then I wrote a first-cut of the original application that I had envisioned for the SB72EX. As it turned out, two of the example applications formed the core of the networking capabilities of my application. The bulk of my time was spent implementing a simple command interface to cause the device to interrupt its default IP-to-serial modes to receive and store a stream of data and then FTP that data to a remote server.

The NNDK interface to the IP/socket facilities of the SB72EX proved to be the simplest part of the application - the interface is straightforward and the documentation is excellent (the large collection of application examples didn't hurt). Midway through my efforts I had a couple of questions that I elected to send to the NetBurner support group - the response was almost immediate, the person(s) responding were obviously knowledgeable and experienced - how refreshing!

The SB72EX device and (perhaps more importantly) the NetBurner tool-set have proven to exceed our immediate needs and have started the gears turning as to how we might use them to address other requirements.

Fred Craft, Software Engineer

NetBurner
Networking in One Day!

Information and Sales | sales@netburner.com
Web | www.netburner.com
Telephone | 1-800-695-6828

freescale™
Alliance Member

BASIC, LUA & PYTHON Supported!

Micromint USA

Web & SSH Interfaces for remote access!

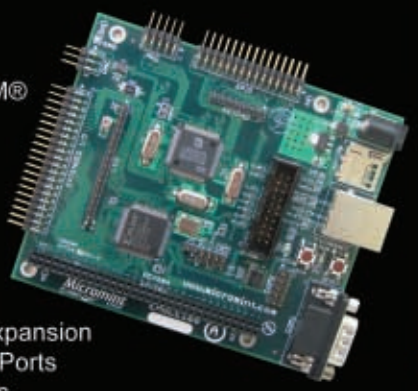
Introducing Micromint's All New ARM Based Controllers

$89.95*

Micromint Eagle 100

Featuring the Luminary Micro ARM® Cortex™

FEATURES



- 50 MHz 32-bit ARM® Cortex™-M3
- 256 KB Flash
- 64 KB SRAM
- 10/100 Ethernet
- Micro-SD
- LCD Port
- 4x4 Keypad Port
- PC/104 8-bit I/O expansion
- Two RS232 Serial Ports
- I^2C, SSI (SPI) Ports
- 8-ch. 10-bit ADC, 4-ch. 10-bit DAC
- Reconfigurable I/O with Xilinx CPLD
- Up to 52 - 5V tolerant GPIOs, 80 without PC/104
- 4 general purpose timers, 1 watchdog timer
- Support for GNU and IAR compilers.
- Thumb2® instruction set for smaller object code
- Basic and LUA supported
- +5V power supply required
- Dimensions: 3.8 inches x 4.4 inches

Visit our website for webcast schedules

Reduce Your Time to Market.

Minimize your time from conception to production by utilizing one of Micromint's market-proven controllers. Whether your concerns are digital or analog, inputs or outputs, Micromint has a product to fit your needs. Order quantities of one to thousands. Custom design and configurations are available.

$129.95*

Micromint Electrum 200

Featuring the AT91SAM9260 ARM9

FEATURES



- 180 MHz 32-bit ARM9®
- 1 GB NAND Flash
- 64 MB SRAM
- 10/100 Ethernet
- MicroSD
- LCD Port
- 4x4 Keypad Port
- PC/104 8-bit I/O expansion
- Two RS232 Serial Ports
- I^2C, SSI (SPI) Ports
- USB Host and device ports
- Optional 8-ch. 12-bit ADC, 4-ch. 12-bit DAC
- Reconfigurable I/O with Xilinx CPLD
- Up to 72 - 5V tolerant GPIOs, 100 without PC/104
- 6 timers, 1 watchdog timer
- Linux 2.6 with realtime extensions
- Support for GNU compiler
- Basic, LUA and Python supported
- +5V power supply required
- Dimensions: 3.8 inches x 4.4 inches

With over 500,000 controllers in the marketplace, Micromint has been providing innovative, turn-key solutions to the OEM market for 29 years-from design through production, as well as packaging and shipping the final product. Our broad line of embedded controllers and turn-key solutions can turn your imagination into reality.

Visit our website @ www.micromint.com to see our complete line of OEM Solutions.

Delivering Embedded Controllers for 29 Years

www.micromint.com

111 Commerce St | Lake Mary, FL 32746 | 800-635-3355 | 407-333-4799 | Fax 407-333-4788 | sales@micromint.com

* Qty 1 pricing. Limited time offer, restrictions may apply.

The Newest Interconnects

Micro USB
Industrial Ethernet
Hard Metric
Fiber Optic
MicroTCA
HDMI
SFP
Mini SAS
SATA



molex
HDMI Cable Assemblies
www.mouser.com/molex/a

LEMO
S and K Series Multi-Pole Connectors
www.mouser.com/lemo/a

HARTING
IP67 Data 3A Connectors
www.mouser.com/harting/a



www.mouser.com
Over A Million Products Online

New Products from:



Tyco Electronics
Authorized Distributor
QSFP Pluggable I/O Interface
www.mouser.com/amp/a



PHOENIX CONTACT
INSPIRING INNOVATIONS
M12 Ethernet Cables
www.mouser.com/phoenix/a



FCI
MicroTCA
Card Edge Connectors
www.mouser.com/fci/a

The ONLY New Catalog Every 90 Days

Experience Mouser's time-to-market advantage with no minimums and same-day shipping of the newest products from more than 366 leading suppliers.

MOUSER ELECTRONICS®
a tti company

The Newest Products For Your Newest Designs

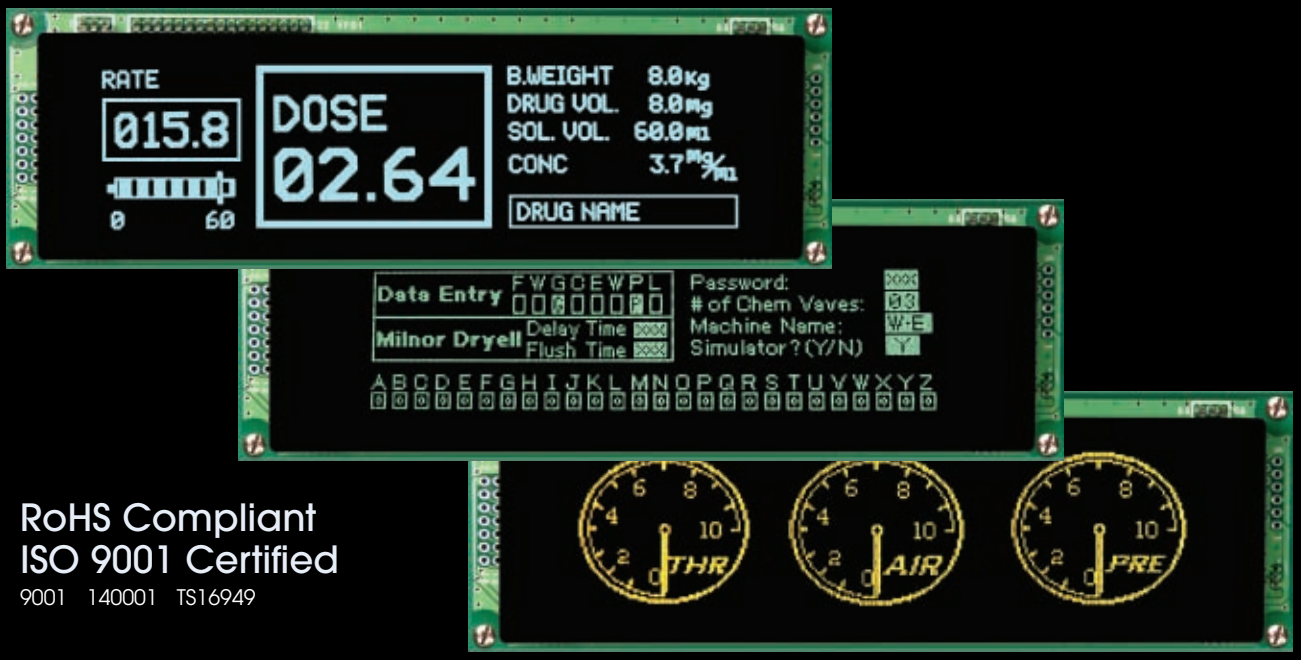
(800) 346-6873

Noritake

Vacuum Fluorescent Displays vfd

Graphic Display Value Approaching Character Display Prices

A Display You Can Read From Any Angle



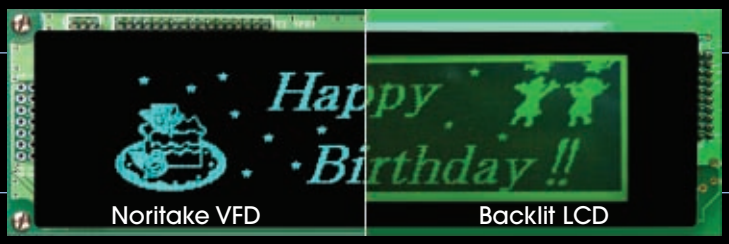
Noritake 7000 Series

RoHS Compliant
ISO 9001 Certified
9001 140001 TS16949

MORE BUILT-IN FEATURES THAN YOU MAY EVER NEED...

- Up to 4 Independent Windows with Individual Control
- Many Int'l Font Sets with 16 User-Defined Characters
- Easily Combine Text & Graphics on One Screen
- Horizontal & Vertical Scrolling
- Proportional Fonts & Font Magnification
- 8 Levels of Brightness Control
- Built-in Screen Savers
- Many Filters Available to Match Application
- Development Kit Available

DON'T COMPROMISE.
Your application deserves
VFD quality ...



www.noritake-elec.com/51m

Noritake Co., Inc. 2635 Clearbrook Dr., Arlington Heights, IL 60005 phone **1-800-779-5846** e-mail **electronics@noritake.com**

itron

Over 100 Years in Business

TASK MANAGER

Circuits & Acquisition

Data acquisition and transmission devices are essential tools for cultivating the Average Joe's personal and professional life. When he needs a data-acquiring device like a simple digital thermometer, he drops his debit card and buys one. You, however, aren't an Average Joe, at least when it comes to electronic technologies, because you can design custom systems to meet your data acquisition needs. This issue features articles by engineers who did just that.

Starting on page 15, James Blackwell describes a novel data acquisition system that gathers targeted information from the Internet. The design—which is built around a Microchip Technology PIC microcontroller and a WIZnet WIZ810MJ Ethernet module—collects and displays news content from the BBC's web site.

In a two-part series titled "Dynamic Animatronic Remote," Peter Montgomery describes how he developed a different sort of data acquisition system. This month, he explains how he designed and programmed a remote controller for a complex animatronics system (p. 32). The design enables him to acquire and transmit data quickly and accurately from different locations. Mobility is key.

If you've been reading *Circuit Cellar* for the past few years, you know that several of Bruce Land's current and former Cornell University students have published informative articles about well-designed applications. A good example is the February 2007 article by Craig Ross and Ricardo Goto titled "RFID Security System" (*Circuit Cellar* 199). This month, Bruce presents an article on the subject of fast digital filtering, a topic both students and professional engineers should understand (p. 40).

Helicopters require accurate vibration-measurement technology. Turn to page 48 to read about an MCU-based, real-time system that Juan Tapia Farias designed for the Chilean Air Force (Fuerza Aérea de Chile, FACH). The system was tailored for a Bell Helicopter Model 206 JetRanger helicopter, but you can customize a similar design for the mechanical system of your choice.

Columnist Jeff Bachiochi presents the last data acquisition-related article in this issue. Starting on page 68, Jeff introduces you to one of the newest advances in biometric technology. He explains how Atmel's AT77C104B sensor chip and the AT91SAM7FP105 FingerChip security processor make fingerprint data acquisition and processing easy.

On page 22, George Martin begins where he left off in July by tackling the important topic of energy acquisition. In this article, he explains how he installed his photovoltaic system. He now lives and works "off the grid."

To learn about about a more "micro" application, refer to Naubert Aparicio's article, "Real-Time Operating System Design" (p. 56). The purpose of the article is to teach you how to customize an operating system for an MCU. He designed his "ArmExe" RTOS to harness the power of Luminary Micro's Cortex-M3 family of microcontrollers.

Did you make it to the 2008 Embedded Systems Conference in San Jose, CA? If not, don't worry. Tom Cantrell describes his experiences at the conference (p. 76). His presentation of new embedded gadgetry is sure to encourage you to begin a new application.

cj@circuitcellar.com



CIRCUIT CELLAR®

THE MAGAZINE FOR COMPUTER APPLICATIONS

FOUNDER/EDITORIAL DIRECTOR
Steve Ciarcia

MANAGING EDITOR
C. J. Abate

WEST COAST EDITOR
Tom Cantrell

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

NEW PRODUCTS EDITOR
John Gorsky

PROJECT EDITORS
Gary Bodley
Ken Davidson
David Tweed

ASSOCIATE EDITOR
Jesse Smolin

CHIEF FINANCIAL OFFICER
Jeannette Ciarcia

MEDIA CONSULTANT
Dan Rodrigues

CUSTOMER SERVICE
Debbie Lavoie

CONTROLLER
Jeff Yanco

ART DIRECTOR
KC Prescott

GRAPHIC DESIGNERS
Grace Chen
Carey Penney

STAFF ENGINEER
John Gorsky

ADVERTISING

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

PUBLISHER
Sean Donnelly
Direct: 860.872.3064, Cell: 860.930.4326, E-mail: sean@circuitcellar.com

ADVERTISING REPRESENTATIVE
Shannon Barraclough
Direct: 860.872.3064, 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
860.875.2199, Fax: 860.871.0411, E-mail: info@circuitcellar.com
Editorial Office: Editor, Circuit Cellar, 4 Park St., Vernon, CT 06066, E-mail: editor@circuitcellar.com
New Products: New Products, Circuit Cellar, 4 Park St., Vernon, CT 06066, E-mail: newproducts@circuitcellar.com

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

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

CIRCUIT CELLAR®, THE MAGAZINE FOR COMPUTER APPLICATIONS (ISSN 1528-0608) is published monthly by Circuit Cellar Incorporated, 4 Park Street, Vernon, CT 06066. Periodical rates paid at Vernon, CT and additional offices. **One-year (12 issues) subscription rate USA and possessions $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 © 2008 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.



Touch the Future of User-interface Design

Create innovative and elegant interfaces with Cypress CapSense™ Solutions

Replace mechanical switches and knobs with industry-leading PSoC®-based CapSense™ touch-sensing solutions from Cypress. They allow you to quickly and easily develop elegant, innovative interfaces for cell phones, computers, consumer electronics, white goods, automotive solutions, industrial applications — and more.



Get your Development Kit today:
www.cypress.com/go/design



SAMPLE CAPSENSE APPLICATIONS

CYPRESS
PERFORM

© 2008 Cypress Semiconductor Corporation. All rights reserved. Cypress, the Cypress logo, and PSoC are registered trademarks and CapSense is a trademark of Cypress Semiconductor Corporation.

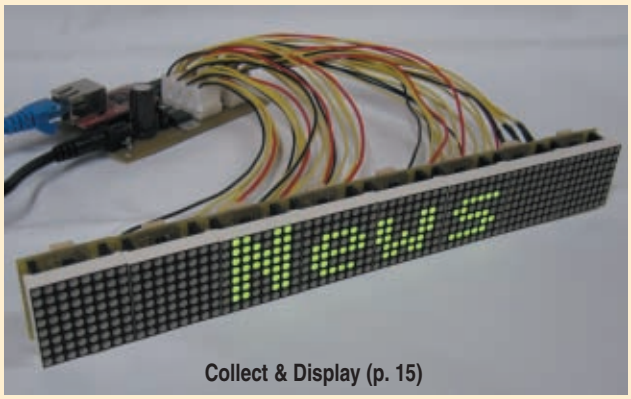
September 2008: Data Acquisition

FEATURES

15 **Content Collection and Display**
Build an Internet-Connected News Ticker
James Blackwell
Second Place WIZnet iEthernet Design Contest 2007

32 **Dynamic Animatronic Remote (Part 1)**
Build a Custom Controller with a Color LCD
Peter Montgomery

40 **Fast Digital Filtering**
Bruce Land



Collect & Display (p. 15)


A Handy Controller for an Animatronic System (p. 32)

48 **MCU-Based Flight Data Acquisition**
Real-Time Vibration Measurement for Aircraft
Juan Tapia Farias

56 **Real-Time Operating System Design**
Customize an OS for a Powerful Embedded MCU
Naubert Aparicio

62 **Prototyping with QFP Parts**
Stuart Ball

COLUMNS

22 **INTELLIGENT ENERGY SOLUTIONS — LESSONS FROM THE TRENCHES**
Living and Working Off the Grid
Part 2: Installation and Performance
George Martin

68 **FROM THE BENCH**
Fingerprint Data Acquisition
Print Sensing with Embedded Technology
Jeff Bachiochi

76 **SILICON UPDATE**
Embedder Than Ever
ESC Delivers the Goods
Tom Cantrell


PV System Installation (p. 22)

DEPARTMENTS

4 **TASK MANAGER**
Circuits & Acquisition
C. J. Abate

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

14 **TEST YOUR EQ**

93 **CROSSWORD**

94 **INDEX OF ADVERTISERS**
October Preview

96 **PRIORITY INTERRUPT**
Something Old, Something New
Steve Ciarcia

Hammer Down Your Power Consumption with picoPower™!



THE Performance Choice of Lowest-Power Microcontrollers

Performance and power consumption have always been key elements in the development of AVR® microcontrollers. Today's increasing use of battery and signal line powered applications makes power consumption criteria more important than ever. To meet the tough requirements of modern microcontrollers, Atmel® has combined more than ten years of low power research and development into picoPower technology.

picoPower enables tinyAVR®, megaAVR® and XMEGA™ microcontrollers to achieve the industry's lowest power consumption. Why be satisfied with microamps when you can have nanoamps? With Atmel MCUs today's embedded designers get systems using a mere 650 nA running a real-time clock (RTC) and only 100 nA in sleep mode. Combined with several other innovative techniques, picoPower microcontrollers help you reduce your applications power consumption without compromising system performance!

Visit our website to learn how picoPower can help you hammer down the power consumption of your next designs. PLUS, get a chance to apply for **a free AVR design kit!**

http://www.atmel.com/picopower/

Everywhere You Are®

© 2008 Atmel Corporation. All rights reserved. Atmel®, logo and Everywhere You Are® are registered trademarks of Atmel Corporation or its subsidiaries.

NEW PRODUCT NEWS

Edited by John Gorsky

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

NEW STM8S MICROCONTROLLERS

The **STM8S** family is a series of 8-bit MCUs using the next-generation STM8 core specified for the industrial temperature range. The new MCUs combine the core's high speed, performance, and code efficiency with versatile peripherals, including a number of specific features for robustness and reliability. Integrated memory, including true EEPROM, simplifies emulation.

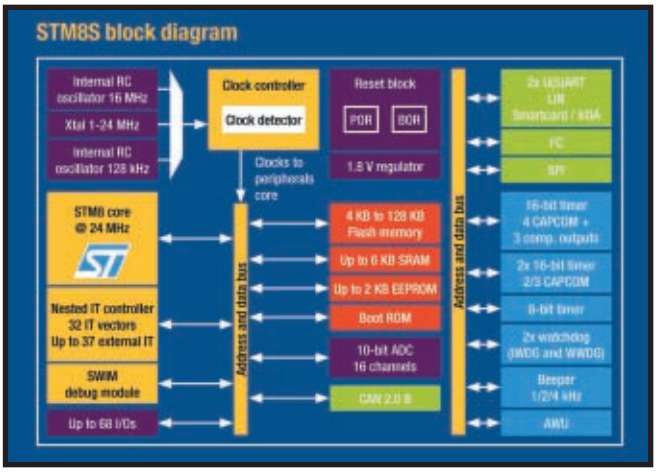
The STM8 core has a 32-bit memory interface and a three-stage pipeline, achieving 20-MIPS peak processing performance at 24 MHz. A stack pointer and 16-bit index registers enhance manipulation of tables. The core's 16 MB of linear memory address space streamlines paging operations above 64 KB. In addition, improvements to stack-pointer operations, supplemental addressing modes, and new instructions boost support for C programming and real-time performance to increase code density and processor efficiency.

Alongside the advantages of the STM8 core, the STM8S family includes large on-chip flash memory with variants from 4 to 128 KB. The integrated EEPROM delivers performance comparable to external EEPROM, with real-time read-while-write capability and minimum 300,000 write-cycle durability.

The STM8S family is available in a wide range of package styles including 32- to 80-pin LQFPs, 20- to 48-pin QFNs, and a 20-pin TSSOP.

The microcontrollers begin at less than **$1** for the LQFP32 package with 32 KB of on-chip flash memory in quantities of 10,000.

STMicroelectronics
www.st.com





5 Competitive Advantages

Overseas Manufacturing

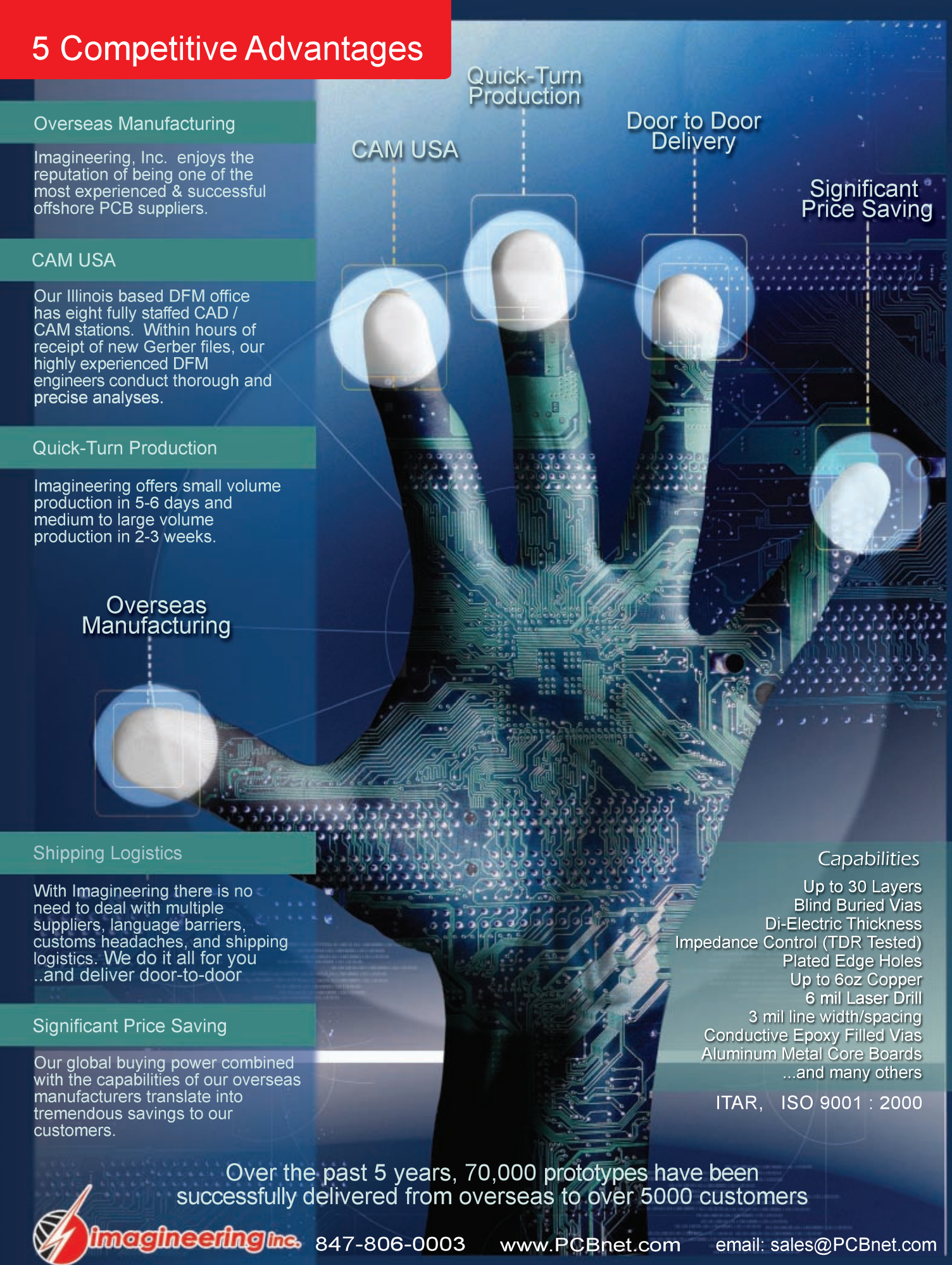
Imagineering, Inc. enjoys the reputation of being one of the most experienced & successful offshore PCB suppliers.

CAM USA

Our Illinois based DFM office has eight fully staffed CAD / CAM stations. Within hours of receipt of new Gerber files, our highly experienced DFM engineers conduct thorough and precise analyses.

Quick-Turn Production

Imagineering offers small volume production in 5-6 days and medium to large volume production in 2-3 weeks.



Shipping Logistics

With Imagineering there is no need to deal with multiple suppliers, language barriers, customs headaches, and shipping logistics. We do it all for you ..and deliver door-to-door

Significant Price Saving

Our global buying power combined with the capabilities of our overseas manufacturers translate into tremendous savings to our customers.

Capabilities

Up to 30 Layers
Blind Buried Vias
Di-Electric Thickness
Impedance Control (TDR Tested)
Plated Edge Holes
Up to 6oz Copper
6 mil Laser Drill
3 mil line width/spacing
Conductive Epoxy Filled Vias
Aluminum Metal Core Boards
...and many others

ITAR, ISO 9001 : 2000

Over the past 5 years, 70,000 prototypes have been successfully delivered from overseas to over 5000 customers

imagineering inc. 847-806-0003 www.PCBnet.com email: sales@PCBnet.com

23 YEARS IN BUSINESS...AND STILL GOING STRONG

SEGGER

Software Trials Available

JTAG (J-Link™)

+++ ARM7/9 Cortex M3 +++

- USB to JTAG
- Fast 720kb/s Download Speed
- Serial Wire Debug (SWD) Support
- Multicore Debugging Support
- Auto JTAG Speed Recognition

The J-Link can be coupled with a number of available software modules to fit your application needs.

J-Flash is a stand-alone application used with the J-Link to program internal and external flash devices.

J-Link RDI permits the use of the J-Link with an RDI compliant debugger.

J-Link GDB Server is a remote server for GDB.

J-Link Flash Download is a module used to download your program into flash even if your debugger does not have a flash loader.

J-Link Flash Breakpoint permits you to set an unlimited number of software breakpoints while debugging in flash.

J-Link SDK is a standard DLL that extends the full functionality of the J-Link to your proprietary application.



JTAG Isolator

We also offer a JTAG isolator which can be used to offer electrical isolation between your target hardware and the J-Link. This is essential when the development tools are not connected to the same ground as the application. It is also useful to protect the development tools from electrical spikes that often occur in some applications, such as motor control applications.



Special Offer

Includes the J-Link, J-Link GDB Server, and the J-Link Flash Download. www.segger-us.com/ncu.html

J-Link Non-Commercial (NCU) Bundle

www.segger.com

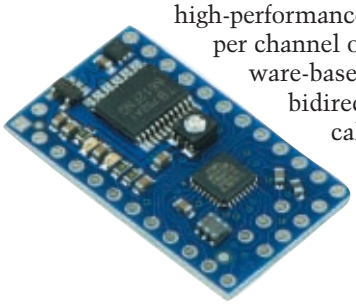
NEW PRODUCT NEWS

ROBOT CONTROLLER

The **Baby Orangutan B-168** robot controller is the latest addition to the Orangutan robot controller line. The compact module has dimensions of 1.2″ × 0.7″, and it can be configured to fit in a solderless breadboard or a 24-pin dual-in-line package socket. For applications with low I/O usage, the Baby Orangutan B-168 board can also be configured with pins on just one side of the module for use as a single in-line package. The diminutive size of the Baby Orangutan B-168 makes it well-suited for primary control of miniature robots or for auxiliary control on larger robots.

The Baby Orangutan B-168 is based on an Atmel ATmega168 microcontroller running at 20 MHz with 16 KB of flash program memory and 1-KB data memory. The use of the ATmega168 microcontroller makes the Baby Orangutan B-168 compatible with the popular Arduino development platform. Free C and C++ development tools and libraries are also available.

Integrated motor control sets the Orangutan family of controllers apart from other small microcontroller boards, and the Baby Orangutan B-168 features dual-high-performance, MOSFET-based H-bridges to deliver up to 1 A per channel over the 5- to 13.5-V operating range. With hardware-based ultrasonic PWM generation, two independent, bidirectional DC motors can be controlled symmetrically and without any processor overhead.



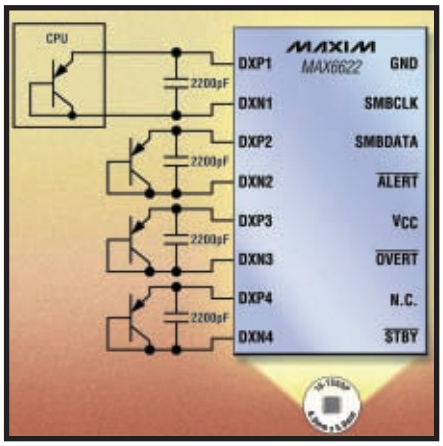
The Baby Orangutan B-168 costs **$29.95** per unit.

Pololu Corp.
www.pololu.com

HIGH-ACCURACY FIVE-/SEVEN-CHANNEL TEMPERATURE MONITORS

The **MAX6622/MAX6636** circuits are ±1°C accurate, five-/seven-channel temperature sensors. The MAX6622 accurately measures its own temperature and the temperature of up to four external locations—such as CPUs, memory, GPUs, and other heat-sensitive components—while the MAX6636 accurately measures its own temperature and up to six external locations. In both the MAX6622 and the MAX6636, one remote channel is optimized to provide accurate temperature measurements, even when used with thermal diodes on ICs operating from power supply voltages as low as 0.3 V. Offering the industry's highest accuracy, these devices are ideal for applications in which precise temperature monitoring of multiple locations is critical.

The devices increase flexibility by allowing you to program the temperature threshold of alarm outputs. These outputs can serve as interrupts, they can be connected to a system fan, or they can be used for other thermal-management circuitry—further enhancing flexibility. Additionally, these devices offer a hardware-standby function that eliminates false alerts at system boot.



The MAX6622 and MAX6636 communicate through a two-wire SMBus-compatible interface. They operate from a 3- to 5.5-V supply voltage range, and they consume only 500 µA of supply current and 30 µA in Standby mode. Packaged in small 16-/20-pin TSSOP packages, these devices are fully specified over the –40° to 125°C temperature range. The MAX6622 starts at **$3.00** and the MAX6636 starts at **$3.75** for (1,000-up, FOB USA).

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

NEW PRODUCT NEWS

PULSE-TIMING MODULES CONTROL SENSOR PULSE TIMING

The **PTM-300 Series** pulse-timing modules solve timing issues by providing precise timing control of proximity and photosensor output pulses in process control, conveyor, and machine applications. The PTM-300 input circuitry automatically detects and determines signal polarity from any standard open-collector type sensor output, such as NPN (sinking) and PNP (sourcing). The timing module then produces a timed output with the same polarity.

The PTM-300U models include an LCD and user controls for selecting the desired timing function (fixed pulse length, pulse extended, motion detection, delay-on, delay-off, buffer, and more). Pulse timing is from 10 ms to 60 s with timing resolutions of 1 ms (from 10 ms to 999 ms) and tenth seconds (from 1 to 60 s). Other features include "fault-protected" output circuitry and encapsulated timing circuitry. Selectable input modes are also provided, including compliment signal polarity and "debounce" of pulses from dry contact input sources.



Standard models operate from 10 to 30 VDC and provide "real-time" bicolor LED indication of the input and output signals' polarities. PTM-300U models are user configurable. Both are packaged in a (2" sq.) low-profile, eight-pin octal plug-in. The models start at **$83.50** and **$57.25**, respectively in one- to nine-piece quantities.

Brentek International, Inc.
www.brentek.com

ACCELERATION SENSOR MODULE WITH USB

The **JoyWarrior24F8** is a three-axis acceleration sensor with a USB interface. It is available as a complete, ready-to-use package based on a three-axis MEMS sensor with 10-bit resolution in a range of ±2 g, ±4 g, or ±8 g at 125 values per second. The JoyWarrior24F8 also offers eight inputs for use with push buttons or other digital inputs.

The JoyWarrior24F8 modules work as joystick devices. This allows easy access to the acceleration data via the joystick API of the respective operating system and eliminates the requirement of a device-specific driver. By using the system drivers, software installation is reduced to a minimum. Direct access to the detailed sensor data and setting of sensor parameters is possible via a second virtual interface. A software tool is available that allows direct access to the sensor parameters. Sample code is provided to allow the end user to write their own software. A calibration program is provided to compensate for mechanical tolerances due to mounting.

The USB interface allows for easy use with any computer. Software support is available for Mac OS X and Windows. Linux is under preparation. A configuration tool for setting the sensor parameters and a calibration tool as well as sample code is part of the software support. A low mass of less than 5 g and a small form factor of just 30 mm × 33 mm × 5 mm enables you to use it in many applications.



The modules can be used to measure acceleration or tilt angle as well as low-frequency vibration. With a unit cost of approximately **$26** in 100-piece quantities a whole lot of new applications are possible.

Code Mercenaries
www.codemercs.com



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
voltage regulator
dual H-bridge
trimmer pot (on ADC7)
power LED (green)
red user LED (on PD1)
ATmega48/168 microcontroller
programming connector
20 MHz clock
Solder Paste Stencils
Use our low-cost solder paste stencils to quickly assemble your surface-mount designs.
From $25
Custom Laser Cutting
From $25
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

NEW PRODUCT NEWS

WEATHER-RESISTANT WIRELESS THERMOCOUPLE AND RTD TRANSMITTERS

The new **UWTC-2-NEMA** and **UWRTD-2-NEMA** wireless temperature transmitters feature a NEMA 4X enclosure that is perfect for industrial applications that involve falling dirt, rain, sleet, snow, wind-blown dust, splashing water, or hose-directed water. You can program each thermocouple model to work as a type J, K, T, E, R, S, B, C, or N transmitter.

RTD models are factory programmed for PT100 sensors. Each unit comes with a lithium battery that will power the unit for over three years. Free PC software allows the product to measure and transmit process temperature, ambient temperature, RF signal strength, and battery condition in real time. This proprietary, CE-compliant product works with all Omega Engineering UWTC series receivers, WiSeries meter/controller/scanners, and DIN rail receivers.



The transmitters start at **$165** each.

Omega Engineering, Inc.
www.omega.com

LOW-POWER, AUTO-ZERO OP-AMPS

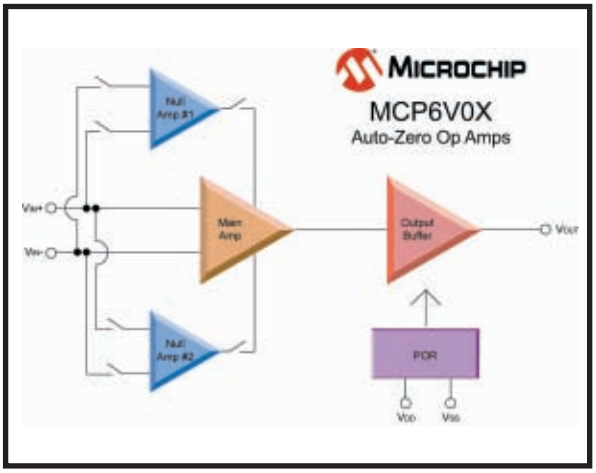
The **MCP6V01/2/3UTH (MCP6V0X)** are low-power, auto-zero op-amps. The op-amps feature a unique self-correcting architecture that enables ultra-high precision, with an input offset voltage of just 2 µV maximum. They also feature high common-mode rejection, rail-to-rail input/output, and low quiescent current; making them ideal for portable, battery-powered instrumentation devices, such as those used in the industrial and medical markets.

With their low operating-voltage range of 1.8 to 5.5 V and low typical quiescent current of just 300 µA, the MCP6V0X op-amps can run off two 1.5-V cells with full battery utilization. The devices feature a self-correcting architecture that results in fewer errors, especially at higher gains. Additionally, their rail-to-rail input/output structure enables greater dynamic range. The combined result of these features is better system performance at all operating voltages.

The MCP6V0X family is supported by the no-cost FilterLab analog filtering software tool and the Mindi online simulator tool. The FilterLab analog filtering software tool provides full schematic diagrams of the filter circuit with component values and displays the frequency responses. The interactive Mindi simulator tool enables designers to quickly generate circuit diagrams, simulate circuits, and specify passive components for a variety of power and battery-charger applications. Circuits developed using the Mindi tool can be downloaded to a PC or workstation. They are often ported directly into system diagrams.

The MCP6V0X op-amps are available for sampling and purchase today. All of the new op-amps are available in eight-pin SOIC packages, with the MCP6V02 also offered in a 4 mm × 4 mm DFN package.

The MCP6V01 costs $1.26 each in 10,000-unit quantities. The MCP6V02 costs **$1.92** each in 10,000-unit quantities. The MCP6V03 costs **$1.31** each in 10,000-unit quantities.



Microchip Technology, Inc.
www.microchip.com

NEW PRODUCT NEWS

MOTOR MOUNT AND WHEEL KIT

It's time to give your robot the mobility and style it deserves with the new **Motor Mount and Wheel Kit with Position Controller** (part number 27971) from Parallax. Powerful 12-VDC, 150-RPM motors are combined with precision-machined 6061 aluminum hardware to provide enough power, strength, and beauty to make other robots jealous. Conveniently positioned screw holes in the bearing block make mounting this kit a breeze, and the included 6" pneumatic rubber tires perform well on a variety of smooth or rugged terrains.

The kit includes two position controllers, which use a quadrature encoder system to reliably track the speed and position of each wheel with 36 positions/rotation resolution and report the data on command via a 19.2-kbps serial bus. The position controllers also can be interfaced with HB-25 motor controllers (sold separately) to automatically provide user-definable smooth speed ramping and accurate position control, which frees up the main processor to handle more important tasks.

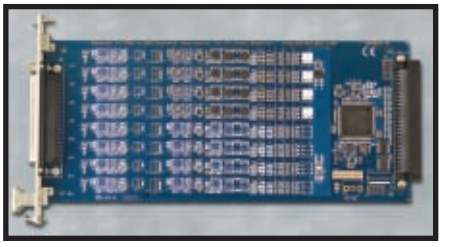


The kit costs **$279.95** and is available from stock.

Parallax, Inc.
www.parallax.com

ANALOG OUTPUT ISOLATION BOARD

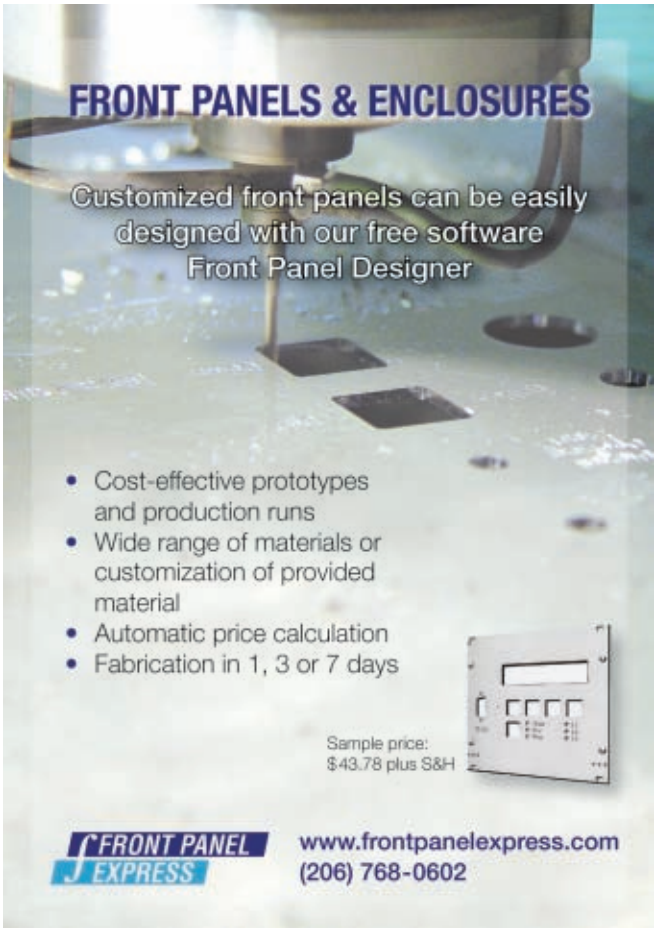
Designers can now avoid two serious problems that may affect analog outputs: ground currents and phase errors. The new **MSXB 075** board offers analog output isolation at a reasonable cost per channel, even for a few outputs. The board offers isolated analog expansion for data acquisition processor (DAP) boards. The new board includes four single-ended analog outputs, provides an isolated ground for each output, and updates all channels simultaneously.



The MSXB 075 boards fit into a backplane in a standard industrial enclosure or in a DAPserver. The DAPserver also contains one or more DAP boards and can contain other expansion boards for isolated analog input and isolated digital input and output. It then forms a ready-to-run package for applications that require complete protection from ground currents and phase errors. An included cable lets you run a DAPserver from any PC with an Ethernet connector. The MSXB 075 board is available now. The MSXB 075 costs **$475**.

Microstar Laboratories, Inc.
www.mstarlabs.com





CIRCUIT CELLAR

Test Your EQ

Edited by David Tweed

Problem 1—When processing a video signal, what is the process known as "gamma correction," and what is its purpose?

Problem 2—With modern solid-state image sensors, an operation known as "demosaic" is often required. What is this?

Problem 3—When processing video signals in the digital domain, sometimes you need a line buffer and sometimes you need a frame buffer. What's the difference, and why would you need one versus the other?

Problem 4—What do the acronyms NTSC and PAL stand for? What is their significance?

Contributed by David Tweed

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

You may contact the quizmasters at eq@circuitcellar.com

GHz BGA/QFN Sockets
0.3mm to 1.27mm

Industry's Smallest Footprint



- Up to 500,000 insertions
- Bandwith to 40 GHz
- 2.5mm per side larger than IC
- Ball Count over 3500, Body 2 - 100mm
- Four different contactor options
- Optional heatsinking to 100W

Our GHz BGA/QFN sockets provide excellent signal integrity in a small, cost effective ZIF socket for prototyping and production test use.

Request our free catalog today!

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

Expand Your I/O with Rabbit® RIO

Versatile Device with I/O Options

- Add functionality without costly processor changes
- Multiple Processor Interfaces
- Add 38 I/O
- Configure I/O for PWM, TRIAC, input capture, or decoder



Rabbit RIO™ Programmable I/O Application Kit for $299

RABBIT

Order Online At rsappkits.com

2007 WIZnet iEthernet DESIGN CONTEST
SECOND PLACE CONTEST WINNER

Content Collection and Display

Build an Internet-Connected News Ticker

James no longer has to turn on a TV or computer to get news updates. His innovative design retrieves news headlines from RSS feeds and constantly scrolls them across a dot-matrix LED display. The system checks for updates every 15 minutes.

When I first heard about WIZnet's iEthernet 2007 design contest, I was really excited about all of the possibilities an integrated Ethernet chip would open up to smaller microcontroller-based systems. I knew there were tons of networking applications that I could create (e.g., web servers, remote sensor networks, and home automation/control systems). But many of the designs that first came to mind had already been done to death, and when it came down to finding a new project, I struggled a bit. Having missed the last few *Circuit Cellar* contests, I knew I had to get an entry done for this one.

At the time, I was messing around with interfacing dot-matrix LED displays to microcontrollers. I showed a friend how to make customized messages scroll across a display by programming a computer a certain way. (He wasn't exactly a *Circuit Cellar* reader.) He was really interested in the stock market, so he thought it would be nice to have a scrolling stock ticker similar to those shown on some financial television programs. I thought it was a really good idea, but rather than scrolling stock information, I decided to scroll up-to-date news headlines.

Luckily, there are many news organizations that provide free news updates via the Internet. Nearly all of the major ones provide really simple syndication (RSS) feeds for news headlines as well. RSS gives content providers a way to distribute periodically updated material. Conveniently, RSS uses XML, so the relevant data is well-structured for processing by clients.

To retrieve the news headlines, I wrote a cut-down HTTP client and XML parser. I wanted the device to operate without any action by the user, so I implemented a domain name service (DNS) and dynamic host configuration protocol (DHCP) client in order to use dynamic IP addressing. Every 15 min., the device renews its client IP, requests the IP of the BBC News RSS server, parses the XML for news headlines, and scrolls them across the display (see Photo 1).

The system features a Microchip Technology PIC18F2525, PIC18F2221, and a WIZnet WIZ810MJ Ethernet module. In this article, I'll describe how I used these building blocks to design the news ticker.

SOME CHALLENGES

I wanted the design to be as simple and inexpensive as possible. The WIZnet

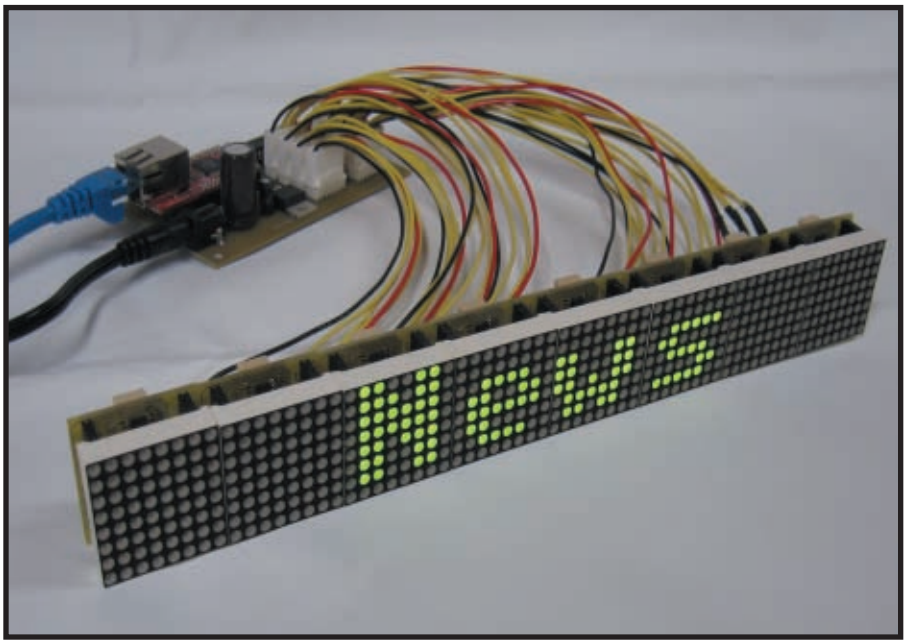


Photo 1—Check out the LED news ticker in action. Actually, you can't really see the full extent of the action. Still pictures don't show scrolling very well. Use your imagination!

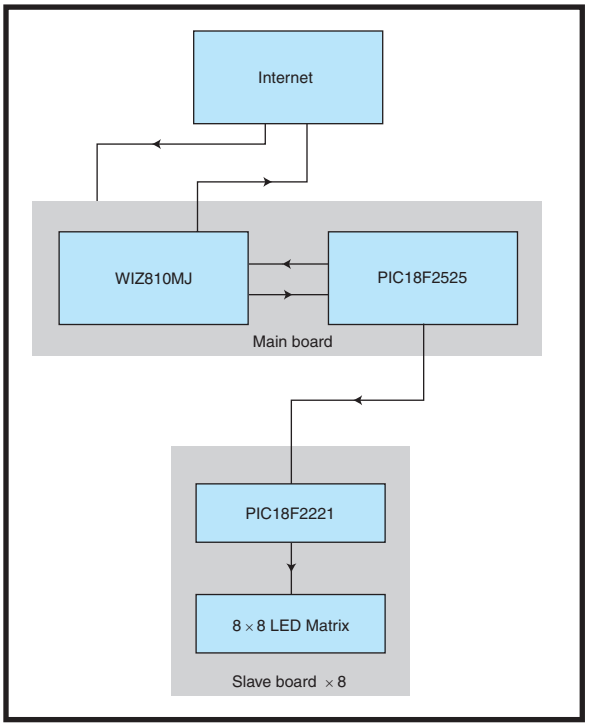


Figure 1—This is a block diagram of the system. On the main board, a Microchip Technology PIC18F2525 communicates with a WIZnet WIZ810MJ to access the Internet. The slave boards receive data from the main board while updating their LED display.

chip handles most of the network processing, so a cheaper, less complex microcontroller could be used for controlling the system. Knowing that these products have a limited amount of internal RAM, I considered adding some external storage for buffering the incoming XML data; but ultimately, I decided against that. Without a buffer large enough to store the entire web page, I was limited to using only the integrated receive buffers of the WIZnet chip. Although the software would be more complicated, the need for an additional external component was eliminated.

Controlling all of the LEDs presented another challenge. Nearly all dot-matrix LED displays are common-anode or common-cathode. Each row of the matrix is connected to the anode or cathode of every LED in that row. The columns of the matrix are connected to the opposite end of every LED in that column. Each row is then sequentially enabled while manipulating the columns of the matrix in order to illuminate the specified LEDs in that row. This process is called "scanning." If the matrix is scanned at a high enough frame rate, persistence of vision will be maintained. The purpose of scanning is to reduce the number of control pins required, along with the amount of power needed to illuminate the display.

Continuously scanning the display takes a lot of processing time, so a master/slave arrangement was used. In this setup, the master tells the slaves what image to draw while the slaves draw the image. This allows a large amount of I/O manipulation to be offloaded to the slaves, leaving the master free to do more important things.

A side effect of matrix scanning is a reduction in display brightness. Because each row of the display is on only once per frame refresh, and you have eight rows in the frame, the average current through an LED will be 12.5% of its instantaneous current. If the LEDs and their current-limiting resistors are specified for 20 mA (a pretty standard LED current), the average current through them will be only 2.5 mA due to the low duty cycle of matrix scanning.

Fortunately, the LEDs are still bright enough with the reduced current. I actually halved the recommended LED current from 20 to 10 mA. With the average current through each LED equal to 1.25 mA, the display was still easy to read in daylight.

A nice feature of LEDs is that they can withstand pulses of current much higher than their recommended average current. If the display is not bright enough using the 20-mA recommended average current, it can be raised according to the specification of the LEDs. For the LEDs in my displays, the maximum pulse current is 100 mA. With a 12.5% duty cycle, this gives an average current of 12.5 mA through each LED—still well within the recommended average range.

Make sure to select an appropriate current for your LEDs. There are 128 LEDs in each display, and I used eight displays in this system for a total of 1,024 LEDs. If 12.5 mA was selected as the average current through each LED, the maximum average current draw for the entire display would be 12.8 A! Using an average LED current of 1.25 mA

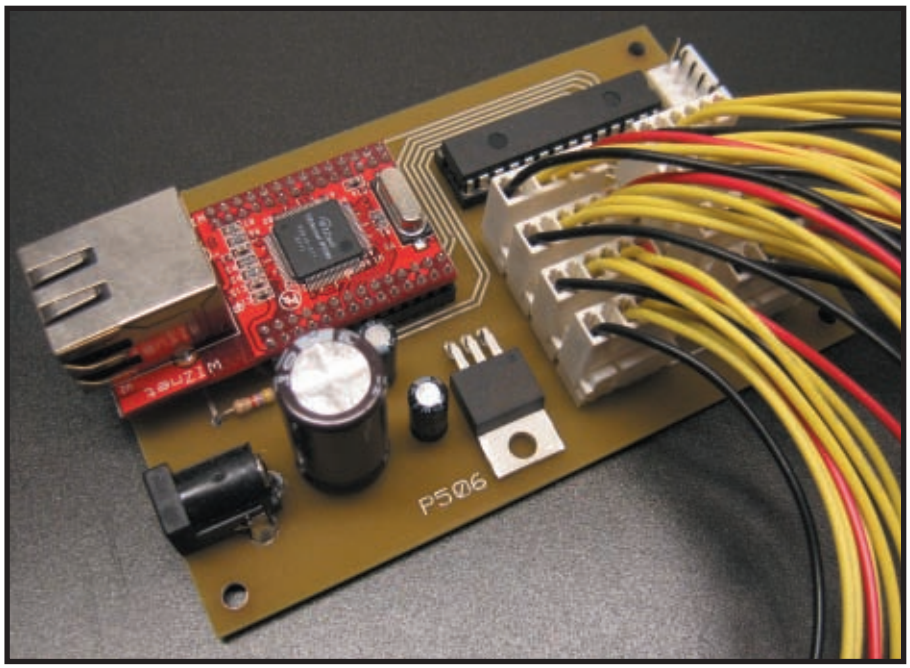


Photo 2—This is the main board. Can a PCB get any simpler?

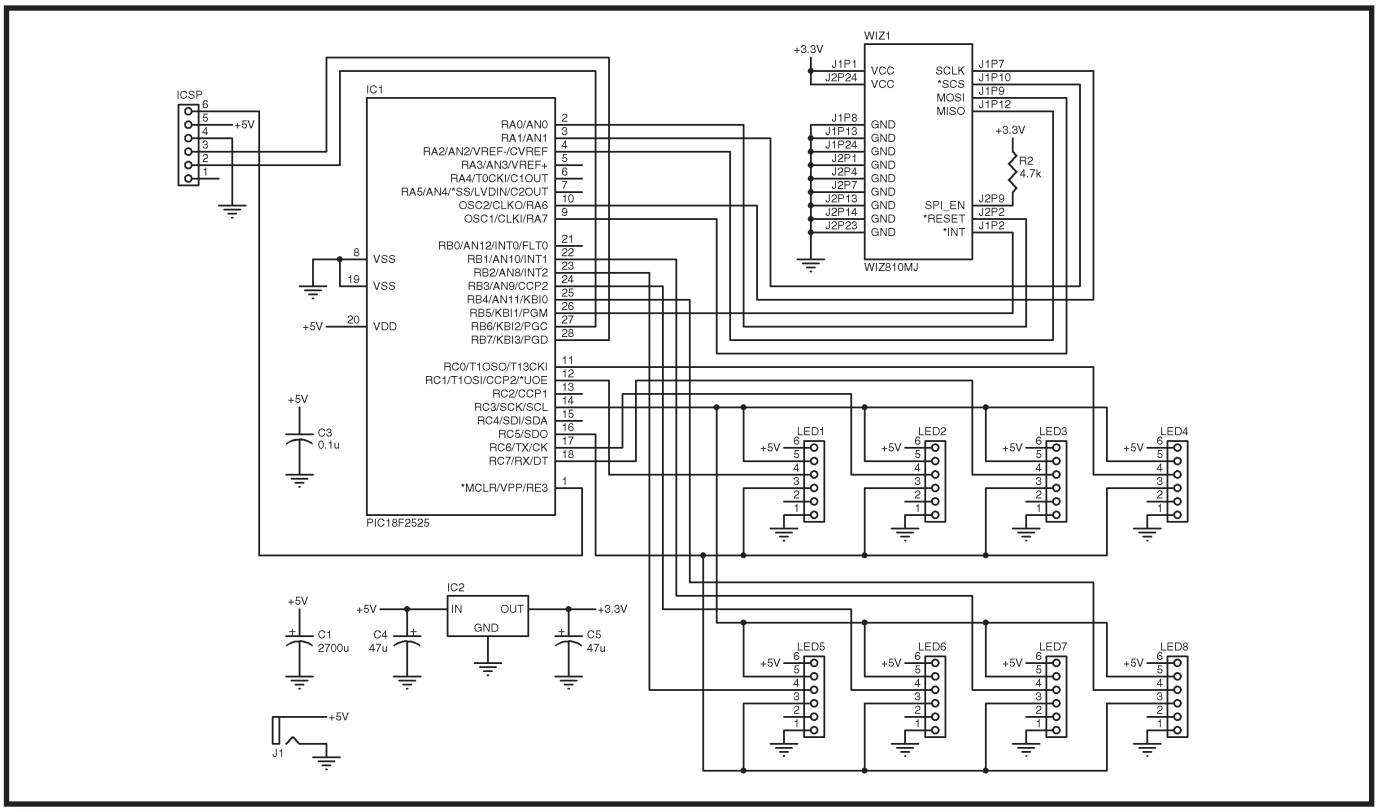


Figure 2—There's not much to the main board. The Microchip PIC18F2525 and WIZnet WIZ810MJ are the only major components.

reduces the total current required by a factor of 10.

You're now familiar with many of the gotchas I encountered. Let's move on to the hardware behind the project.

THE HARDWARE

As you can see in Figure 1, the design consists of a main board and eight slave boards that piggy-back to bicolor dot-matrix LED displays. The main board is responsible for handling networking-related tasks, updating the frame buffer for the display, and sending each slave the current frame it should be drawing. The slave boards continually paint their frame buffers to their respective LED matrix while waiting for the main board to send them updated display data.

Each LED matrix contains 64 pixels, organized as an 8 × 8 square. Each pixel is represented by a red and green LED, for a total of 128 LEDs per matrix. A maximum of eight characters can be fully displayed at one time.

The main board is fairly simple (see Photo 2 and Figure 2). It features a PIC18F2525 at 32 MHz, a



WIZ810MJ Ethernet module, power supply circuitry, and headers for connecting the slave boards. The WIZnet module houses a WIZnet W5100 integrated Ethernet controller and an RJ-45 jack with integrated magnetics.

The eight slave boards are also simple (see Photo 3 and Figure 3). Each one uses a PIC18F2221 at 32 MHz, a 1.32″ 8 × 8 bicolor, common-anode dot-matrix LED display, a pair of Texas Instruments TPIC6C596 power shift registers, and four Fairchild Semiconductor FDC6312 dual-channel P-FETs. Shift registers are used to further reduce the number of I/O pins required to interface to the matrices. Each shift register is 8 bits wide, so two are used to control the 16 LEDs that make up the columns of each matrix (one green, one red, eight of each). They also provide a high-power current sink for the LED cathodes. The P-FETs act as high-current, high-side drivers for the common-anode rows.

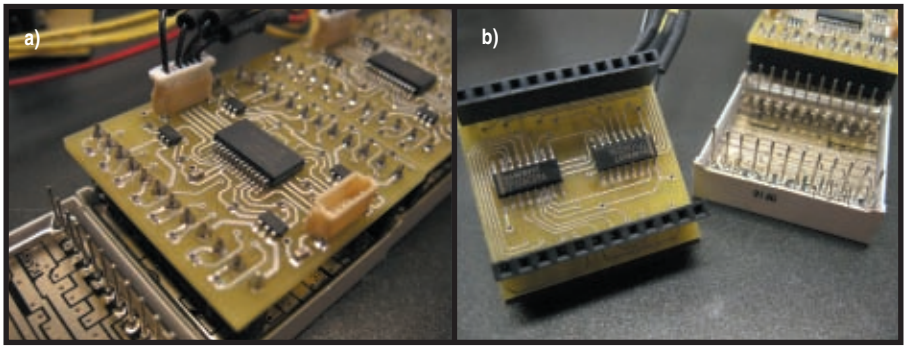


Photo 3a—Take a look at the slave boards. Soldering eight of these by hand took forever. **b—**This is a close-up of how an LED matrix and a slave board snap together.

A 5-V, 2.5-A, regulated switching power supply provides power to the system. Even with the current reduction from matrix scanning, a fairly large amount of current is required—roughly 1.3 A with the display fully illuminated. In reality, the device draws only around half that because I don't use the bicolor capability. There are a lot of unlit pixels as well due to the shapes of each letter and spaces between words. Still, I have some wiggle room in case I ever decide to increase the display brightness.

The microcontrollers and LED circuitry are powered by the 5-V rail provided by the regulated switching power supply. A linear regulator on

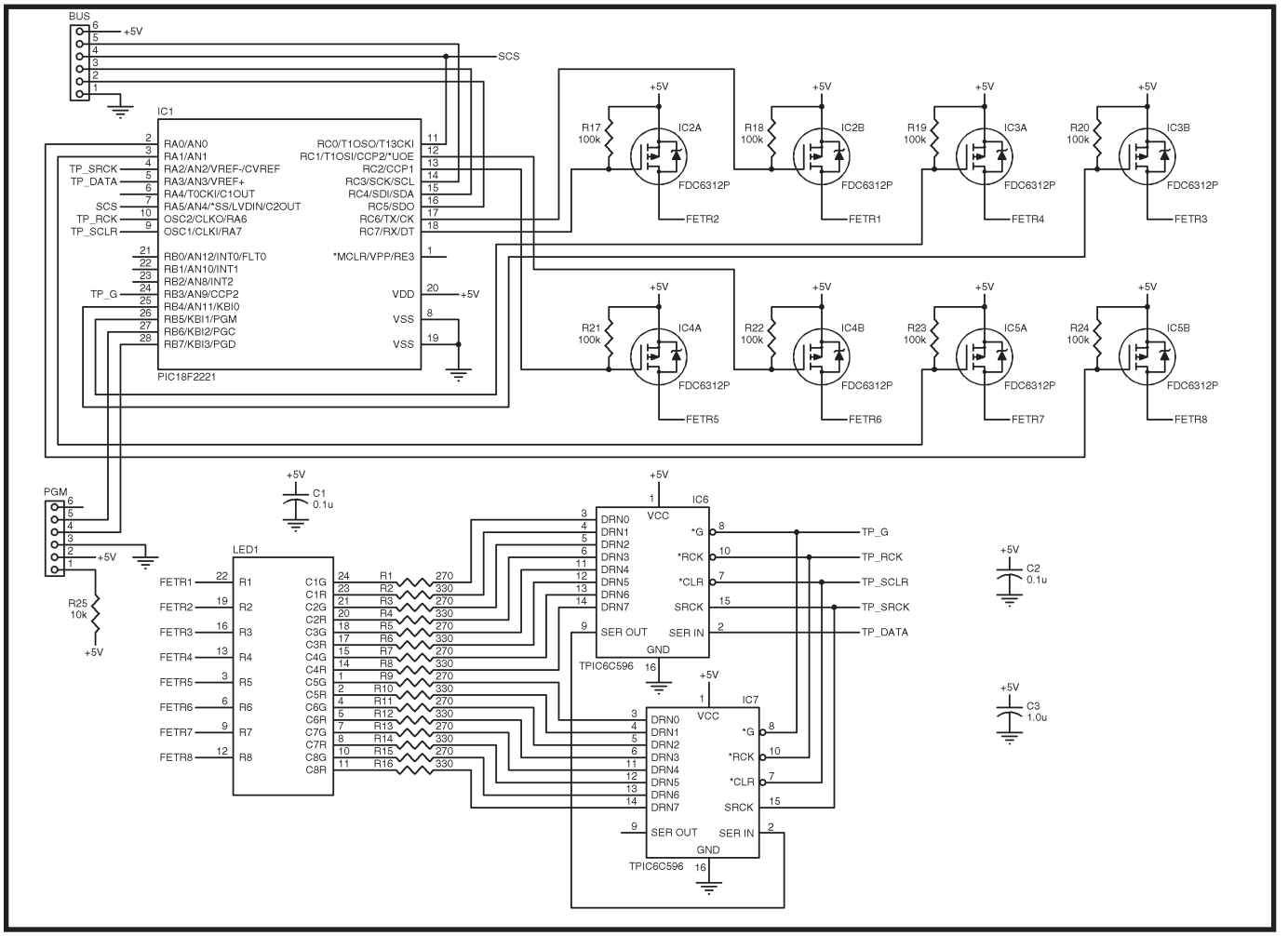


Figure 3—Here are the slave boards in all of their glory. Most of the components are for interfacing to the LED matrix.

the main board provides the 3.3 V required by the WIZnet module. The module and the PIC18F2525 are 3.3-/5-V-tolerant, so level-shifting circuitry isn't required.

The master communicates with the slaves via a SPI bus. Each slave has its own chip-select line. The bus's data rate is kept at a minimum due to the distance between the main board and the slaves. A SPI bus is also used to communicate with the WIZnet module.

WIZnet MAKES IT EASY

Having to deal with implementing a software TCP/IP stack with fairly limited resources is a pain, so a fully featured drop-in solution was attractive. There was also the added bonus of offloading a nontrivial amount of processing to an external chip, leaving the master to skip a lot of the grunt work. The WIZnet chip greatly reduced the complexity of the main board's firmware.

The WIZnet chip supports many protocols, including the well-known TCP and UDP among others. These two protocols are used for all of the networking services that the main board implements (DHCP, DNS, and HTTP). The W5100's socket interface enables connection to other nodes on the network.

You probably read Fred Eady's article, "iEthernet Bootcamp: Get Started with the W5100" (*Circuit Cellar* 208, 2007). Fred explained the socket interfaces better than I can, so refer to his article if you want to delve into the details of setting up the W5100 registers correctly. Suffice it to say, depending on the protocol used, all that's needed for a socket connection are the IP addresses of the client and host and the source and destination port. Once these are established, you can begin sending and receiving data.

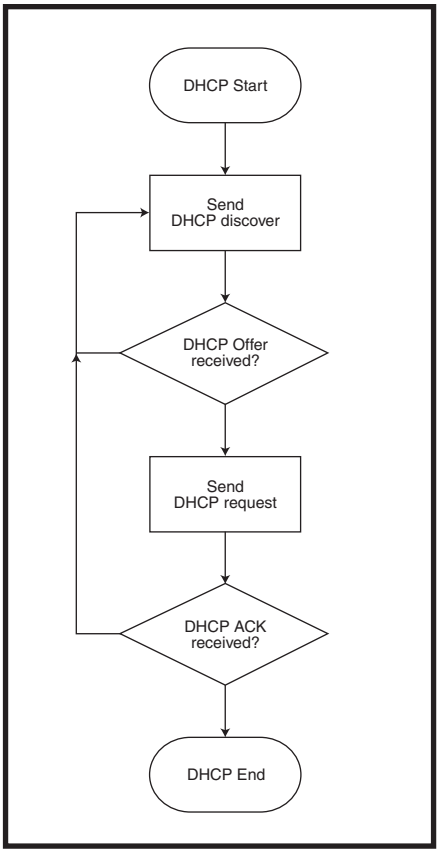


Figure 4—Packets are exchanged by the DHCP client and server to obtain local network information. The DHCP client will repeatedly request an IP until one is leased by the server.

MAIN BOARD FIRMWARE

The firmware for the main board was written in C using Microchip's C18 compiler and MPLAB IDE. Nearly all 4 KB of RAM provided by the PIC18F2525 is used. Less than half of the 48 KB of program memory was used, so some functionality can be added.

The program itself is separated into several different tasks and is entirely interrupt driven. Short tasks are executed in the interrupt, while long tasks poll status flags and execute in the main loop. The main board's primary tasks are the news update task, the display task, and the "updating message" task.

NEWS UPDATE TASK

The news update task is responsible for DHCP leasing, DNS querying, downloading news updates via HTTP, and formatting them for display. Three out of the four sockets are used on the WIZnet module (one for each of the previous services). The task is called approximately every 15 min. While it is running, it calls the "updating message" task to display a message while the headlines are downloaded from BBC News.

Every time the task is run, a DHCP lease is requested and a DNS query is made to resolve the IP address of newsrss.bbc.co.uk. Although caching and lease times are provided from the queries, these are ignored for the sake of simplicity.

DHCP

DHCP is a convenient feature to have on a device like this. It enables the automatic retrieval of local network information, including a free client IP address, the subnet mask, the gateway IP address, and the DNS server IP address. All of these are required to establish a connection on the Internet. Without DHCP, this information would have to be manually entered periodically because the addresses are dynamic. I wanted the device to operate without user interaction, so implementing a DHCP client was required.

DHCP uses UDP to facilitate client and server communication and is implemented on its own socket (socket #1). Figure 4 shows the process of obtaining local network information using DHCP.

The client first broadcasts a DHCP Discover packet, which alerts DHCP

Listing 1—The description tag stores the news headlines you want to extract. Everything else can be ignored.

```
height="49"
url="http://newsimg.bbc.co.uk/media/images/44531000/jpg/_44531986_usflag66i_ap.jpg
"/></item> <item><title>Amnesty pressures IOC over
China</title><description>Amnesty International calls for condemnation of China's
handling of protests in Tibet.</description><link>http://news.bbc.co.uk/go/rss/-
/2/hi/uk_news/7325754.stm</link><guid
```

servers on the network. Multiple DHCP servers may respond with a DHCP Offer packet. This packet notifies the client of an IP if one is available for lease. The client responds to the first server that responds. The client then sends a DHCP Request packet to the first DHCP server found, requesting the IP that was previously received in the DHCP Offer. If the server accepts the request, a DHCP ACK packet is sent. This confirms with the client that the IP is now assigned. If an IP is not served, the process will repeat until a client IP is obtained.

DNS

DNS is another useful feature to have. Instead of having to remember the IP address of every web site you visit, DNS allows familiar hostnames (e.g., www.circuitcellar.com) to be used to connect to remote hosts. Many host IP addresses are dynamic, so DNS is another must-have (unless you want to update the IP manually, and that's no fun).

DNS uses UDP for communication between servers and clients. It is implemented on its own socket (socket #2). The process is straightforward and has only two steps. The client sends a DNS query packet to the DNS server that was identified through DHCP. The query packet contains the requested domain name (newsrss.bbc.co.uk in this case) in the form of questions. Multiple questions, and hence domain names, can be sent, but you need to look up only one domain name in this case. Following the question section is the answer section. This area will be used by the DNS server's reply, but it is unused in queries.

The DNS server sends a DNS response after receiving a query. It is noteworthy that the format for queries and responses is identical. The DNS response contains the original question sent by the client along with an answer. Multiple answers can be sent even if only one question is asked. Some of the answers may be redirected domain names, which can safely be ignored. The IP address of the server is the only data you're looking for. The DNS client parses the answer section until it finds the required IP address of the host.

HTTP

HTTP is used to obtain the XML data from the BBC RSS news feed. It uses TCP and is implemented on its own socket (socket #0). HTTP supports several actions that enable data transfers from the host to the client, or vice versa. In this case, the only action you're concerned with is `GET`. There's nothing complicated here. The `GET` action retrieves a web page. The only data needed for the `GET` action is the URL of the web page you want. Once a `GET` action is received by the host, it starts transmitting the requested page to the client.

The XML data must be parsed to extract the desired news headlines. Fortunately, XML provides a nicely structured set of data by enabling providers to define their own elements. Elements are basically data grouped around "< >" tags. The BBC RSS feed provides the news headlines in elements named description. The tags will appear as `<description>` in the XML. Listing 1 shows a sample of a description element and the news headline it contains.

The XML data is parsed while it is still in the WIZnet chip's receive buffers. The parser steps through the buffer, looking for the beginning of a description tag. Once one is found, it stores the following text until the end of the description tag is found (`</description>` in this case). The news headlines are stored in the PIC18F2525's internal memory and most of it is used for that purpose. About 20 headlines can be stored, each with a maximum of 145 characters. BBC headlines are generally less than 120 characters. If a headline is longer than 145 characters, additional characters are discarded.

Once all of the headlines are stored, they must be parsed again. XML uses special sequences of characters to represent quotation marks, apostrophes, and ampersands (`"`, `'`, and `&`, respectively). The parser must find them and convert them to the correct characters.

DISPLAY TASK

The display task updates the slaves with display data and maintains the frame buffer. It is called approximately every 16 ms.

Compared to the news update task, the display task is relatively simple. It shifts all of the data in the frame buffer left one pixel every time it is run. When the frame buffer has been shifted eight pixels, it loads the next character into the buffer. Once the last headline is displayed, the task wraps around to the first headline.

The frame buffer holds pixel data for 10 frames. The display itself shows only eight of those frames (one frame for each LED matrix). An extra frame is on both ends of the buffer and they're there to facilitate scrolling. Each frame in the buffer stores 8 bytes of information. Each byte represents a column of the LED matrix. Each bit then determines whether that pixel is on or off. Sixty-four bits are used for each frame (8 bytes × 8 bits), with each bit corresponding to one pixel on the LED matrix.

To convert news headlines to pixels, an 8 × 8 font table is used. The table maps ASCII characters to pixel data for the LED displays. Eight bytes are used for each ASCII character. Like the frame buffer, each byte represents a column and each bit represents a pixel.

SLAVE BOARD FIRMWARE

The firmware for the slave boards was written in C using Microchip's C18 compiler and MPLAB IDE. Few resources are used on the slaves because their responsibilities are limited.

Compared to the main board, the slave firmware is pretty dumb. The program sits in a loop, continually refreshing the display, while waiting for the SPI interrupt to trigger. The SPI interrupt routine simply loads the data received from the master into a frame buffer.

The infinite loop continuously refreshes the LED matrix. Only one row is illuminated at a time, and rows are illuminated sequentially. Each byte in the 64-byte frame buffer holds

the color information for a single pixel. The data corresponds to the binary representation of which LEDs should be illuminated. For each row, a 16-bit number is generated holding the bit pattern for the LED columns. For example, the binary bit pattern B"11111111 11111111" would turn on every LED. The bit pattern B"01010101 01010101" would turn on every red LED. The pattern B"00000000 00000000" would turn off all LEDs, and so on. When the entire frame has been refreshed, the process begins again.

HAVE FUN

This project was a success. It is still sitting on my workbench, so I can read news headlines when I am not near a computer.

You learn a lot by doing a project like this because there's so much functionality involved. There's also a lot of room for new features that I didn't have time to implement. One that I will eventually integrate is the ability to upload a customized message through a web page that the main board will host. I may even add the ability to scroll stock information in the future.

James Blackwell (j.blackwell@azosoft.com) graduated from Cleveland State University in 2007 with a Bachelor's degree in Electrical Engineering. He currently works as a design engineer for Delta Systems in Streetsboro, Ohio. When James isn't perfecting his perpetual motion machine, he enjoys playing bass guitar and drums.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2008/218.

RESOURCES

F. Eady, "iEthernet Bootcamp: Get Started with the W5100," *Circuit Cellar* 208, 2007.

WIZnet, Inc., "Application Note – DHCP," 2008, www.wiznet.co.kr/rg4_board/view.php?&bbs_code=en_pds&ss%5bst%5d=1&ss%5bsc%5d=1&kw=dhcp&bd_num=15832.

———, "Application Note – DNS," 2008, www.wiznet.co.kr/rg4_board/view.php?&bbs_code=en_pds&ss%5bst%5d=1&ss%5bsc%5d=1&kw=dns&bd_num=15833.

———, "W5100 Datasheet," 2008.

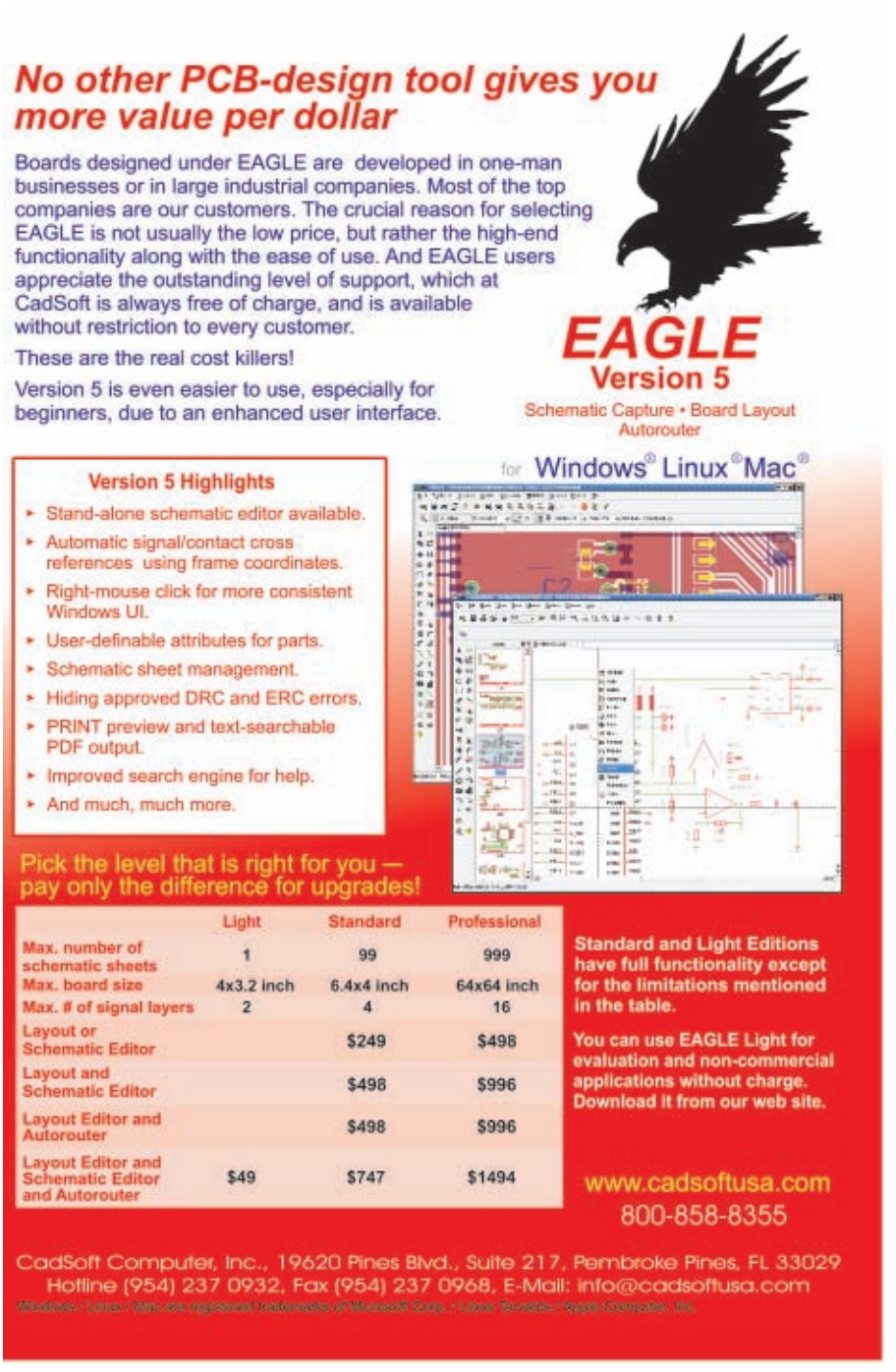
SOURCES

FDC6312 MOSFET
Fairchild Semiconductor Corp.
www.fairchildsemi.com

MPLAB C18 Compiler, PIC18F2221 microcontroller, and PIC18F2525 microcontroller
Microchip Technology, Inc.
www.microchip.com

TPIC6C596 Shift register
Texas Instruments, Inc.
www.ti.com

WIZ810MJ Ethernet module
WIZnet, Inc.
www.wiznet.co.kr/en



Living and Working Off the Grid

Part 2: Installation and Performance



George continues describing how he installed an off-grid photovoltaic (PV) power-generating system in New Mexico. He details how he mounted the panels, connected the batteries, and monitored the system's performance.

LESSONS FROM THE TRENCHES

by George Martin

Last time, I described the planning and design phase of our off-the-grid solar system in Silver City, NM. This month, I will cover the installation process and explain how the system operates.

All the mounting, wiring, and enclosures had to comply with the latest National Electrical Code (NEC) and any local variations. The code is there for a purpose. There may be variation in the details with your local installer or inspector, but approaching them in a professional manner will go a long way. Remember, as Al Capone once said, "You can get much further with a kind word and a gun than you can with a kind word alone."



Photo 1—The plywood board is painted white per local code. Here you see the mounting panels and the DC distribution box.



Photo 2a—This is the starting assembly with the major pieces in place. **b—**Here you can see more wiring. **c—**This is a close-up of the AC side. **d—**This is a close-up of the DC side.

Refer to the National Fire Protection Association's web site for information on the 2008 National Electrical Code (NEC2008), which is the latest version (www.nfpa.org).



Photo 3—Here you can see the battery stand, the engine hoist, the 700-lb. battery, and my able-bodied assistant. Watch those fingers!

The building where this equipment was to be installed was under construction during the planning stages of this project, so I was free to configure things as I saw fit. Actually, one of my original requirements was that the system wouldn't impact our lifestyle. As a result, the interface to the house from the solar equipment panel looks just like power came in from a utility. I have normal outlets throughout the house. No strange DC outlets or any goofy low-voltage wiring. I use normal 110/220 all the way.

Let me also add that it soon became clear that I did not want to try to read and understand the code. So, I approached the electrician who won the bid on the house's electrical work. TAZ Electric (Tim Clark and son) used to be located in Alaska. Installing solar systems was what Tim did for a living. Jackpot! He agreed to wire up the equipment. What a big load off my mind. Also, he was comfortable with me doing as much or as little of the work as I wanted. All of the pieces were falling into place.

MAIN OUTBACK MOUNTING PANEL

I purchased the inverters, charge controllers, AC distribution box, DC distribution box, and mounting panel from OutBack Systems through The Solar Biz (www.thesolar.biz). The folks there had good prices, were knowledgeable about solar installation, and agreed to stage the delivery of equipment. Plus, it was only about a 2-h drive to their shop across some beautiful wilderness country.

The interface between the solar AC (output) and the house AC (input) connects to a normal 200-A circuit panel that you might find in your home. We should have come right from the OutBack AC panel and saved this box. Chalk this up to beginner's error. The equipment mounts on two metal panels. These can hold four inverters. (I'm using only two.) I could have gotten away with only one of those metal panels, but the dealer and electrician both agreed that for the price it was better to have a lot of space to work with.

Photo 1 shows a plywood board (painted white per local code) with the mounting panels and the DC distribution box installed. I did this during one of my cross-country trips. It was actually painted in December 2006. The height of these panels is also part of the code, so be careful here (41″ off the ground, but check and double check this number). You do not want to have to move all

of the equipment.

MOUNTING EQUIPMENT

I had a platform to mount the equipment. I started mounting the inverters in January. I was there by myself. Each inverter weighs 62.6 lb! There was no way that I could have lifted that over my head to bolt it down. Even if I had had help to install it, I wouldn't have been able to remove them for service. So, I gave up and went back to Connecticut. We've got a bunch of Yankees who hang out at the local coffee shop who would surely have the answer. Block and tackle was the immediate reply from the committee at the coffee shop. Of course, I went down to the boat yards and found $100 units. It was too much for me, so I went online and found a $20 unit with a 9:1 mechanical advantage. Refer to Photo 2b in my Issue 216 article to see the block and tackle used to lift one of the inverters.



Photo 4—Three panels are wired in series and run into the MPPT controllers. Their capacity is 170 W each, 510 W total, to charge the batteries and put off running the generator.

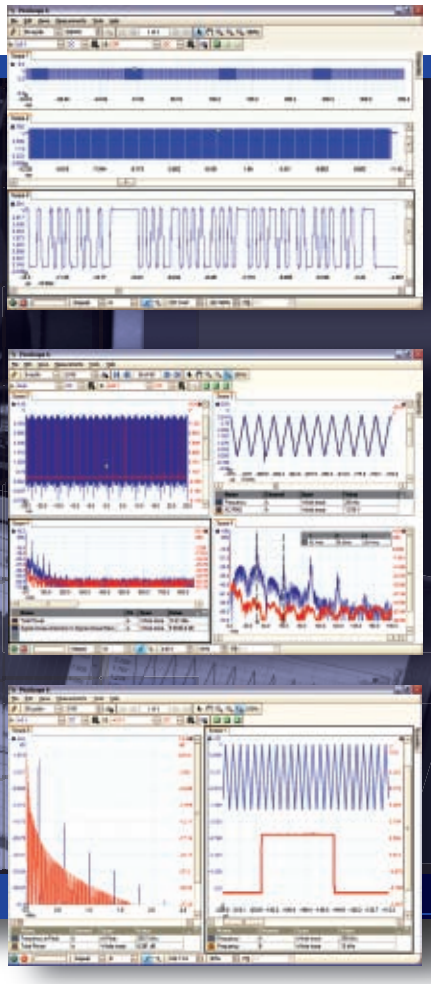
Now both the AC and DC distribution boxes are mounted along with both inverters. Time to start the wiring. The connections were simple. I suspect that if I did the wiring it would look like a mess, so I called Tim the electrician to see if he was interested. He was surprised that the equipment did not come prewired. It seems that labor is so



Photo 5a—The poles are in place and one POWER-FAB unit was installed. Oops! **b—**Here, Steve Becerra and son are lifting out a pole. **c—**The poles are in their final place with a conduit between them. **d—**These are the trenches from the poles to the house.

From the market leaders in pc oscilloscopes...

USB Connected High Speed PC Oscilloscopes and Spectrum Analyzer



5000 SERIES

PicoScope 5000 Series

The No Compromise PC Oscilloscopes

250 MHz bandwidth
1 GS/s real-time sample rate
128 Megasample record length

With class-leading bandwidth, sampling rate, memory depth and an array of advanced high end features, the PicoScope 5000 PC Oscilloscopes give you the features and performance you need without any compromise.

Advanced Triggers
In addition to the standard triggers the PicoScope 5000 series comes standard with pulse width, window, dropout, delay, and logic level triggering.

250MHz Spectrum Analyzer

High speed USB 2.0 Connection

Automatic Measurements

Arbitrary Waveform Generator
Define your own waveforms or select from 8 predefined signals with the 12 bit, 125MS/s arbitrary waveform generator.

Waveform Playback Tool
PicoScope software now allows you to go back, review, and analyze up to 1000 captures within its waveform playback tool.

BEST in TEST 2008
TEST & MEASUREMENT WORLD *Award Winner*

3000 SERIES

The 3000 Series of oscilloscopes from Pico Technology include general purpose and high resolution models: With 12 bit resolution and 1% accuracy, the 10MHz PicoScope 3424 is able to detect changes as small as 0.024% (244ppm) – making it the ideal 4-channel oscilloscope for analog design and analysis. The higher speed 8 bit models in the PicoScope 3000 series feature sampling rates up to 200MS/s and up to 1 MS/s record lengths for general purpose and portable applications.



2000 SERIES



The PicoScope 2000 series oscilloscopes offer single and dual channel units that offer highly portable/low cost solutions to general purpose testing. The award winning 25MHz handheld 2105 fits comfortably into the palm of your hand yet still includes the powerful features found in larger oscilloscopes.



pico® Technology

VISIT **www.picotech.com/pco420**

to check out our full line of pc based instruments or call 1.800.591.2796 for information and a product catalog



Photo 6—This is the maximum power point tracker in operation.

expensive and hard to find in Alaska that all equipment is prewired and then shipped to the site. He agreed to get all of the wire and do it during his next visit to the site. Photos 2a–d show the wiring internal to the AC and DC boxes.

BACKUP GENERATOR

I have a generator for the times when we have a week of clouds or the batteries are so cold they lose their capacities. To get a partial inspection approval, we connected the generator, bypassed the inverters, and got the inspection for the generator running the house (less batteries and solar panels). This was a good deal in that the contractor and electrician's work was declared complete and I could get them paid. And it gave me time to set up the panels. My wife and I were only coming out to the site for a couple of weeks about every three months. So work was in small increments.

TEMPORARY BATTERY INSTALLATION

The next step was to temporarily connect the batteries. Photo 3 shows my wife with one of the battery modules, the engine hoist, and a stand for the batteries. Each module is a 12-V battery weighing 700 lb. The 48-V battery consists of four of the battery modules connected in series. The stand was built to get the engine hoist under the batteries for lifting. I wanted to keep the batteries out of the way to make solar panel wiring easier, and then move them to their final position. In the final position, they need to be covered and vented to the outside because of outgassing during charging.

I now had a generator, inverters, and batteries in place. I could run the generator for 2 h, charge the batteries, and run the house for two to three days. Water was next. While water is not strictly part of the solar design, I installed a 3,000-gallon holding tank approximately 60′ above the house. I used gravity to provide water pressure until I could get the well work completed. So no water pumping is required. The well will also be a solar-power unit, but that's another article.

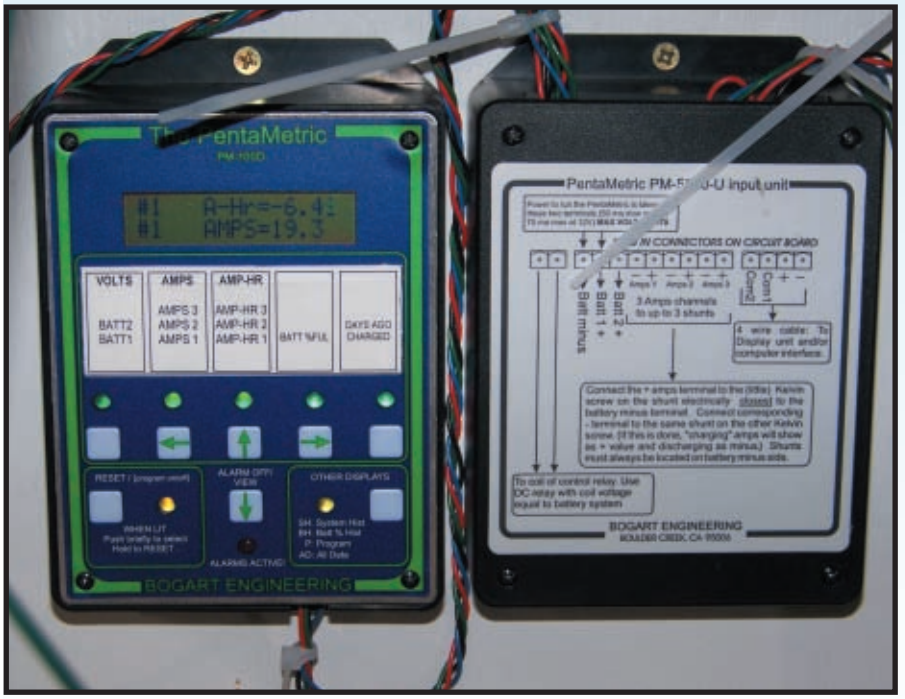


Photo 7—This is the PentaMetric system.

SOLAR PANELS

The next and last major items to install were the solar panels (see Photo 4). Remember: 18 panels, 170 W each, mounted on three poles. I used pole mounts from POWER-FAB (www.power-fab.com). At about this time, Steve Ciarcia passed me a copy of his article about his PV system, including information about all of the fuss he made wondering if his poles were strong enough (S. Ciarica, "Solar-Powering the Circuit Cellar," *Circuit Cellar* 209, 210, and 211, 2007–08). Thanks for the ground work, Steve.

I concluded that my panels were smaller in square surface area, closer to the ground, and they would never see a hurricane. I hope two out of three is sufficient. Actually, the installation instructions were clear on the



Let your geek shine.

Meet Pete Lewis, lead vocalist for the band Storytyme. Pete recently created the RS1000, a new personal monitor system for performing musicians. It was SparkFun's tutorials, products and PCB service that enabled him to take his idea to market in less than a year.

The tools are out there. Find the resources you need to let your geek shine too.

sparkfun ELECTRONICS

Sharing Ingenuity

WWW.SPARKFUN.COM

©2008 SparkFun Electronics, Inc. All rights reserved.

Hear music from Storytyme at **www.storytymeband.com**, or check out Pete's RS1000 at **www.rockonaudio.com**.

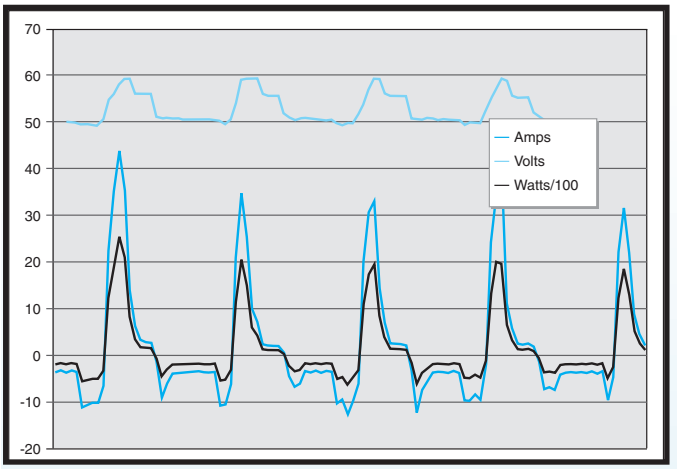


Figure 1—This is the output of the solar path program for my location.

amount of concrete needed. So, I opened the POWER-FAB packages, assembled what was mounted on each pole, and was comfortable about the next step, locating and digging the holes. With a trusty compass in hand, I located magnetic north, corrected for magnetic deviation (14° at my location), and tested my results against noon shadows. I was planning to place the three poles on an East-West line with sufficient spacing (approximately 8″) so that the panels wouldn't touch or cast shadows on each other.

THE BIGGER THE BACKHOE, THE BIGGER THE MESS

I dug the holes with my backhoe, placed the poles, lined them up, braced them, and called for the concrete pour. I then had to wait a day for the concrete to cure. I took the crew out to dinner. Boy, we were proud of our accomplishments. The next day, I set up the panel mounts.

Photo 5a shows the poles with one of them having the POWER-FAB mount and one solar panel installed. What's wrong with this picture? Notice how the solar panel overhangs the mounts. I never figured on this overhang. My spacing was out the window; the poles were too close. Boy was I sad and disappointed. There was only one thing to do: move them. So, we dug new holes at the proper spacing and pulled the pole and concrete base. Photo 5b shows Steve Becerra and his son helping move a pole and base. Well, we did it. We moved two of the poles and bases farther apart so that the panels would mount without touching. It took all day, but they are in a straight East-West line with sufficient spacing.

I placed all of the pole mounts on the poles and mounted all of the panels. At the same time, Steve Becerra was digging the trenches to connect the pole wiring to the house. There are junction boxes on the two outside poles. The 18 panels are connected in six strings of three panels to a string. Three of the strings run into each of the junction boxes. Photo 5c shows the panels from the pole along with three of the panels from the center pole all wired to the junction box. Photo 5d shows the runs from the junction box to the house. I used PVC pipe to encase the wires. Schedule 40 for above ground and schedule 80 for buried pipe.

Tim Clark came out and wired the panels and junction boxes to the charge controller. Again, neatness and code compliance was of major concern to us.

CHARGE CONTROLLERS

The solar panels connect to the OutBack MX60 charge controllers. The charge controllers then go directly to the batteries. The MX60 MPPT controllers adjust the amount of current received from the panels to maximize the power (volts × amps). Inside is a buck converter switching controller to convert solar panel voltage to required battery voltage (see Photo 6).

Typical numbers on a bright day in February are 100 V and 5 A into the MX60 and 50 V and 10 A out to the batteries. Charging 48-V batteries, these converters operate at 97% or better efficiency.

WIRE SIZING

It's an analog world, especially when it comes to wire size. Small wire is less expensive, but larger wire is more efficient (less power loss). As you can see in the photos, the battery cables are huge (actually 2/0 or 00 AWG) and only about 0.2 Ω per 1,000 m. That's well-suited for charging the 3,000 W of solar power and 8,000 W of generator power that could be going into the batteries. So, 11,000 W and 48 V with two inverters is about 230 A into the batteries or about 115 A in each cable.

The other main area of concern is connecting solar panels to the MX60. Each panel has 12-gauge wire leads attached. I kept that wire size to the combiner boxes on the poles. Each solar panel is 35 V and 4.8 A at a maximum power of 170 W. Three panels wired is 105 V and 4.8 A. Three of these stacks come into each on the combiner boxes for a total of 14.4 A to the charge controller. This wire run would be about 70′ to 80′ and consist of two wires (power and return). From tables in the MX60 manual, for less than 1.5% voltage drop, #8 wire would take you 75′ and #6 would take you 110′.

CONFIGURATION & PERFORMANCE

The inverters came with a predetermined battery voltage. I needed to set the charging voltages and charging times, low and high battery voltages, and current limits. The charge controllers first needed to be set up for the proper battery voltage. (All of this is quite detailed and would change with units from different manufactures. If you are really interested, download the manuals from the OutBack Power Systems web site.)

I added a separate monitoring system. The monitoring system is from Bogart Engineering. I selected the PentaMetric system (see Photo 7). I've just begun to learn all of its capabilities. It is connected to the battery voltage and current shunt measuring current in and out of the batteries. The shunt came with the OutBack panel. You can refer to it in the system diagram in my July article. The PentaMetric also has the computer interface. And Bogart Engineering supplies a GUI interface to set up the data that is to be monitored and saved. Using the default settings, I started logging



COMPETITION ECLIPSED

Imagine an industry first MCU capable of operating down to 0.9 V. Imagine solutions that are more power efficient than you thought possible. Imagine smaller form factors and lower system costs. Imagine best-in-class tools that enable immediate system design. Thanks to the remarkable engineering gains of our new line of MCU's, you can stop imagining and start creating your next great product.

WWW.SILABS.COM/POINT9

LOWEST VOLTAGE / GREATEST POWER EFFICIENCY
HIGHEST FUNCTIONAL DENSITY / BEST-IN-CLASS TOOLS



INTRODUCING THE C8051F9xx: THE FIRST LINE OF MCUs
DESIGNED TO OPERATE FROM A SINGLE-CELL BATTERY (0.9–3.6 V)

SILICON LABS

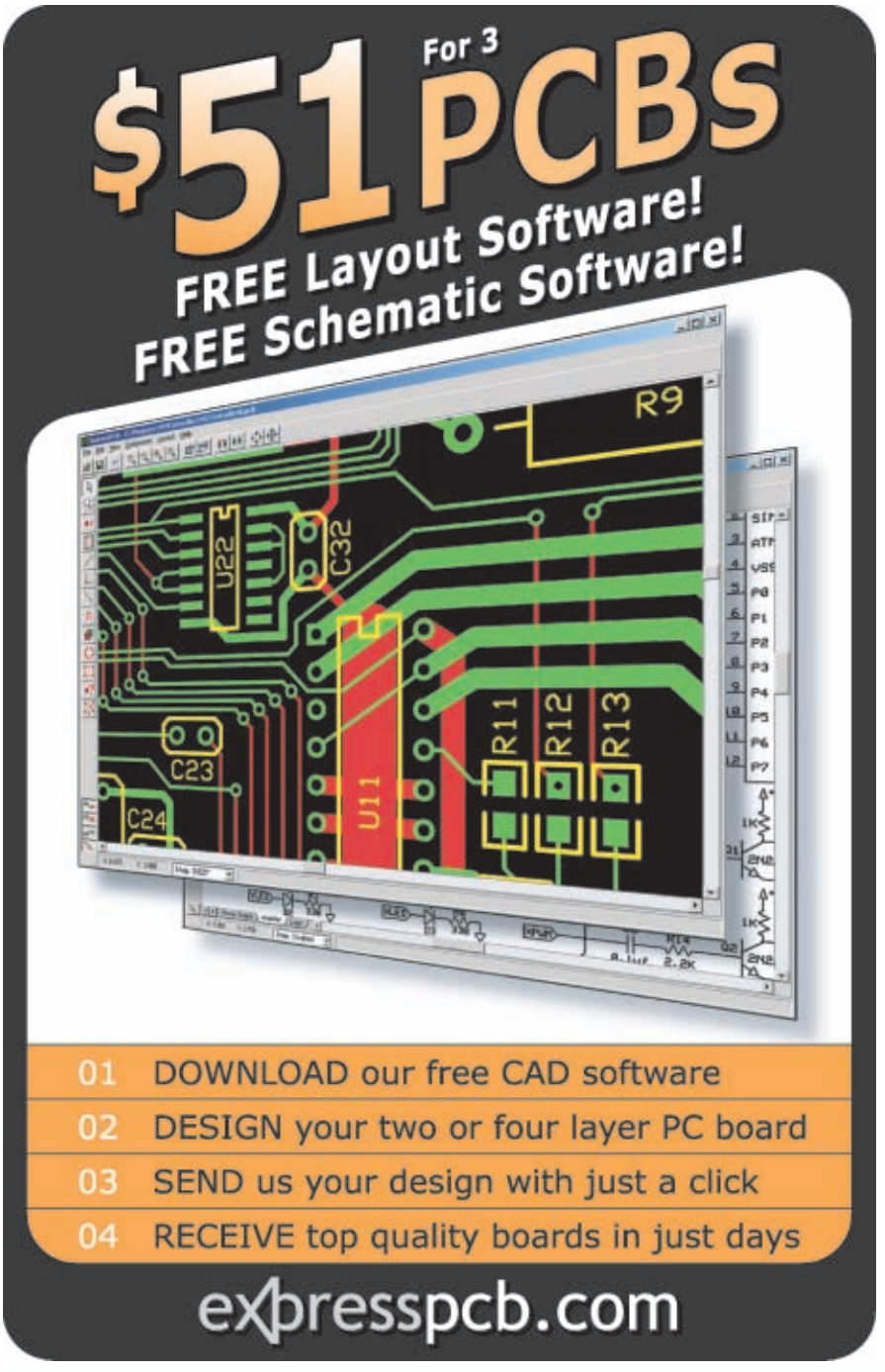
data. For the most part, this was a boring exercise because the days were so sunny and the routine about the house was so predictable.

The spreadsheet labeled PM_Mar_22_2008_PeriodicData.xls on the *Circuit Cellar* FTP site contains several days worth of data. The columns titled No Label are unused variables that I never did anything with. I was first interested in battery amps and battery volts. The column titled Watts/100 was calculated from amps and volts and scaled to fit in the graph.

Let's look a bit more into the data. First, notice that the amps line starts out negative at about 4 and continues until it goes negative to about 10. This is the early morning, which means no activity. When the house starts to wake up, the television, lights, microwave, hair dryer, and all of the other appliances come on.

Next, the sun comes up, the current swings to over 40 A, and the system is charging. The peak current is midday sun. That current profile is not the sine wave that you might expect. Once the batteries are charged, the charge controllers limit the battery current so they don't overcharge the batteries. That's the reduction of current from the high to about 5 A positive.

Later, the sun sets and normal evening activities take place, which force the current into the negative area, draw stored power from the battery until bed time, and then the cycle repeats.



Look at the graph of the voltage. Over night, it's about 50 V slowing, going lower as power is pulled from the battery. Then, while charging, the battery voltage increases to close to 60 V during the bulk portion of the charge cycle and then to 56 V during the float portion of the charge cycle. And then, as the sun sets and no type of charging takes place, the battery is supplying all of the power.

The power graph in Figure 1 is calculated from the volts and amps. I scaled it to Watts/100 to fit into the graph area. All of the power going into the battery should equal all of the power coming out of the battery. As you look at the data, you might suspect that more power is put into the batteries than is taken out of them. In fact, I added a column to the spreadsheet for cumulative watts (on the *Circuit Cellar* FTP site). Sure enough, the data supports this suspicion. What's going on? Well, first, the batteries are not 100% efficient. I have no facts to support this, but I bet you can look up different battery types and their efficiencies on the web. Second, the charging cycle is a controlled overcharge. I've read that is to keep the chemistry mixed up and to reduce the formation of sulfates on the plates. For whatever the reason, overcharging also contributes to some power loss.

I intend to add a second current shunt to separate the current the solar panels are providing from the current the inverters are delivering. I will also tune the sampling rates to see if I'm missing any interesting events.

FINDINGS

My solar system works well. Given the amount of solar input at my location, I

could probably reduce the system's size. But I'm holding off until I've seen the system operate for a complete 12-month cycle. Remember that being more careful when selecting your equipment and location can reduce installation costs.

Monitoring solar systems is boring. Nothing is going to happen overnight, but I suspect you might be able to detect performance reductions such as battery connections beginning to corrode. And this may be a lifesaver.

Batteries take maintenance. I need to add water about every two or three months. I have an automatic watering system that I need to get installed.

I've left some solar out of the house equation. I should have solar-heating water to reduce the hot water fuel (propane) and possibly supply some household heating as well. Look at solar hot water as part of any solar installation.

Yes, it's just another engineering project, so let's get going.

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

PROJECT FILES

To download the additional file, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2008/218.

REFERENCE

[1] Wikipedia, "Solar Cell," http://en.wikipedia.org/wiki/Photovoltaic_cells.

SOURCES

PentaMetric PM-100D Battery monitor
Bogart Engineering
www.bogartengineering.com

SX 170B Photovoltaic module
BP Solar
www.bp.com

MX60 Charge controller and VFX3648 inverters
OutBack Power Systems, Inc.
www.outbackpower.com

Pole mounts
POWER-FAB
www.power-fab.com

SO-6-85-17/48 Battery
Solar-One
www.hupsolarone.com

SAVANT
COMPANY INC

EARLY BIRD REGISTRATION OPEN! REGISTER TODAY & SAVE!

Platinum Sponsors

tensilica

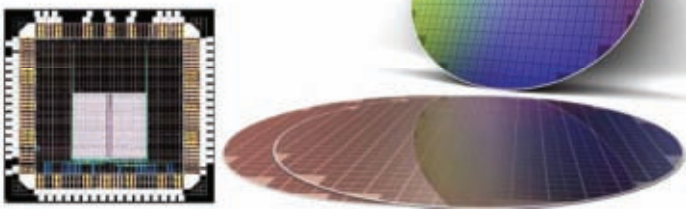
UK TRADE & INVESTMENT

novelics

Gold Sponsor

Mentor Graphics

6th International System-on-Chip (SoC) Conference, Exhibit & Workshops
November 5 & 6, 2008 — Radisson Hotel Newport Beach, California
www.SoCconference.com



Questions, Please Contact: SoC@SavantCompany.com
or (949) 851–1714 — www.SavantCompany.com

The Most Informative, Targeted & Affordably Priced SoC, ASIC, CSSP, FPGA, and Foundry Technology Conference, Exhibit & Workshops of the Year
Don't Miss Out!

www.SoCconference.com

Dynamic Animatronic Remote (Part 1)

Build a Custom Controller with a Color LCD

Peter built a custom controller that enables him to remotely control his animatronic systems. Each piece of hardware is controlled by a separate computer program. The remote polls his laptop and builds menus that show and control the state of each program.

Every year, I put on an elaborate Halloween show in my front yard (see Photo 1). Being electronically minded, my fully automated show includes a talking skeleton and crow, sound effects, and more. (In *Circuit Cellar* Issue 188, 2006, I described the animatronic controller I built for operating and controlling the props.) My system uses a laptop to coordinate the show, and it talks to several distributed microcontrollers via an RS-485 serial link. A custom packet format and messages enable PC programs to control the distributed hardware. It works great once it's dialed in, but getting it dialed in can be difficult.



Photo 1—Each year, my Halloween display gets bigger and bigger. It's amazing what you can build with Styrofoam, cardboard, and plaster (if you have the right friends)!

Last Halloween, I experienced some problems. The show wasn't working properly and I was going nuts trying to fix it. I found myself running maniacally back and forth from my house to the yard. Why? The computer used to control the show was inside and the hardware was outside. Plus, because I had darkened and covered my windows to make the house look spooky, I couldn't see or hear what was going on outside. Thus, trying to set the appropriate levels for sound and lighting was a nightmare.

Prior to last year's show, I had been thinking about building a remote control that would enable me to modify the running show from outside by plugging into the same RS-485 network that controls everything. However, I knew building such a system would be tricky. One of the hallmarks of the Halloween control system is that there is not one monolithic program running on the computer that controls everything. Instead, individual programs run independently that conform to the rules of my Halloween serial network. This enables them to send and receive packets, control hardware nodes, and play sound files as needed. Thus, while considering my options, it became obvious that building a "hard-wired" remote control to handle the show wouldn't work. I knew I needed to be able to add and subtract programs at any time, so designing a



Photo 2—Packaging is important. The finished remote's case is from OKW Enclosures. The custom panel was machined by Front Panel Express. The result looks slick and polished.

a)

Level 0: Get programs
Level 1: Display programs
Level 2: Show controls for a chosen program

b)

Level 0: Get programs
Level 1: Display programs
Level 2: Show controls for a chosen program
Sub-menu: Show additional elements that make up a group

Figure 1a—The first concept for the menu hierarchy was this simple and limited approach. Level 0 was hard-coded in the flash EEPROM. Level 1 was whatever programs were running on the PC. Level 2 was all of the controls for the current selected program presented as a single group. **b—**The second concept for menu hierarchy was similar, except it had a single level of submenus to allow for grouping related elements for a PC program. This would eliminate having all of the controls for a program in a single list on the remote.

remote dedicated to a specific program (or programs) would mean I'd have to constantly modify the firmware to keep it in sync with the ever-changing computer software.

I recently designed a dynamic remote control that meets all of my requirements (see Photo 2). The remote can connect to the network, control programs on the computer, and handle new programs. It can poll the computer, find running programs, poll the PC again after program selection, and then build menus on the fly to show program status and enable modifications.

In this article, I'll describe how I designed the custom controller with its color LCD. Next month, I'll cover the software and menu code.

DYNAMIC SYSTEM

I wanted to design a dynamic system. Years ago, Apple created a specification called Publish/Subscribe. I always liked the sound of that because it seemed clear what was going on just by the name. I co-opted the name and created a Publish/Subscribe system that integrates into my Halloween serial network.

When I started this project, I figured that I could bang out the design in a few weeks, especially because I was leveraging so much software that was already written. Right. It took me more like three months. It was trickier than I thought for a number of reasons.

My original plan was to support only a few GUI elements on the PC, such as the scrollbar, checkbox, and combo box. I also decided at the beginning that the menu system displayed on the remote would have a limited depth and hierarchy. My initial plan was to have a top-level menu that enables me to tell the remote to find programs running on the PC and then click down a level to see them. It would look something like what you see in Figure 1a.

That changed. Once I got the initial system up and running, I realized that I needed to support a few more GUI elements. I also realized that I wanted to have a submenu of parameters. This way, I figured, I would be able to group things more logically. I decided that a single submenu down would be sufficient to let me group together various elements. I thought this would be relatively simple to implement because the specifications were limited (see Figure 1b). That also changed. Again, once I got the basic two-level system up and running, I

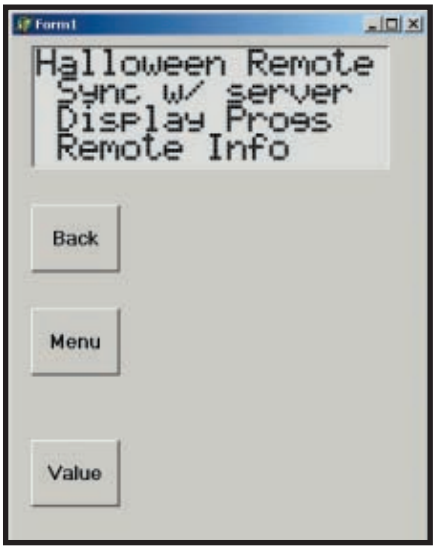
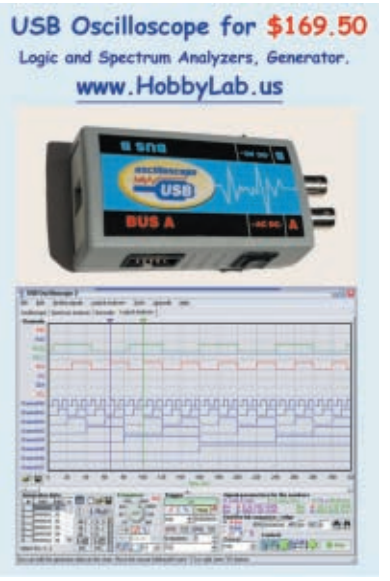
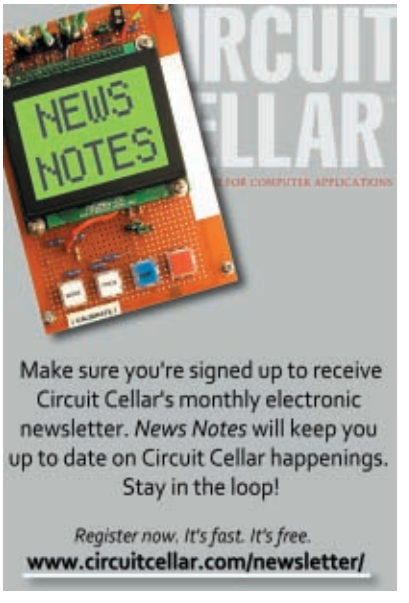


Photo 3—CodeGear's Delphi IDE was used to mock up ideas (number of buttons, placement, and more) for the physical layout of the remote. This is the final design.







Thinking about your next scope?

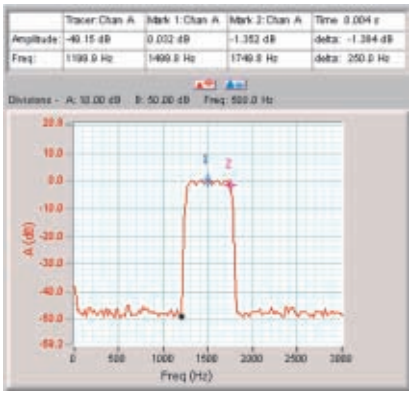
Compare the specs:

Item	Tektronix TDS2012B	Agilent MSO6012A	Pico 3206	Link DSO-8502	Bitscope USB 310	TiePie HS3-100	cleverscope CS328A
Analog Input FSD	20 mV – 50V	20 mV – 50V	±100 mV – ± 20V	50 mV – 50V	±513mV – ± 10.8V	±100 mV – 80V	**18 mV – 80V**
Overload Protection	**300 Vrms**	**300 Vrms**	±50V	±150V	±100V	200V	**300 Vrms**
Analog Channel bit resolution	8	8	8	8	8	8 or 12	10, 12 or 14
Can do hardware trigger referenced averaging	Yes (10 bit)	Yes (10 bit)	-	-	-	-	**Yes (14 bit)**
Analog Input Offset	±2V:→200mV then ±50V	**±25V:→200mV then ±100V**	None	±5 divisions	None	None	±3.5V → 200mV then ± 80V
External trigger sampled, stored & displayed?	No	No	No	No	No	No	**Yes**
Samples of storage, all inputs recorded	2.5k	2000k or 8000k	500k	512K	64k	128k	**4000k or 8000k**
Resolution on 2V offset signal	0.078 mV	0.078 mV	15.6 mV	15.6 mV	15.6 mV	15.6 mV	**0.018 mV**
Tracking graph with independent time base	No	No	No	No	No	No	**Yes**
Dynamic Range (raw unprocessed)	48 dB	48 dB	48 dB	48 dB	48 dB	48 dB	**60 dB or 72 dB or 84 dB**
Maximum time resolution	1ns	**0.5ns**	10ns	4ns	50ns	10ns	10ns
Resolution at 20ms width	8 us	**10 ns**	40 ns	40 ns	320 ns	200 ns	**10ns**
Width at max sample rate	2.5us width	1 ms width	5 ms width	320 us width	640us width	1.3 ms width	**20 or 40 ms**
Peak Captured to eliminate aliasing	**Yes**	**Yes**	**Yes**	No	No	No	**Yes**
Trigger delay	**Yes**	**Yes**	No	No	**Yes**	No	**Yes**
Protocol Analysis	No	**Yes**	No	No	No	No	**Yes**
Period Trigger	Yes	**Yes**	Yes	**Yes**	No	No	**Yes**

the review continues on our website at *www.cleverscope.com/review/*

- 100 MHz Bandwidth
- Dual mixed sig trigger
- 2 analog, 8 digital inputs and ext trig
- 10,12 or 14 bit sampling
- 4 or 8 MSamples/ channel storage
- 100 MSamples/sec
- Battery option
- USB or ethernet
- Data projector display
- Regular software and firmware updates
- Simple cut and paste into documents

Use the synchronous signal generator module to display filter, equalisation and amplifier transfer responses



Bode Plots

Transfer response for a 500 tap bandpass filter

USA master distributor

Saelig unique electronics

The CS328A is an engineer's toolbox

Oscilloscope—Tracking Graph—Spectrum Analyzer
Protocol Analyzer—Multimeter—Logger—Result Storage
Maths Equation Builder and Display—Function Generator
XY Graph—Drivers for C, Dephi and Labview
Custom Names, Units and Scaling

cleverscope

plug & play instruments

www.cleverscope.com

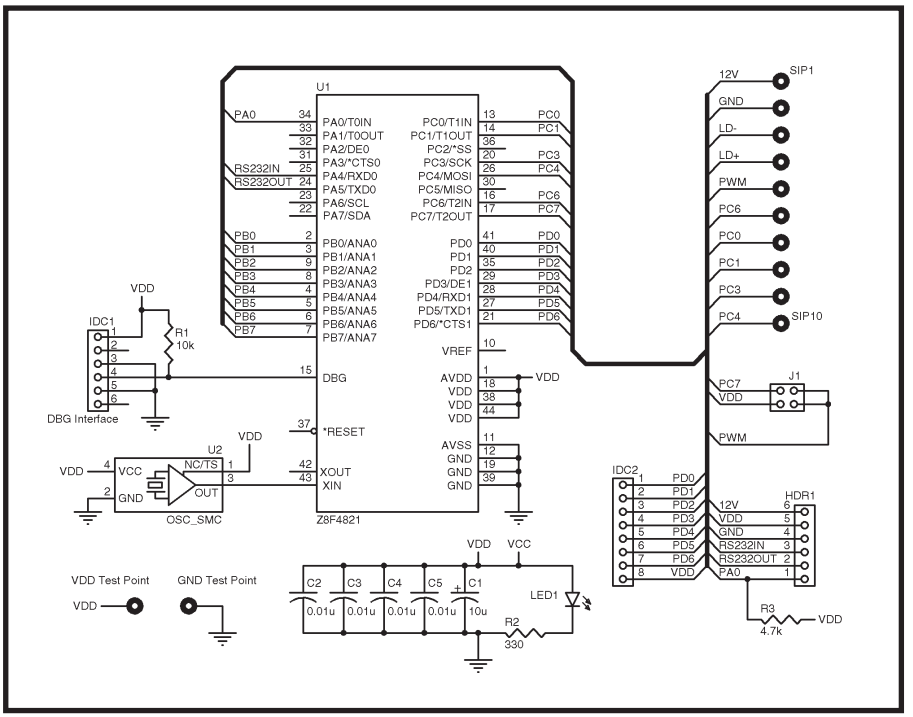


Figure 2—This is the LCD Backpack for the Display3000 display.

quickly realized how useful it would be to have a much deeper and more flexible menu system. I wanted the flexibility to lay out any menu system I pictured in my head. Tests with some fake prototype PC programs showed me that I would need to expand the number of GUI elements supported. So, I plunged into the third rewrite of the menu system.

Some engineers would see this as "feature creep." I disagree with this assessment because the original problem was that I thought I could get away with something that was simple to program, when in reality, I needed something that was simple to use. Forcing all of the controls onto a single scrolling menu page would mean a lot of scrolling up and down when looking for parameters to modify. There is often an inverse relationship between ease of use and ease of programming. This was just another example of this paradox.

DESIGNING THE REMOTE

One of the first tangible things I did was mock-up the remote using CodeGear's Delphi (see Photo 3). The interface builder in Delphi is simple to use. (There are others, but I prefer Delphi.) It enables me to play "what-if" games quickly. My goal was to figure out what the remote's front panel was going to look like. I knew that it would have an LCD. I also knew it would have buttons. The questions were: How many buttons, and what would they do? Would I need a full 10-key keypad? What about an Enter key notifying the remote about a chosen value?

After some experimentation, I came up with the design shown in Photo 2. The remote has an LCD at the top and three buttons arranged vertically below it. The buttons are labeled Back, Menu, and Value (from top to bottom). What isn't shown is that the actual remote also features a scroll wheel on the side. It's implemented as a knob attached to a rotary encoder. Ironically, some time after I finalized this design, my wife gave me an Apple iPod nano. I realized after playing with it that the Apple designers used a similar concept. It's always nice to know you're in good company.

No aspect of the interface was random. I put the scroll wheel on one side and the buttons on the other. So, I don't have to choose between the buttons or the scroll wheel. I can work both at the same time. The buttons are arranged vertically so I can reach them by simply moving my thumb without shifting my grip. As for moving through the menu, I descend down and ascend out of it. Thus, the Back button is at the top so I "ascend" to return. The middle button is the Menu button, which lets me descend down to deeper menus. The physical relationship between these two buttons suggests the up/down nature of traversing the menu hierarchy.

The Value button at the bottom enables me to modify parameter values. The button is implemented as a shift style key because it modifies the behavior of the scroll wheel. If the key is up, the scroll wheel moves a highlight up and down the menu to let you know which line is active. If you hold the Value key when you're located on a parameter, moving the scroll wheel modifies the parameter's value. To keep the remote simple, I didn't include Enter and Cancel keys. This isn't a fighting video game, so elaborate key combinations are not the order of the day. I wanted a an easy-to-use, clean, simple, elegant design. I succeeded.

THE DISPLAY

I have been using LCDs in my projects for years; however, they've always been black and white character displays. For this project, I upped the ante. I used a graphical color LCD.

Prior to starting this project, I'd seen some reasonably priced LCDs advertised in magazines and on the Internet. How hard could it be to find the right one? It was hard—well, at least harder than I thought it would be. The problem with many advertisements is that the fine print tells you that the great price is available only if you're buying 1,000 or more units. There are a lot of interesting, inexpensive surplus LCDs for sale on the Internet. The problem, however, is that you get only an LCD. In some cases, they come with PCBs with LCD driver chipsets. That's a move in the right direction, but it still requires you to locate the chipset's datasheet, figure out how to program it, and then write code from

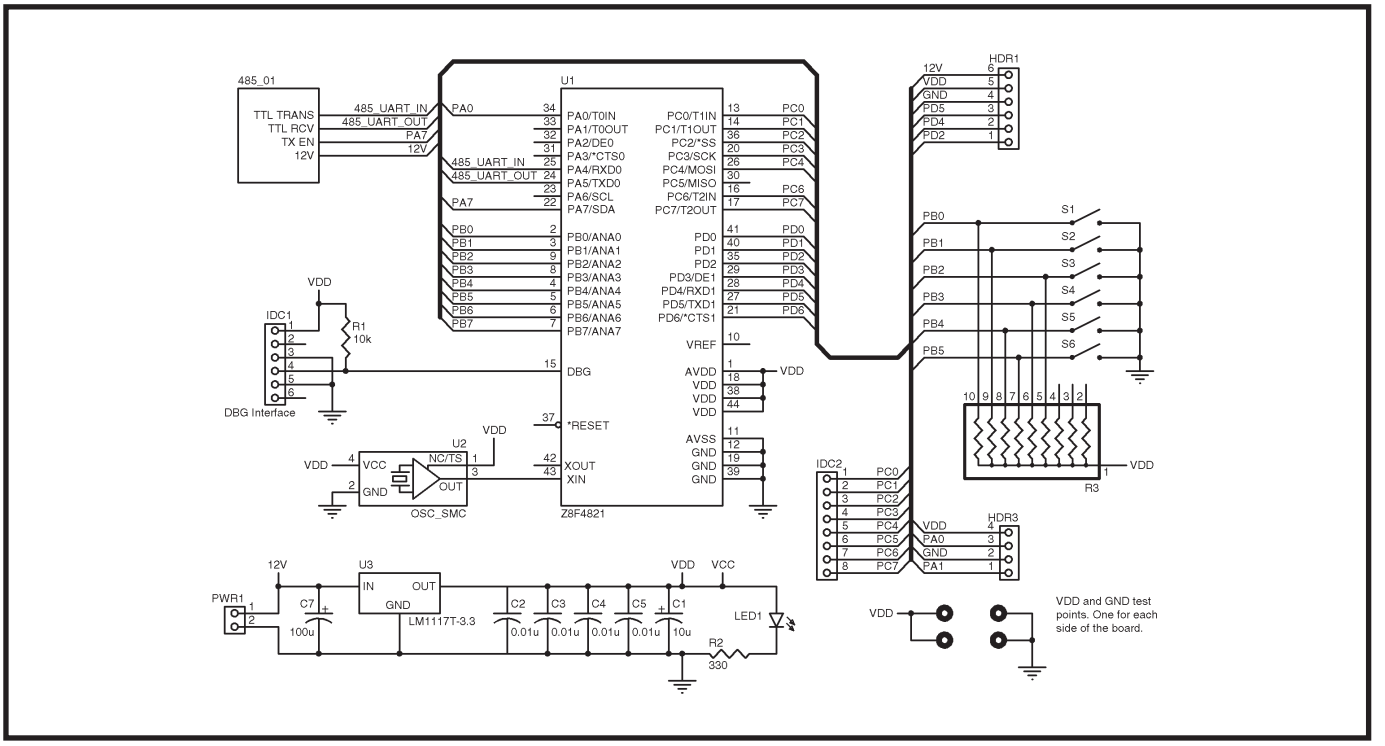


Figure 3—This is the top-level schematic for the remote.

scratch just to do things like draw a straight line. You can see why black and white character LCDs are so appealing.

Fortunately, I discovered Display3000, an Internet LCD vendor located in Germany. While the name may sound like something from an infomercial, the web site promised a good product at a good price. Specifically, I got a 2.1″ diagonal, 16-bit color LCD with a backlight. The display came with a small carrier board that did all of the required level shifting, enabling me to send data signals from 2.5 to 15 V and power the backlight with anything from 5 to 18 V. It also came with source code in both BASIC and C, which implemented a driver that included support for lines, circles, rectangles, bitmaps, and fonts. It was driven via a SPI, so it didn't take too many I/O lines. It cost me about $100 with shipping.

I was pleased when the LCD arrived. In addition to the full driver, I got a super-simple driver that implements the SPI as a few bit-banged I/O lines and displays only a yellow background with a blue line in the center. It really helps to have something dead easy to help get the display up and running on your platform. In my case, I was using a Zilog Z8 Encore. The C code from Display3000 was specifically written for the Atmel

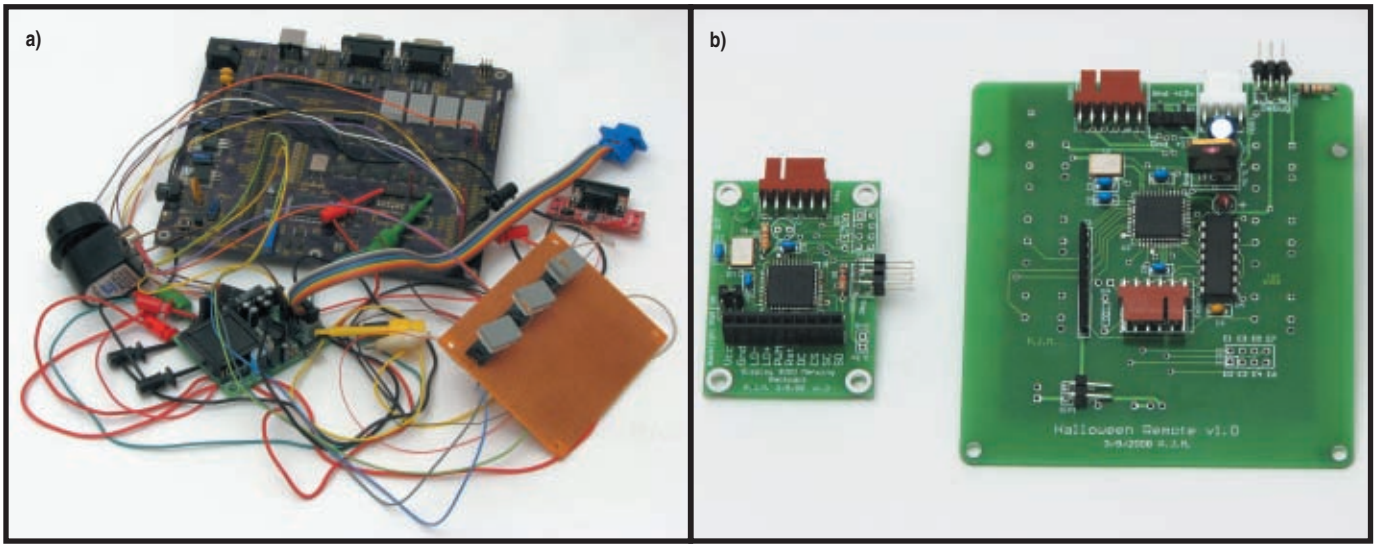


Photo 4a—This pile of parts was used to develop the remote. **b—**Sometimes it's easy to forget that a messy prototyping stage (using a Zilog development board, a custom development board, and various other bits of hacked hardware) will ultimately be reduced to a couple of clean custom PCBs.

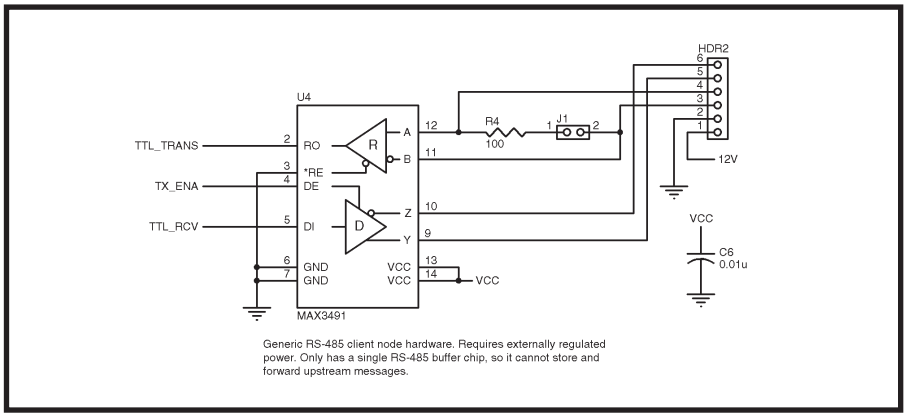


Figure 4—This is the schematic for the RS-485 circuit for the remote.

processor line. After using Google to figure out what some of the Atmel-specific macros were doing, I was able to get the simple driver up and running in a short amount of time.

The display was kind of slow due to the bit banging, but it enabled me to have a framework that I knew worked. From there, I switched it over to using the Z8 hardware SPI. When that was working, I took the SPI-specific code (software and hardware versions) and swapped them into the full driver code. I then got the software SPI version of the full driver working and finally switched it over to the hardware SPI. It was a nice, logical progression that took me from simple to full-blown over a couple of sessions of work without getting me too frustrated in the process.

Once I knew how to control the Display3000 LCD, I decided to create a serial backpack for it (see Figure 2). Rather than have the LCD attached directly to the project via the SPI lines, I wanted a separate processor to run it via high-level commands sent over a UART. A big part of my reason for doing this was that I knew the menu system of the remote was going to require a lot of RAM, and the display system for the LCD would require the same. By splitting it into two separate projects, I got the full complement of RAM for each project because each had its own Z8 running it.

Before I got the color LCD, I borrowed an old Scott Edwards Electronics serial black-and-white graphical LCD from my friend Jon Williams. I used it for early prototyping. It was a 128 × 64 pixel display that also implemented a 4 × 16 text font and ran at 9,600 bps. It didn't take long to get it up and running. I decided early on to isolate the details of the LCD commands from the overall remote project by creating an LCD abstraction library. The idea was to have the remote call the library for doing things like clearing the display, setting a cursor position, writing text, and more. The library would take care of the LCD-specific commands needed to implement numerous functions. It would enable me to swap in a different LCD by simply taking the existing library and modifying it to speak whatever native commands the new LCD required.

This abstraction layer turned out to work beautifully. Once I had the Display3000 serial backpack code working, I took the LCD abstraction library for the Scott Edwards LCD and, in literally 10 min., converted it to use the commands needed for my Display3000 LCD serial backpack.

There was more work required after that to add features (such as setting colors, something the black and white display didn't need), but it was cool to have the color LCD showing the menus for the remote in such a short amount of time.

THE MAIN BOARD

While I was waiting for the LCD to arrive from Germany, I used the Scott Edwards display and forged ahead with my main project board. A month earlier, I had created a simple demonstration PCB for the 64K series of Zilog Z8 Encore chips. (The chips are a perfect fit for most of my projects because they have a lot of RAM by embedded standards.) I used a Z8F421, which has 48 KB of flash EEPROM and 4 KB of RAM (see Photos 4a and 4b). Most similar controllers from other companies tend to have only 2 KB of RAM. Because this project involved a lot of serial communication (multiple RAM ring buffers) and a dynamic menu system (the menus were created in RAM), I needed as much RAM as possible.

It's a bonus that Zilog offers development kits at amazing prices ($50). The development kit I used comes with a serial or USB programmer/debugger, along with a full-blown IDE and C compiler. Unlike many other companies, the Zilog compiler doesn't become feature-limited after a certain number of days. It offers free upgrades, and Zilog has upgraded the IDE and compiler multiple times over the years. This is not a dead-end product.

After getting the code working and stable on my demonstration board, I created a new hardware design just for the remote. I found an extremely

Listing 1—The static array `lg_QuadDecode` is used to determine if the encoder has turned forward, reverse, or just had a glitch to be ignored. The code runs in the interrupt and uses the current bit values of the encoder along with the left-shifted values from the previous interrupt as an index into `lg_QuadDecode`.

```
static char lg_QuadDecode[16] = {0, 1, -1, 0, -1, 0, 0, 1,
                                 1, 0, 0, -1, 0, -1, 1, 0};

// The encoder is attached to bits 0 and 1 of port E
CurrQuadInput = PEIN & 0x03;
Decode Entry = CurrQuadInput | (PrevQuadInput << 2);
lg_CurrEncoderPos += lg_QuadDecode[DecodeEntry];
// Remember what the encoder reading was when we come back next time...
PrevQuadInput = CurrQuadInput;
```

cool-looking case from OKW Enclosures (www.okwenclosures.com), and I wanted a board that was custom fit to it. I decided to go for broke and use the surface-mount version of the Z8, even though it also comes in PLCC and DIP packages. The Z8 runs on 3.3 V, so there is a simple 3.3-V linear regulator, along with the Z8F421, an 18.432-MHz oscillator, a Maxim Integrated Products MAX3491 RS-485 transceiver, various I/O connectors, and the push buttons for the interface. The main schematic for the remote is shown in Figure 3. The RS-485 circuit for the remote is shown in Figure 4.

Like most of the boards I lay out, I put in some "future proofing." Even though I needed only a single column of three buttons, I put a matching column on the other side of the PCB in case I want more buttons. I also added a 2 × 8 0.100″ center header that carries the I/O pins for port E. It takes a little extra time to route these devices on the PCB, but they come in extremely handy after the PCB is fabricated and you want to add a new feature.

One element of this PCB made it slightly tricky. I checked and found out that there wasn't a lot of room between the PCB and the top of the OKW case. While there was plenty of space for the switches and ICs, things like jumpers or bulky connectors could get dicey. So, I decided to lay out the board with the switches and power LED on top and all of the other components on the bottom. Routing got a little weird. I had to keep reminding myself that even though two package outlines crossed each other, it wasn't a problem because they were on opposite sides of the board.

I/O CODE

I wrote my own serial library to run on the Z8. It understood how to send and receive the packets used in my Halloween network long ago. With some basic modification, I also created a text-specific serial library. The first bit of truly new code was written to support the scroll wheel. I perform quadrature decoding of the encoder using two I/O lines and a hardware timer running at 30 kHz. Whenever the timer generated an interrupt, the interrupt handler reads the state of the encoder, merges it with the state of the encoder from the previous interrupt shifted to the left two bit positions, and feeds that into a look-up table. The look-up table and code to use it look like Listing 1.

The look-up table and the timer effectively oversample the encoder output. They also determine not only which way the encoder is turning, but also if the transition from the previous state to the current state is legal or bogus. This eliminates any problems that occur if the encoder stops with the disk partially covering the optical sensor. This can create a condition where the clock toggles high/low, making it seem as though the encoder is moving when it's simply the electronics being unable to settle on a logic state. Similar problems occur with contact bounce on a mechanical encoder. Any time this happens, the look-up table will simply return zero. Thus, the encoder position variable doesn't show a change in position. A nice plus is that every clock edge generates a pulse, multiplying the resolution of the encoder four times.

After that was working, I wrote the code to handle the debouncing of the three button inputs. With the input side of the hardware and software working, I started tackling the remote code—specifically, the menu system needed for this project. I did some research on the Internet and in *Circuit Cellar*. Aubrey Kagan wrote an excellent article about implementing a menu system for projects using embedded processors and LCDs ("Hierarchical Menus in Embedded Systems," *Circuit Cellar* 160, 2003). His system was designed to be a general-purpose menu system where the menus were hard-coded into the source code. While I needed some menus embedded in the source, I knew most of my menus would be generated on the fly from information sent by the PC to the remote. I gleaned a number of useful points from his article. But ultimately, I implemented a different system written from scratch that had different goals in mind.

Finally, the foundation for this project is finished. My hardware is up and running, and I can read the scroll wheel and buttons. I also have a basic strategy for my menu code. Next month, we'll walk through the land of software. I will tie everything together and explain how the menu code actually works. I'll also describe the design decisions I made while I was trying to create a flexible and dynamic menu system.

Author's note: For videos and additional photos of the show in action, go to my web site: www.socalhalloween.com.

Peter Montgomery (servoanimation@socalhalloween.com) spent 12 years working as a visual effects supervisor on films such as Mortal Kombat *and* Ace Ventura: When Nature Calls *before becoming a director. He has directed dozens of commercials and made the transition to episodic television with The Disney Channel's* Lizzie McGuire. *Self-taught in both programming and digital hardware design, Peter uses his super powers only for good, not evil.*

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2008/218.

SOURCES

Delphi IDE
CodeGear
www.codegear.com

D012 Display module
Display3000
www.display3000.com/html/english.html

Custom panel
Front Panel Express
www.frontpanelexpress.com

Z8F4821 Microcontroller
Zilog, Inc.
www.zilog.com

Monitoring Applications Just Got Easier



Smarter Embedded Design

The RCM4300 RabbitCore® module gives you more processing power and memory for your demanding embedded applications that require web interfaces, graphics or encryption. Data logging applications will also take advantage of the miniSD™ memory cards with up to 1GB of storage. It's easy to add the RCM4300 – or any RabbitCore module – to your pin-compatible design.



From **$80** qty. 1000

Superior Performance and Mass Storage for:

- Web interfaces and graphics
- Data logging
- Security and encryption

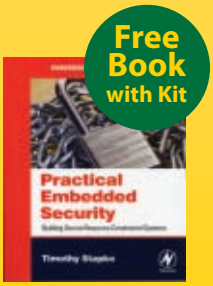
RCM4300 RabbitCore®

- **Rabbit® 4000 @ 58.98 MHz**
- **10/100Base-T Ethernet**
- **Up to 1GB of miniSD™**
- **8 channel analog input, 12-bit**

Get Your Development Kit Today for $259

Development kits include everything to start your development, Dynamic C® software, RCM4300 RabbitCore module, 512MB miniSD™ card, development board, programming cables and accessories.

www.rabbitkits.com



RABBIT™

2900 Spafford Street, Davis, CA 95618 Tel **530.757.8400**

08081

Fast Digital Filtering

Digital filtering is never fast enough. As Bruce explains, optimizing filtering on 8-bit microcontrollers, with no floating-point hardware, using 16-bit fixed-point arithmetic with 8 bits of integer and 8 bits of fraction (8:8 notation) is a good start.

Digital filtering is like auto racing: No matter how fast you can go, you would like to go faster. This observation suggests that optimization of filtering algorithms for speed of execution is always in good taste. For 8-bit microcontrollers, with no floating-point hardware, optimization of filtering requires trade-offs between accuracy, ease of programming, and speed.

For ease of use and design, floating-point arithmetic is the best. But when using an Atmel ATmega32 microcontroller and CodeVision C, a floating-point multiply takes around 400 cycles and a floating add takes 120 cycles. To analyze different techniques, compare how many two-pole infinite impulse response (IIR) filters you can run in real time at an 8-kHz sample rate. An 8-kHz sample rate translates to 2,000 cycles between samples at a maximum CPU clock rate of 16 MHz. As I will explain, a two-pole IIR filter requires five multiplies and four adds per sample for a total of 2,480 floating-point cycles. So, at an 8-kHz sample rate, you cannot even run one filter in real time using floating-point arithmetic.

Going to the other extreme of simple 8-bit integer arithmetic, a 1-byte 8-bit × 8-bit multiply is performed in hardware in two cycles but takes a total of eight cycles to fetch and store. An 8-bit add is about the same speed. We could run about 40 two-pole filters in real time, but an 8-bit format does not have the dynamic range necessary to build accurate filters. As you shall see, building accurate filters means subtracting two numbers, which are almost equal, so 8-bit numbers suffer because so few bits are left after the subtraction.

A compromise is to use 16-bit fixed-point arithmetic with 8 bits of integer and 8 bits of fraction (8:8 notation). As you shall see, a 16-bit fixed multiply and add (called a MAC operation) can be done in 24 cycles, including loading parameters. By the time registers are saved and restored in a C function call, the two-pole IIR calculation takes 182 cycles. You can therefore perform about 11 filter operations per sample time! Its accuracy is good enough for filters up to about four poles, but some care is needed. Narrow bandwidth may cause errors, the filter coefficients may need to be scaled, and actual filter responses should always be measured. I chose 8:8 notation in spite of its accuracy limitations for three reasons: One, it is fast to compute. Two, we use 8:8 notation for other applications where a larger integer dynamic range is handy, specifically for video games. The integer part of the notation maps nicely into video pixels, while the 8-bit fraction is accurate enough to represent fractional pixel-change speeds. Third, the Matlab design tools that the students are used to using produce filter coefficients of up to ±7 or so for some frequency ranges; but for smaller coefficients, it is easy to scale the Matlab-generated filter coefficients by 16 to increase fixed-point accuracy without changing the filter frequency response. Scaling requires just one small change to the assembler filter code.

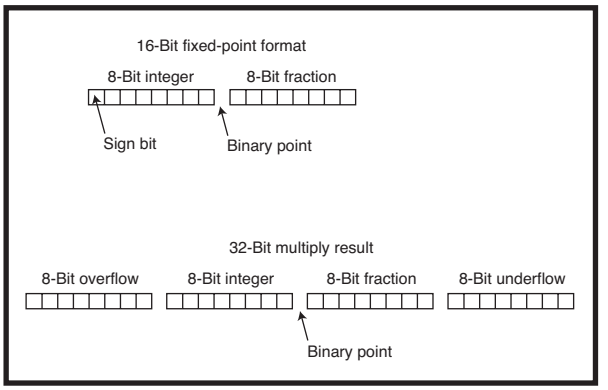


Figure 1—This is the fixed-point arithmetic scheme used for filtering. There are 8 bits of integer and 8 bits of fraction (8:8 notation). The format is signed, 2's complement, so that the assembler add/adc instructions can be used directly.

Students in my ECE 476 microcontroller class at Cornell University have used these digital filters for several applications over the last two years. (See the

We add value to PCBs
when others just sell them.

- One Stop Manufacturing Service
- Free Electronic Components
- Free Prototyping Assembly
- Professional Consultant



Blank PCBs don't work

We make working PCBs

WWW. EZPCB. COM

PCB+FPC+Rigid Flex PCB+Assembly+Keypads+Enclosures+Components+Cables+Metal Fab+Consultant

www.EzPCB.com
Email: sales@ezpcb.com

LUMINARY MICRO®

Connected. Versatile. Cost-effective.

Stellaris means:

- The most advanced integrated MCUs with high-performance Ethernet, USB, CAN and serial communications
- Complete open-tooled, ready-to-implement reference design kits for connected applications
- Cost-effective modules for rapid time to market
- Advanced interrupt-driven software with Stellaris Driver Library support
- Entry into "$1 to 1GHz" ARM architecture compatibility





RDK-IDM
Intelligent Display Module

Featuring color touch QVGA display with Ethernet and Power-over-Ethernet or direct power input

$ 219.00

RDK-S2E
Serial-to-Ethernet Module

Featuring two serial UARTs to 10/100 Ethernet, with software

$ 139.00

www.luminarymicro.com

LUMINARY MICRO®

STELLARIS® microcontrollers

FAIRCHILD SEMICONDUCTOR the power franchise Cortex ARM

Listing 1—The heart of any digital filter is the multiply-and-accumulate (MAC) operation. This assembler macro does an 8:8 MAC in 16 machine cycles, while maintaining a 24-bit intermediate result.

```
.macro mult_acc        ;r31:r30:r24 += r23:r22 * r21:r20 += p1*p2
    muls   r23, r21    ; (signed)p1-high * (signed)p2-high
    add    r31, r0
    mul    r22, r20    ; p1-low * p2-low
    add    r24, r0
    adc    r30, r1
    adc    r31, r27
    mulsu  r23, r20    ; (signed) p1-high * p2-low
    add    r30, r0
    adc    r31, r1
    mulsu  r21, r22    ; (signed) p2-high * p1-low
    add    r30, r0
    adc    r31, r1
.endm
```

Resources section.) Examples include a music-controlled stepper-motor-driven Marionette, a MIDI synthesizer, a guitar tuner/trainer, a music-controlled solenoid-driven eight-channel water fountain, a voice-command car (which worked best in Hindi), and a voice-command lock.

DIGITAL FILTERS AT A HIGH LEVEL

At a high level, digital filtering attempts to use the current input sample, past inputs, and past filter outputs to select certain frequencies of the input signal. You will be restricted to a subset of digital filters, which are referred to as linear, time-invariant filters. With this restriction, you can use powerful design tools available on the Internet or in languages such as Matlab to build filters. (See the Resources section.) Any linear, time-invariant filter can be written as:

$$a(1) \times y(n) = b(1) \times x(n) + b(2) \times x(n-1) + \ldots + b(nb+1) \times x(n-nb) - a(2) \times y(n-1) - \ldots - a(na+1) \times y(n-na)$$

where y(n) is the newly computed output at time n, x(n) is the current input, x(n – 1) is the previous input, y(n – 1) is the previous output, with other past samples needed up to the number of b(nb) and a(na) coefficients used. The coefficient a(1) is generally set to unity in the design software. This form is referred to as "Direct Form II Transposed." The Matlab filter command uses this form. Determining which values for the as and bs to use has been the object of person-centuries of research, all of which is available in any digital filtering text, from Matlab help or from the Internet. To scale the filter coefficients for better accuracy, set a(1) to a power of two (say 16) and multiply all of the other coefficients by 16 also. This operation enables you to specify four more bits in the coefficients and adds only two cycles per bit of shift to the execution time of the filter.

For a two-pole filter, the equation reduces to:

$$a(1) \times y(n) = b(1) \times x(n) + b(2) \times x(n-1) + b(3) \times x(n-2) - a(2) \times y(n-1) - a(3) \times y(n-2)$$

which has five multiplies and four adds, as mentioned in the introduction. This formula requires that at any time, the last two past filter outputs and last two past inputs need to be available. If we used the scaled form with a(1) = 16, then we will need to divide the new output value by 16, which is just four arithmetic-shift-right operations.

FIXED-POINT ARITHMETIC

Before we talk about the details of the filters, we need to talk about fixed-point arithmetic. Figure 1 shows the bit layout for the number format we use. The 16-bit number is stored in a normal, signed, 2s-complement integer variable, but the bits are interpreted (from left to right) as a sign bit, 7 bits of integer (–128 to 127), and 8 bits of fraction. The left-most digit of the fraction has a weight of 0.5 while the right-most digit has a weight of 0.00391 = 1/256. Therefore, you can represent any number in the range of –128 to 127 as a resolution of 0.00391.

Adding two of these numbers is easy. The 2s complement add/subtract hardware simply adds them in two 8-bit pieces using an add followed by an add-with-carry instruction. Multiplication requires a little more care. Doing a 16-bit × 16-bit multiply results in a 32-bit result (see Figure 1). But the output value of the fixed multiply must be another 16-bit fixed-point number. The top byte of the 32-bit result is treated as an overflow and discarded. But where is the binary point in the result? Because there are two 8-bit fractions, there must be 16 bits of fraction in the output of the multiply. The lowest 8 bits is underflow. Therefore, the desired 16-bit result is the middle 2 bytes of the 32-bit result. The process can be summarized as a C language macro where x and y are 16-bit integers:

```
#define multfix(x,y)((int)((((long)(x))*((long)(y)))>>8))
```

This macro does the multiply. Although it is faster than a floating multiply, it is slow compared to an assembly version of the algorithm for a couple of reasons. The first is that we never have to actually compute the top byte of the 32-bit result. The second is that the 32-bit shift right can be replaced in assembler by simply copying the middle 2 bytes to the desired destination. Because the operation we actually want for digital filtering is a multiply followed by a running sum of five terms, we can economize further by combining the two operations into a MAC.

The MAC operation, which I wrote in assembler, was based on an Atmel application note titled "Using the Hardware Multiplier" (AVR201). Listing 1 shows the details. The AVR multiply opcodes operate only on a few registers,

Filter	Execution time
Two-pole general IIR	182 Cycles
Two-pole Butterworth low/high-pass	148 Cycles
Two-pole Butterworth band-pass	140 Cycles
Four-pole Butterworth band-pass	228 Cycles

Table 1—The execution times for the various filters implemented are displayed. Times are in machine cycles. At a 16-MHz clock rate, the general two-pole IIR takes 11.3 µs and the four-pole Butterworth band-pass takes 14.3 µs.

so placement of operands is pretty much fixed with one 16-bit value in registers r22 (low byte) and r23 and the other 16-bit value in r20 (low byte) and r21. The MAC operation saves the low-order result byte in r24 so underflow values may be added in through all five MAC operations needed for a single-filter operation. The 16-bit results are in registers r30 and r31. All of the multiply opcodes put the results in r0 and r1. First, a signed multiply is used to compute the most significant byte of the result, which is added to r31. Then an unsigned multiply is used to compute the product of the lowest 2 bytes. This is followed by two signed-unsigned multiplies to compute the remaining cross terms. The MAC was written as an assembler macro to minimize execution time.

With the macro written, all that had to be done was to load five sets of samples (input and output) and filter parameters (`as` and `bs`) into the appropriate registers and then save the updated sample history for the next sample.

When specifying coefficients for the filters, it is convenient to have a couple of utility macros, which convert floats to fixed point and back again. These macros would never be used in a fast loop, but to define only constants at the initialization or readout stages of a program where execution speed is not so important. `float2fix` converts a float to a fix by shifting 8 bits to the left using a multiply by 256. `fix2float` reverses this:

```
#define float2fix(a) ((int)((a)*
    256.0))
#define fix2float(a) ((float)(a)/
    256.0)
```

IMPLEMENTING THE FILTERS

I implemented a general two-pole IIR filter, as well as specialized two-pole low-pass, high-pass, and band-pass Butterworth filters. I also implemented four-pole Butterworth band-pass filters. Butterworth filters have maximally flat frequency response and reasonable roll off rates. Table 1 shows execution times for the filters implemented. Symmetry in the Butterworth design enables one or two multiplies to be combined for extra efficiency.

Listing 2—This is a complete listing of the second-order IIR filter function. Note that the function uses the MAC macro. Near the bottom of the listing are eight commented lines which implement a 16-bit divide by 16. The lines would be uncommented if you use scaled coefficients for more (12-bit) accuracy.

```
int IIR2(int xx)
// xx is the current input signal sample
// returns the current filtered output sample
begin
    #asm
    push r20                ;save parameter regs
    push r21

    clr r27                 ;permanent zero
    clr r24                 ;clear 24 bit result reg; msb to lsb => r31:r30:r24
    clr r30
    clr r31

    lds  R22, _b1           ;load b1 from RAM
    lds  R23, _b1+1
    ld   R20, Y             ;load input parameter xx from stack
    ldd  R21, Y+1
    mult_acc                ; b1*xx

    lds  R22, _b2           ;load b2 from RAM
    lds  R23, _b2+1
    lds  R20, _xn_1         ;load x(n-1) from  RAM
    lds  R21, _xn_1+1
    mult_acc                ; b2*x(n-1)

    lds  R22, _b3           ;load b3 from RAM
    lds  R23, _b3+1
    lds  R20, _xn_2         ;load x(n-2) from  RAM
    lds  R21, _xn_2+1
    mult_acc                ; b3*x(n-2)

    lds  R22, _a2           ;load -a2 from RAM
    lds  R23, _a2+1
    lds  R20, _yn_1         ;load y(n-1) from  RAM
    lds  R21, _yn_1+1
    mult_acc                ; -a2*y(n-1)

    lds  R22, _a3           ;load -a3 from RAM
    lds  R23, _a3+1
    lds  R20, _yn_2         ;load y(n-2) from  RAM
    lds  R21, _yn_2+1
    mult_acc                ; -a3*y(n-2)

    lds  R20, _xn_1         ;load x(n-1) from  RAM
    lds  R21, _xn_1+1
    sts  _xn_2, r20         ;store x(n-2) to  RAM
    sts  _xn_2+1, R21
    ld   R20, Y             ;load input parameter xx from stack
    ldd  R21, Y+1
    sts  _xn_1, r20         ;store x(n-1) to  RAM
    sts  _xn_1+1, R21
    lds  R20, _yn_1         ;load y(n-1) from  RAM
    lds  R21, _yn_1+1
    sts  _yn_2, R20         ;store y(n-2) to  RAM
    sts  _yn_2+1, R21
;to scale the filter by 16, uncomment the next 8 lines
  ;   asr  r31              ; divide by 16 for coeff prescale
  ;   ror  r30
  ;   asr  r31
  ;   ror  r30
  ;   asr  r31
  ;   ror  r30
  ;   asr  r31
  ;   ror  r30
    sts  _yn_1, r30         ;store new output as y(n-1) to  RAM
    sts  _yn_1+1, r31

    pop r21                 ;restore parameter regs
    pop r20
    #endasm
end
```

The details can be found on the fixed-point math web page noted in the Resources section of this article. About eight four-pole filters will run in real time at an 8-kHz sample rate.

Listing 2 gives the source for the general two-pole IIR filter. The Codevision C compiler enables inline assembly code, which is indicated by the `#asm` directive. The input parameter `xx` is passed on the data stack, which is pointed to by the hardware registers r28 and r29. These two registers are together referred to in the assembler as the Y register. A variable declared in C may be referenced in assembler by prepending an underscore to its name. So, if the variable `b1` was declared in C to be of type integer, `_b1` is the address at which the low byte of b1 is stored and `_b1+1` is the address of the high byte. The routine pushes a couple of registers on the hardware stack, clears the MAC accumulator, performs five load-and-MAC operations, and ends by updating the sample history and restoring the two registers. The previously defined macro is inserted five times. If coefficient scaling is used, eight lines near the end should be uncommented, and the input coefficients should be multiplied by 16 before calling this routine.

TESTING THE FILTERS

I wrote a test program to verify the frequency response of the filters. The program consists of a timer interrupt service routine (ISR) running at 8 kHz and a main routine running in the background. The ISR synthesizes a sine wave at a frequency selected by the main routine. The sine wave is produced using direct digital synthesis (DDS) with a 16-bit accumulator. (Refer to the documents listed in the Resources section of this article for more information about DDS.) At every sample, either a two-pole or four-pole filter is executed using the sine wave as input. The ISR also measures the peak amplitude of the filter output and maintains a counter so that the main routine can measure real time. The main routine initializes the sine table necessary for DDS, the timer ISR, and the UART. It then steps through a list of test frequencies. For each frequency, main computes the DDS increment, then waits a few tenths of a second for transients to settle, reads the maximum amplitude measured by the ISR, and prints the maximum. The programs are available through a link noted in the Resources section of this article.

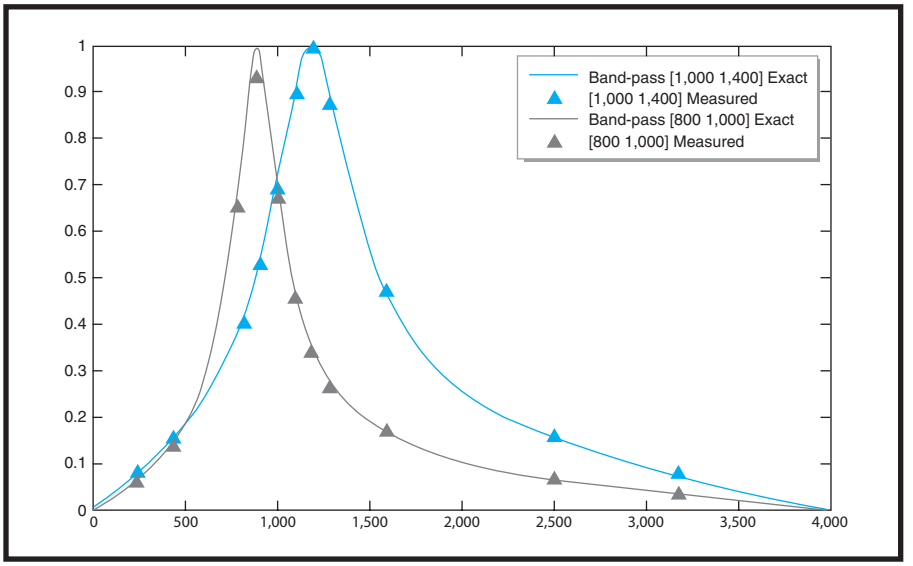


Figure 2—This is the frequency response of two two-pole filters showing a small inaccuracy for narrower bandwidth filters due to 8:8 round-off errors. Coefficient scaling can reduce errors.

Because of the limited accuracy of a 16-bit fixed-point system (relative to the floating design tools), it is important to characterize the limitations on filter performance. Figure 2 shows the results of testing two two-pole Butterworth band-pass filters. With a bandwidth of 400 Hz (0.1 of the Nyquist frequency), the filter computed by Matlab on the PC matches the response measured on the ATmega32 closely. However, with a bandwidth of 200 Hz, there is a 6% gain error (–0.5 db) because some of the coefficients are so small that few bits are left in the terms multiplied by b1. Scaling the coefficients by 16 and using the scaled filter code reduces error to less than 1% for the narrower bandwidth. The filter test code may be found on the *Circuit Cellar* FTP site.

Figure 3 shows the results of testing

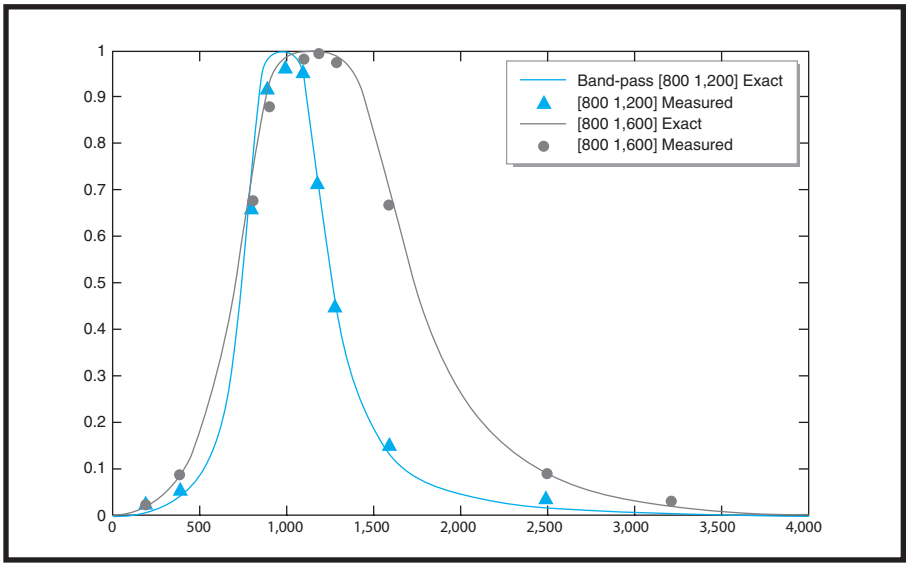


Figure 3—These are the results of testing two four-pole Butterworth band-pass filters. The filters exhibit sharper cutoffs but more gain errors. With a bandwidth of 800 Hz (0.2 of Nyquist frequency), the filter computed by Matlab matches the response measured on the ATmega32 within 2% near the peak. At a bandwidth of 400 Hz, there is an 8% gain error (0.7 db), still small enough for most applications.

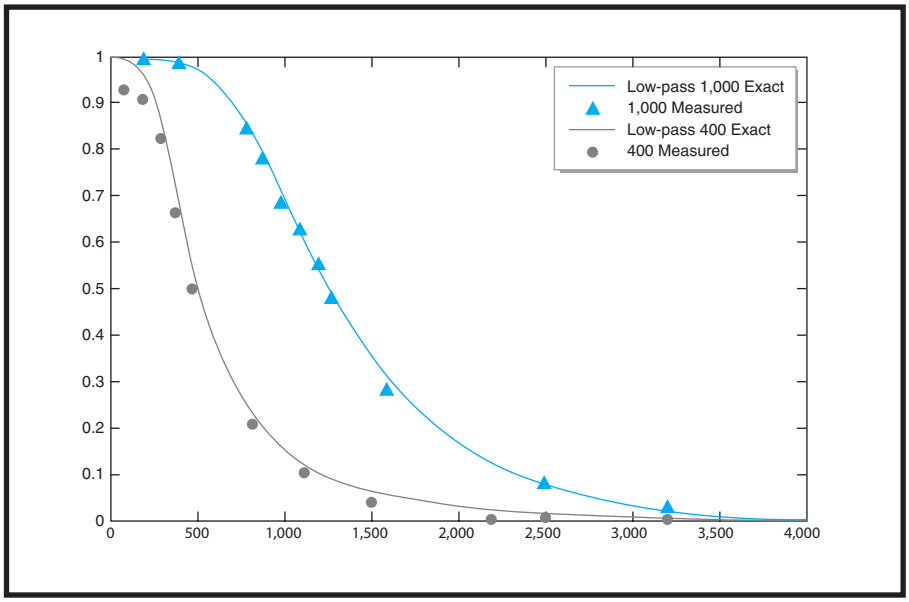


Figure 4—These are the results of testing two two-pole Butterworth low-pass filters. With a bandwidth of 1,000 Hz (0.25 of the Nyquist frequency), the filter computed by Matlab on the PC matches the response measured on the ATmega32 closely. At a bandwidth of 400 Hz, there is a 7% gain error (–0.6 db) at low frequencies. Scaling the coefficients by 16 reduces the error to less than 1%.

two four-pole Butterworth band-pass filters. The filters exhibit sharper cutoffs, but more gain errors. With a bandwidth of 800 Hz (0.2 of Nyquist frequency), the filter computed by Matlab matches the response measured on the ATmega32 within 2% near the peak. However, with a bandwidth of 400 Hz, there is as much as an 8% gain error (0.7 db) at some frequencies and 4% near the peak. Scaling the coefficients by 16 reduces the error to less than 1% for the narrower bandwidth.

Figure 4 shows the results of testing two two-pole Butterworth low-pass filters. With a bandwidth of 1,000 Hz (0.25 of the Nyquist frequency), the filter computed by Matlab on the PC matches the response measured on the ATmega32 closely. At a bandwidth of 400 Hz, there is a 7% gain error (–0.6 db) at low frequencies. Scaling the coefficients by 16 reduces the error to less than 1%.

The filters have been used successfully in several of my students' projects (see the Resources section). The filters execute quickly enough to allow students to use 10 general two-pole IIR filters or eight four-pole Butterworth band-pass filters (at a sample rate of 8 kHz). This is enough to perform some elementary voice processing and identify features in music.

Even the 16 bits used in fixed-point format with 8 bits of fraction do not provide enough accuracy for narrow bandwidth filters. As a rule of thumb, if you do not scale the filter coefficients, you should limit the bandwidth of two-pole filters to greater than 200 Hz (at an 8-kHz sample rate) and four-pole filters to greater than 400 Hz, in order to maintain less than 1-db gain errors. By scaling filter coefficients, you can make narrower filters. Scaling by 16 allows band-pass filters as narrow as 250 Hz at an 8,000-Hz sampling rate with 1-db error for four-pole Butterworth filters. With scaled filters, dynamic range may suffer for larger valued inputs. Using only inputs with values less than one (8-bit accuracy) minimizes the chance of overflow. Narrower bandwidth filters should always be tested for adequate filter performance and for numerical overflow.

Author's note: For more information about designing digital filters, refer to the "IIR Digital Filter design applet," www.dsptutor.freeuk.com/IIRFilterDesign/IIRFilterDesign.html. Note that the definitions of as and bs are reversed and that the indexes are zero-based rather than one-based. This applet assumes a sample frequency of 8,000 Hz, but you can scale to other sample rates by making the normalized cutoff frequency the same:

(cutoff frequency)/(Nyquist frequency) = (cutoff frequency)/4,000

Bruce Land (brl4@cornell.edu) is a senior research associate in both Neurobiology and Behavior and Electrical and Computer Engineering at Cornell University. He teaches two courses in Neurobiology and Behavior and two in Electrical and Computer Engineering. Bruce also provides general research support in electronics design and computer techniques.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2008/218.

RESOURCES

Atmel Corp., "AVR201: Using the Hardware Multiplier," 2002, www.atmel.com/dyn/resources/prod_documents/DOC1631.PDF.

Cornell University, ECE476 Course and Student Projects, http://instruct1.cit.cornell.edu/courses/ee476/FinalProjects/ and http://instruct1.cit.cornell.edu/courses/ee476/.

R. Lacoste, "DDS-GEN: Part 1: A High-Performance DDS Generator," *Circuit Cellar* 129, 2001.

B. Land, "Electrical Engineering 476: Fixed Point mathematical functions in Codevision C and assembler," Cornell University, 2007, http://instruct1.cit.cornell.edu/courses/ee476/Math/index.html.

M. Pechanec, "Direct Digital Synthesis: DDS theory and simulator," www.geocities.com/CapeCanaveral/5611/dds.html.

L. E. Turner, "Finite Precision Effects in Digital Filters," 1999, www2.enel.ucalgary.ca/People/Turner/fpeffects/fpeffects.html.

SOURCE

ATmega32 Microcontroller
Atmel Corp.
www.atmel.com

RENESAS M16C Platform

M16C - Protecting code investment with true code & pin compatibility

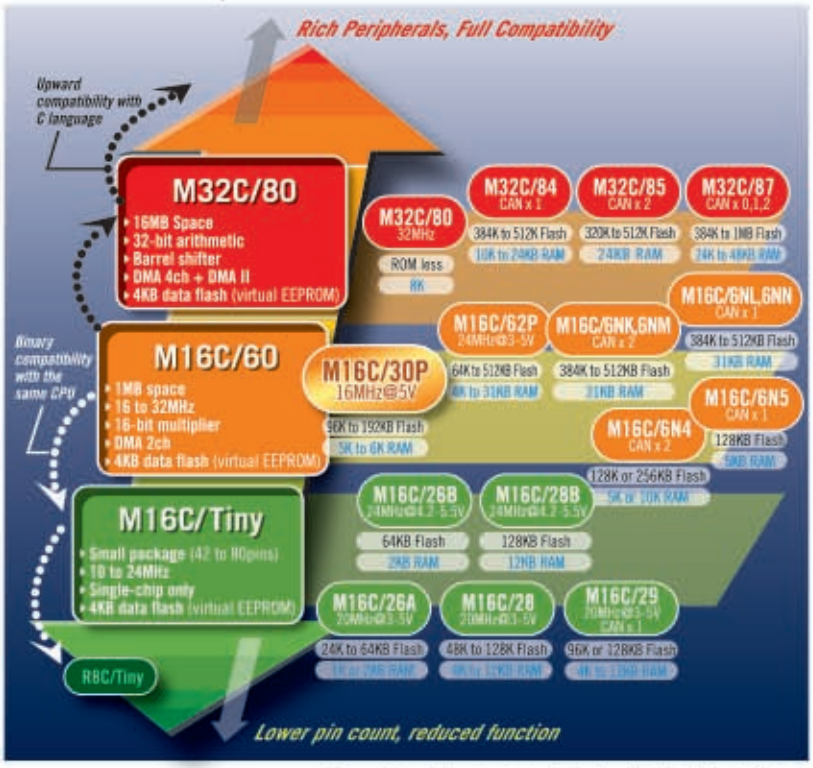
Enables connectivity with CAN, LIN, PLC, ZigBee and DMAC

Renesas Technology

No. 1* supplier of microcontrollers in the world

presents the M16C Family, the industry's only fully code compatible platform that addresses the entire 8-bit through 32-bit price/performance application space. The M16C Family of devices protects your code investment with seamless code and pin compatibility across the platform. The M16C platform features optimized peripheral mapping and on-chip memory that permits applications ranging from 24K bytes to 1M bytes, with between 42 and 144 pins, in an efficient 16-bit CISC architecture. Reduce your overall development time and cost with a cross-platform code base and development tools from Renesas Technology.

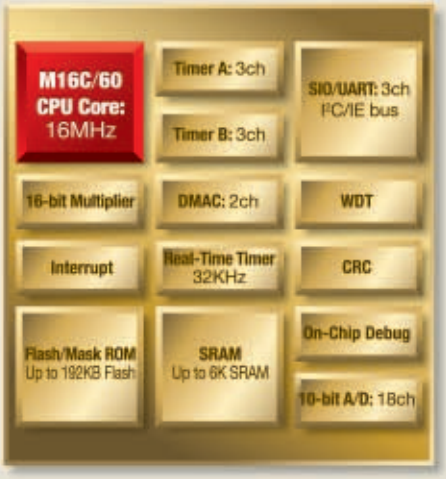
M16C Product Lineup



HOT Products	M16C/30P Group
	3 cents per pin*

* M30302FCPGP#U5C: 100 pin, LQFP, 128K Flash / 5K RAM

M16C/30P Block Diagram



Top Reasons to select M16C MCU

- **Broad Platform & Compatibility**
  - Wide selection: 24KB to 1MB Flash, 42 to 144 pin
  - Up/downgrade in same package, same board design
  - Saves costs and protects code investment
- **Powerful and Easy**
  - Banked registers & high speed interrupts (M32C)
  - Optimized instructions for 1cycle operations
  - High-speed hardware multiplier
  - Common Tool Chain across the platform
  - Proven middleware: ZigBee, CAN, motor control
- **Versatile & Efficient**
  - Specialized peripherals: CAN, PLC, motor control
  - Efficient mapping for practical application
  - Flexible DMAC for efficient connectivity
  - Multiple clock sources and power saving modes
- **Quiet EMI/EMS**
  - Built-in noise cancellation, fewer components
- **Reliable and Secured**
  - Trusted flash and protection registers
  - Fail-safe features: oscillator-stop, non-stop WDT

*Source: Gartner "Semiconductor Applications Worldwide Annual Market Share: Database" Hiroyuki Shimizu, 27 March 2008, GJ08218 * This is 2007 ranking

#1 MCU REACH FURTHER

Get Started Today -

Go online and register to be eligible for a FREE M16C Starter Kit *

www.america.renesas.com/ReachM16C/d

*Only available to customers in North and South America



DEVCON Developers Conference Oct. 13-15 San Diego www.RenesasDevCon.com

Renesas Technology Corp.

Everywhere you imagine. RENESAS

MCU-Based Flight Data Acquisition

Real-Time Vibration Measurement for Aircraft

Excessive rotor vibration can endanger a helicopter and its crew. That's why Juan developed a system that measures a helicopter's tail rotor vibration in real time. The inexpensive design determines the maximum vibration amplitude in a Bell 206 JetRanger helicopter's tail rotor. An accelerometer and an optic sensor in the tail rotor collect data, which is later processed by a Cypress Semiconductor CY8C27443 microcontroller and displayed on a Hitachi LCD.

Helicopter pilots and technicians expect their aircraft parts to vibrate. However, excessive rotor vibration is dangerous and requires attention. Checking helicopters for excessive vibration is costly in terms of man hours, so I developed a system that can measure a helicopter's tail rotor vibration in real time (see Photo 1 and Figure 1).

The inexpensive design I'll describe in this article is used to determine the maximum vibration amplitude in a Bell 206 JetRanger helicopter's tail rotor. A Wilcoxon Research 991D internally amplified accelerometer and a Banner Engineering optic sensor in the tail rotor collect the data, which is processed by a Cypress Semiconductor CY8C27443 microcontroller and displayed on a Hitachi HD44780 LCD (see Photo 2 and Figure 2).

Note that the parameters, filter configurations, and internal components that I will describe were calculated and adjusted in relation to the turning speed of a 206 JetRanger helicopter's tail rotor. You must adjust the parameters and component configurations for other aircraft.

VIBRATION 101

Vibration affects helicopters because of the large number of rotating components that are involved. If vibration is not controlled, the increased stresses on components throughout the aircraft will shorten their useful lifespans. Vibration also interferes with communication equipment and turbines. Therefore, vibration must be maintained at low levels to avoid catastrophic consequences.

In order to fly, a helicopter needs all of its turning components to be balanced (in a static and dynamic manner). Static balancing of the main and tail rotors is accomplished during periodic maintenance by making sure that the center of gravity of each is aligned with its rotation

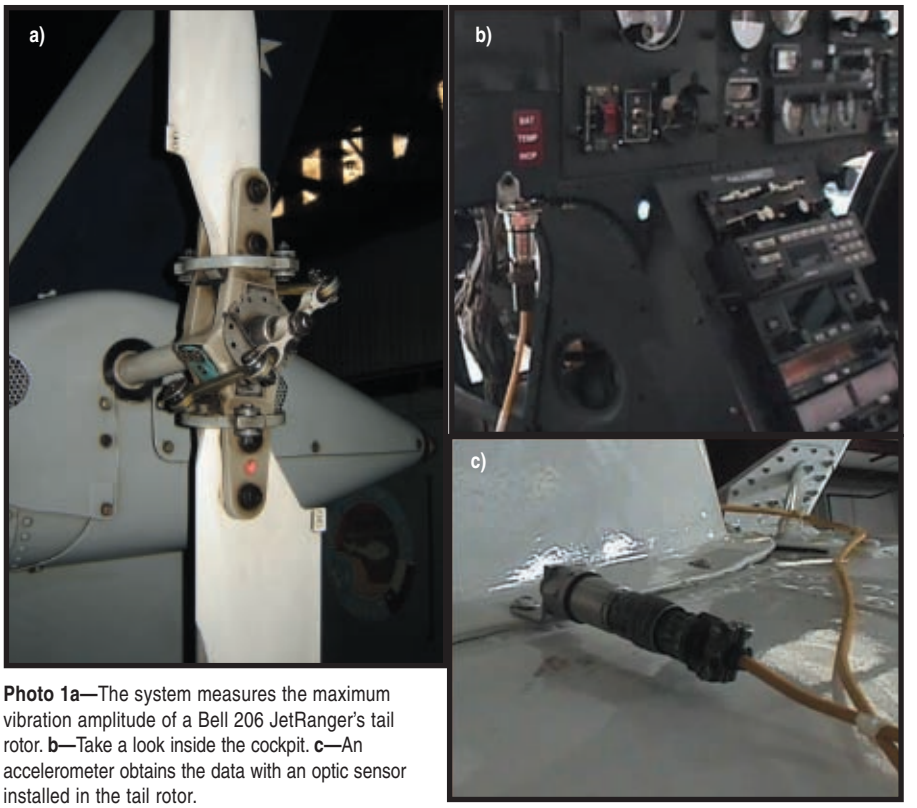


Photo 1a—The system measures the maximum vibration amplitude of a Bell 206 JetRanger's tail rotor. **b**—Take a look inside the cockpit. **c**—An accelerometer obtains the data with an optic sensor installed in the tail rotor.

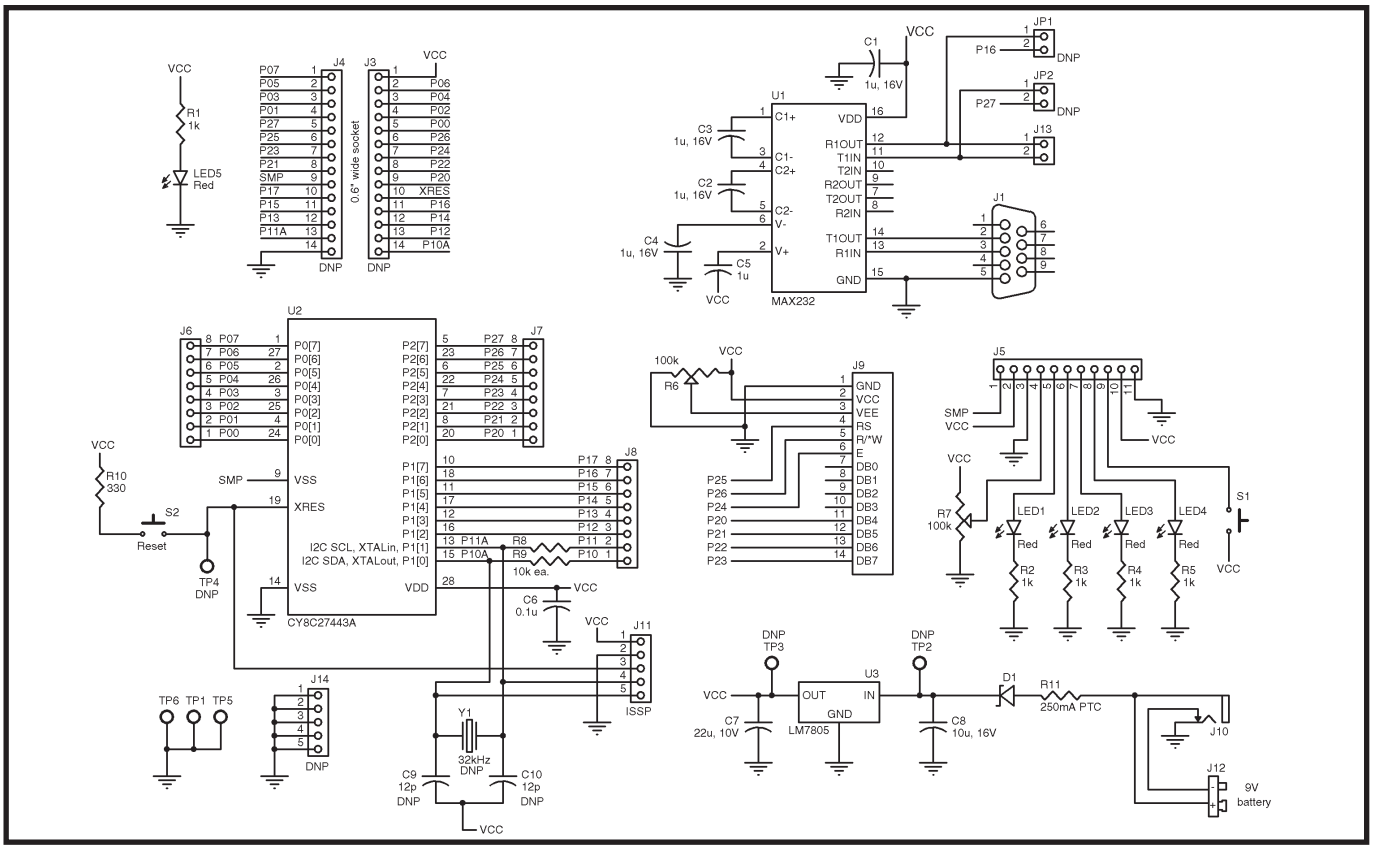


Figure 1—I used a Cypress Semiconductor CY3210-MiniProg1 kit. The CY8C27443 microcontroller is the heart of the design.

axis. Dynamic balance is obtained after analysis. This application is important because any small change in a component can alter the dynamic balance, thus causing vibration throughout the aircraft.

ISO 2041 defines vibration as "all vibrations in time or in magnitude that describe the movement or position of a mechanic system, when this magnitude is alternately higher or lower than an average value."[1] Harmonic vibration is the simplest form of vibration characterized by a sinusoid. It is generated in linear systems because of the presence of potential problems such as unbalance. This movement is analyzed in terms of a rotating vector with a constant angular speed (ω) through which the oscillation frequency (f) is in hertz that is different from the angular frequency expressed in 1/s. All of this leads to the mathematical representation of this phenomenon:

$$y = Y\sin(\omega t + \varphi) = Y\sin(2\pi f t + \varphi) \quad [1]$$

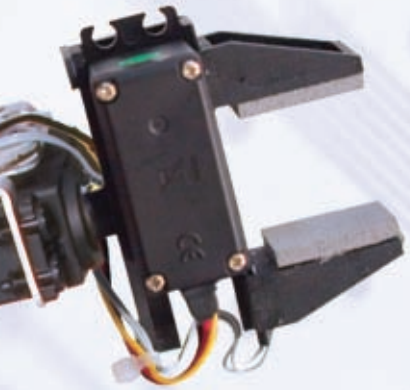
where:

$$\omega = \frac{2\pi}{T} \text{ and } f = \frac{\omega}{2\pi} = \frac{1}{T} \quad [2]$$

φ is the vibration phase.

Vibrations can be analyzed in two





ROBO DEVELOPMENT CONFERENCE & EXPO '08

NOV. 18-19, 2008

SANTA CLARA CONVENTION CENTER

SANTA CLARA, CA

Join the International Technical Design and Development Event for the Personal, Service, & Mobile Robotics Industry

EXCLUSIVE OFFER:

USE PRIORITY CODE RDCC AND

SAVE $300 *ON A CONFERENCE PASS*

WWW.ROBODEVELOPMENT.COM ■ 800-305-0634

Featured Keynote Presentations



Sebastian Thrun
Professor of Computer Science and Electrical Engineering
Director, Artificial Intelligence Laboratory
Stanford University
"Making Cars Drive Themselves"

Plus...

Maja J. Matarić
Founding Director
USC Center for Robotics and Embedded Systems
Director
USC Robotics Research Lab

Jeanne Dietsch
CEO
MobileRobots Inc
"Opportunities at the Uptick"

- 30+ Cutting-edge, relevant conference sessions covering:
  - Systems and Systems Engineering
  - Tools and Platforms
  - Enabling Technology
  - Achieving Autonomy
  - Design and Development
- Keynote sessions delivered by internationally recognized industry leaders
- Exposition floor featuring what's new and what's next in robotics design and development
- Valuable networking opportunities that put you in touch with peers, industry experts and up-and-coming talent

FOR COMPLETE EVENT DETAILS VISIT *WWW.ROBODEVELOPMENT.COM*

JOIN THESE LEADING COMPANIES AT 2008 ROBODEVELOPMENT

Founding Sponsor

Microsoft

Gold Sponsor

MOBILEROBOTS

Silver Sponsors

ADEPT Robotics

ADVANCED MOTION CONTROLS

AMOTION

BOSTON ENGINEERING™

Elmo Motion Control

ENERGID

NATIONAL INSTRUMENTS

Analyst, Association & Academic Co-Sponsors

OMG OBJECT MANAGEMENT GROUP

Media Co-Sponsors

CIRCUIT CELLAR

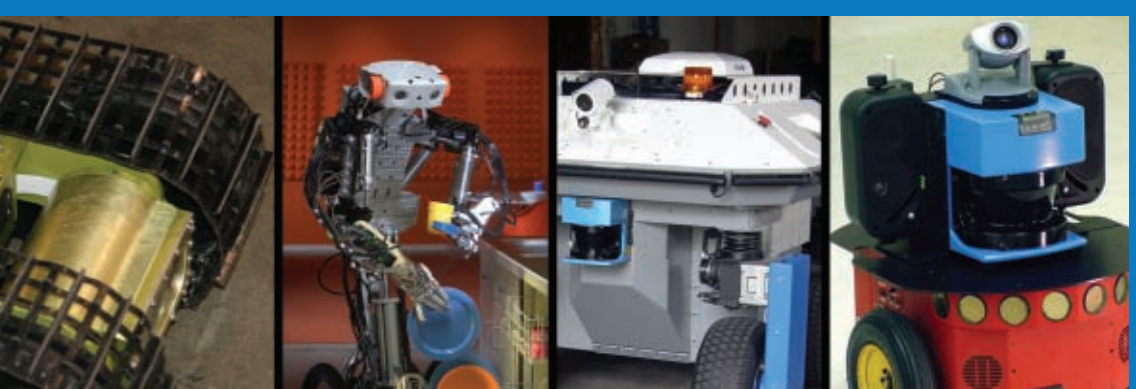
LINUX JOURNAL

Entertainment Engineering

NUTS AND VOLTS

SERVO

Listing as of July 17, 2008. For a current list of participating companies, please visit *www.robodevelopment.com.*



For Information on ***Sponsorship and Exhibiting Opportunities,*** *contact Ellen Cotton at ecotton@ehpub.com or 508-663-1500 x240*

basic domains: the time domain and the frequency domain. These have a direct bond. Electronically speaking, vibration is registered as an electronic signal proportional to the physical phenomenon being quantified. This is recorded fundamentally in the time domain. To obtain the spectrum of the vibrations, pass the signal through a filter that separates each one of the vibration components by frequency. When measuring the vibration level, define what physical magnitude you want to use to describe the vibration, such as movement, speed, or acceleration.

ISO 2041 introduces a few definitions regarding the magnitudes. Movement is a vector quantity that describes the change of position of a body or particle regarding a reference system. Velocity is a vector that specifies the derivative of the movement in time. Acceleration is a vector that specifies the derivative of speed in time. These magnitudes are expressed in the following way:

$$\begin{aligned} y(t) &= Y\sin(2\pi ft) \\ v(t) &= (2\pi f)Y\sin\left(2\pi ft + \frac{\pi}{2}\right) \\ a(t) &= (2\pi f)^2 Y\sin(2\pi ft + \pi) \end{aligned} \quad [3]$$

SYSTEM OVERVIEW

A basic vibration measurement system includes sensors for capturing the signal, a preamplified stage, a filter, and a microprocessor (see Figure 3). At least two sensors are required to build this prototype. I used vibration-measuring sensors (accelerometers) and speed-measuring sensors (optical types).

A Banner Engineering SM312LVMHS optic sensor (MINI-BEAM model) with a visible red sensing beam (650 nm) measures revolutions. (A photo of the sensor is posted on the *Circuit Cellar* FTP site.) The sensor measures the total revolutions of the tail rotor by means of a light beam projected onto the surface of an adhesive reflective tape (3M 7610). This tape has a strong reflection, a light weight, and a low profile. Plus, it neither seriously affects the rotor's balance nor dislodges in flight.

The sensor is mounted in the tail rotor at a maximum distance of 1 m. The gain can be adjusted so that it works up to 5 m, depending on the reflecting material and the supply voltage. The mechanism must be aligned perpendicular to the reflector. After installation, the gain is adjusted by means of a 15-turn screw until the LED indicator located at the base is illuminated (see Figure 4).

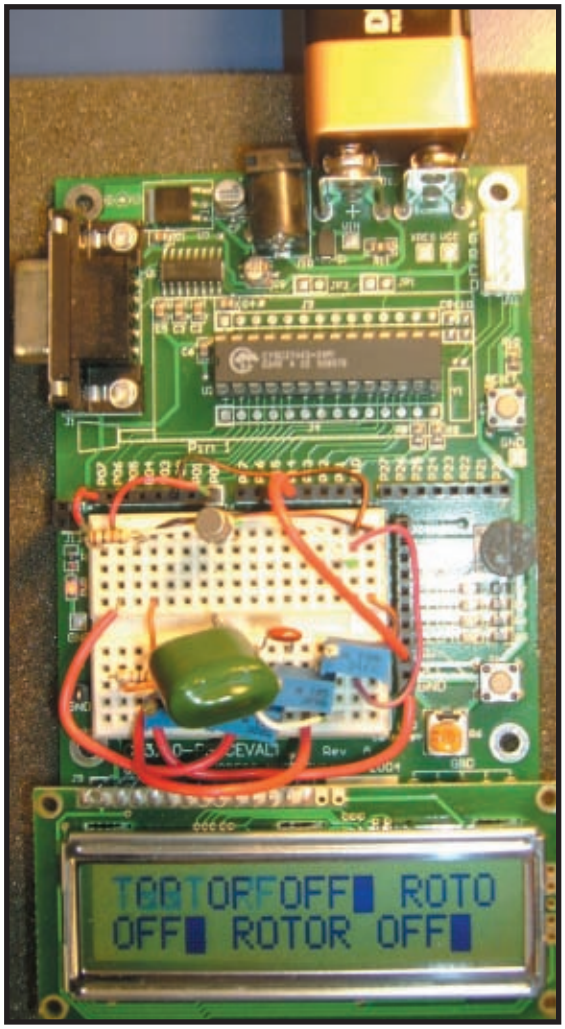


Photo 2—This is the circuit mounted in the breadboard (according to AN2320) for amplified signal input (accelerometer).

An accelerometer measures the magnitude of the vibration amplitude. It is also synchronized with the optic sensor to calculate the phase where the maximum vibration occurs. I used a Wilcoxon 991D internally amplified accelerometer. It is specifically used in helicopters to maintain the structure of the aircraft through a wedge located on the tail rotor. The calibrations are shown in Figure 5.

The voltage obtained at the accelerometer's output has a relatively

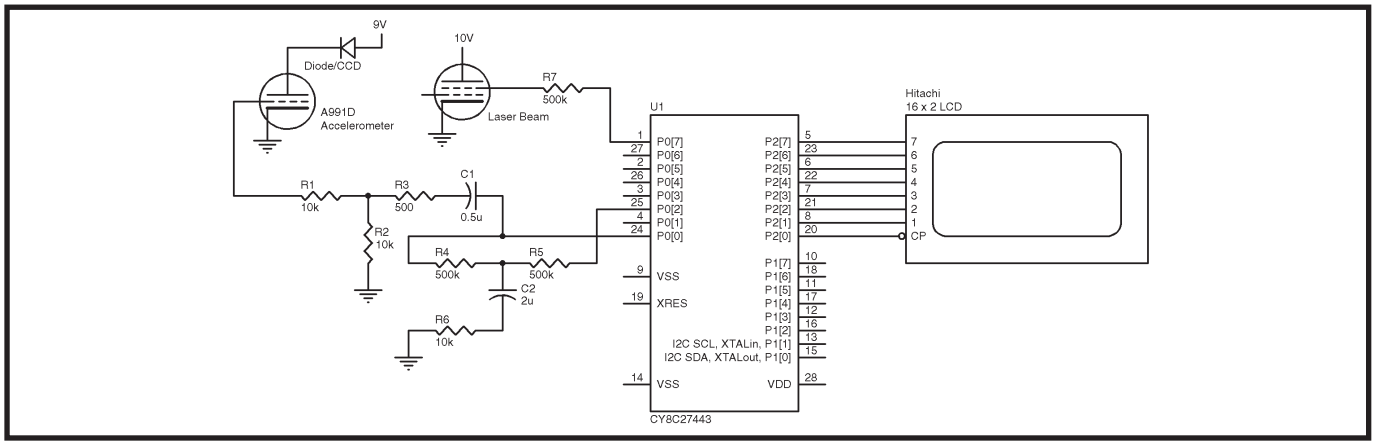


Figure 2—The system features a Cypress CY8C27443 and a handy LCD. It collects RPM data, vibration amplitude in gravitational units (g), and velocity (IPS).

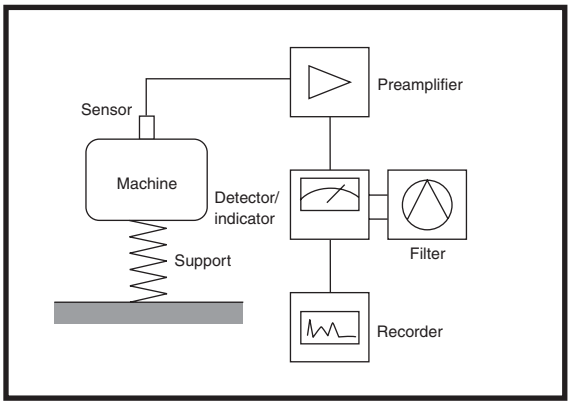


Figure 3—A good vibration measurement system includes several parts: sensors for capturing signals, a preamplified stage, a filter, and a microprocessor.

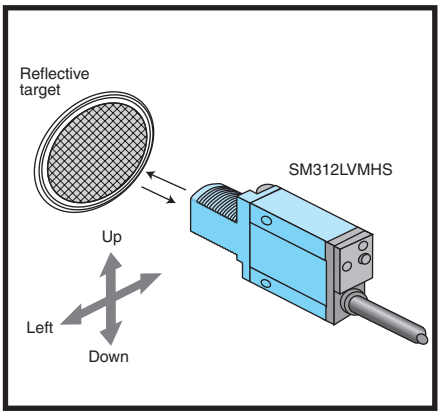


Figure 4—The sensor is aligned. You can move the sensor or the target to obtain the fastest receiver LCD pulse rate.

high value compared to the sensitivity of modern electronic measuring equipment. For example, it is not strange to have 20 mV/g at the output of a piezoelectric accelerometer. If an element subject to crash were measured, it would not be strange to measure an acceleration of up to 1,000 g, which is equivalent to an output voltage of 20 V. However, the impedance of the output of the transducers is tens of gigaohms. Any measuring system connected to the output of the accelerometer will make the voltage decrease unless it has an input impedance that is much higher than this. Therefore, to minimize the effect indicated in the previous section, the preamplifier is located between the accelerometer and the measuring system. In addition, the preamplifier offers other possibilities, such as the amplification of the calibrated variable admitted by another instrument, the adjustment of normalized gains according to the sensitivities of the accelerometers, and the integration of the signal that permits velocity and movement measurements.

You must amplify the signal with an op-amp. This is because the accelerometer has a sensitivity of 40 mV/g, which means that the signals of interest will be buried in the quantization nose of the ADC. A programmable gain amplifier (PGA) user module is commonly used for weak signal conditioning. Problems arise when the desired gain is greater than 100. This problem is caused by the PSoC's internal amplifier input offset voltage, which may be as much as 10 mV. The amplified offset voltage distorts the symmetry of the AC output signal and decreases the dynamic range of successive signal processing stages (ADC and more). Refer to Cypress Semiconductor's application note titled "Offset Compensation for High Gain AC Amplifiers" (AN2320) for additional information.

The signal filter is not executed in an analog way. It is executed through algorithms constructed in software. This is done with recursive digital filters that calculate high speeds at high resolution.

This design uses two recursive IIR filters, a 25-Hz high-pass filter followed by a 45-Hz low-pass filter. The tail rotor turns an average of 2,560 RPM. (Refer to Cypress's application note AN2099 for additional information.) Hence, the transfer function is calculated for a single-pole, high-pass and low-pass recursive filter defined by the following equation:

$$f_0 = \frac{f_s}{2\pi a} \quad [4]$$

The sample frequency, the roll-off frequency, and the attenuation values are combined to define a value for a:

$$a = \frac{f_s}{2\pi f_0} \quad [5]$$

For example, for a sample of 7,812 kilosamples, and to obtain a low-pass filter of 45 Hz, the value of "a" is calculated and later defined and used in the project program in C language. For the 45-Hz simple-pole, low-pass filter, a equals 27.6. For the 25-Hz high-pass filter, a equals 49.7. The previous information defines the following lines in the project program:

```
#define aVALUE 27.6
#define bVALUE 49.7
```

Apply these definitions to the filter:

```
fLowPass += ((Vin-fLowPass)/aVALUE);
fhighPass += ((fLowPass-fhigPass)/bVALUE);
```

The order of the filters changes according to the results. In order to supply the circuit and make the system portable, the equipment is configured to operate with a 9-VDC battery, which also supplies the sensors.

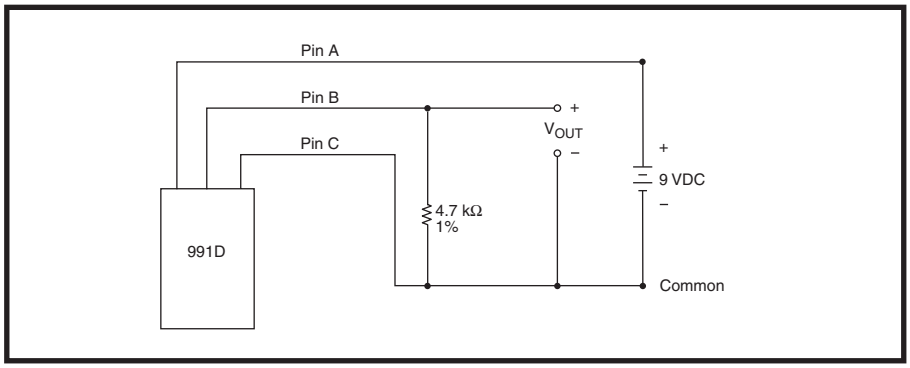


Figure 5—This is the Wilcoxon 991D accelerometer's calibration power diagram.

USER MODULES

The system features several important user modules. A PGA user module is used as a buffer to enable the device input. The signal can be referenced to the internal analog ground

AGND (2.08 V).

The CMPZC (comparator) user module is used to configure a zero-crossing detector and synchronize each cycle of the input signal. It also enables you to send the digital output bit for each signal to an external pin for observation and testing.

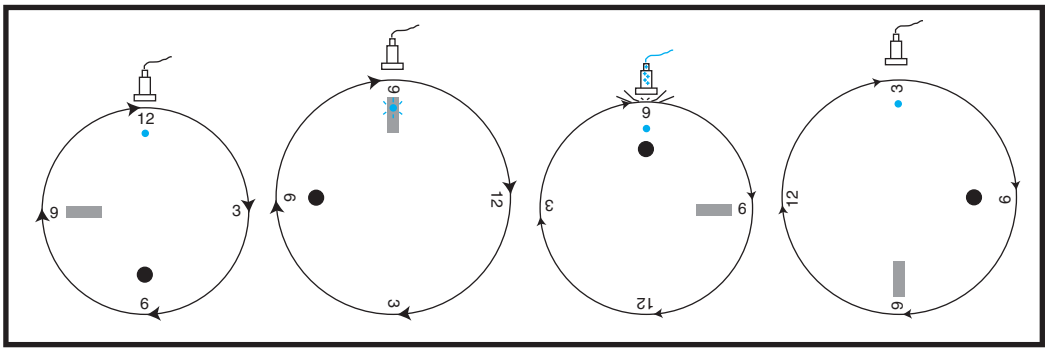


Figure 6—This is the path of an unknown mass in 1 RPM.

The LPF2 (low-pass filter) user module uses two analog blocks to configure a low-pass filter to eliminate noise and avoid the false signals coming from the optic sensor.

Two SCBlock user modules are for general use. They are used as a buffer to send the signal from the PGA to the Schmitt trigger shooter. They are also used as a trigger to send the interrupt signal to the timer, which is used to count the time intervals between adjacent blades (RPMs).

The TIMER16 user module enables you to calculate the quantity of pulses in a determined interval of time that is activated by the triggers. Another timer is configured to detect a decrease in RPMs, interrupting the measuring in case there are less than 12 RPMs. This reinitiates the pointers and variables and displays the message "Rotor Off."

Several user modules are used for the accelerometers. A PGA user module is used together with an inverting amplifier to minimize noise, which is described in Cypress's application note AN2320. The inverting amplifier (AMPINV) is also used as an accelerator.

The ADC in the DELSIG11 ADC user module provides 11-bit resolution and a maximum sample speed of 7,812 ksps. A conversion rate of 128 µs is obtained, but it requires a minimum of three samples for valid data.

The final result of the sample speed is determined by the data clock that depends on the associated clock and resolution:

$$\text{Sample rate} = \frac{\text{Data clock}}{1{,}024} \qquad [6]$$

The selected data clock is 8 MHz, so a user module provides an 11-bit output and a total of 2,048 counts. It is based on a (1.6-Bandgap ±1.6 Bandgap) input range centered on a user-selected AGND (2.08 V).

The TIMER 16 user module is in charge of producing the interval of sample time for the ADC, generating sample intervals of 500 µs.

All of the information collected previously—RPMs, vibration amplitude in gravitational units (g), and velocity inches per second (IPS)—is displayed on a HD44780 LCD.

The accelerometer delivers the amplitude in Gs at a determined RPM, from which I deduced that I could calculate the vibratory acceleration of a harmonic signal as follows:

$$A = \left(\frac{2\pi \times \text{frequency} \times V}{1{,}000}\right) \qquad [7]$$

Because the acceleration A, the frequency (RPM/60), and the value of π are known, you can obtain the speed in millimeters per second. The speed of the vibrations is measured in IPS. This is multiplied by 2.54 (1″ = 2.54 cm) to get the speed in cm/s. The user module placement is shown in the file labeled Additional_Figures_Photos.pdf on the *Circuit Cellar* FTP site.

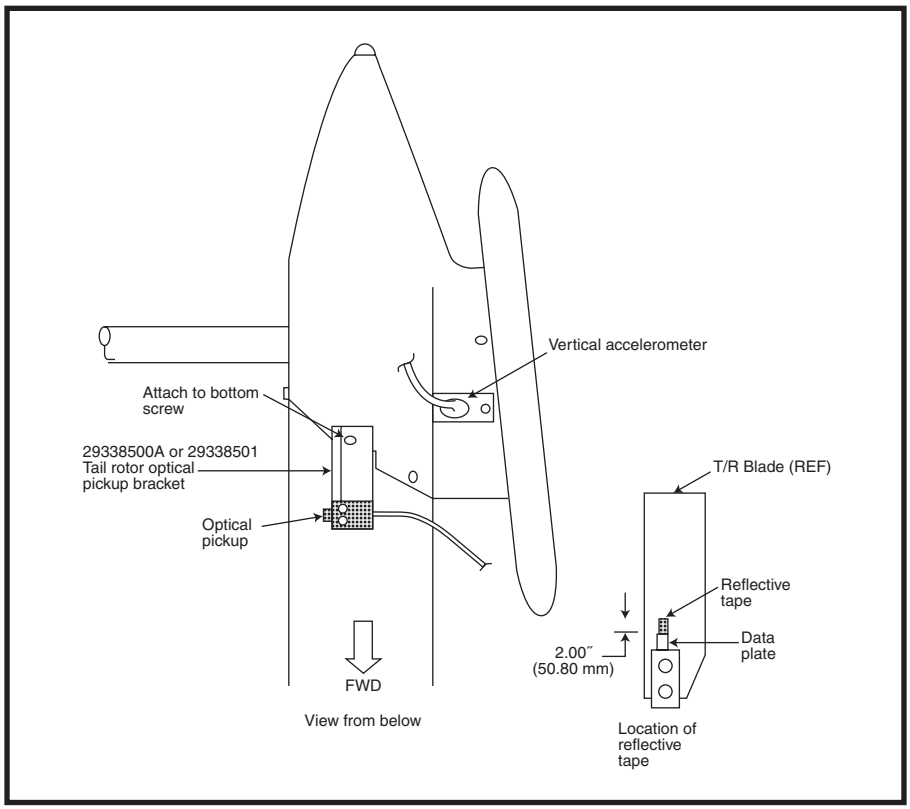


Figure 7—These are the sensor locations.

OPERATION

The rotor turns in a circular pattern. It is always referenced through

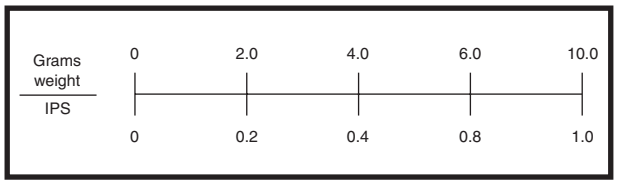


Figure 8—Here you see the relationship between speed (IPS, velocity) and weight (lb).

a time designation (see Figure 6). The signal of the accelerometer (located vertically at the 12 o'clock position) is adapted and amplified in a mixed manner (externally and internally) to later sample with the ADC. The ADC conversions are triggered by a timer. The output of the ADC is filtered by a recursive (IIR) filter (implemented in the PSoC software) that has a cutoff frequency that is close to the rotation rate of the tail rotor.

Definite marker passes are obtained from the optic sensor (laser beam) by the tail rotor revolutions and the reflecting tape. This is controlled and synchronized with a timer and a Schmitt trigger. It is filtered and corrected to determine the maximum amplitude of the vibration and its phase. The information is displayed on an LCD.

INSTALLATION

Figure 7 shows where the data-reading sensors are located. You can obtain power from an adapter on the aircraft that delivers 24 VDC through a cable, although the prototype obtains its power from a 9-VDC battery, which powers both the sensor and the PSoC device.

In the case of the optic sensor, which requires 10 VDC, it was necessary to adjust only the gain to achieve good performance. When installing the cable that transmits the energy to the sensors (which is at least 25′ long), special precautions must be taken to guide it in a secure manner so it doesn't jam the rotating elements of the aircraft.

After the measurements are taken, the values of the magnitude of vibration and the angle or phase of the maximum amplitude are obtained. High-frequency vibrations in the tail rotor are the unbalanced result of a sector of the disc that's heavier than the other. Hence, corrections are made by adding or eliminating weight (see Figure 8).

The location, method, and type of weight added depend on the helicopter manufacturer. A photo of the test results is in the file labeled Additional_Figures_Photos.pdf on the *Circuit Cellar* FTP site.

Figure 6 shows the path followed by an unknown mass (black dot) in an RPM. This produces the imbalance in one of the tail rotor's blades (gray tape) and the way of detecting it. The location of the accelerometer and the optical sensor (blue dot) is also shown.

THE VERSATILE PSoC

The PSoC worked well in this project. I think it would work well in various other applications, such as turbines or conventional motor systems. As you'll see, you can obtain satisfactory results when you compare them to standard reference equipment. Table 1 is a comparison of test results between a PSoC-based system and a rotor analysis diagnostic system (RADS). (Refer to www.smiths-aerospace.com for more information.)

This project is the result of one year of research and development. Therefore, many steps have been omitted so I could demonstrate the power and versatility of a PSoC microcontroller.

	Test 1	Test 2	Test 3
Reference equipment (RADS)	1,472 RPM	2,010 RPM	1,534 RPM
	0.78 g	0.81 g	0.76 g
	1.95 IPS	1.49 IPS	1.90 IPS
PSoC	1,469 RPM	2,005 RPM	1,554 RPM
	0.75 g	0.82 g	0.78 g
	1.92 IPS	1.49 IPS	1.87 IPS

Table 1—This is a comparison of test results between a PSoC-based system and a rotor analysis diagnostic system.

Besides measuring vibrations, this device can be used for turbines, helicopter rotors (shaft, tail, and main), and conventional engines. It's an affordable alternative to the standard equipment that's currently on the market.

Juan Tapia Farias is an electronics engineer with 13 years of experience maintaining hydraulic systems and helicopters for the Chilean Air Force (FACH). He also works independently for PSoC-Chile. You may contact Juan at juantapia@kenter.com.

PROJECT FILES

To download code and additional files, go to ftp://ftp.circuit cellar.com/pub/Circuit_Cellar/2008/218.

REFERENCE

[1] International Organization for Standardization, "2041: Vibration and Shock," www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=6818.

RESOURCES

V. Grygorenko, "Application Note AN2320: Offset Compensation for High Gain AC Amplifiers," Cypress Semiconductor Corp., 2005.

D. Van Ess, "AN2099: Single-Pole IIR Filters: To Infinity And Beyond," Cypress Semiconductor Corp., 2003.

SOURCES

SM312LVMHS Optic sensor (MINI-BEAM model)
Banner Engineering Corp.
www.bannerengineering.com

CY3210-MiniProg1 kit and CY8C27443 microcontroller
Cypress Semiconductor Corp.
www.cypress.com

HD44780 LCD
Hitachi
www.hitachi.com

991D Accelerometer
Wilcoxon Research, Inc.
www.wilcoxon.com

USB Connectivity for Embedded Designs



If you need Full-Speed USB 2.0 device, embedded host, dual role and On-The-Go solutions, Microchip Technology has them available today. We offer 8-, 16- and 32-bit MCUs with USB connectivity, providing easy migration with a single development environment. This maximizes pin compatibility and seamless code migration from 20 to 100 pins, enabling you to scale your USB design with ease.

Download FREE USB software including source code:

- Host Stack
- OTG Stack
- Device Stack
- Class Drivers (HID, Mass Storage and CDC Drivers)
- Thumb Drive Support (Mass Storage Driver, SCSI Interface, 16-bit and 32-bit File Management, Application Software)

USB Starter Kits accelerate development of USB designs using 8-, 16- or 32-bit MCUs starting at only $59.98



Core	Flash Program Memory	Pins	USB Type
8-bit	Up to 128 Kbytes	20 - 80	Device
16-bit	Up to 256 Kbytes	64 - 100	Device, Embedded Host, Dual Role, OTG
32-bit	Up to 512 Kbytes	64 - 100	Device, Embedded Host, Dual Role, OTG

The Microchip name and logo, the Microchip logo and PIC are registered trademarks of Microchip Technology Incorporated in the USA and in other countries. © 2008, Microchip Technology Incorporated. All rights reserved.

Real-Time Operating System Design

Customize an OS for a Powerful Embedded MCU

Naubert's ArmExe is an RTOS for the Luminary Micro Stellaris ARM Cortex-M3 family of microcontrollers and Keil's µVision environment. The multitasking preemptable executive uses task priorities, separate task and kernel stacks, and execution privileges. It handles all interrupts with priorities, supports nested interrupts, and implements binary semaphores and events as the resource-sharing mechanisms.

When I first received my Luminary Micro evaluation board for the Luminary Micro Designstellaris2006 contest, I couldn't believe how much power the little Stellaris chip had. The demonstration program that came with the LM3S811 evaluation board displayed an impressive game in a low-cost, 32-bit platform. Thirty-two bits? Yes, 32 bits in a microcontroller! The Luminary Micro Stellaris ARM with its first implementation of ARM's new Cortex-M3 and its Thumb-2 instruction set is a 32-bit core designed to substitute for 8- or 16-bit microcontrollers in embedded applications. It introduced me to a world of new possibilities of applications that would be difficult to develop with lower-end chips. But I feared this architecture because I didn't know too much about ARM. In addition, to take advantage of such a powerful microcontroller, I needed an operating system. So, I decided to build my own real-time operating system (RTOS). I call it the "ArmExe."

The ArmExe is a compact RTOS designed to take advantage of all of the features of the Luminary Micro Cortex-M3 family of microcontrollers and the ARM's µVision development environment. This multitasking preemptable OS supports up to 254 tasks with 250 priorities each, different task and kernel stacks, execution privileges, interrupt handlers with priorities, and binary semaphores and events as the resource-sharing mechanisms.

The ArmExe was highly integrated with the µVision environment. The C libraries were tailored to work in Thread Safe mode, and I used several of the compiler features, including SVC calls, weak functions, and code-optimization declarations. In addition, I used the µVision Configuration Wizard for easy kernel configuration.

In this article, I will describe the features of the ArmExe and how it was created. The concepts here will help you design a new OS for almost any powerful embedded microcontroller.

RTOS FEATURES

The ArmExe was designed to have a platform to develop multithreading applications in the new Cortex-M3 architecture. The ArmExe kernel permits interrupt nesting with a single interface to the user, while other RTOSes require a different set of system calls when the code is executing in an interrupt handler. Using the same interface makes the ArmExe much easier to use. It also takes advantage of the new Cortex-M3 interrupt optimizations for nested interrupts. When an interrupt arrives, most of the registers are pushed in the stack automatically. If a new, higher-priority interrupt arrives during this interrupt handling and before the beginning of the interrupt code, the Cortex-M3 microcontroller won't need to push a new set of registers because they are unchanged. It will simply jump to the higher-priority interrupt handler and save only the return pointer. The ArmExe recognizes these events and permits the nesting of the priority-based interrupts, which will also simplify the user coding.

The Stellaris processor, with its Cortex-M3 architecture, was designed to be used with an OS. The clever designers included two stack pointers: one for a privileged or handler mode and another for the user or task mode. By using the CPU privileged access and the two stack pointers, the ArmExe simplified the context save and restore routines, achieving fast task switching. The use of the privileged modes also improves the kernel's security by not allowing some instructions to be used in Task mode. For instance, it is not possible to disable interrupts while in Task mode. The ArmExe kernel always runs in Handler mode with full access to all CPU resources, but user code runs in protected Task mode. If

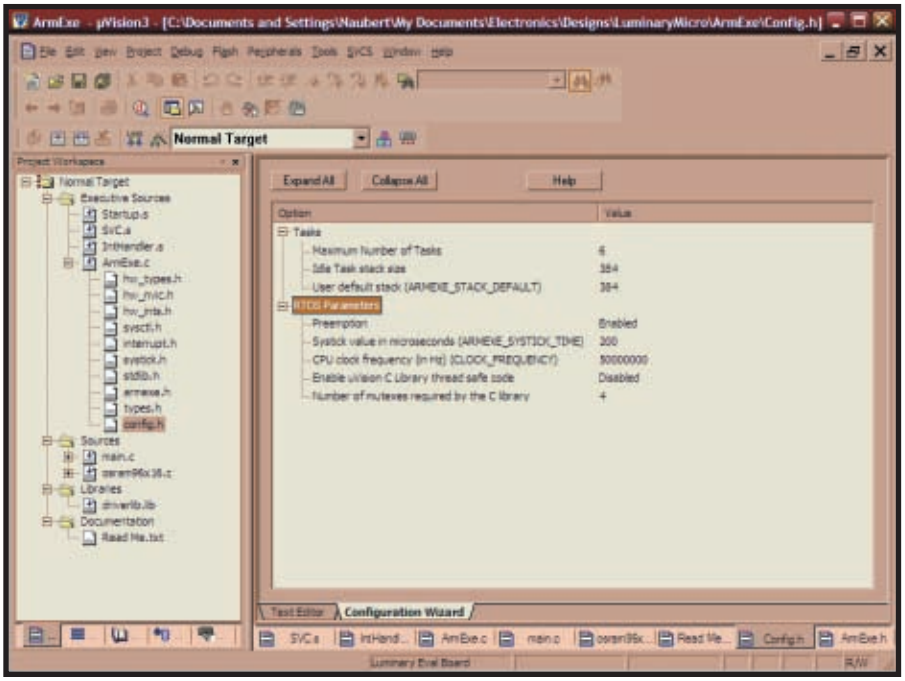


Photo 1—The ARM µVision environment supports configuration wizards, which the ArmExe uses to make the kernel configuration easy for the programmer. Here you can see the kernel parameters in the configuration wizard associated with the ArmExe config.h file.

the user code has to perform a privileged operation, it must do it through a call to an OS function. The Stellaris and other higher-end CPUs have specific instructions for system calls that can switch from User mode to Kernel mode and make the kernel regain full control of its resources.

The ArmExe offers dynamically changing interrupt priorities. The interrupt priorities are heavily used in multithreading applications to guarantee the right interrupt nesting and control resource-sharing access. Common systems use the ability to disable interrupts to access shared resources and interrupt code run with interrupts disabled. The ArmExe is different. There are no interrupt-disabled code regions. Critical sections are handled by raising the CPU priority. Interrupts that arrive with a lower priority than the CPU cannot interrupt the current flow. An interrupt handler code can be interrupted by higher-priority interrupts without affecting the task scheduler. Of course, interrupt code shouldn't invoke blocking kernel calls because they will block the kernel forever. This is a normal programming consideration in RTOSes. The SVC call, which is the only way to call the kernel, is set in the highest possible interrupt priority. When the kernel is executing, nobody can interrupt it. The interrupt handler kernel code is protected, raising the running priority to avoid switching. It is lowered to the corresponding interrupt priority only when the user interrupt handler is called.

The ArmExe also implements task priorities, which are handled entirely in the kernel code. Ready tasks with a higher priority will preempt any lower-priority task that is running. If preemption is enabled, tasks with the same priority will time slice each systick interrupt. If preemption is disabled, the current running task must block itself before the kernel schedules the next same or higher-priority ready task.

Finally, two user-level resource-sharing control mechanisms were implemented in the ArmExe: mutexes and events. Mutexes are a mechanism to control exclusive access. A task must request the ownership of a mutex to have exclusive access to an associated resource. Only one task can be the owner of a mutex at a time. If another task wants to access the same resource, it first has to request the same mutex. The kernel will block this task because another task has the mutex ownership. The blocked task

Function	Description
ArmExeInit	This routine starts the ArmExe kernel and defines the initial systick period with systick_period.
ArmExeTaskCreate	Creates a new task. This routine requires a storage for the stack, and a pointer to a reserved 32-bit aligned storage must be provided. The function returns the task ID, which is used as an argument of task-related kernel calls.
ArmExeTaskSleep	This routine sleeps a task for a specified time. It will behave as a normal delay function with one difference being that it will guarantee only the delay to be at least the number of ticks specified. But it could take more time. Like any other RTOS, the higher-priority task will run and no guarantee can be made that the newly awakened task will run. If the task ID is zero, the current task will sleep.
ArmExeMutexGet	Get the specified mutex. If the mutex is not owned by another task, the current task will acquire ownership. If the mutex is owned by another task, this task will be blocked and queued in the mutex wait list.
ArmExeMutexRelease	Release the specified mutex. This function should be called only by the owner of the mutex. It will free the oldest blocked task waiting on the mutex and will transfer the ownership of the mutex to it. Other tasks blocked on the mutex will remain blocked until the last owner releases the mutex. When this happens, the mutex will be free of ownership and no task will be waiting on it.
ArmExeEventWait	Wait for the event to happen. If the event didn't occur before the call, the task will block waiting for it to occur. If the event already happened and has not been cleared, the call just returns without blocking.
ArmExeEventSignal	Signal that the event happened. It will flag the event and release all blocked tasks on the event.

Table 1—These are some of the ArmExe kernel system calls. With only 28 system calls, the ArmExe is easy to use. The kernel can also be easily extended, a feature that will let the programmer add code that needs to be run in Privileged mode.

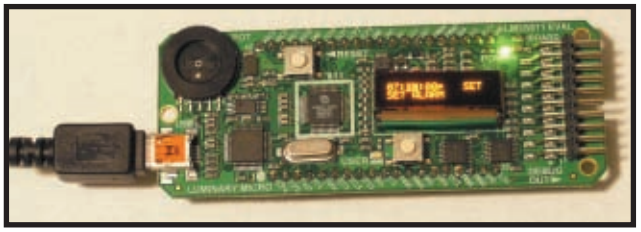


Photo 2—Here the EKK-LM3S811 evaluation board is running the multithreaded alarm clock demo application. A task is in charge of the user input. Another task handles the display. A third one is the alarm task.

will not have CPU time until the first task releases the mutex. Once this happens, the blocked task will gain ownership to the mutex and will have exclusive access to the shared resource. The ArmExe can have an unlimited number of mutexes. The ArmExe also implements events. An event is a mechanism where tasks wait for something to happen and they will block waiting for it. When the event occurs, usually in interrupt code, the event is signaled and all tasks that were blocked on it are unblocked.

The ArmExe was designed and compiled with the ARM µVision 3 development environment in C and in assembler language. It uses the Luminary Micro Stellaris driver library. It was tested with the 1003 release, but it should work with any subsequent version. The user code can use the driver library in multiple tasks if the `ARMEXE_KEIL_LIBRARY_SAFE` constant is set in the config.h configuration file. All of the files related to this project are posted on the *Circuit Cellar* FTP site.

USING ArmExe

The ArmExe is easy to use. A single-kernel configuration file has to be modified to change the kernel parameters to adjust its memory footprint. The parameters include the maximum number of tasks, the idle task stack size, the default stack size for user tasks, a preemption enable/disable option, the systick period, the clock frequency (if the thread-safe C library is needed), and the number of required C library mutexes. Photo 1 shows the configuration wizard of the ArmExe config.h file where these parameters can be easily adjusted thanks to this feature of the µVision environment.

The kernel system calls are separated into functional groups: the Kernel initialization and configuration group has two system calls to initialize the kernel and set the systick period; the Task Management group, has all task creation and destroy calls; the task Scheduling group encloses the calls related to task suspension, resuming and priorities; the Mutex Handling group, has the calls related to Mutexes; the Event Handling group has the calls related to Events; the Interrupt Handling group has the calls to enable, disable, and manage interrupts; and the miscellaneous group has additional utility system calls. Table 1 shows some of the most important kernel calls. A full table is posted in the *Circuit Cellar* FTP site in the file labeled ArmExe Full System Calls.doc.

Programming with the ArmExe is simple: just add a reference to the ArmExe.h include file, program a `main()` routine that calls ArmExeInit(TICK_TIME), and define the `init_tasks()` function to initialize and create the tasks. Optionally, you may define `signal_error()` to capture fatal kernel error messages. In the µVision project properties it is required that the parameter `--entry Reset_Handler` is defined in the linker options section and that fixed address compilation is used. The following files must be added to the project, which are part of the kernel: Startup.s, SVC.s, IntHandler.s, and ArmExe.c. The Luminary Micro DriverLib library revision 1003 or later must also be added to the project.

An alarm clock demo program was developed to test some of the ArmExe's functions. As simple as it may sound, the clock program is a three-task multithreaded code running flawlessly in the EKK-LM3S811 evaluation board (see Photo 2). The code is fully documented. I invite you to download the main.c file from the



Circuit Cellar FTP site.

ArmExe INTERNALS

An RTOS simulates the concurrent execution of several tasks or processes. This simulation is achieved by assigning a bit or slice of CPU time to every task that is ready for execution. A task is told to be ready if it can run without having to wait for any other resource but CPU time. A task is considered to be blocked if it is waiting for a resource (i.e., waiting for an interrupt). By knowing that a task is just waiting, instead of leaving it running in a loop, an RTOS will hold the task from running and assign the CPU time to other ready tasks. In a single-threaded CPU, such as a Luminary Micro microcontroller, only one task can be in the running state. If several tasks are ready, the highest priority task will have the CPU. If preemption is enabled, all tasks with the same priority will be given slices of CPU time in a round-robin fashion. Figure 1 shows the ArmExe task states.

The first step to build an RTOS is to write routines to save and restore the state of a running task. Saving the task state will put the task in "hibernation." Restoring its state will make that task run again, as if it was never suspended. The state of a task or its context comprises all of the user-level registers, the stack, and other system status. This information is saved in a structure called the task control block (TCB). Task switching is the transfer of CPU time from a task to another and it is achieved by saving the context of the current running task and restoring the context of the new selected ready task.

The ArmExe performs its duties as a standard RTOS: it has a task control block (TCB structure in ArmExe.c), which defines a task. The ArmExe TCB also has other variables, such as the task status, task ID, and task priority. When the ArmExe needs to change context, it will do it only from Handler mode—that is, from inside interrupt code. When the Cortex-M3 interrupts a task, it will automatically save the R0 through R3, R12, LR, PC, and PSR registers. They are stored in the task stack. The ArmExe kernel then saves the missing R4-R11 registers also in the stack, thus saving the entire context for that task. The task switching happens when returning from an interrupt handler. Instead of just restoring the last pushed registers, the kernel will restore the corresponding registers of the new task that should run. Because all of the registers are saved in the task stack, the corresponding task will run at the interrupt exit by just changing the task stack pointer.

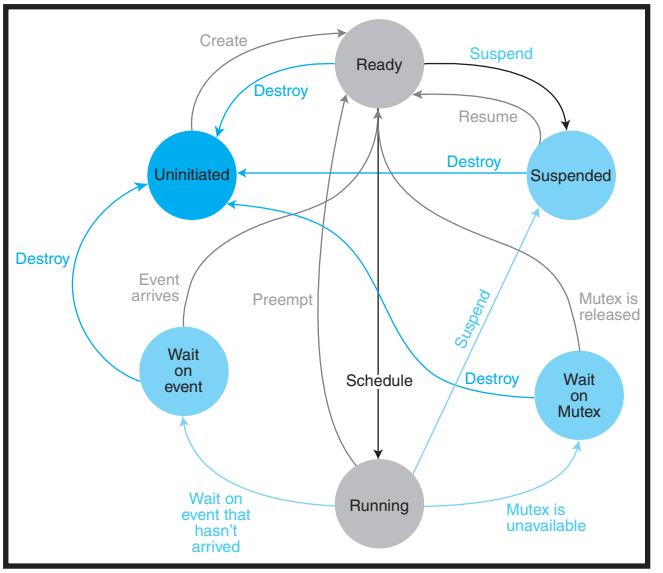


Figure 1—These are the ArmExe possible task states. A new task is normally put in the ready state. It will run when it's scheduled by the kernel according to its priority. If a task performs a sleep or a wait for mutex or event, the kernel will put that task in a wait state and won't have CPU time. It will be put in the ready list again when the block condition ceases.

This task-switching logic is programmed in each of the three main interrupt entry handlers: in SVC.s for the SVC kernel calls, in the `SysTick_Handler` in ArmExe.c for the `systick` interrupt handler, and in IntHandler.s for the rest of the IRQ handlers. Each of the interrupt handlers can switch a task, so the entry and the exit code are similar. The interrupt entry code increments the `ISR_counter` variable. This is used to check the interrupt nesting. If the counter returns to zero at ISR exit, it will be recognized as the last interrupt handler—and thus, a context restore is forced. However, if it isn't, then it means that the return must be done to another interrupt handler, in which case the context is not restored from the thread stack. This simple logic mixed with the right interrupt priorities is what makes the interrupt nesting support a

Listing 1—This shows how the context is saved at the interrupt handler entry. If an interrupt arrives while a previous handler is running, it is not necessary to save the context. This way, the ArmExe takes advantage of the optimizations for nested interrupts available in the Cortex-M3 architecture.

```
PUSH        {R1,LR};              save R1, LR, as used in this handler

; increment ISR_counter, this counter maintains the ISR nesting level
LDR         R12,=ISR_counter
LDR         R1,[R12]
ADD         R1,R1,#1
STR         R1,[R12]

; check if in first interrupt
CMP         R1,#1                 ; get ISR_counter, if 1 this interrupted user code
BNE  do_not_save

; save context if first interrupt
LDR         R12,=task_running     ; current task
LDR         R12,[R12]             ; get TCB
LDR         R1,=old_task          ; get old_task pointer
STR         R12,[R1]              ; save current task into old_task
MRS         R1,PSP                ; get thread SP into R1
STMFD       R1!,{R4-R11};         save R4-R11 registers into thread stack
STR         R1,[R12,#0];          store new SP in TCB offset 0 (tcb->sp)
```

reality in the ArmExe. In the interrupt handler entry code shown in Listing 1, notice how easily the context is fully saved with only seven instructions.

The interrupt exit code is shown in Listing 2. The `ISR_counter` is decremented and compared with zero. If so, the interrupt handler exits from the last ISR so a context restore must be performed. In the SVC handler, additional code was added to return the R0 value to the calling thread because the SVC calls implement C function calls with a return value. The actual context restore is performed in the following six instructions. The new task TCB is retrieved from the `task_running` variable. The SP of the new thread is used to restore R4-R11 and then the routine saves the resulting stack pointer into the PSP, or process stack. At this point, returning from the interrupt handler (with `POP {PC}`) will automatically restore the rest of the registers and return to the new selected task code.

Once the user task is run, the kernel will regain control only when an interrupt arrives. At that point, the kernel should assess the tasks and determine which one will run next. This is the scheduler's job. The scheduler in the ArmExe runs in Privileged mode. This mode is automatically set by the CPU when an interrupt arrives. The schedule decision normally runs at the end of each interrupt handler, when the kernel finishes its duties and returns the CPU to a user task. It is assumed that the interrupt could change the status of another task so the kernel must check if a higher priority task must run after the interrupt is processed.

Listing 2—This is the interrupt handler exit code where a task context is restored. After checking that the interrupt is the last one, the registers are restored and the stack pointer is set to run the task selected by the `task_running` variable.

```
            ; check if this is the last ISR
            LDR     R12,=ISR_counter
            LDR     R1,[R12]
            SUB     R1,R1,#1
            STR     R1,[R12]
            CMP     R1,#0
            BNE     return          ; this is not the last ISR, return

            ; restore context
last_isr    LDR     R12,=task_running
            LDR     R12,[R12]       ; get new TCB
            LDR     R12,[R12,#0]    ; get new thread SP into R12
            LDMFD   R12!,{R4-R11}   ; restore R4-R11 from the thread stack
            MSR     PSP,R12         ; set thread SP, context will change at
                                    interrupt return
return      POP     {PC} ; return
```

The ArmExe scheduler is implemented in the `schedule()` routine (in the ArmExe.c file), which will switch the `task_running` variable to the highest priority ready task (which is the first element in the ready list, because it is a priority ordered list). The task scheduling is also called each systick interrupt to implement preemption. The `SysTick_Handler` interrupt routine will call the `SysTick_Interrupt()`, which calls `schedule()`.

The `schedule()` routine is straightforward. It will check the next ready task queued in the ready list and validate that if its priority is greater than the current running task. If it is, that task will be the next running task by changing the `task_running` variable, and the old running task will be queued in the ready list at its corresponding position depending on its priority. If preemption is enabled and a task is ready with the same priority as the current running task, then a task switching will be forced, producing a round-robin time-slice of tasks with the same priorities.

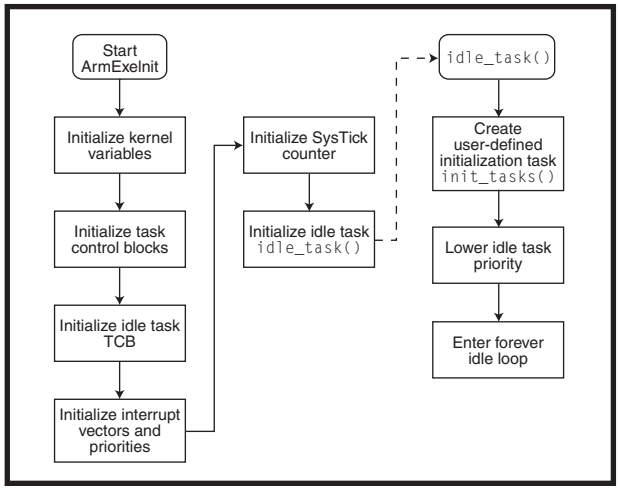


Figure 2—The ArmExe kernel initialization starts when the user code calls `ArmExeInit()`. An interesting part of this initialization is that a higher priority task is created to initialize the user tasks. When this task finishes, the kernel lowers the priority and that is when the user-created tasks can have the CPU.

The ArmExe has implemented seven task states: Uninitiated, Ready, Running, Sleeping, Suspended, Waiting on mutex, and Waiting on event (see Figure 1). The task control blocks are stored in a static array called `tcb_array`. All of the entries are initialized with the `TASK_UNINITIATED` state.

When a new task is created, it can be initialized as Ready or Suspended. If the task is Ready, it will be queued in the ready list so the next `schedule()` call considers it to run, depending on its priority. If the task is suspended, it will remain in that state until resumed with the `ArmExeTaskResume()` call. A single task will be running at a time. If no tasks are ready to run, then the idle task will run, which does nothing. If a task calls a kernel function that requires the kernel to complete some operation before returning to the task, the task will be blocked until that operation is finished. There are several blocked states in the current implementation of the ArmExe, including sleeping, waiting on mutex, and waiting on event. A task that is blocked will be queued in a list associated with the operation that must be finished.

Sleeping tasks are queued in the sleeping list, which is a time-ordered list of tasks that is checked every systick interrupt to awaken tasks. If the time that a task requested to sleep passes, the task status is changed to Ready, the task is queued again in the

ready list, and a reschedule is requested. Waiting for mutex tasks are queued in a list associated with the mutex. The tasks remain blocked until the mutex is released, where the first blocked task is released, queuing it into the ready list, and making it the next owner of the mutex. This pattern remains until all waiting for mutex blocked tasks are released. Waiting for event tasks are queued in a list associated with the event. When the event is signaled, all of them are moved to the ready task at once.

KERNEL INITIALIZATION

The ArmExe has to be initialized by the user code before creating the tasks. The initialization is done by calling `ArmExeInit()`. The routine will perform several internal housekeeping routines, initialize the systick counter, and start the idle task (see Figure 2).

The idle task gets CPU time when no other task is ready. It spares CPU time by cycling in an infinite loop. But before entering in this loop, the idle task must initialize the user tasks. It creates a new task that will run the user-defined `init_tasks()` routine. This task will run with a priority value, which is higher than any user-selectable priority.

Because none of the user-created tasks can have a higher priority than this initialization task, no task will run until the priority is lowered. This occurs automatically when the `init_tasks()` ends, leaving the CPU to the user-defined tasks starting with the higher priority user task first. This mechanism simplifies the user tasks start and makes the ArmExe easier to use.

TASK INITIALIZATION

By now you should have a question. How will a task run the first time if its context was never saved before?

Well, the kernel has to initialize the task context the first time to make the scheduler think that the task was previously suspended. This is performed in the task creation routine `ae_task_create()`. The routine will find an available task ID. It will then search for a free Task Control Block (TCB) slot in the tasks array and initialize it according to the specified parameters. It then calls the `stack_init()` routine, which will initialize the context as if the task was previously suspended by an interrupt. This means that the task stack has to have pushed the register values in the right order, that the process status register has the Thumb bit on (as the Stellaris CPUs can only run Thumb instructions), and that the return value in the stack points to the start of the corresponding task code. The LR register is set to the `ArmExeTaskExit()` code. So, if the task routine ends, it will jump to the exit code by default. Finally, the resulting stack pointer is stored in the TCB. So, it is ready to be used by the interrupt exit routine to start the task whenever the scheduler put this task to run.

PROJECT COMPLETE

I finished this project with all of my goals satisfied. I learned a lot about the new ARM Cortex-M3 architecture, how the nested vectored interrupt controller (NVIC) worked, the details of Cortex-M3 interrupt optimizations, how to handle privileged instructions, user and handler stacks, and everything about interrupt handling with priorities.

I am especially pleased that I completed a fully customizable real-time operating system that will simplify my developments on the Luminary Micro Stellaris family of microcontrollers. This was my first ARM development. The result is not bad for just one month of dedication. It certainly demonstrates how a good architecture, the right tools, and an excellent MCU implementation can accelerate your project development to achieve a fast time to market.

Editor's note: Naubert Aparicio's "ArmExe" design won Honorable Mention in the Luminary Micro Designstellaris2006 Contest. For more information about the contest and to learn more about the winning projects, visit www.circuitcellar.com/designstellaris2006.

Naubert Aparicio (naubert.aparicio@usa.net) is a computer science engineer from Simon Bolivar University in Venezuela. He works as the technology director for Quantum Business Engineering in Puerto Rico, managing the design of mission-critical Unix systems architectures. When Naubert has free time, he enjoys learning about robotics, advanced digital design, and embedded systems.

PROJECT FILES

To download code, go to ftp://ftp.circuitcellar.com/pub/Circuit_Cellar/2008/218.

RESOURCES

ARM, "ARM v7-M Architecture Application Level Reference Manual," ARM DDI 0406A-01, 2006.

———, "Cortex-M3 Technical Reference Manual," ARM DDI 0337B, 2006.

———, "Procedure Call Standard for the ARM Architecture," GENC-003534, 2006.

———, "RealView Compilation Tools: Assembler Guide," Version 3.0, DUI 0336B, 2006.

———, "RealView Compilation Tools: Compiler and Libraries Guide," Version 3.0, DUI 0337B, 2006.

———, "RealView Compilation Tools: Linker and Utilities Guide," Version 3.0, DUI 0338B, 2006.

Luminary Micro, Inc., "Clocking Options for Stellaris Family Microcontrollers," Application Note, AN01240-01, 2006.

———, "LM3S811 Microcontroller Data Sheet," DS-LM3S811-00, July 2006.

———, "Stellaris Driver Library: User's Guide," SW02034-697, 2006.

———, "Stellaris LM3S811 Evaluation Board User's Manual," EK-LM3S811-01, 2006.

SOURCES

µVision 3 IDE
Keil
www.keil.com

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

Prototyping with QFP Parts

Fed up with trying to socket QFP parts, Stuart came up with a solution. He built a PCB with the QFP part he wanted and used it as a prototyping module. The system turns a QFP part into a socketable device that you can use for prototyping. The concept can be used for FPGAs, microprocessors, and other parts in QFP packages.

In the old days of integrated electronics, integrated circuits came in DIP packages. It was easy to build a hand-wired prototype of your design. But as technology advanced and pin counts increased, DIPs gave way to PLCC parts. You could still prototype those by socketing them, but now, even newer QFP parts are difficult to socket and hard to prototype with.

Typically, the way to work with a QFP or QFN (or other surface-mount quad package) is to design your circuit, make a PCB, and start debugging. If the part is a microcontroller or FPGA, you sometimes have the option of buying a development board from the manufacturer and developing your design from that. These options have some drawbacks, though.

If you want to make a one-time circuit, a full PCB is often expensive. Development boards are pricey and they often do not have what you really need to prototype your design. Finally, sometimes you want to build something that needs a high-pin-count device, but you don't want to go through the PCB design/layout/fab cycle just to test a concept or build one.

Another option is to use an off-the-shelf QFP-to-PLCC or QFP-to-DIP adapter. They work, but they are universal, which means they don't have specific characteristics for the part you want to use. This can be problematic, for example, if the trace length between the power pins and the nearest bypass capacitor is long.

The approach described here is a "QFP module." The basic concept is that you build a PCB with the QFP part you want and use that PCB as a prototyping module. The module contains all of the essential parts to make it usable in any design. Essentially, the module turns a QFP part into a socketable device that you can use as a prototyping device, similar to a DIP IC.

The example used here is a 64-pin TQFP board for an Atmel ATmega325, ATmega645, or ATmega128 microcontroller. The board includes pads for the QFP device, bypass capacitors, an ISP programming header, and a crystal. Photo 1 shows the front and back of the board.

The top of the board holds the QFP, crystal, and a programming adapter. The bottom of the board has bypass capacitors, capacitors for the crystal, and some resistors. Each of the four connectors around the edge is for a single-row header that enables the board to be plugged into a perfboard for prototyping.

Although this example is for a specific family of microcontroller parts, the general concept can be extended to FPGAs, microprocessors, and other parts that are available in QFP packages. To provide maximum effectiveness, you should include parts that are likely to be used in all designs on the board.

Although this approach requires making a PCB, you will be making a generic PCB that you can use to prototype any circuit using the target part. You could use this to prototype a circuit that uses a microcontroller and an FPGA, or just an FPGA. You could even build a few of these and keep them around for any quick microcontroller/FPGA prototypes that you need to build up when you want to try out a concept.

DESIGNING THE PCB

There are a number of ways to design a PCB. If you have access to professional layout packages like Cadence Design Systems's Orcad PCB design tools or Altium Designer or Altium P-Cad, it is just like any other simple PCB design. For experimenters or others without access to a professional CAD application, there are lower-cost alternatives such as Lab Center Electronics's Proteus PCB design software (www.labcenter.co.uk) and Cadsoft Computer's Eagle (www.cadsoftusa.com). Finally, ExpressPCB (www.expresspcb.com) and PCB123 (www.pcb123.com) both offer free PCB layout software on their web sites. These packages are tied to their ordering processes.

Generally, you will want to include on the board the common components that you would include on a typical

design. This could include crystals or oscillators, programming connectors (ISP, JTAG, and more), bypass capacitors, serial EEPROMs, ADC references, voltage regulators (3.3 V, 2.5 V, and more), and an LCD connector (for LCD-specific microcontrollers). This is not an exhaustive list, of course, but it is typical of what you might find on such a board. Note that you don't have to actually install all of the parts, but you will want to put pads for them on the board. For example, if you are making a board for a microcontroller that has multiple crystal inputs (such as the Atmel AT32 family), you could put pads on the board for all of the possible crystal connections, but install only those that you need.

The example used for this article includes a programming (ISP) connector, crystal (and associated capacitors), and bypass capacitors. There are also four resistors that can be installed (two at a time) to use the board for either the ATmega325/645 or the ATmega128. The resistors direct the appropriate microcontroller pins to the ISP connector.

Generally, you will want to bring all of the microcontroller pins out to the headers on the perimeter of the board, even if they are connected to one or more connections on the board. The example board has the ISP pins connected to the ISP header, but also to the perimeter pins. This enables them to be used both for in-circuit programming and for other functions in the target application.

Important: make sure the headers are on 0.1″ centers across the board. If the headers are not on a multiple of 0.1″ spacing, the board won't fit into a standard perfboard. On the example board, J1 and J3 are 1.7″ apart. If they were placed 1.75″ apart, they would not plug into a standard 0.1″ perfboard without bending some pins. Keep this in mind when placing the connector footprints.

Figure 1 shows the schematic of the example board. J1–J4 are the perimeter headers. J5 is the QFP part. It is shown as a dual-row header, but the PCB layout program is told that it is a QFP part to get the right footprint. J6 is the ISP header. Finally, R2–R5 are used to program the board for the ATmega325/645 versus the ATmega128. When you design the PCB, use the right footprint—QFP packages come in various sizes, and even parts with the same number of pins can be different if they have different pin spacings or package widths.

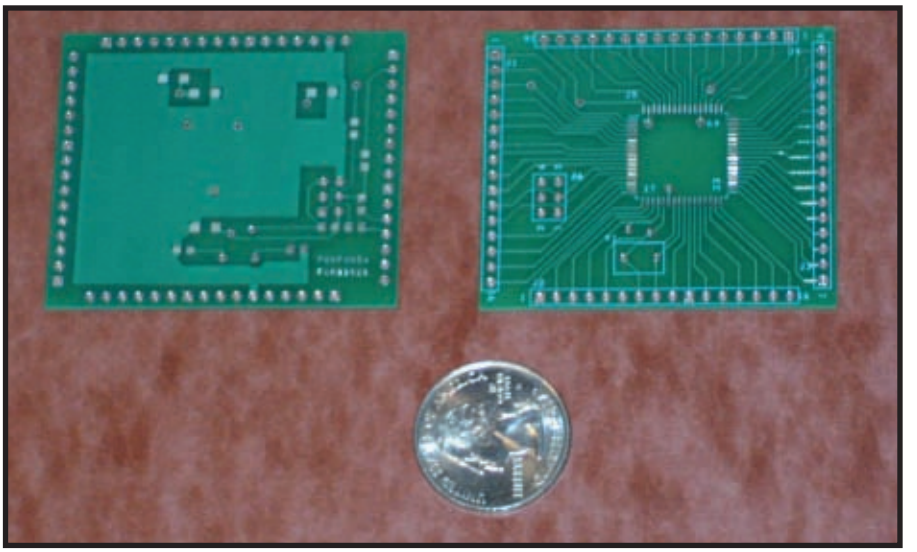


Photo 1—Take a look at the front and back of an example PCB. The top of the board (right) has the QFP area, programming header, and crystal. The bottom of the board (left) has resistors and capacitors.

The back of the board has a large ground area (see Photo 1). This provides a low-impedance ground path between the microcontroller, bypass capacitors, and the ground connections at the edge of the board. It is not essential that you have a large ground area, but you will want wide ground traces to create a low-impedance path for these parts. If you need the back of the board to route traces, you could run a wide ground trace around the

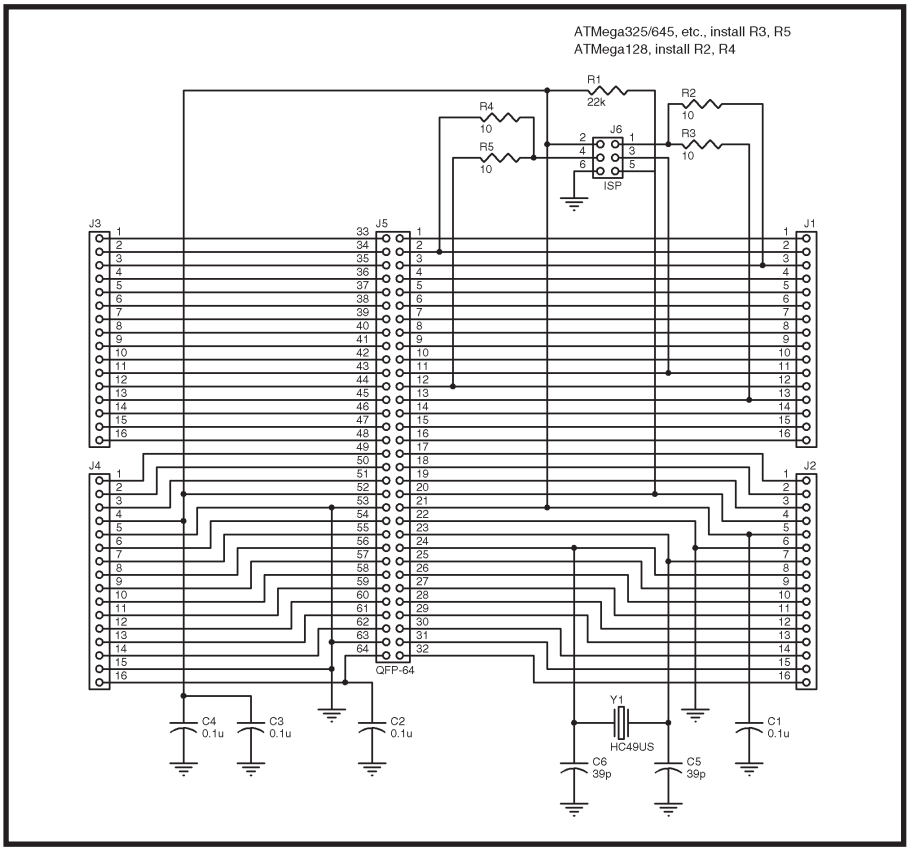


Figure 1—This is the PCB module. J5 is the QFP part. It is shown as a connector to make the schematic easier to read, but it is configured on the PCB as a QFP footprint.

outer perimeter of the board, outside the headers, at the expense of making the board larger.

Bypass capacitors are a significant advantage to this approach. Standard prototyping adapters that fit any device of the right size and pin count cannot be designed for specific power and ground pins. Consequently, it is difficult to connect bypass capacitors to those pins on the board itself. By designing a board for your specific part, you can add bypass capacitors near the power pins and connect the other side to a low-impedance ground connection. This will minimize problems caused by power noise and switching transients.

Recommended values for bypass capacitors are typically 0.01 to 0.1 µF. If you expect to use the completed module on a breadboard with long power and ground leads, you could also include a bulk electrolytic capacitor on the board.

Use a two-layer board whenever possible to keep the cost low. However, you may run into cases where four layers are needed. An example of this would be if you were using a microcontroller with a USB 2 interface. In that case, you may need to design the traces for a specific differential impedance, and that may require a ground plane on an inner layer. Design of controlled impedance traces is beyond the scope of this article, but there are a number of Internet resources that describe it.

If you are going to hand-assemble the board, I recommend parts no smaller than 0805. Maybe it's just my age, but I find 0603 and smaller parts to be difficult to assemble by hand. In most cases, it will be easiest to put capacitors and resistors on the bottom of the board, leaving the top free for traces to the perimeter headers. On the example boards, the QFP, crystal, and ISP header are on the top of the board; all of the resistors and capacitors are on the bottom.

Some QFP parts have a ground pad on the bottom of the package; if so, make a large pad or copper area on the PCB, with one or more vias to the back of the board. Use a fairly large hole—0.040″ or so.

I ordered the example boards from PCB FAB Express (http://pcbfabexpress.com). However, a number of PCB prototyping suppliers will take orders over the Internet and make boards for less than $20 each in small quantities. The pricing is competitive with off-the-shelf QFP adapter boards. The key is that the boards are small (example boards are a bit under 2″ × 2″), making them relatively inexpensive.

ATTACH PARTS

The best way to attach parts to the board is, of course, to use a professional PCB assembly shop. There are a number of contract manufacturers who will fab the board and assemble it for you. However, these boards would typically be made in small enough quantities to make that cost-prohibitive.

The example board was assembled by hand. By far, the hardest part to put on the board is the QFP device. Several steps are required to assemble the boards. First, place the QFP on the board. It is extremely important that

Cellular and GPS capable
Data Mover


• Flash file System
• 4 Chan 12-Bit A/D
• 1MB SRAM
• 512KB FLASH
• 4 Isolated Inputs
• 4 Hi Current Outputs
• 2 External RS-232
• 2 Internal RS-232
• Bat Backed Clk/Cal
• Cell Modem Option
• Internal GPS Option
• Metal Case Option
• 1ma Standby Option
It's easy and cost-effective to do mobile or solar-powered data collection and asset monitoring with the JK micro's Data Mover.
With the ability to integrate a Cellular Modem, GPS and DOS-based embedded controller in a single rugged enclosure, you can capture and transmit your data quickly, easily and at low cost. Inexpensive development kits including Borland C/C++ and PowerBasic are available now. Call or email us for more details.
Call 530-297-6073 Email sales@jkmicro.com
On the web at www.jkmicro.com
JK microsystems

NEW Production Mate
the PIC® MCU
PRIME 8
Production Programmer
8 Port In-Circuit
Gang Programmer
· SD Card Reader
· 2MB Internal Memory
· Voltage Configuration
· Stand-Alone LCD & Control Buttons
· Automatic Serial Numbering
· Free Control Software
$950
www.ccsinfo.com/gangcc
CCS
Discount available with software purchase

Need something more portable... LOAD-n-GO $199
PIC is a registered trademark of Microchip Technology, Inc.

you align the pins with their respective pads. While holding the part down, solder a few pads on one corner of the QFP device. Don't use excessive solder, but don't worry about shorting pins either. I held the part with my thumb while soldering with a wetted soldering tip, but you could also use masking tape, copper/aluminum tape, or a spot of glue under the device. Be sure you have the QFP oriented the right way. There is no polarizing key or anything other than the visual pin 1 indicator to prevent you from rotating the part.

After soldering one corner, recheck the pins to ensure they are aligned with the pads on all four sides. This is extremely important because it will be difficult to fix later if you get it wrong. Unless you have bionic eyes, you will need a magnifier. I recommend a jeweler's loupe or another high-power magnifying lens so you can see the leads. After verifying lead alignment, solder a few pins on the opposite corner from the first one you soldered.

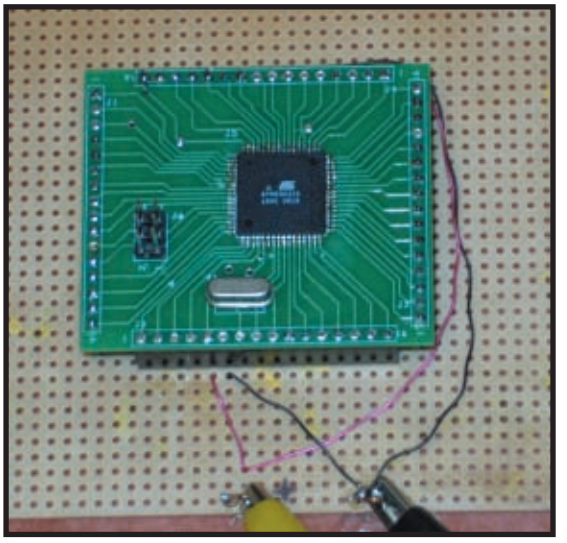


Photo 2—This is an example PCB plugged into a perfboard for prototyping. At this point, just power and ground are connected, so the test program can be run to verify all of the connections.

Using a magnifier, make sure the pins are touching the pads. If the part isn't level on the PCB, it will be difficult to make connections on the board. If you use tape to hold the QFP down for soldering, peel the tape off in a high-humidity area (or wet the board) to avoid static damage to the QFP device. Solder the remaining pins. Start on one of the two unsoldered corners so you don't loosen the part when you solder the corners where you started. Again, don't use too much solder, but don't worry about shorting pins together.

Once all of the QFP pins are soldered to the board, use rosin-impregnated copper solder wick (solder braid) to remove the excess solder shorting the pins together. Heat the solder wick and use it to wick away the excess solder. To avoid damaging the part, this should take no more than a few seconds per side. Don't be too frugal with the solder wick. Cut off the used portion and throw it away to minimize the time needed to wick solder off the pins.

You might think that solder wick will pull away the solder from under the QFP leads as well, but this doesn't happen.



The solder wick works by capillary action, and won't pull the solder from under the leads. However, if you tried this with a suction solder remover, you would remove that solder. As a result, many of the leads wouldn't make contact with the board.

Again, using a magnifier, examine the pins for solder bridges missed and any leads that appear to be separated from the board. If you have solder shorts, you may be able to eliminate them by touching one of the leads with the tip of the iron. If that doesn't work, use the solder wick again. If the QFP has a ground pad in the middle, solder it to the PCB from the back of the board through the vias.

Once the QFP is soldered into place, solder the remaining parts. I recommend soldering the headers to the board last. Having the headers out of the way makes it easier to solder everything else. However, if you are worried about static or handling damage to the QFP, you could solder that part last.

Once everything is soldered to the board, use flux remover to clean the flux from the board. The rosin from the solder wick will make a rosin mess on the board. On the example boards, I cleaned the QFP before soldering everything else, but that was mainly to make it easier to inspect for solder bridges and unsoldered pins. You can clean the entire board after soldering.

Photo 2 shows the assembled board attached to a perfboard for testing. The perfboard will be used to add other parts for prototyping a circuit. It is also possible to attach two modules to one perfboard. For example, you might have a design that uses a QFP microcontroller and a QFP FPGA.

TESTING

Once all of the parts are soldered to the board, test it. This serves two purposes. One, you can make sure that the QFP part isn't damaged. Two, you can make sure all of the QFP pins are making connections to their respective pads and not shorted to each other. Testing consists of generating a known output from the QFP pins and checking it at the pin headers.

For the example boards, a simple program was written for the ATmega device and programmed using the ISP header. Pins on the ATmega microcontrollers can be programmed as inputs or outputs, as well as to perform other functions. The test program configures all of the pins (except reset) as outputs and then pulses a "one" on each pin, one at a time. The program loops forever. This enables you to check if the pins are working by just probing them with an oscilloscope to ensure that the positive output pulse occurs on the pin. Using two channels of an oscilloscope, you can probe adjacent pins to be sure the signals are different. (If they are the same, the pins are shorted together.) Usually, though, if you have two pins shorted, it will be obvious just by probing one of them.

The ATmega devices enable all of the pins to be programmed as outputs. Some devices have pins that are input only. For those pins, you could wrap one of the outputs back to that input and do something with it (drive another output pin when the input pin is low and more) to verify that the input





works correctly. You would make this connection on the perimeter connectors rather than make it a permanent connection on the PCB.

For an FPGA or CPLD, you could program the device as a large shift register, coupling all of the bits to the output pins and using an oscilloscope to test. Like the microcontroller case, if the device has some pins that are just inputs, you could connect one of the outputs back to the input and generate a second output from that pin to test if the input pin is working. If the device has multiple clock inputs, you may need more than one test program.

After you verify that nothing is broken and everything is connected, you can program your final application into the board and start testing with that. However, the simple test program will give you confidence that the part and the module are working correctly so you don't look in the wrong place for any problems you find.

PROTOTYPING WITH THE MODULE

The module is built with headers on 0.1″ centers that are soldered to the bottom of the board, around the perimeter. This enables you to plug the board into a standard perfboard with pins on 0.1″ centers for prototyping. You can insert the module pins directly into the perfboard for the lowest-profile solution, but this makes it difficult to reuse the module for a different application. However, for a permanently wired project, that may be the best solution.

An alternative approach is to use 0.025″ female sockets on the perfboard and plug the module into that. The advantage of this is that the module is removable, and can be changed if it is damaged, or it can be used on another breadboard. This makes the final assembly taller, but it also enables parts to be mounted on the perfboard under the module. You can actually mount components under the board even when sockets are not used, but they can't be tall. If you use the space under the module, avoid placing hot components (e.g., voltage regulators) in that space. Besides heating the QFP on the module, it may be difficult for heat to escape from under the socketed module.

Another approach is to use strip sockets and pins. These have pins that are like machine-pin IC sockets, and they give a lower profile while still allowing the module to be socketed. However, these pins are more fragile than the 0.025″ headers, so be careful when inserting and removing the module.

ALTERNATIVES

There are some off-the-shelf prototyping solutions. Schmartboard has various QFP prototype boards (www.schmartboard.com). They are designed to handle multiple sizes of QFPs, so they are typically larger than a board you design yourself. Bellin Dynamic Systems has QFP to DIP adapters, which are suitable for use with machine-pin socket strips, and let you prototype like we used to do with DIP packages (www.beldynsys.com). Finally, note that EZPrototypes (www.ezprototypes.com) has several QFP adapters.

All of these, and some others that are also available, will work. However, I like the modular approach because it is custom for the specific part and it enables me to put the bypass capacitors close to the device power pins. Plus, I can control the quality of the ground connections.

GO MODULAR

Prototyping with QFP parts is not easy, but sometimes you really need to use a high-pin-count device in a QFP or similar package. This modular approach minimizes the difficulty and provides a way to customize your QFP design or development system for your applications, while remaining generic enough to permit the same board design to be used in many projects.

Stuart Ball (stuart@stuartball.com) is a registered professional engineer with a BSEE and over 20 years of experience in electronics design. He is currently an engineering manager at Seagate Technologies.

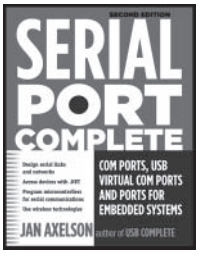
SOURCE

ATmega128, ATmega325, and ATmega645 Microcontrollers
Atmel Corp.
www.atmel.com



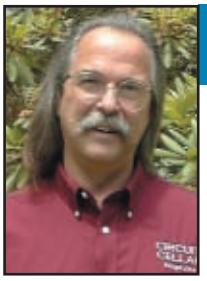


THE SERIAL PORT LIVES!



Everything you need to know about COM ports, USB virtual COM ports, & asynchronous serial ports for embedded systems.
Hardware & software for RS-232 & RS-485. Wireless options and more.

Serial Port Complete
Second Edition
Jan Axelson
ISBN 978-1-931448-06-2 $39.95
Lakeview Research LLC www.Lvr.com
From the author of USB Complete



by Jeff Bachiochi

Fingerprint Data Acquisition

Print Sensing with Embedded Technology

Identifying and matching fingerprints is a tedious process that involves huge databases. Today, a few companies are using embedded microcontrollers to create stand-alone print-sensing systems. Jeff examines the latest fingerprint security technology from Atmel.

I thought the popular television show *CSI: Crime Scene Investigation* would have run its course by now. However, even with two spin-offs, it still isn't losing its appeal. Because I stare at a monitor all day long, I try to limit my evening's radiation intake to *CSI* (or an occasional episode of *MythBusters* or *American Chopper*). Each high-tech *CSI* lab (Las Vegas, Miami, and New York) uses virtual screens and some of the newest sensors for analyzing smells, sounds, chemicals, DNA, and bodies. Databanks are available for almost everything imaginable: motor vehicles, military personnel, criminals, glass, fabrics, feathers, and more. One of my favorites is the fingerprint databank—or should I say "databanks"?

In the real world, there is not a sole repository of prints at this point in time. Thus, it can be a challenge for crime fighters to find matches. But as the United States and other countries become more closely linked, shared databases will reduce this to a nonissue.

The fingerprint has been studied throughout history. First, it was analyzed on clay tablets and seals. Later, it was studied on paper documents. During the 1680s Marcello Malpighi—an Italian anatomist and professor at the University of Bologna in Italy—may have been the first to discuss the make-up of the fingerprint. However, it wasn't until the mid- to late-1800s that fingerprint styles were classified and recognized as unique. Prior to using fingerprints (as a means of identification), Alphonse Bertillon—a French law enforcement officer and biometrics researcher who created

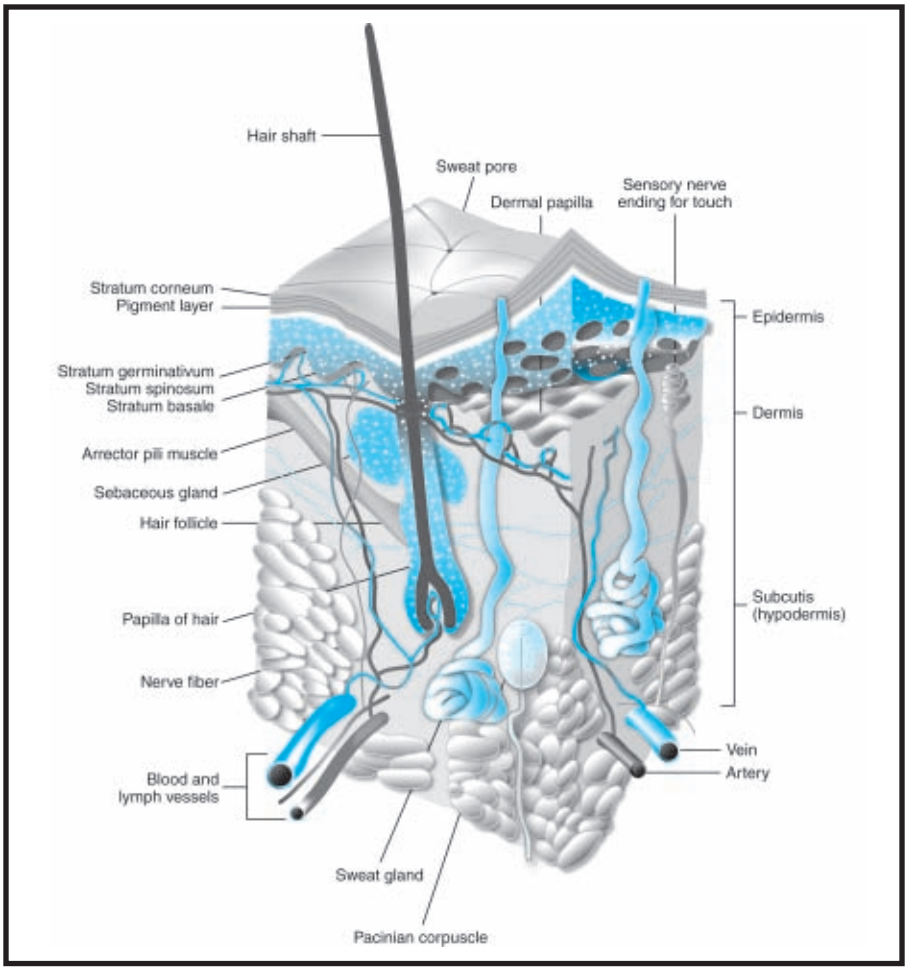


Figure 1—The dermal papilla or upper-most layer of the dermis is responsible for the ridge patterns known as fingerprints. These patterns, formed randomly before birth, remain the same throughout life. (Source: Wikipedia, http://en.wikipedia.org/wiki/Image:Skin.jpg)

anthropometry—devised a system of (adult) identification that compares measurements of certain body dimensions. It isn't difficult to see why this idea was not infallible. Just prior to the 1900s, fingerprints were used to prove the identities of murderers.

We now know that the odds of two fingerprints being the same is about one in 64 billion. (There are about 6 to 7 billion people alive today.) A 1921 Act of Congress established the Identification Division of the FBI. By 1950, the FBI had manually collected 100 million fingerprint cards. Today, the use of an automated fingerprint identification system (AFIS) enables the digitized collection of prints and database searches. Are the writer's of *CSI* telling us the real story, or is what we see science fiction?

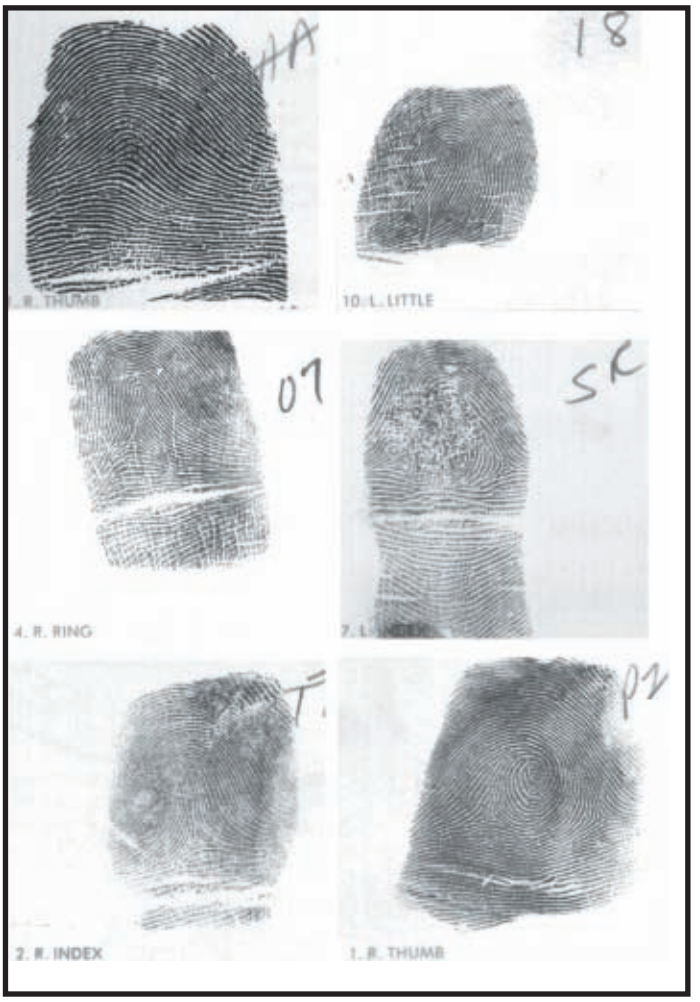


Photo 1—While each fingerprint can be categorized as an arch, loop, or whorl, many prints are combinations of these three main types. Here are samples of arch, left loop, right loop, scar, tented arch, and whorl (left to right, top to bottom). (Source: National Institute of Standards and Technology, www.itl.nist.gov/iad/894.03/nigos/NBIS/nbis_non_export_control.pdf)

ANATOMY OF A PRINT

In biology class, you learned about the dermis and epidermis, the body's outer protective layers. The dermis contains hair follicles, sweat glands, sebaceous glands, apocrine glands, and blood vessels. The dermal papilla, or the finger-like outer surface of the dermal layer, helps cushion and bond to the epidermis layer (see Figure 1). The epidermis, which doesn't contain blood vessels, receives nutrients by diffusion from the dermal layer. Hairless areas of the body (palms, fingers, soles, and toes) are designed for rough activities. The papillae in these areas give the epidermis its bumpy friction ridges that we call fingerprints.

The ridge pattern begins to develop in the dermis during the first trimester of pregnancy. Random factors during growth allow even identical twins to develop different dermal papilla. Because the ridge pattern is not a function of the epidermis, if the epidermis is damaged or scarred, once healed, the pattern takes on its original form. It is thought that these ridges aid the hands and feet by providing a less slippery surface.

A fingerprint is an impression made by the ridge pattern of the finger, palm, toe, or sole. Of all of the ridge patterns extending over your hands and feet, your finger tips (and toes) have the most unique properties. A ridge can have only a few distinct characteristics. It can be a segment with a beginning and an end or it can be continuous with no beginning or end. It can also split into two ridges (bifurcation), or two ridges can converge into a single ridge, depending on how you look at it.

Using these ridge characteristics, there are three main classification patterns: the arch, the loop, and the whorl. Each fingerprint can have one or a combination of these patterns. Photo 1 shows six plain and complex patterns.

LEAVING YOUR MARK

Fingerprints can be left behind in one of three ways. One, every time you touch something you leave behind natural body secretions. These secretions leave invisible (latent print) ridge details on the item. Two, when you have dirty hands, the details might actually be visible (patent print). Three, if the item you touch is soft, like wax, the ridge details may be left by deforming the surface of the material (plastic print).

Latent prints are invisible, so an investigator must find a way to make it visible. The method will depend on the material being explored. An image might become visible optically via diffusion, luminescence, absorption, or reflection. You can also use chemicals to create a reaction. Changes in color will mark the results.

The oldest method is one of the simplest. You can lightly brush a colored powder on a surface. The powder then sticks to the secretions.

Whether the print is plastic, patent, or latent, the idea is to make it stand out from the surface color. Doing so enables you to photograph the print and use it for comparison.

MINUTIA

Comparing a fingerprint to prints on file is a tedious process. You can skip those that have different characteristics, but how do you determine if the found "plain loop" print is the same as all of the other "plain loop" prints on record? You must compare smaller features or minutia. Earlier I discussed ridge characteristics. If the

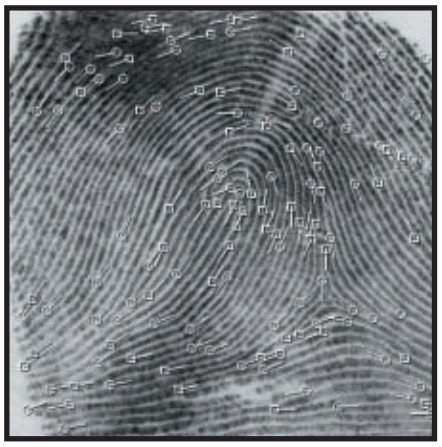


Photo 2—This image of a plain loop print has its detected minutia labeled. A circle identifies ridge ends, while bifurcations (or ridge splits) are identified by squares. (Source: NIST, www.itl.nist.gov/iad/894.03/nigos/NBIS/nbis_non_export_control.pdf)

features were located and identified for a print, then any matching print would need to have a list of identical features.

In order for this to work, the scale of any print must be known. The American National Standards Institute (ANSI)/National Institute of Standards and Technology (NIST) standard unit is 0.01 mm (one unit = 0.01 mm). A point in the lower-left corner becomes coordinate 1, 1. Each minutia point is given an index number with an associating coordinate (x, y), orientation (degrees), and type. The type in most cases is either ridge ending (A) or bifurcation (B). The orientation of all minutia is measured in degrees counterclockwise from horizontal (three o'clock) to a handle of the minutia. In a type-A minutia, the handle begins at the ridge endpoint and proceeds away from the endpoint along the ridge. In a type-B minutia, the handle begins at the split and proceeds between the two splitting ridges. Photo 2 shows a print with identified minutia points.

If the print image colors are reversed, ridges turn into valleys. Thus, a type A turns into a type B (and vice versa). It's all in your point of view.

A print can be rotated by adding an angle value to each of the indexed orientations. Although the physical X, Y coordinates change, the important information (distance between minutia) remains unchanged and can still be calculated using the original coordinates.

PRINT CARD

Many occupations require fingerprint records. While you may consider this a breach of your civil rights, companies requiring it consider it a form of protection for them. Identification through fingerprints is only as good as the database. It takes a trained eye and a lot of time to compare two prints and determine that they are indeed identical. The classification of prints helps narrow down which prints have to be compared. However, with potentially millions of print cards, this becomes an almost impossible feat.

Several decades ago, the tedious manual searching process consumed a lot of time and required vast amounts of manpower. As a result, fingerprint comparisons were used for confirmations between knowns, rather than all out searches for unknowns.

In the late 1970s, the FBI began using the computer to capture, store, and search fingerprint records. However, it wasn't until the early 1990s that the Integrated Automated Fingerprint Identification System (IAFIS) was developed (along with the NIST) as a way to exchange fingerprint data between agencies. A seven-point application package is available from the Department of Homeland Security (DHS), the NIST, and the FBI that is the culmination of 10 years of work. It contains utilities (in C) to segregate individual fingerprints from a four-finger slap (one-time impression), classify a print, detect minutia, establish a quality factor, compare for match, handle data files, and manipulate images. For more information, visit the NIST's web site (http://fingerprint.nist.gov/NFIS).

FINGERPRINT SENSORS

Many companies provide packaged systems for fingerprint recognition. While most can be used to obtain a print and pick out minutia, the systems are basically used with local databases for security verification. They may use a variety of methods for obtaining a live print. Optical devices use light absorption or scattering to capture finger images. Thermal devices sense heat from the finger to capture an image pattern. Capacitive devices sense an electrical charge from ridge lines to the surface of a sensor. RF, or acoustic signals, look beyond the outer layer of the skin to construct an image pattern. Each has its advantages and drawbacks.

There are a number of high computational functions that you must run on an image (collect the image, analyze for minutia, and compare minutia for a match). You need much more than a sensor to make a useful device. Simple sensors generally have a USB (or equivalent) connection and require a PC's computational power to handle the real functions. That's hardly cost-effective for an embedded project.

Several companies have applied sensor technology to an embedded microcontroller to create a stand-alone system. After you stop gagging on some of the pricing, you might end up looking at Atmel's AT77C104B sensor chip and the AT91SAM7FP105 FingerChip security processor (FSP). These are used in the Atmel AT77DK100 development kit.

A PICTURE IS WORTH 1,000 WORDS

Well, actually, in this case, a 232 × 232 pixel image is worth approximately 13,456 words. But who is counting? To obtain that image, Atmel developed the AT77C104B fingerprint sensor, which is based on FingerChip thermal technology. It is a linear sensor that captures fingerprint images by sweeping the finger over the sensing



Photo 3—The Atmel AT77C104B thermal sensor's hard protective coating is weatherproof and can withstand more than 1 million swipes. (Source: Atmel, www.atmel.com.

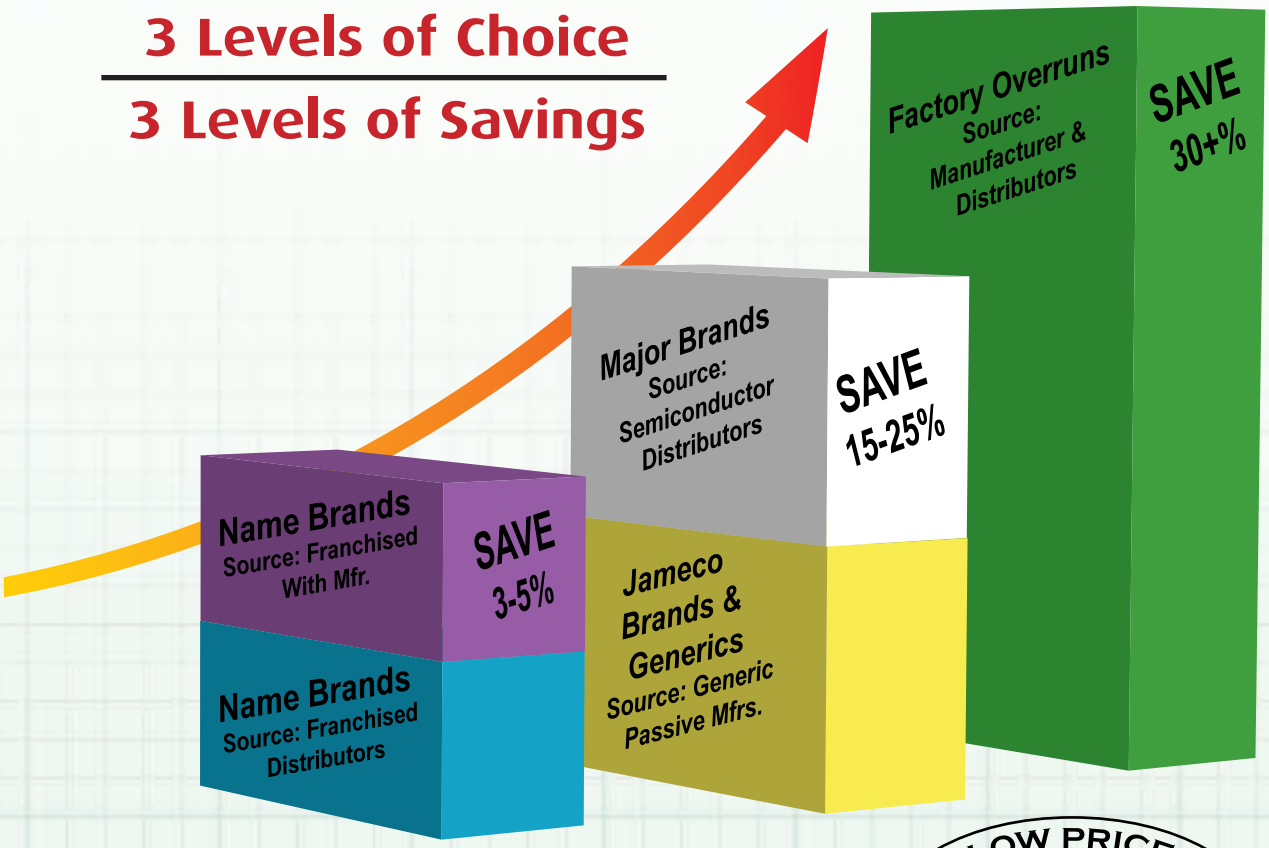
Exclusively at Jameco... The Lowest Prices... Guaranteed!

Save 3% to 5%
Looking to save on thousands of Name Brand components? Jameco offers their popular products for 3% to 5% below market price, and we guarantee it!

Save 15%-25%
Save more when a specific manufacturer is not required. Major Brand semi's are sourced from 5-6 major manufacturers, plus we offer Jameco brand and generic passives for even greater savings.

Save 30% or more!
Looking for killer deals? Jameco buyers often find Factory Overruns from some of the industry's biggest names. That means savings of 30% and more to you!

3 Levels of Choice
3 Levels of Savings



Other Jameco Advantages:

- We offer over 300 Major Manufacturers
- ***99%*** of catalog products ship the same day.
- Lowest prices guaranteed, or we pay 10%.

LOW PRICE GUARANTEE

If you find a lower print-advertised price for a product we offer, just let us know. We'll offer you that product for 10% less than the price you found, right down to our cost!

JAMECO® ELECTRONICS

Order 24 hours a day, 7 days a week

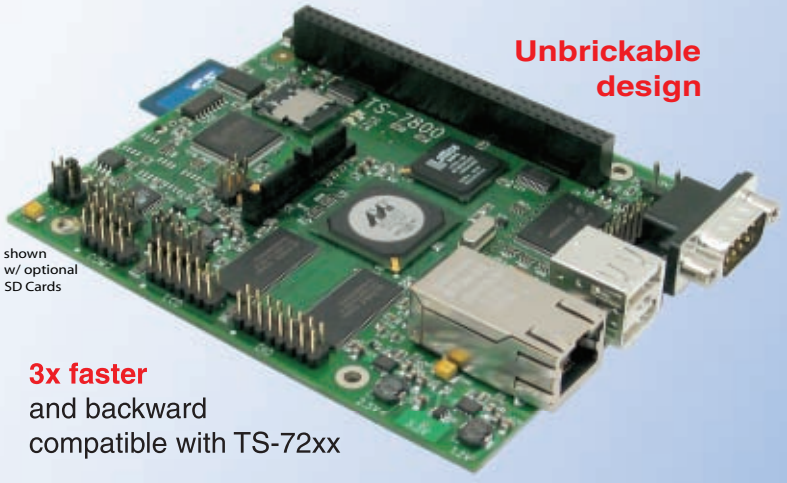
www.Jameco.com

Or call800-831-4242 anytime

SEMICONDUCTORS • PASSIVES • INTERCONNECTS • ELECTROMECHANICAL • POWER

Embedded Single Board Computers

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
- 12K LUT programmable FPGA
- Internal PCI Bus, PC/104 connector
- 2 host USB 2.0 480 Mbps
- Gigabit ethernet
- 2 SD sockets
- 10 serial ports
- 110 GPIO
- 5 ADC (10-bit)
- 2 SATA ports
- Sleep mode uses 200 microamps
- Boots Linux in < 2 seconds
- Linux 2.6 and Debian by default

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, USB WiFi

200 MHz ARM9
Power as low as 1/4 Watt

- 8 boards, over 2000 configurations — as low as $99 qty 100
- Fanless, no heat sink
- SDRAM - up to 128MB — $129 qty 1
- 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
- SD card option
- VGA video
- LCD ready
- Linux, Real Time extension, Debian

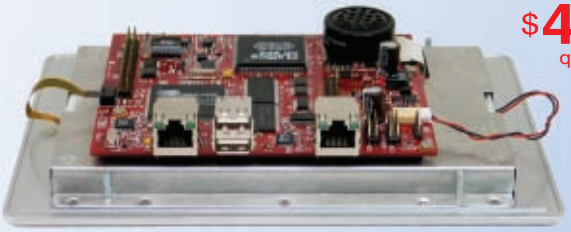
- Over 20 years in business
- Open Source Vision
- Never discontinued a product
- Engineers on Tech Support
- Custom configurations and designs w/ excellent pricing and turn-around time
- Most products stocked and available for next day shipping

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

Featured Products and PC/104 Peripherals

NEW!





TS-TPC-7390

200MHz ARM9 Touch Panel Computer

$449 qty 1

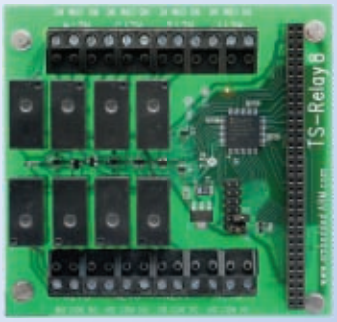
- Low Power, Industrial Quality Design
- Mountable aluminum frame
- 64MB SDRAM (128MB opt)
- 512MB Flash w/ Debian Linux
- Programmable FPGA- 5K LUT
- 7" Color TFT-LCD Touch-Screen
- 800x480 customizable video core
- Dedicated framebuffer- 8MB RAM
- Audio codec with speaker
- Boots Linux 2.6 in about 1 second
- Unbrickable, boots from SD or NAND
- Runs X Windows GUI applications
- Runs Eclipse IDE out-of-the-box

NEW!

TS-RELAY8

Eight Software Controlled Relays

- 8 SPDT relays
- Up to 277 VAC @ 5A
- Up to 30 VDC@ 5A
- Software controlled
- 40mA draw per coil
- I/O jumpers



$89 qty 1

NEW!

TS-ADC16

ADC, DAC and Digital I/O

- 16 16-bit ADCs
- Up to 100Ksps (10us)
- 1KB ADC RAM-FIFO
- 4 ADC voltage ranges
- Prog. pacing clock
- Externally triggered
- 4 inputs, 1 output
- 4 12-bit DACs
- 4 16 bit counters

$169 qty 1



see our website for x86 SBCs, peripherals and option details

Technologic Systems

We use our stuff.

Visit our TS-7800 powered website at

www.embeddedARM.com



Photo 4—This view of the FSP demo software shows you the image built from the sensor slices it received of my finger's swipe across the sensor. The image on the right has been digitized and shows the minutia it located in the image.

area (see Photo 3). The AT7710413 uses a SPI to ask for and retrieve data. Three main modes of operation are available: Standby, Pointing, and Acquisition. As you might expect, the Standby mode turns off some internal operations requiring less than 10 µA of current. The device can simulate the functions of a mouse by using the Pointing mode. In this mode, a finger scan is internally interrogated and using threshold data the user provides, it will calculate the x and y movement values. The Acquisition mode allows a slice (or frame) of data to be clocked out by the master. The slice consists of eight rows of 232 columns covering an area of 0.4 mm × 11.6 mm. Each pixel is a 4-bit representation of the amount of heat sensed over the integration period (the time between frame acquisitions). This means that clocking the pixel data must be done quickly at regular intervals.

Let's suppose you want to be able to accept a finger speed of approximately 8″/s. It is recommended that the host use a SPI clock speed of approximately 10 MHz (100 ns). At that rate, you can grab a frame in about 750 µs, have 250 µs in between slices for work, and get approximately 1,000 slices/s. At roughly 800 bytes/slice, 1,000 slices would be 800 KB. Whew, that's a lot of data! Luckily, you're producing meaningful data for only about one-tenth of that time. If you could process the slices as they came in, you would require only approximately 30 slices to produce a 12.6-mm square image. No mater how you slice it, this requires more data space than your average 8-bit microcontroller has available.



Photo 5—Once all of my fingers were enrolled, verified, and stored in the Atmel AT91SAM7FP105, it could identify each of my fingers while disregarding the fingers of others.

KEIL™
An ARM® Company
Software Development Tools
The Leader in Microcontroller Development Solutions
C/C++ Development Kit including best-in-class compilers, genuine Keil µVision®, and royalty-free RTX RTOS.
ULINK®2 Adapter for target debugging and Flash programming.
A/D
I/O Ports
Run-Control
Timers
Interrupts
Debug Channel
PWM
Flash ROM
CPU
UART
RAM
I²C/SPI
RTC
DMA
Ethernet
SD
USB
CAN
Keil RTOS and Middleware components are specifically optimized for embedded systems and include TCP/IP, Flash File system, USB and CAN support.
ARM
www.keil.com/arm
Cx51
www.keil.com/c51
C166
www.keil.com/c166
Out-of-the box support for more than 1,400 Microcontroller devices.
Call 1-800-348-8051 for a free demo CD.
www.keil.com

How can all of these slices be assembled into a single image? Think of this as taking a panoramic shot of the surrounding countryside from the top of a mountain. You would take a picture, rotate a bit, take another picture, and continue this until you went 360°. You might take 20 or so pictures. Back at home, with the pictures in hand, you could take a pair of scissors and trim off the overlapping parts on the left side of each picture. By gluing the trimmed side of a landscape on top of the next (matching) picture you can recreate the original scene based on recognizing the redundant parts of each picture. If you did this in Adobe Photoshop, you could save and print out the panoramic photo as one continuous image. You can begin to see the computational requirements of just collecting a fingerprint image using this technique, not to mention the other functions that need to be implemented to make a useable product.

IMAGE FILES A NO-NO

Although complete image files of fingerprints may be stored in large

databases for visual confirmation when used as evidence in a trial, this is unnecessary in many limited-security applications. As you know, it is print minutia that points out the features used to make comparisons. Therefore, all that's required for most comparison databases is the minutia list of each print.

Note: Most people will rebel against having their fingerprint stored or circulated, but a list of minutia doesn't seem to be as offensive, even though it contains enough information for identification purposes.

BLACK (AND WHITE) MAGIC

A useful fingerprint identification system will require some high-powered data shuffling. Atmel's advanced RISC machines (ARM) 32-bit microprocessors are up to the task. Integral USB, USART, and SPI ports give this microcontroller three possible communication media. Biometric routines handle slice and slice reconstruction, image and minutia extraction, template and record creation, and verification and identification. Because all functions (including the record database) are embedded in the hardware, the information is considered protected. To protect any data moving external to this microcontroller, data security can be enabled using the advanced encryption standard (AES) and 128-bit keys.

Although you won't be searching through a database of thousands of fingerprints, data collecting will be necessary. On-chip nonvolatile storage can hold up to 25 print records. The collection process is called enrollment. An enrollment record contains a template of the minutia gathered from a fingerprint's image, along with some data used to associate the record with the fingerprint image. It is a good idea to enroll a number of fingerprints from each person that will be using the system. You might choose to enroll the pointer finger from each hand, just in case you get a blister or some other surface damage. The enrollment process requires you to provide multiple swipes to assure that the minutia data created from a single swipe isn't abnormal.

Following the enrollment of a finger's print, a verification process is used to make sure that the record's template will match the appropriate finger's print by requiring the user to swipe their finger. There must be at least one enrollment record for each user who will require identification. Once this has been completed, all of the records are saved in nonvolatile memory for use (even after the power has failed). Database maintenance is also supported so you can easily delete or add records to the database.

Finally, you have the identification process. This is the normal mode where an application will look for a finger swipe, extract minutia, and search the database for records that match. The application can then determine if any action is required based on the search results.

All of this is done via one of the communication channels. Your application, whether it is on a PC or a microcontroller, uses the AT91SAM7FP105's built-in functions for support.

FSP DEMO

The AT77DK100 comes with a PC application that enables image viewing, enrolling, identification, and database management via the USB interface. I switched from raw to digitized images between the left and center images in Photo 4. The left image shows the fingerprint image built from the sensor's slice inputs. The center image shows a digitized image with the minutia identified. The print's template is made up of these minutia points and would be stored in a record if this were an enrollment and verification process.

Photo 5 shows the FSP matching a finger swipe to one of the 10 records I previously enrolled (one record for each finger of both hands). Let's say you want to be able to identify a member of your immediate family. You can use this PC program to enroll all of the users and save the records to the on-chip nonvolatile memory. The sensor system would then be ready to support your application. Your application might be used simply to enable a door-lock solenoid whenever a swipe matches one of the family member's prestored prints.

I hope your brain is beginning to churn out possibilities. Atmel has given designers a great peripheral to work with.

One last note: Whatever process you use to obtain a fingerprint image, you will most likely get slightly different results with each image. This means that the minutia extracted may be slightly different. If you are dealing with images of slightly different areas of the same print, the minutia will not be 100% identical. Comparing templates will therefore have some percentage of matching minutia that is less than 100%. Most likely, you are not just picking the database record with the highest percentage; otherwise, there would always be a match. It is just as important that wrong matches are rejected as it is that correct matches are accepted.

When you watch the *CSI* lab obtain a match on a latent print pulled from a crime scene, you can assume that there is a lot that they are not showing you. You must assume that once a computer has pulled up a potential match it is visually scrutinized by a professional before it is entered as evidence. The smaller the partial print, the fewer minutia, which can lead to a higher number of hits. And where does this leave investigators if the real print isn't among the database? Can you say "DNA"?

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

SOURCE

AT77C104B FingerChip Fingerprint sweep sensor, AT77DK100 FingerChip development kit, and AT91SAM7FP105 processor
Atmel Corp.
www.atmel.com



by Tom Cantrell

Embedder Than Ever

ESC Delivers the Goods

It's been a great year for embedded systems designers. You have plenty of new technologies and design options to consider. This month, Tom showcases the best and brightest on display at the 2008 Embedded Systems Conference in San Jose.

This year's Embedded Systems Conference (ESC) in San Jose, CA, marked the twentieth year of the affair. Of course, embedded applications go back more than 20 years, indeed all the way back to the dawn of silicon. Nevertheless, ESC has been invaluable in giving blue-collar technology the respect it deserves.

As with all things silicon, the challenge for designers remains the same. You've got to anticipate where technology is headed, paying special attention to inflection points where the merely "possible" becomes "practical." Today's bleeding edge is tomorrow's leading edge.

Let's check out some of the new gizmos and gadgets I came across at the show. Ready or not, they're coming to a design near you.

LIQUID CRYSTAL DELUGE

Just what is that "liquid crystal" goop they put in LCD panels? I get the concept of the polarizing shutters and such, but I haven't got a clue where you go to buy a jug of that stuff. Whoever's cooking this stuff up, I hope they've got a lot of it. With all of the LCD TVs, monitors, notebook PCs, cell phones, MP3 players, digital picture frames, and so on, half the tech world is already hooked. Now it's embedded's turn.

Anyone remember when PCs had a monochrome display option? Yeah, tossed it long ago. Anyone else have a drawer full of those old two- or four-line alphanumeric LCDs? Time to toss them too. Let's face it, we're in the era of eye candy. Like a red sports car or shot of Botox, a whizzy LCD can be just what the doctor ordered for an embedded design facing a midlife crisis.

Absent some kind of "Organization of LCD Exporting Countries," the explosion of LCD production means lower prices for small and mid-size bit-mapped LCDs. Just Google "surplus LCD" and you'll see what I mean. The other piece of the puzzle is the silicon to drive the display, where it's "Moore for less" business as usual.

As a designer, you've got the choice of designing your own LCD subsystem or buying a preintegrated module. For example, over at EarthLCD, you can find a wide variety of raw glass, including surplus 128 × 128 64K color cell phone displays for just $15 (www.earthlcd.com). They also offer more turnkey "ezLCD" modules starting at $149 that combine the display electronics and driver software with various screens (e.g., 2.7", 2.5", 5.6", and 8" models), options (e.g., a touchscreen) and popular interfaces (serial, parallel, and USB).

Ditto for Reach Technology who offers a line-up of boards, modules, and enclosed units (www.reachtech.com). At the show they were offering a 4.3" widescreen quarter VGA(WQVGA) (i.e., 480 × 272) development kit including a controller board (serial and



Photo 1—The latest generation of 32-bit MCUs has the horsepower it takes to drive high-resolution LCDs. Luminary Micro's Intelligent Display Module (IDM) highlights the opportunity to spiff up an embedded design with fancy graphics.

USB), power supply, cables, and driver software for $279.

Software is half the battle when it comes to crafting a whizzy GUI. Going beyond the typical "driver" software (i.e., graphics primitives like DOT, LINE, and CIRCLE), Amulet Technologies (www.amulettechnologies.com) pioneered the intriguing concept of using web page authoring tools (i.e., HTML). However, when I covered their initial offering back in 2001 ("Liquid Crystal Delight," *Circuit Cellar* 132), the technology could handle only monochrome displays. But now, thanks to an Atmel ARM7-based CAP chip (www.atmel.com), they've got custom-chip performance to handle full-color (i.e., 24-bit) displays up to SVGA (800 × 600) resolution. Indeed, the CAP chip not only does the heavy lifting graphics-wise, but also has plenty of cycles left over to run application code.

For those inclined to roll their own, there's no shortage of chip suppliers willing to give you a hand. Thirty-two-bit MCU upstart Luminary Micro (www.luminarymicro.com) continues their impressive ARM Cortex-M3 MCU roll-out (right now their web site shows more than 100 different parts) with an Intelligent Display reference design kit (see Photo 1). Based on their LM3S6918 MCU (50 MHz, 256-KB flash memory, and 64-KB RAM), the kit includes a 64K color QVGA touchscreen display, Ethernet connectivity, a power supply (including a Power-over-Ethernet option), cables, and a debug adapter for just $219.

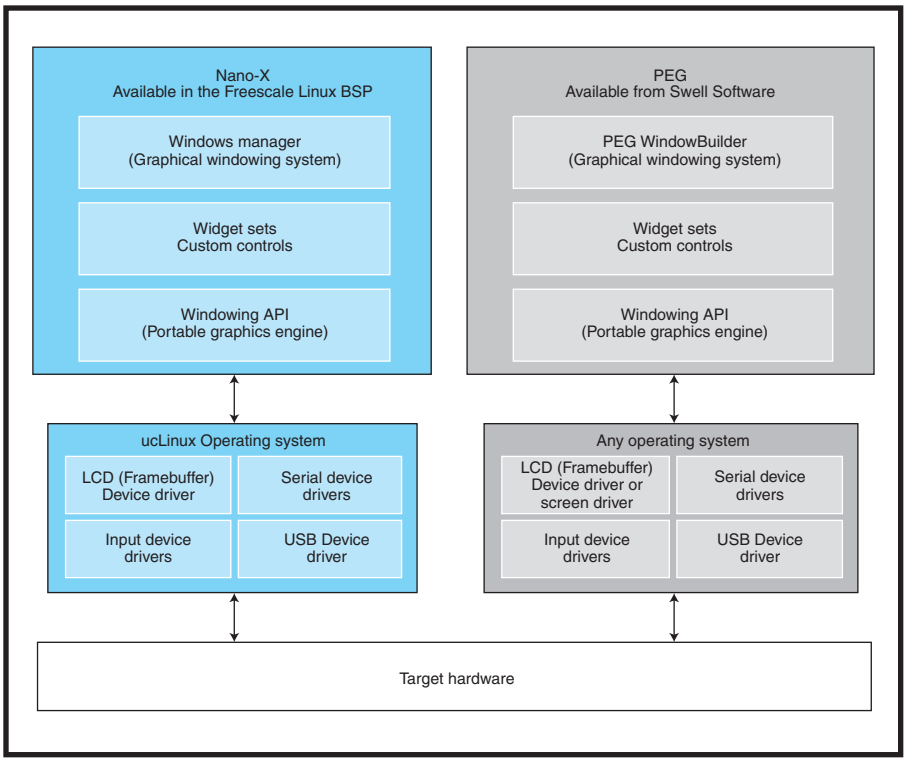


Figure 1—To OS or not to OS, that is the question. And it's your choice with two graphics software solutions for the Freescale Semiconductors MCF5227x LCD-enabled ColdFire MCU. There's a free Nano-X library that works with µCLinux and a commercial suite (PEG from Swell Software) that works with any (or no) OS.

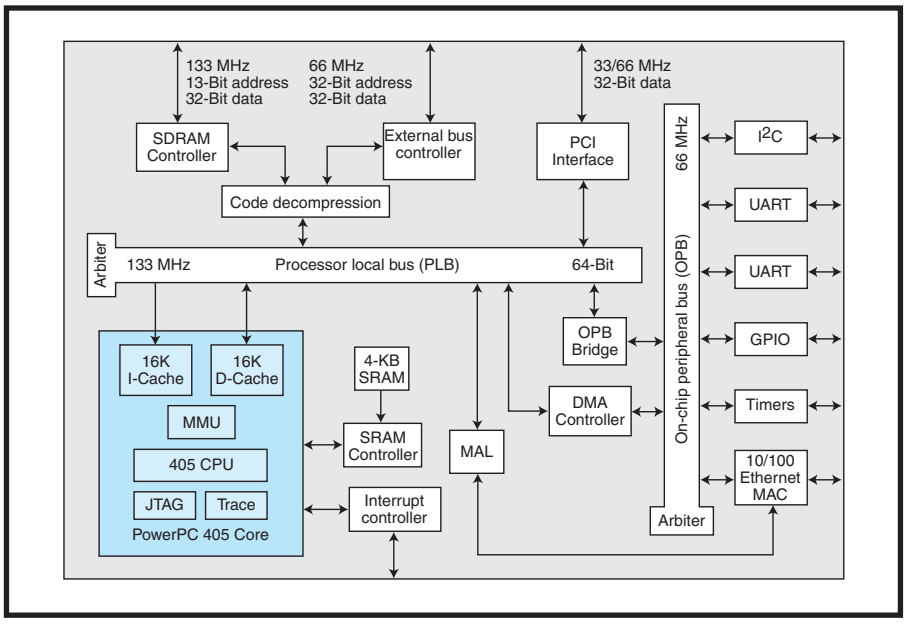


Figure 2—On paper, PowerPC processors like this PPC405 from AMCC look a lot like 32-bit flash MCUs based on ARM and MIPS architectures. Just lose a few pounds (note the code decompression) and toss in a bunch of flash memory and it would be good to go for single-chip applications.

Recently spun out from Philips, NXP Semiconductors (www.nxp.com) has been successful with their practical line-up of ARM7-based, 32-bit flash memory MCUs. Now free to fly, NXP wasted no time expediting a move into higher-end applications by acquiring Sharp Electronics' "Blue Streak" catalog. In one fell swoop, the deal gives NXP an entire line of high-performance ARM9 chips well-suited for LCD applications, not to mention an established customer base to grow from.

Not to be outdone, Freescale Semiconductor extends their venerable ColdFire/68K line-up with the MCF5227*x*, their first ColdFire chip to include an integrated SVGA LCD touchscreen controller (www.freescale.com). The LCD won't be lonely, because the chip has plenty of other transistors to keep it company, including USB, CAN, serial of all flavors (UART, I²C, and SPI), 128 KB of SRAM, DDR SDRAM interface, and more. Running at up to 160 MHz, the '5227*x* has the horsepower to handle some heavy-duty operating system and graphics software (see Figure 1).

The message for embedded designers is to go graphic—and the glitzier, the better—or go home.

32-BIT ON THE MARCH

As much as I've been writing about 32-bit MCUs over the past few years, you'd think the story must be over.

Yes, there's no doubt that ARM has been in the spotlight. Thanks to an "open" architecture and compelling chip offerings from premier MCU suppliers (e.g., NXP, Atmel, STMicroelectronics, and Luminary), ARM has been on quite a roll in the mass-market 32-bit MCU biz.



Photo 2—One step closer in the quest for tone, courtesy of the Coyote-1 stompbox and plenty of Parallax Propeller multicore MIPS. The Coyote-1 serves as an open-ended platform for a virtually unlimited repertoire of programmable effects (www.openstomp.com).

The march of good news for ARM continued at the ESC with announcements from both NXP and Atmel of plans to license the Cortex-M3, notably an upcoming revision of the core freshened with improved power management. There's even a new MCU startup, Energy Micro, wrapping their business plan around the Cortex-M3 as a way to deliver "the world's most energy friendly microcontrollers" in 2009 (www.energymicro.com).

As you may recall, herding the cats (i.e., existing ARM7/9 licensees) to the new M3 architecture wasn't easy. Indeed, it might not have happened at all without a kick-start from Luminary, though even then the situation remained fragile. But now, with all of the heavyweights onboard, there's no doubt that the Cortex-M3 is the way the ARM MCU wind is blowing. Nevertheless, it will be interesting to see how suppliers and designers finesse the "upgrade path" considering the ongoing demand for existing ARM7/9-based MCUs.

ARM Wins. End of story, right? Well, it ain't over until it's over—and it ain't over!

Leave it to Microchip Technology, one of the few companies strong enough to challenge ARM's plans for MCU domination, to do just that. The PIC32 represents worthy competition, but the importance of Microchip's move goes beyond the chip itself. In fact, it's a reminder that the 32-bit MCU story isn't over, it's just beginning.

Ever heard of Renesas Technology (www.renesas.com)? They have a lot of great chips. And, oh yeah, they happen to be the number one global MCU supplier.

How about Freescale? Boardroom distractions aside, they ship a heck of a lot of chips every day.

Could there be other shoes to drop? Maybe what the world needs is a PowerPC MCU? Freescale and AMCC (www.amcc.com) would be likely candidates (see Figure 2), and maybe even Apple, who recently bought boutique PowerPC chip supplier P.A. Semi, could have a hand in it. Sure the architecture might have to go on a diet to better fit in tighter apps, but so did ARM and MIPS with their code compression schemes (Thumb and MIPS-16 respectively). Surely a company with the technical might of IBM is capable of having some good ideas. Indeed, given their heritage of research and development leadership, chances are they've already had

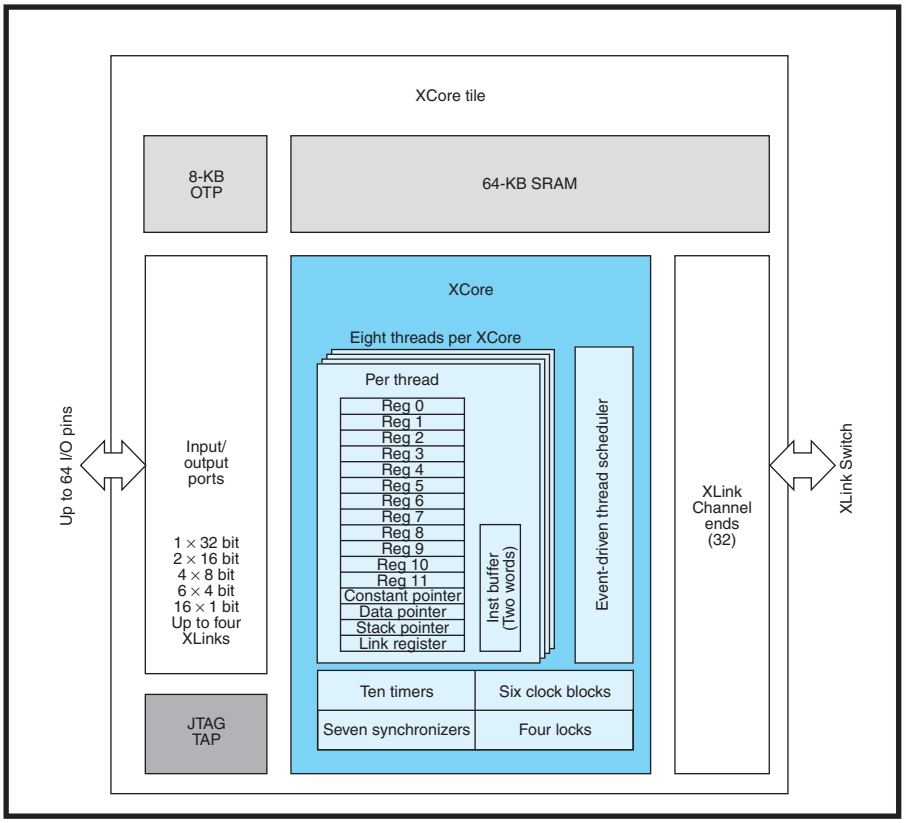


Figure 3—XMOS now says the time and silicon are right to carry forward multicore techniques (i.e., extensibly linked 32-bit processors) first pioneered in the 1980s-era Transputer.

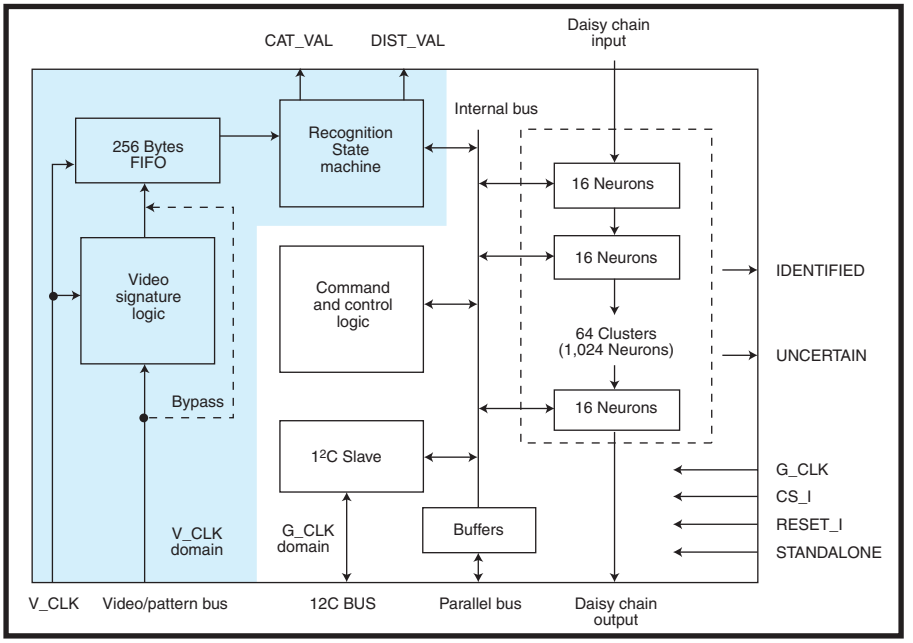


Figure 4—Recognetics says silicon should work smarter, not just harder—and for certain applications (e.g., real-time pattern recognition), their CM-1K neural network does just that.

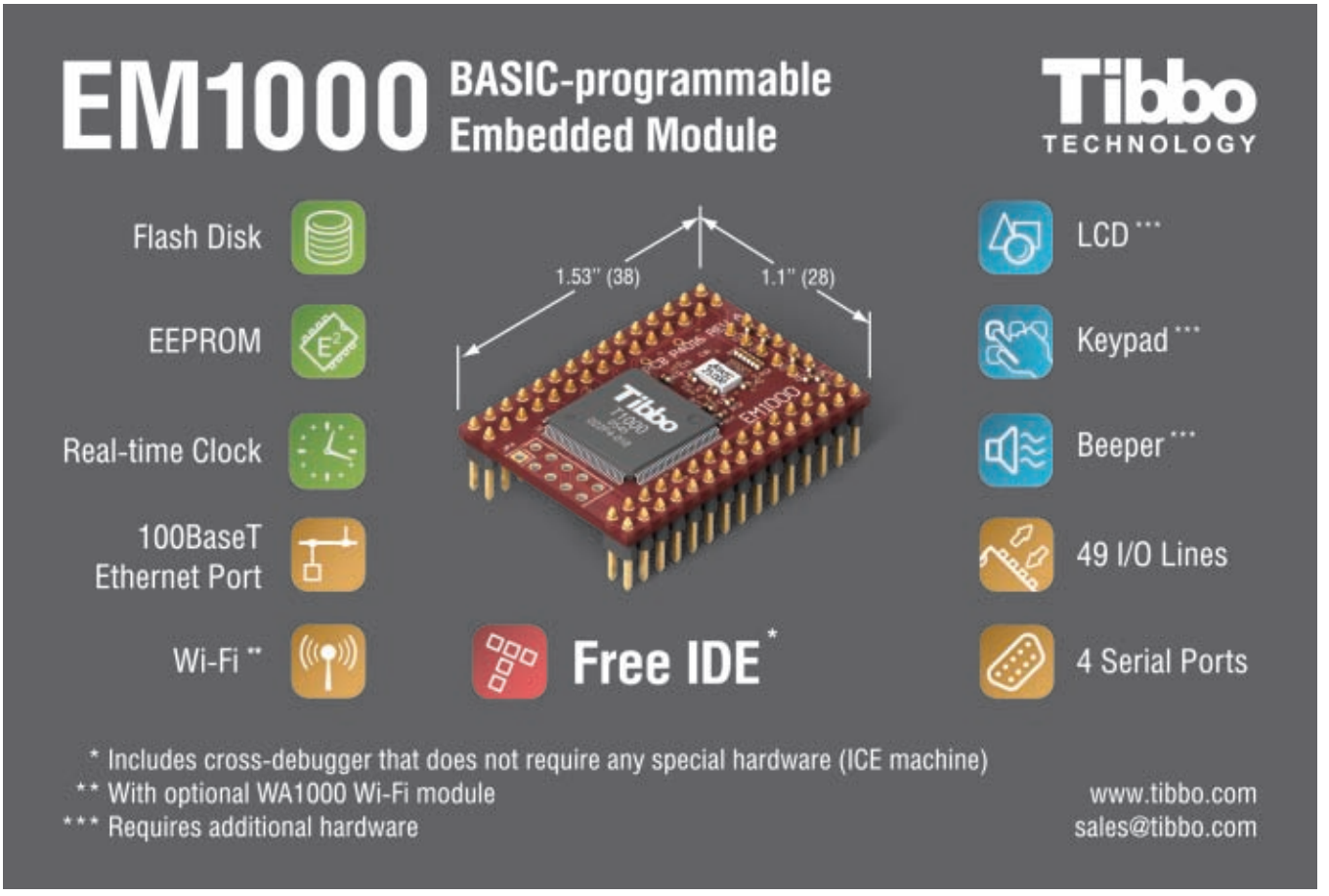
them.[1]

CORE WARS

The proliferation of 32-bit MCUs is a good example of the way yesterday's bleeding-edge technology becomes mainstream tomorrow. Right now multicore processors are state-of-the-art and all the rage for PCs and video games. Presumably, that means it's only a matter of time before they find their way into embedded applications.

Actually, it's already happening courtesy of Parallax and their Propeller chip, the multicore for the masses (www.parallax.com). Parallel programming isn't a new concept, but it's a truism that nothing happens until a chip gets shipped. While experts debate the subject, Parallax is shipping Propeller chips, software is getting written, and cool applications are real (see Photo 2).

By the time you read this, startup XMOS Semiconductor should be shipping their first chip (www.xmos.com). The XS1-G is a four-core unit designed to craft what the company calls "software defined silicon" using "C" with parallel extensions. As shown in Figure 3, each core (or "tile" as XMOS calls them) is a 400-MHz, 32-bit processor with 64 KB of SRAM and 8 KB of OTP. Each tile has hardware support for up to eight threads and access to up to 64 pins with autonomous I/O timing capability (i.e., time-stamped inputs and scheduled outputs). In turn, the processing elements (tiles and threads)



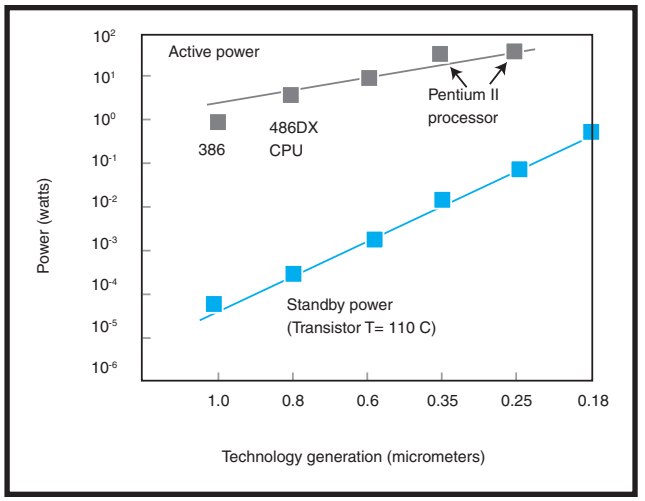


Figure 5—MCU suppliers should think twice before marching like lemmings to the latest-generation fab processes. Standby power consumption is the problem because there's a lot of current leakage going on.[3] Short of stocking up on Depend underwear, the best way to avoid leakage problems is to hang back on the process curve.

communicate via programmable XLINK connections that provide deterministic connection at 1 Gbps on-chip. Furthermore, XLINK connections can cross chip boundaries, enabling designers to bring multiple chips to bear on larger applications.

If this all seems vaguely familiar, it's because the founders of XMOS are many of the same folks who brought you the Transputer back in the 1980s. That chip was based on many of the same worthy concepts, but was simply too far ahead of its time both in terms of silicon and customer interest. Now the silicon is up to snuff and everyone "gets it" when it comes to multicore. Once more, with feeling!

Refer to Figure 4 for another back-to-the-future chip, the CM-1K neural-network from Recognetics (www.recognetics.com). With 1,024 neurons working in parallel, the chip offers some astounding pattern-recognition capabilities. It is said by the company to be nearly 100 times faster than a 300-MHz DSP. Although this isn't the first time AI concepts have found their way into silicon, maybe this time it's déjà vu, not deja-voodoo.

LITTLE CHIPS THAT COULD

So with all of the talk about 32-bit MCUs and multicore, you might be thinking mere 8-/16-bit chips are dead. Well, I've heard folks giving that eulogy for many years, and it's as wrong today as it's ever been.

Indeed, I had to smile when I peeked under the hood of the NS9215 NET+ARM chip from Digi International (www.digi.com). Inside you'll find the not unexpected ARM9 core with an Ethernet MAC and the usual I/O suspects (timers, serial ports, ADC, and more). Oh yeah, it also includes two "Flexible Interface Modules" (i.e., programmable I/O processors), which just happen to be "DRPIC 165*XX*" 8-bit MCUs. It's 32-bit, 8-bit, and multicore all in one chip!

Thirty-two-bit MCUs are a blessing when you need lots of headroom, a





BitScope USB Mixed Signal Oscilloscope

Analog + Digital

Digital Storage Oscilloscope
✓ Dual Channel Digital Scope with industry standard probes or POD connected analog inputs. Fully opto-isolated.

Mixed Signal Oscilloscope
✓ Capture and display analog and logic signals together with sophisticated cross-triggers for precise analog/logic timing.

Multi-Band Spectrum Analyzer
✓ Display analog waveforms and their spectra simultaneously. Base-band or RF displays with variable bandwidth control.

Multi-Channel Logic Analyzer
✓ Eight logic/trigger channels with event capture to 25nS.

DSP Waveform Generator
✓ Optional flash programmable DSP based function generator. Operates concurrently with waveform and logic capture.

Mixed Signal Data Recorder
✓ Record to disk anything BitScope can capture. Supports on-screen waveform replay and export.

User Programmable Tools and Drivers
✓ Use supplied drivers and interfaces to build custom test and measurement and data acquisition solutions.

Inventing the future requires a lot of test gear...

...or a BitScope



BS100U Mixed Signal Storage Scope & Analyzer

Innovations in modern electronics engineering are leading the new wave of inventions that promise clean and energy efficient technologies that will change the way we live.

It's a sophisticated world mixing digital logic, complex analog signals and high speed events. To make sense of it all you need to see exactly what's going on in real-time.

BS100U combines analog and digital capture and analysis in one cost effective test and measurement package to give you the tools you need to navigate this exciting new frontier.



Standard 1M/20pF BNC inputs

Smart POD Connector



Opto-isolated USB 2.0

12VDC with low power modes

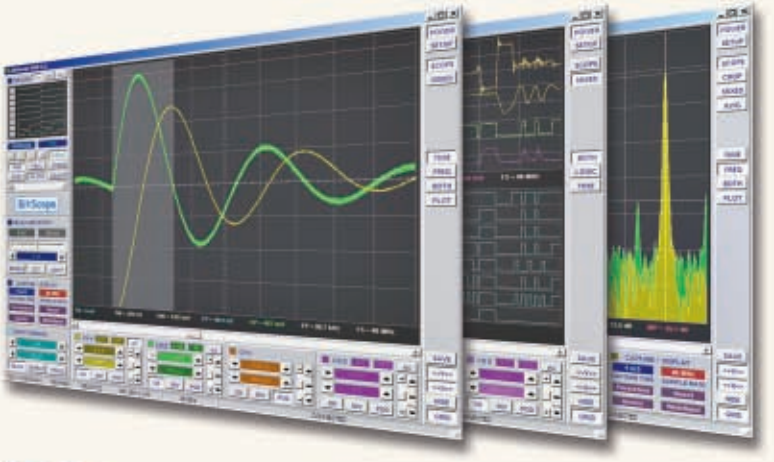
BitScope DSO Software for Windows and Linux

BS100U includes BitScope DSO the fast and intuitive multichannel test and measurement software for your PC or notebook.

Capture deep buffer one-shots, display waveforms and spectra real-time or capture mixed signal data to disk. Comprehensive integration means you can view analog and logic signals in many different ways all at the click of a button.

The software may also 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 another PC as if a BS100U was locally connected.





www.bitscope.com

EARLY BIRD SPECIAL!
$200 off
Extended thru Sept. 19
use code: ARMEBM2

find yourself [HERE]



REGISTER NOW www.arm.com/developersconference

drive your business [FORWARD]

A full 3-days of priceless information and contacts you can't afford to miss if you're currently designing or evaluating an ARM Processor-based application!

Take advantage of the only design conference dedicated to a single architecture - ARM. Communicate, Connect and Create together with the largest partner community in the industry.

> For engineers designing products with ARM Processor-based devices
> 3 days intensive training - over 100 track sessions
> Exhibitions - industry leaders & ARM Connected Community members supporting ARM products
> Forums and special analyst presentations on industry trends

October 7–9, 2008
Santa Clara Convention Center
Santa Clara, CA USA

ARM | 2008
Developers' Conference

Exhibitors / Sponsors as of August 1st:

Platinum | Gold | Silver

cadence ATMEL MAGMA NXP Actel American Arium pls prolific Quadros symbian SYNOPSYS

Alchip Technologies, Inc. Arrow Electronics Carbon Design Systems, Inc. CodeSourcery, Inc. Common Platform Coverity Coware, Inc. EVE USA, Inc. Fraunhofer Institute for Integrated Circuits Green Hills Software IAR Systems Lauterbach Luminary Micro, Inc. Macraigor Systems LLC Mentor Graphics Corporation Micrium Micro Digital, Inc. Open Kernel Labs Signum Systems STMicrosystems Texas Instruments TRANGO Virtual Processors VaST Systems Technology videantis GmbH VirtualLogix VisualOn Inc. Wipro Zilog

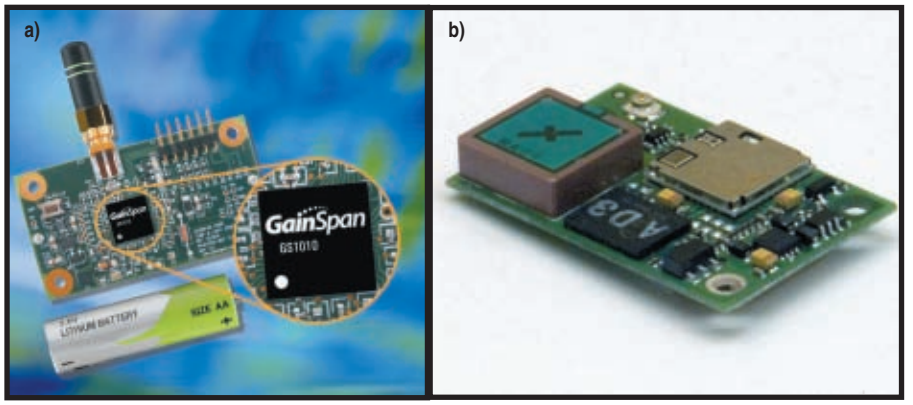


Photo 3—"Standards? We've already got stinkin' standards." So say outfits like GainSpan **(a)** and Laird Technologies **(b)** who see wireless sensor applications hitching a ride on the already pervasive installed base of Wi-Fi networks.

classic example being the graphics LCD GUI applications discussed earlier. Nevertheless, 8-/16-bit chips still have legs and, according to the market stats I've seen, demand is higher than ever. We're talking about 8-/16-bit MCU production volume heading towards a million units an hour, if it isn't there already. Our little friends may not get many headlines, but there are still plenty of sockets to fill. And there are some wonderful new 8-/16-bit chips to fill them.

No topic is more timely than "green," and 8-/16-bit chips are as green as can be. Using generally less advanced manufacturing processes means less leakage and lower standby power (see Figure 5). The processes are further optimized for real-world designs with features like extended temperature range, robust flash memory, and precision analog.

Consider the latest entry in Texas Instruments's popular MSP430 line-up, the MSP4305*xx* (www.ti.com). Of course, it's faster (up to 25 MHz) and improved with more memory (up to 256 KB of flash memory and 16 KB of SRAM), flexible clock options and pin configuration (pull-up, drive strength), comprehensive analog capability (ADC, DAC, op-amps, comparators), and even a 32-bit multiply instruction.

But the thing that caught my eye was '5*xx* power specs. Running on anything between 1.8 and 3.6 V, power consumption is just 220 µA/MHz across the entire frequency range. You can squeeze a few 16-bit MIPS from less than 1 mA! Or burn rubber at 25 MHz for just 5 mA plus change.

Standby power is the other part of the "green" equation, and TI is paying attention like everyone else. For the '5*xx*, you're looking at just 1.5 µA, and that covers not just state retention and wake-up capability, but also running the real-time clock.

Data Acquisition & Control Computer
iPac 9302
• ARM9 200 Mhz Processor
• Floating Point Math Engine
• 2 USB 2.0 Host Ports
• SD/MMC Flash Disk Slot
• 48 Digital GPIO Lines
• 10/100 Base-T Ethernet port
• 5 channels of 12 bit A/D & 2 PWMs
• 1 RS232 & 1 RS232/422/485 Serial Port
• Battery Backed Real Time clock/calendar
• OpenEmbedded Based Linux Distribution
• Eclipse Linux Development Environment
.net Micro Framework
Microsoft Windows CE
2.6 Kernel
The iPac has enough I/O for demanding applications & with a size of 3.5" x 3.8" it can fit almost anywhere. Prices start at $150.00. For more information please visit our website:
www.emacinc.com/sbc_microcontrollers/ipac_9302.htm
Since 1985 OVER 23 YEARS OF SINGLE BOARD SOLUTIONS
EMAC, inc.
EQUIPMENT MONITOR AND CONTROL
Phone: (618) 529-4525 • Fax: (618) 457-0110 • Web: www.emacinc.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

XMEGA sounds like the name of a comic book superhero. But with 32-MHz performance and up to 384 KB of flash memory, the latest addition to the Atmel AVR line-up is no joke. As with the TI parts, power consumption is remarkably low at about 300 µA/MHz (e.g., 3.6 mA at 12 MHz). The XMEGA standby power specs are even more impressive with the lowest of low-power modes (RAM retention) consuming a mere 100 nA! Go ahead and throw in power-down extras (like the RTC and brown-out detection) and you're still looking at just a couple of microamps.



Photo 4—"Standards? We don't need no stinkin' standards." This tiny wireless module mates ANT's proprietary (low-power and flexible topology) protocol with a custom 2.4-GHz radio from Nordic Semiconductor.

Even high-performance MCUs can be brought to their knees by the demands of I/O and interrupts. By offloading the I/O chores, DMA can drastically cut processing overhead and increase overall throughput. A good DMAC is like getting a bunch of extra processor MIPS for free.

Both TI and Atmel include DMACs, and Atmel takes the concept a step further with programmable "Event Controller" hardware. Thanks to dedicated peripheral-to-peripheral connections, I/O transactions can take place autonomously with zero CPU overhead or interference. At Atmel's ESC booth, they showed off a demo of an XMEGA doing streaming audio with the CPU asleep.

It's a good thing that XMEGA can handle I/O, because there's a ton of I/O to handle with up to 100 pins. This chip is fully loaded with eight USARTs, four I²C, four SPI, eight 16-bit timers, megasampling 12-bit ADCs and DACs, and even an AES/DES encryption engine.

Over the years, I've seen the roll-out of a lot of "new-and-improved" 8-/16-bit MCUs. I've always been impressed with the advances overall, but there was invariably some feature I felt could have been added or improved. But at this point, I have to concede that chips like the TI '5*xx* and Atmel XMEGA are pretty darn close to perfect (although I'm sure they'll somehow come up with even "perfecter" chips next time).

NET WORKING

"Wireless standard" is kind of an oxymoron. Indeed, the fact that different radios and protocols can coexist is a great thing. When it comes to wireless, we can all get along. It just means that, like the United Nations, you need a lot of translators (i.e., "gateways" in radio-speak).

Connect One covers all of the bases with a range of chips and modules encompassing a variety of networking and security schemes (www.connectone.com). For instance, you could turn your car into a "hotspot" by lashing together an iChip-based Wi-Fi-to-cellular gateway. Gee, now folks can surf the web while they drive, truly a "killer app."

Another outfit exploiting the ubiquity of Wi-Fi is GainSpan, a new startup spun out of Intel (www.gainspan.com). Their GS1010 combines a pair of ARM7 MCUs (application and protocol) along with an 802.11b/g radio on a single chip (see Photo 3a). Similarly, Laird Technologies (www.lairdtech.com) offers a chip-size (25 mm × 35 mm) 802.11 module including a UART host interface, 12 GPIOs, and two 10-bit ADCs (see Photo 3b). These are good middle-of-the-road options for applications that need a chip-scale solution but don't have enough volume to get the attention of big shot PC market Wi-Fi chip suppliers.

Over the years, ZigBee has given new meaning to the phrase "feature creep," and folks are no doubt wondering "Where's the Beef?" But finally, ZigBee is coming to an application near you. Notable examples are smart energy meters, where market watchers forecast ZigBee in up to 33% of the meters shipped.[2] These meters are just the first step in migration to a smarter and greener consumer energy regime that will create a lot of demand for "smart appliances," "smart houses," and the like.

At the heart of ZigBee is one of the more standard standards, namely the IEEE 802.15.4 radio. At this point, it's probably easier to count the number of chip suppliers that don't offer one than those that do. But just because ZigBee uses 802.15.4 that doesn't mean 802.15.4 needs ZigBee. In particular, I note continued interest in "6LowPAN" IP-over –802.15.4 from a growing list of key players including Atmel and TI.

That's not to say that 802.15.4 is the only game in town. Nothing is stopping anybody from rolling their own radio such as the Nordic Semiconductor nRF24AP1 (www.nordicsemi.com). As you can see in Photo 4, it combines a proprietary 2.4-GHz radio with the ANT protocol stack on one chip for an easy to integrate drop-in solution (www.thisisant.com).

With all of the radios blaring away, the fact that wires are still useful gets lost in the noise. Notably, wires are still the best way to ship power, so I wasn't surprised to see "Power-over-Ethernet" (PoE) gaining traction. I came across an Ethernet device server (the "Owl" from Geist Technology, www.geisttek.com) that has a PoE option, as does the Luminary graphics LCD controller mentioned earlier. It would be great if Ethernet gadgets could be powered over the wire. Death to the wall warts I say!

Another thing a wire can do is deliver data quickly, reliably, and on schedule. To that end, I noticed growing support for IEEE 1588 on the show floor. The standard utilizes a "Grandmaster" reference clock and time stamping to give Ethernet a better shot in applications that require precise, high-speed timing. At the show, Imsys Technologies was showing a development kit for IEEE 1588 applications based on their IM3220

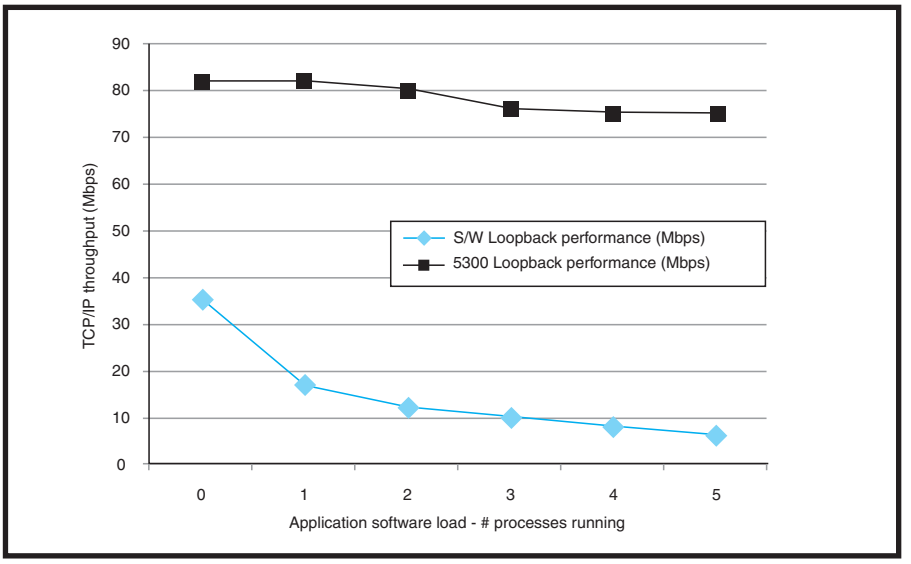


Figure 6—Sometimes, a little hardware, as in a WIZnet 5300 chip, goes a long way. These performance measurements show how a so-called "TCP/IP Offload Engine" (ToE) like the 5300 boosts performance and stability compared to a conventional software stack.

Ethernet IC with built-in '1588 support (www.imsys.se).

WIZnet was showing a new incarnation, the 5300, of their novel "TCP/IP-in-Hardware" approach (www.wiznet.com). The premise exploits the fact that hardware can make things that are really cheap and things that are really fast.

The first-generation WIZnet 5100 chip proved to be a natural for adding Internet capability to blue-collar embedded designs. (Check out the results of the iEthernet WIZnet Design Contest 2007 at www.circuitcellar.com/wiznet.) Combining a 5100 with an entry-level MCU delivers good networking performance, yet costs less than a typical 32-bit MCU + OS + software stack solution.

The 5300 focuses on the performance side of the equation with more (and more flexibly allocated) buffer memory and faster I/O throughput



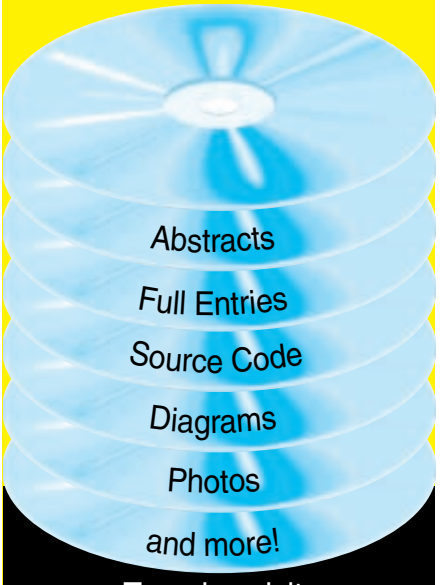
Photo 5—In the old days, "Type "H" for help" was all of the encouragement you'd get from the typical demo board. According to Renesas, it's time to move on.

Why fight with downloads?

Access thousands of Circuit Cellar contest project files in seconds rather than minutes. Ensure that these application libraries remain safe within your personal archive.

CD-ROMs now available:

- WIZnet iEthernet Design Contest 2007
- Microchip Technology 16-Bit Embedded Control Design Contest 2007
- Atmel AVR Design Contest 2006



To order, visit
www.circuitcellar.com/CD

(wider host bus and DMA). At the show, WIZnet had a demo centered on a 32-bit Linux SBC, comparing the performance of a traditional software TCP/IP stack with and without 5300 acceleration. The 5300 option is much faster, and better yet, TCP/IP throughput is virtually unaffected by varying application software load (see Figure 6). The performance and stability under load makes the 5300 a natural for high-traffic multimedia apps (e.g., set-top box, media center).

BRAVE NEW WORLD

I've fired up hundreds of evaluation boards over the years, but never one like the Subatomic Particle Board from Renesas. It's a multimedia experience with an ever cheerful and encouraging Renesas lady holding your hand all the way (see Photo 5). Indeed, thanks to built-in audio on the Atom board, she gives you some soothing words of advice even before you connect the board to your PC.

But there's more than a pretty face here. Under the hood, the "Activity Center" scripts have access to the innards of the hardware and software tools (e.g., compiler and debugger). That means a tutorial can walk you through some sophisticated operations (e.g., finding a bug and rebuilding the project) far beyond the typical "Quick Start" or "Hello World."

Being an old dog, I'm generally skeptical of new tricks. On the other hand, it isn't rocket science to imagine that things will change for the next generation of embedded engineers raised on MTV and video games. Give Renesas credit for thinking way outside the "ROM Monitor" box. This definitely isn't your father's evaluation board.

So, what is next? You get to choose your own favorite spokesperson/guide/avatar? "Hacker Hero?" Commercials?

You betcha! Once again, the ESC reminded me that when it comes to silicon, it's never a matter of if, only of when.

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.

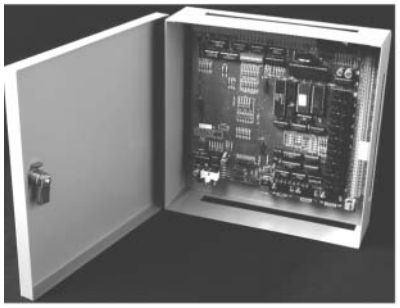
REFERENCES

[1] T. M. Kemp, R. K. Montoye, D. J. Auerbach, J. D. Harper, and J. D. Palmer, "A Decompression Core for PowerPC," *IBM Systems Journal*, Vol. 42, Number 6, 1998, www.research.ibm.com/journal/rd/426/kemp.html.

[2] P. Harper-Slaboszewicz, "ZigBee's Growing Footprint," UtiliPoint International, IssueAlert, Dec. 12, 2007, www.zigbee.org/imwp/idms/popups/pop_download.asp?contentID=12361.

[3] S. Thompson, P. Packan, and Mark Bohr, "MOS Scaling: Transistor Challenges for the 21st Century," *Intel Technology Journal*, Vol. 2, Number 3, 1998, http://download.intel.com/technology/itj/q31998/pdf/trans.pdf.

HomeVision-Pro



CONTROLS:
- Lighting
- HVAC
- Security
- A/V
- Much more...

HomeVision-Pro is the newest, most powerful automation controller in the award-winning HomeVision® product line. It's configured from a PC, but can then run standalone. Features:

- Two-way X-10 & IR
- Video screens on a TV
- 8 relays
- 16 digital & 4 analog I/O
- 64 digital temp sensors
- 1 USB & 3 serial ports
- Powerful scheduling (>8000 lines, >2000 objects)
- Clock
- PC web server
- Expandable
- Software included
- 3-year warranty

Custom Solutions, Inc.
800-398-8882, Web: www.csi3.com, E-mail: csi@csi3.com

CIRCUIT CELLAR®
back issues available as
Searchable Archives on CD-ROM

CD-ROM #12 2007 Issues 198-209
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

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. E-mail adcopy@circuitcellar.com with your file and digital submission or send it to IDEA BOX, Circuit Cellar, 4 Park Street, Vernon, CT 06066. For more information call Shannon Barraclough at (860) 875-2199.

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

NEW
RS232 to TCP/IP

- *TCP/com*™ v2.0, RS232 to TCP/IP software. Plus TCP/IP to RS232.
- *WinWedge*™ RS232 or TCP/IP data direct into any Windows app. - Excel, Access, etc.

TALtech
Free 30 day evals at
www.taltech.com

CUSTOM MEMBRANE KEYBOARDS / SWITCHES



- 1 TO 2 WEEKS TURNAROUND
- VERY COMPETITIVE PRICING
- Ex.: (5) 4-switch keyboards for $395.00
- PCB backed switches
- Custom metal backplates/assemblies
- Electronic assemblies/graphic overlays
- Electronic file transfer capabilities

Picofab Inc.
47808 Blvd. Henri-Bourassa
Charlesbourg, Quebec, Canada G1H 3A7
Tel: (418) 622-5298 • Fax: (418) 622-9996
Email: sales@picofab.net

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

DLP Design

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

LPC3180 Dev Kit



The LPC3180 Development Kit provides a stable platform for building powerful user applications with the NXP Semiconductors LPC3180 within the ARM9 Linux environment.

The kit provides the user with an array of popular mobile technologies, making it an ideal development target for consumer communications software. Using existing Linux hardware drivers, developers can focus on building and improving their user interface applications without worrying about low-level hardware functionality.

The LPC3180 Development Kit may be purchased at Digi-key.com, PN: 622-1018-ND or Mouser.com, P/N: 786-LPC3180-DEV-KIT, and includes the LPC3180 board, the LCD and Keypad Board, and everything else you need to get started.

FDI also provides low cost demo kits for NXP ARM7 LPC2xxx, LPC9xxx and ICP/ISP FLASH programmers.

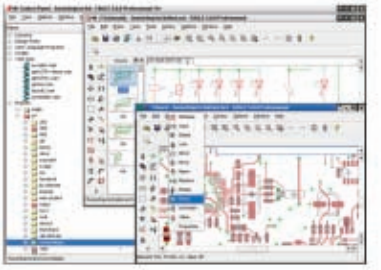
FDI
www.teamfdi.com
VISA/MC/Amex

Future Designs, Inc.
2702 Triana Blvd
Huntsville, AL 35805
(256) 883-1240
Fax (256) 883-1241

EMULATE A USB
Keyboard + Mouse + Joystick

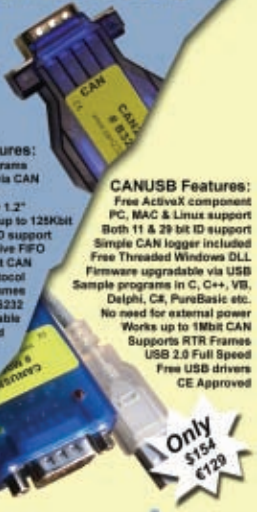
NEW!
MODEL KE-USB108
-108 individual programmable switch inputs
-Produce long macros from a single input
-Send keystrokes,mouse or joystick actions
-Ideal handling of toggle switches
-2 Rotary Encoder inputs
-Potentiometer inputs for Joystick X,Y,Z
888-690-9080
HAGSTROMELECTRONICS.COM
11 FIDDLERS GREEN, LANSING NY 14882

iMAGEcraftC
The Best Selling C Compiler for AVR
• High quality code generation
• First commercial whole program compression optimizer
high powered development, garage powered prices!
starting at $249!
• Application builder
• Multiple versions
• Global optimizer
• Simple to use IDE
• Multiple platform support
• Flexible licensing schemes
www.imagecraft.com
info@imagecraft.com
(650) 493-9326 • FAX (650) 493-9329

The number one choice in PCB-Design just got better.
EAGLE V5.0 Highlights
Display, Zoom and Grid History + Lock function for parts + unlimited name/value attributes for parts + move group to other sheet w/o modifying the board + reorder sheets with drag&drop or insert new sheets + REPLACE function for the schematic editor + selective display of unrouted tracks + time-saving new re-routing feature + minimum clearance for any combination of net classes + add elements to group or subtract elements from group.

www.cadsoftusa.com

Introducing the iLoad Mini™ - low cost, miniature load cell
Digital Core
Analog Compatibility

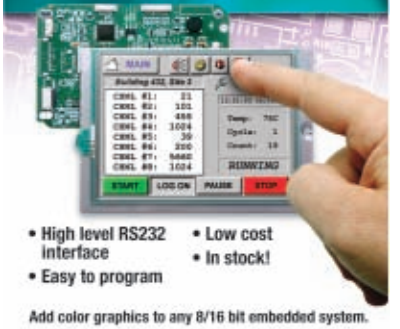

Loadstar SENSORS
www.loadstarsensors.com/cc | 650.938.4282

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

MYLYDIA, INC.
Free one time setup!
(subject to approval)
W/5 double side PCB
2 weeks
Tel: 800.695.9342
sales@mylydia.com
www.mylydia.com

Solve complex signal acquisition problems...

• positioning & control
• environmental
• acceleration
• transients
• pressure
• vibration
• sonar
• GPS
• Linux Driver
• Guaranteed in stock
• Many newly added features
• 16-bit analog inputs and outputs
• Million sample FIFO eliminates interrupts
• Wide analog input and output ranges
• -40°C to +85°C Standard
• Order 24/7, fast and easy.
www.stx104.com
Apex Embedded Systems
help@stx104.com • 608-256-0767 x24

Add a color touch interface to your embedded product!

• High level RS232 interface
• Low cost
• In stock!
• Easy to program
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

Free Training
for Rabbit Developers
Learn More at
Rabbit-U.com
09078

tincan TOOLS
Linux in a 40-pin DIP
- 200 Mhz ARM9 CPU Board
- 0.1" centers - easy to prototype
- use your existing prototype tools
- full Linux with 2.6 kernel
- CD includes source and schematic
HAMMER
Perfect for robotics, web servers, remote monitoring & communications
www.TinCanTools.com

PC/104 PERIPHERALS
SCIDYNE Offers a Full Line of Innovative Modules for PC/104 Applications
ADIO-104 This Module Does it all!
•16 Analog Inputs •5V Power
•8 AnalogOutputs •Pulse Accumulator
•24 Digital I/O lines •Open-Drain Outputs
DIO96-104
•96 Bi-Directional Digital Channels
•Input, Output and Strobed I/O functions
•Uses Familiar 82C55A Chips
XIO-RO8
Add eight High-Power relay outputs to any digital port.
Call or Visit us on the web
SCIDYNE® 1-877-724-3963
www.scidyne.com

iMAGECraftC
C Compilers for:
MSP430
CPU12
ARM
AVR
etc.
high powered development, garage powered prices!
starting at $249!
• First commercial whole program compression optimizer
• Multiple versions
• Global optimizer
• Simple to use IDE
• Multiple platform support
www.imagecraft.com
info@imagecraft.com
(650) 493-9326 • FAX (650) 493-9329

MCS - 51
EXCELLENT noise immunity!
V-E54 v1.0 (USD 59)
Embedded board MPC82E54A, the most efficient MCU from Megawin, 16K Flash speedx12. 8ch. 10bit A/D, 2ch. 12bit D/A
PRO-MISP (USD 99)
Megawin 8051 ISP Programmer, High Speed download via USB Port, 89E58AE, 82E54AE2, developer board included, 3 Pins to target board (+5V, Data, GND)
www.ispmicro.com
sales@ispmicro.com

Tiny Tool. Huge Productivity.
Introducing Logic. The Logic Analyzer 2.0.
• 8 Channel @ 24MHz
• SPI, I2C, Serial Decode
• Best-in-Class Software
• $149 • Try Free
Visit saleae.com for free trial.
Also find Logic on SparkFun.com or google "logic analyzer."
Saleae

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

FlashCore-B [FB]™
Add embedded mass data storage, 16-bit ADCs to your product
$79 Qty 1 $34 OEM
• 2.1" x 2.4", C/C++ programmable.
• CompactFlash interface with FAT file system.
• 16-bit ADCs, DACs, RS232 and TTL I/Os.
• Ultra-low quiescent current for battery power.
50+ Low Cost Controllers with ADC, DAC, 18 UARTs, 300 I/Os, solenoid, relays, CompactFlash, LCD, DSP motion control. 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

Instant LCD



ezLCD-002 - $189

- Versatile Programmable Module
- Serial/I²C/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.™

PHYTEC

phyCORE® SBCs Accelerate 32-bit Designs



phyCORE-LPC3250

CDTS Single Board Computers:
- shorten time-to-market
- reduce development costs
- forgo substantial design issues and risks
- Windows® Embedded CE and Linux Board Support Packages (processor-dependent)
- $170/unit benchmark price at 1K for ARM9-based SBC

ARM: i.MX31 (ARM11), i.MX27 (ARM9), LPC3250 (ARM9), LPC3180 (ARM9); LPC2294 (ARM7)
XScale: PXA320, PXA270, PXA255
PowerPC: MPC5554, MPC5200B, MPC565, MPC555
ColdFire: MCF5485 Blackfin: ADSP-BF537

- Rapid Development Kits start at $399
- include SBC, Carrier Board, software, docu as well as kit-specific cables, adapters and LCD
- SBC module easily ports from Carrier Board to user target hardware
- Carrier Board serves as target reference design



www.phytec.com ■ 800.278.9913 ■ www.phycore.com

Bargain Test Gear !

PDS5022



Portable, benchtop 25Mhz 100MS/s 2-ch color LCD Scope. Download waveforms via USB1.1 to PC. 7.8" 640 x 480 Color LCD 3.5lbs. Volt/Time Automatic Measurement Peak-to-peak Value, Average, RMS, Frequency and Cycle. **FREE** 1x/10x probes, USB cable. Compare with expensive LCD scopes!

PDS5022S $325

SPI Xpress



SPI Bus Analyzer - SPI protocol exerciser and analyzer for standard SPI and non-standard 4-wire and 3-wire serial protocol interfaces up to 50 Mbps. PC control via free GUI Control Panel Software. SPI sniffer, chip-to-chip emulation.

SPI Xpress $783

Saelig com unique electronics

1-888-7SAELIG
info@saelig.com
www.saelig.com

Going Wireless is Easy!

Embedded Bluetooth Module



NEW

Parani-ESD 1000

- Class 1 OEM style
- Bluetooth (v2.0 + EDR)
- Up to 100 meter transmit distance

LEMOS INTERNATIONAL 1-866-345-3667 sales@lemosint.com

www.lemosint.com



Make sure you're signed up to receive Circuit Cellar's monthly electronic newsletter. *News Notes* will keep you up to date on Circuit Cellar happenings. Stay in the loop!

Register now. It's fast. It's free.

www.circuitcellar.com/newsletter/

Overcoming obstacles with...

Mach Design Services, LLC

- Hardware Design
- Software Design
- PCB Design
- PCB Assembly
- & Much More!

Focusing on 8/16 bit control modules for the following apps:

- automotive
- industrial
- consumer

www.machdesignservices.com

NEW

HI-TECH C® PRO for the Microchip

PIC32

MCU Family

C Compiler

FREEWARE

New PRO compiler also operates in Lite mode with no time or size limits:

microchip.htsoft.com/portal/CC_20

HI-TECH SOFTWARE

1,600+ Products

www.DesignNotes.com



Test Equipment
USB PC 2 CH 60MHZ Oscilloscope
$329.99 + free gift



PCSU1000

DesignNotes.com
Affordable Prototyping Products

1-800-957-6867

iMAGECraftC

NEW: the eBox- AVR!

Compact Complete AVR Kit
- 16K ImageCraft ICCAVR compiler
- Mega16 kit with LED, keypad, LCD and more
- 10 example software projects
- Battery INCLUDED

Perfect for learning embedded programming and the Atmel AVR.

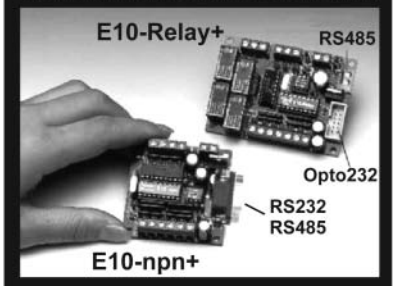


only $199!

www.imagecraft.com
info@imagecraft.com
(650) 493-9326 • FAX (650) 493-9329

The $69 PLC

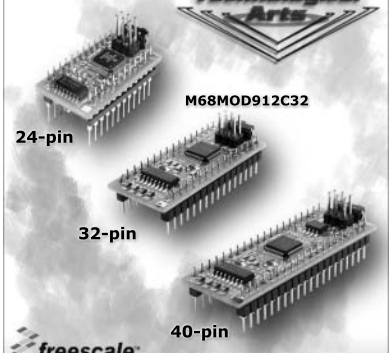
Work as Stand-Alone Ladder Logic PLC.
Or as Smart Remote I/Os of PC/ PLCs.
RS485 allows 256 units to be networked.



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
Triangle Research
Tel: 1-877-874-7527 - PLC specialist since 1993

Freescale 9S12C microcontroller DIP modules

Technological Arts



freescale Alliance Member
www.nanocore12.com
Toll-free: 1-877-963-8996

TStik!™

Rugged TINI Java™ Module with 10/100 BaseT



TStik is a ruggedized TINI400 chipset in the familiar SIMM72 form factor. Upgrade most DSTINI1 (TINI390) systems or use our new TILT socket boards (TILT Pro is shown above).
TStik is about $100, and sockets start at under $60.

SYSTRONIX®
full details at www.TStik.com

Control Your Motor



Use low-cost DC PM brush motors
Proportional PWM Forward/Reverse
Controls 2 motors independently
Or mixed tank steering algorithm
5-30 V, 30A/motor, 70A starting
Adjustable current limiting
Dynamic brake and Clutch output
CDFR for 8 bit parallel port I/O
RDFR for R/C servo command
(800) 882 6832 Visa/MC

VANTEC.COM

Flashlite 186



- 186 processor @ 33 MHz
- DOS w/ Flash File system
- 44 Digital I/O lines w/ CPLD
- Console / Debug Serial Port
- 7-34V DC or 5V DC power • 2 Serial Ports
- Accepts 8MB DiskOnChip • 2 16-bit Timers
- 512K DRAM & 512K Flash • Watchdog Timer
- Expansion options with Peripheral Boards

$69 QTY 1

$99 Development System

Development kit includes:
- Flashlite 186 controller
- Borland C/C++ ver 4.52
- FREE Email Tech Support
- Serial Driver library
- AC Adapter and cable

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

JK microsystems

CROSSWARE®
Tools for Embedded Development

ARM® 8051 COLDFIRE®

Three quality tool suites
- ❑ *C/C++**
- ❑ *Code Wizards*
- ❑ *Debugging*
- ❑ *Simulation*
- ❑ *Support*

* Embedded C++ for ARM and ColdFire only

Advanced software tools since 1984

www.crossware.com
360-812-2397



Weather Instruments for PCs

AAG electrónica

www.aagelectronica.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

***Incredibly Low-Cost* USB Chips at HexWax**

USB-232 USB to serial UART, smarter and cheaper than the FT232R
USB-SPI USB to synchronous serial (I2C, too!)
USB-DAQ Data logger, plugs in like a flash drive
USB-Config Configuration & diagnostic module
USB-TakeOff Power take-off & battery charger
WattLogic Domestic power consumption logger
AnniLogic Anniversary reminder
expandIO-USB I/O expanders
TEAleaf-USB Security dongle

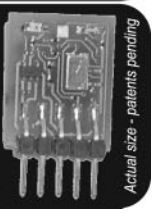
No drivers needed!

Visit www.hexwax.com today!

***Amazing* PIC programmer**

Most devices supported
ICSP, SQTP, & copy limits

$32 at Digikey & Mouser
www.flexipanel.com



Actual size - patents pending

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!

Trace SYSTEMS, Inc.

www.TraceSystemsInc.com

301-262-0300

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

CIRCUIT CELLAR®

back issues available as

Searchable Archives on CD-ROM

CD-ROM #12 2007 Issues 198-209
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

Temperature Monitoring Over the Internet

-View temperatures using a web browser

-Connect up to four temperature sensors

-Send email alerts or alarms

-Two built-in relays



CONTROL by WEB

www.ControlByWeb.com

(435) 750-5999

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

Order online at: www.melabs.com

microEngineering Labs, Inc.

Development Tools for PIC® Microcontrollers

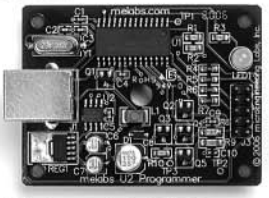
Phone: (719) 520-5323
Fax: (719) 520-1867
Box 60039
Colorado Springs, CO 80960

USB Programmer for PIC® MCUs $89.95 (as shown)

RoHS Compliant

Programs PIC MCUs including low-voltage (3.3V) devices

Includes Software for Windows 98, Me, 2K, & XP



With Accessories for $119.95: Includes Programmer, Software, USB Cable, and Programming Adapter for 8 to 40-pin DIP.

Serial LCDs
2-line x 16 $39.95
4-line x 20 $49.95
Quantity Discounts Available!

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

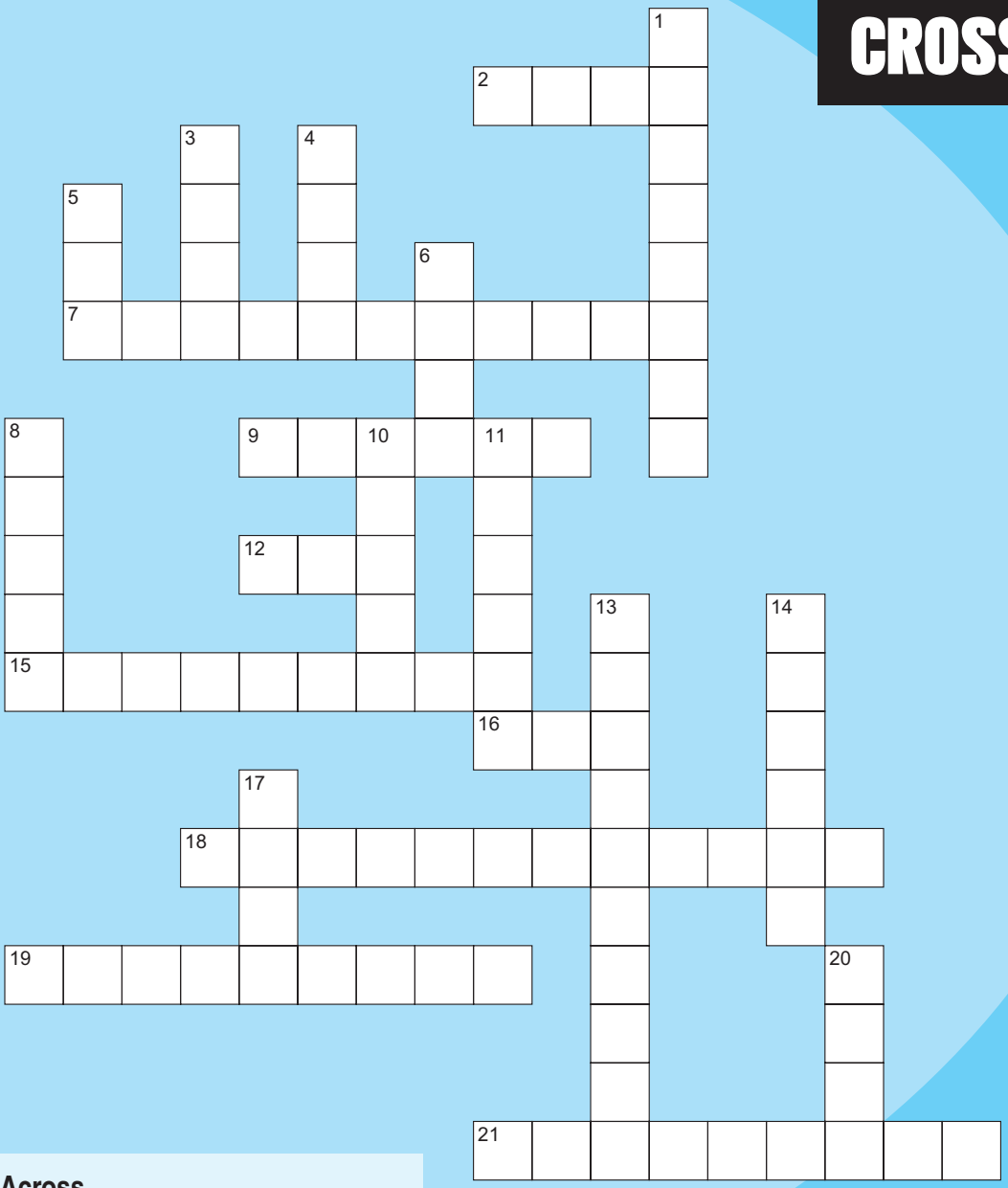
PICBASIC™ Compiler $99.95
BASIC Stamp 1 Compatible
Supports most 14-bit Core PICs
Built-In Serial Comm Commands

PICBASIC PRO™ Compiler $249.95
32-bit signed variables and math operations
Supports Microchip PIC10, PIC12, PIC14, PIC16, PIC17, and PIC18 microcontrollers
Direct Access to Internal Registers
Supports In-Line Assembly Language
Interrupts in PICBASIC and Assembly
Built-In USB, I2C, RS-232 and More
Source Level Debugging

See our full range of products, including books, accessories, and components at:

www.melabs.com

CROSSWORD



Across

2. 10^3
7. Built-in software on a cell phone that enables you to access special Internet sites
9. A pulse used for timing and synchronization
12. 1/1,000 inch
15. 2^{80}
16. A quick way of sharing new content, such as news headlines or podcasts, to web sites and software
18. Someone who owns a copyright
19. A system that examines fingerprints or retinas for security purposes
21. A standard used to assess a computer program's performance

Down

1. A system with one sound channel
3. A template for a blog, web site, or software
4. Binary large object
5. Software designed to prevent the unauthorized duplication of audio or video files
6. Motherboard
8. H
10. Circuit-controlled switch
11. CD writer
13. Altered version of an audio signal caused by a change in the waveform of the initial signal
14. A piece of wire used for finishing a circuit
17. The first American woman (b. 1951) to enter outer space
20. The second stage of testing, where a piece of software is released to a limited number of customers

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	13	Front Panel Express LLC	90	Mach Design Services, LLC	31	SoC Conference
80	AP Circuits	83	Grid Connect, Inc.	55	Microchip	34, 90	Saelig Co.
82	ARM DevCon	88	Hagstrom Electronics	92	microEngineering Labs, Inc.	89	Saleae LLC
92	All Electronics Corp.	90	HI-TECH Software LLC	1	Micromint	89	Scidyne
88	Apex Embedded Systems	33	HobbyLab LLC	2	Mouser Electronics	17	Sealevel Systems
7	Atmel	95	Holtek Semiconductor, Inc.	88	Mylydia, Inc.	10	SEGGER Microcontroller Systems LLC
81	Bitscope Designs	89	ISP Micro	C2	NetBurner	29	Silicon Laboratories, Inc.
65	CWAV	89, 90	IMAGEcraft	3	Noritake Co., Inc.	8	Solder By Numbers (Schmartboard)
21, 88	CadSoft Computer, Inc.	33, 88	IMAGEcraft	67	Nurve Networks LLC	27	Spark Fun Electronics
49	Comfile Technology, Inc.	9	Imagineering, Inc.	91	Ontrak Control Systems	91	Systronix
91	Crossware Products, Inc.	14	Ironwood Electronics	58	PCB-Pool	87	TAL Technologies
64	Custom Computer Services, Inc.	64, 66	JKmicrosystems, Inc.	C4	Parallax, Inc.	C3	Tech Tools
86	Custom Solutions, Inc.	91	JKmicrosystems, Inc.	90	Phytec America LLC	72, 73	Technologic Systems
5	Cypress MicroSystems Inc.	71	Jameco	25	Pico technology, Ltd. USA	91	Technological Arts
87	DLP Design	67	Jeffrey Kerr LLC	87	Picofab Inc.	89	Tern, Inc.
90	DesignNotes	74	Keil Software	92	Pioneer Hill Software	79	Tibbo Technology, Inc.
83	EMAC, Inc.	67	Lakeview Research	12	Pololu Corp.	89	Tin Can Tools, LLC
90	Earth Computer Technologies	88	Lawicel AB	87	Pulsar, Inc.	92	Trace Systems, Inc.
66	Elprotronic	13, 90	Lemos International	14, 39	Rabbit, A Digi International Brand	91	Triangle Research Int'l, Inc.
30	ExpressPCB	80	Linx Technologies	89	Rabbit, A Digi International Brand	91	Vantec
41	ezPCB	88	Loadstar Sensors, Inc.	88	Reach Technology, Inc.	11	WIZnet
87	FDI-Future Designs, Inc.	42	Luminary Micro	47	Renesas Technology	92	Xytronix Research & Design, Inc.
91	FlexiPanel Ltd.	89	MCC (Micro Computer Control)	50	RoboDevelopment Conf & Expo		

Preview of October Issue 219
Theme: Signal Processing

DIY Signal Generation: Create a Signal Source for a Variety of RF Applications

Dynamic Animatronic Remote (Part 2): Software and Menuing Code

Automated Data Mining: Build Your Own Embedded Server Application

Universal Machine Control: How Proper Hardware Emulation Leads to Prototype Development

---

THE DARKER SIDE **Cable Shielding Experiments**

ABOVE THE GROUND PLANE **HT Audio, GPS for APRS, and What Works**

FROM THE BENCH **Serial Terminal Solution**

SILICON UPDATE **Hot Enough For You?:** A Look at Infrared Noncontact Temperature Sensing

ATTENTION ADVERTISERS

November Issue 220 Deadlines

Space Close: Sept. 12
Material Close: Sept. 19

Theme:
Analog Techniques

BONUS DISTRIBUTION:
SoC Conference & RoboDevelopment

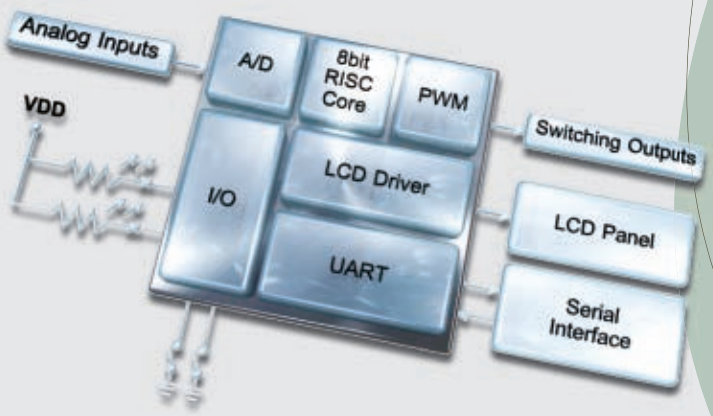
Call Shannon Barraclough
now to reserve your space!
860.875.2199
e-mail: shannon@circuitcellar.com

HOLTEK
www.holtek.com

Intelligent Solutions

Easy connectivity and the responsiveness to match all your intelligent controller needs

The HT46RU2X and HT46RU6X series devices with their fully integrated measurement and connectivity capabilities, embedded A/D converter, PWM and UART functions, significantly reduce external component requirements, reduce costs and increase product reliability, providing excellent solutions for a wide range of control and data transfer applications.





10 - pin MCU | USB MCU | Dual Slope MCU | EEPROM | Power Management | TinyPower™ MCU | Flash MCU | **UART MCU** | Audio MCU | Phone MCU

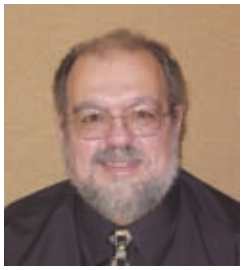
- Wide operating voltage: 2.2V ~ 5.5V
- Industrial specification: - 40 ~ + 85°C
- Integrated UART communication module
- PFD function
- Excellent noise immunity and ESD protection
- Integrated SPI and I^2C functions

Part No.	Program	Data	I/O	Timer	A/D	PWM	I^2C	SPI	UART	Package
HT46RU22	2K x 14	64 x 8	19	8bit x 1	9bit x 8	8bit x 1	V	---	V	24SKDIP, SOP, SSOP
HT46RU232	4K x 16	192 x 8	40	8bit x 1, 16bit x 2	12bit x 8	8bit x 4	V	---	V	28SKDIP, SOP, 48SSOP
HT46RU24	8K x 16	384 x 8	40	8bit x 1, 16bit x 2	12bit x 8	8bit x 4	V	---	V	28SKDIP, SOP, 48SSOP
HT46RU25	16K x 16	576 x 8	48	8bit x 1, 16bit x 2	12bit x 8	8bit x 4	V	---	V	48, 56SSOP
HT46RU26	32K x 16	768 x 8	48	8bit x 1, 16bit x 2	12bit x 8	8bit x 4	V	V	V	48, 56SSOP

Part No.	Program	Data	I/O	LCD	Timer	A/D	PWM	SPI	UART	Package
HT46RU66	16K x 16	576 x 8	32	46 x 4	8bit x 1, 16bit x 2, RTC	12bit x 8	8bit x 4	---	V	52QFP, 56SSOP, 100QFP
HT46RU67	32K x 16	768 x 8	32	46 x 4	8bit x 1, 16bit x 2, RTC	12bit x 8	8bit x 4	V	V	52QFP, 56SSOP, 100QFP

HOLTEK SEMICONDUCTOR INC.

Holtek Semiconductor(USA), Inc.
46729 Fremont Boulevard, Fremont, CA 94538
Tel: (510) 252-9880 Fax: (510) 252-9885
E-mail: sales@holtek.com
http://www.holtek.com/usa_only.html



PRIORITY INTERRUPT

by Steve Ciarcia, Founder and Editorial Director

Something Old, Something New

When I start talking to the average person about my photovoltaic (PV) power installation, most people (both technical and nontechnical) assume I'm talking about an off-grid, battery-backed system. These are people who know where I live and know I have utility lines to my house. Perhaps because they think I'm some kind of maverick, I'd be the one guy they know who would install a solar system simply to ditch the existing power utility/customer rat race. As much as I might like the idea, at my level of peak electric consumption, complete independence is still very cost-prohibitive, but not completely out of mind. ;-)

Although my house is surrounded by trees (yes, I'm still in the middle of a forest despite opening up the area for PV), I've had relatively few power interruptions from bad storms and the like. Fortunately, I don't live too far from main roads, so even if power does fail, it is a fairly quick recovery time. Basically, Connecticut Light & Power is the primary backup to my less than 1-year-old solar PV system.

The "alternative" power backup system that remains unchanged is my 14-kW diesel generator. While I have reduced my power signature considerably to maximize PV energy use, I still have some serious peak power loads. There's really no battery system out there (that I care to pay for and maintain) that would ensure uninterrupted service if the power utility went down. (The PV panels cannot operate without utility power in a grid-tied system.) Knock on wood, but I've bullet-proofed my system to where I am confident of a minimum level of comfort and security. As expensive as diesel may be right now, it's a proven solution and there is no logical reason to dump it in the name of "progress." This slow but continuous style of integrating the best of the old along with something new is my way of advancing my overall level of technology while avoiding the pitfalls from too many changes at once.

I see the tug of online influences (newsletters, blogs, etc.) and *Circuit Cellar*'s print magazine in the same light. You simply don't mess with a proven system despite the apparent benefits of a complete metamorphosis. I've come to the conclusion that everyone is tired of seeing their favorite publications follow the cyber trail. In the minds of subscribers, this trail often means that their print magazines either go away altogether or become mere teaser material for less-inspired, ad-driven online editorial content.

Circuit Cellar magazine maintains a proven path and consistent stability. Still, we're not oblivious to Internet reality and we are integrating new features. When *Circuit Cellar* launched its monthly *News Notes* e-mail newsletter in July, it was greeted with some initial fears. Is this the replacement for the print magazine that subscribers have supported for over 20 years? Some wondered just that, having experienced this type of "new and improved" online publication switch before. All I can say is, relax guys. Not on my watch.

It's not that *Circuit Cellar* readers are in love with paper, but rather, you appreciate what is accomplished on that paper. You cherish the linear flow of the articles found in the paper magazine. You appreciate having large schematics right by your soldering iron. And you realize that it's all too easy for publications to get sloppy with editorial quality when it can get quickly slapped on a web site with little or no technical editing and pulled down just as quickly. You want a print magazine because the success it takes to maintain that level of publishing expense is a measure of credibility by itself. *Circuit Cellar* has never altered from this path.

Electronic publishing at *Circuit Cellar* is meant to complement the print magazine. Our Electronic Edition is an exact replica of the print magazine in PDF format because we have readers who love the ease of reading paper along with those who like the benefits of electronic portability and searchable archives. Certainly, because we don't have the cost and size limitations imposed by print, we could make the electronic version of the magazine be a larger revenue generator with considerably more advertiser input. But, changing *Circuit Cellar*'s basic recipe to emulate a trade magazine is suicidal in the long run. We could head down a slippery slope where articles become mere intros, details become generalities, and invitations to community blogs or advertiser 'Net seminars become the prime editorial generators.

Still, there is room for other *Circuit Cellar* editorial products that do live completely online. *Circuit Cellar*'s *News Notes* is a bit different from what people have come to expect from the impact of cyberspace on print magazine publishing. While the print magazine focuses on detailed application articles, the newsletter addresses items that are outside of that scope but still relevant to the *Circuit Cellar* community. It looks at how the industry interacts with *Circuit Cellar*, describes your impact on Silicon Valley, and presents messages from *Circuit Cellar* that don't fit into the print magazine's long lead times. The free newsletter is also an opportunity to include follow-up insight from authors and columnists. The size restrictions of the print magazine would normally prohibit the inclusion of such material, but *News Notes* specifically tackles it.

So, while our newsletter is still in its infancy and continues to evolve, rest assured its creation is not a harbinger of change to the print magazine. Think of *News Notes* as a conversation between *Circuit Cellar* and its readership, and I look forward to seeing you there. ;-)

Sign up: www.circuitcellar.com/newsletter

steve.ciarcia@circuitcellar.com



PC Based Logic Analyzers

NEW Model DV3400

- Faster Sampling (200/400 Msps)
- More Channels (36)
- More Memory (4x)
- Advanced Match circuits (8)
- ±6V Adjustable Threshold (x2)
- Cascadable Sequencers (4)
- Enhanced Compression

from $499.00

DigiView



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



QuickWriter™

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.






PARALLAX

Helping bring your ideas to reality since 1987.

Microcontrollers and accessories for the engineer, hobbyist, and student.
BASIC Stamp, SX microcontroller, and the Propeller chip.






Explore the Parallax possibilities online at http://www.parallax.com Call our Sales Department toll-free 888-512-1024 (Mon-Fri, 7am-5pm, PDT).

Special thanks to John Twomey, Rich Allred, Andy Lindsay, Ken Gracey, Steve Ainsworth, and MicrobotiX, for the photgraphs. Parallax and the Parallax logo are trademarks of Parallax Inc.

PARALLAX INC.
www.parallax.com