PROJECTS ■ THEORY ■ APPLICATIONS ■ CIRCUITS ■ TECHNOLOGY

NUTS AND VOLTS

www.nutsvolts.com
October 2012

EVERYTHING FOR ELECTRONICS

THE REMOTE DATA LOGGER & Surveillance Cam

Near Field Communications The Solution Looking For A Problem

Build A MIDI TO LOGIC Controller The Easy Way!

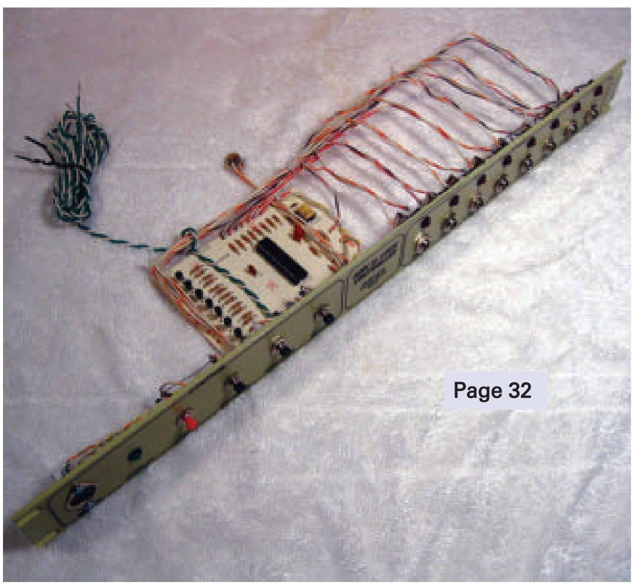
U.S. $6.50 CANADA $7.50
10
0 71486 02421 7

October 2012

www.nutsvolts.com

Projects & Features

Page 60



Page 32

Columns

Departments

Nuts & Volts (ISSN 1528-9885/CDN Pub Agree #40702530) is published monthly for $26.95 per year by T & L Publications, Inc., 430 Princeland Court, Corona, CA 92879. PERIODICALS POSTAGE PAID AT CORONA, CA AND AT ADDITIONAL MAILING OFFICES. POSTMASTER: Send address changes to **Nuts & Volts, P.O. Box 15277, North Hollywood, CA 91615** or Station A, P.O. Box 54, Windsor ON N9A 6J5; cpcreturns@nutsvolts.com.

Simple High-Power Motor Controller 18v25 and 24v23 Pin-Out
USB interface (PC)
RC Interface
(RC Receiver)
TTL Serial Interface
(Microcontroller)
Analog Interface
(Joystick/Potentiometer/Limit Switch)
Battery
SIMPLE MOTOR CONTROLLER?
SERVO CONTROLLER?
USE COUPON CODE! NV8693
WIRELESS ROUTER
SONAR RANGE FINDER??
DISTANCE SENSOR?
SERVO
MOTOR CONTROLLERS
VOLTAGE REGULATOR
Take your design from idea to reality
Pololu
Engage Your Brain
www.pololu.com

by Bryan Bergeron, Editor

DEVELOPING PERSPECTIVES

Full Power to the Shields

Daughterboards — as a means of expanding hardware platforms — have been around for decades. They really came into their own in consumer electronics with the advent of the PC. Daughterboards enabled the expansion of the motherboard and video cards with memory and coprocessors. Unfortunately, most of these arrangements have been proprietary and limited to a single model of motherboard.

Today, thanks largely to the microcontroller market, many daughterboards — better known as shields — are plug-and-play devices. They provide an efficient, cost-effective means of transforming a generic microcontroller into a servo controller, RF transceiver, sensor array, or what have you.

The Arduino exemplifies what a standard shield can do for experimenters. Not only are there hundreds of shields available for the Arduino, but the pin configuration of the Arduino shields (Uno and Mega) has been duplicated for other microcontrollers.

For example, Parallax offers the StampDuino, a BASIC Stamp development board designed to be compatible with most Arduino shields. Parallax also offers a sort of reverse shield — the Board of Education for the Arduino. Whereas most shields are the same size as the Arduino and sit above the microcontroller, the Board of Education is significantly larger than the Arduino and accepts the Arduino from above.

Digilent — my go-to company for high performance Arduino-compatible microcontrollers — also has one of the best assortments of shields on the market. I've used their Basic I/O shield — a real life saver when you have more important things to do than figure out how to connect motors, servos, and I²C devices to your microcontroller.

Digilent also offers 60+ mini-shields — Pmods™ — based on six- and 12-pin connectors, supporting everything from Wi-Fi and organic LED displays to a Bluetooth. I've used the audio input mini-shield which is a real time saver.

Together with the added processing power of the chipKIT microcontroller, the mini-shield and processor hold up to just about anything my PC with sound card can handle. The Pmods — while not a standard shield configuration — are designed to work with their Arduino-compatible shields.

SparkFun — the source of countless red shipping boxes in my workroom — offers some of the coolest shields available, especially if you're into kits. I've built several of their MIDI and joystick shields for various projects. SparkFun also offers the Netduino which — at first glance — could pass for an Arduino Uno but uses the .NET Micro Framework. The hardware is compatible with most Arduino shields.

There are exceptions, of course. Because of size and weight limitations, there are a number of Arduino compatible boards on the market that can't accept shields. For example, I'm working on an ArduPilot Mega 2.0 from 3DRobotics which is essentially a Mega-compatible shield chocked full of SMT sensors for making flying, diving, and walking robots.

The open source board is expensive ($199), but crams an amazing amount of sensor technology into a very small, lightweight package. A regular sized Mega would be too heavy and too large for a small flying vehicle. I'll be reviewing this shield with a DIY quadcopter in an upcoming issue of *SERVO*.

I'm all for shields. Look what the Arduino has done for the add-on industry and for experimenters. There is a downside, of course. There's a lot of unused space on the latest generation of Arduino-compatible shields.

While ASICS and SMT components continually reduce component count and the space requirement, shield size has remained constant. So, even though it's possible to shrink the form factor with no loss in functionality, the standard size and pinout dictate otherwise.

At some point, the standard Arduino will evolve to a new size, and you'll have to buy new shields or some sort of adapter — perhaps akin to the Pmod mini-shields. But that's the price for progress. For now, full power to the shields. **NV**

NUTS AND VOLTS
EVERYTHING FOR ELECTRONICS

Published Monthly By
T & L Publications, Inc.
430 Princeland Ct.
Corona, CA 92879-1300
(951) 371-8497
FAX **(951) 371-3052**
Webstore orders only **1-800-783-4624**
www.nutsvolts.com

Subscriptions
Toll Free **1-877-525-2539**
Outside US **1-818-487-4545**
P.O. Box 15277
North Hollywood, CA 91615

FOUNDER/ASSOCIATE PUBLISHER
Jack Lemieux

PUBLISHER
Larry Lemieux
publisher@nutsvolts.com

ASSOCIATE PUBLISHER/ VP OF SALES/MARKETING
Robin Lemieux
display@nutsvolts.com

EDITOR
Bryan Bergeron
techedit-nutsvolts@yahoo.com

CONTRIBUTING EDITORS
Jeff Eckert, Russ Kincaid, Joe Pardue, Fred Eady, Ron Hackett, Lou Frenzel, Andy Sullivan, Thomas Henry

CIRCULATION DEPARTMENT
subscribe@nutsvolts.com

SHOW COORDINATOR
Audrey Lemieux

MARKETING COORDINATOR WEBSTORE
Brian Kirkpatrick
sales@nutsvolts.com

WEB CONTENT
Michael Kaudze
website@nutsvolts.com

ADMINISTRATIVE ASSISTANT
Debbie Stauffacher

PRODUCTION/GRAPHICS
Shannon Christensen
Sean Lemieux

READER FEEDBACK

So, How Exactly Does the Bell Toll?

I volunteer at the Hagley Museum. At the museum, there is a bell that currently cannot be rung. It is in a tower at the back of the visitor's center, and used to be able to be rung using a rope. An elevator was put in that required fire stops, so now there is no way to get a rope from the tower down to where a human could pull on it. There's only one bell, so there's no need for a controller. A solenoid to ring the bell might be a reasonable thing.

I'd like to find out what sort of solenoid Chris Watson used in his August 2012 article about the church bells.

Dan Caster

Response:

The bells in the St Mark's bell tower are struck by electromagnetic hammers. There is a clapper ball fixed to the end of an armature that is pivoted such that when the 24 volt relay coil is energized, the ball moves and strikes the lip of the bell. Our clappers are mounted inside the bell on a support that is fixed to the point at the top of the bell where the bell itself is hung from. Documentation shows that clappers can also be mounted to strike the outside lip of the bell. Clappers were supplied as original equipment by Van Bergen Bellfoundries, Inc., when the system was installed. Since that time, the company has changed and our bells are now supported by the Verdin Company.

Our contact there is J. Andrew Rebber, Western Regional Manager, Magnolia, TX; phone is 512-301-5987; email is arebber@austin.rr.com.

Good luck with your project.

Chris Watson

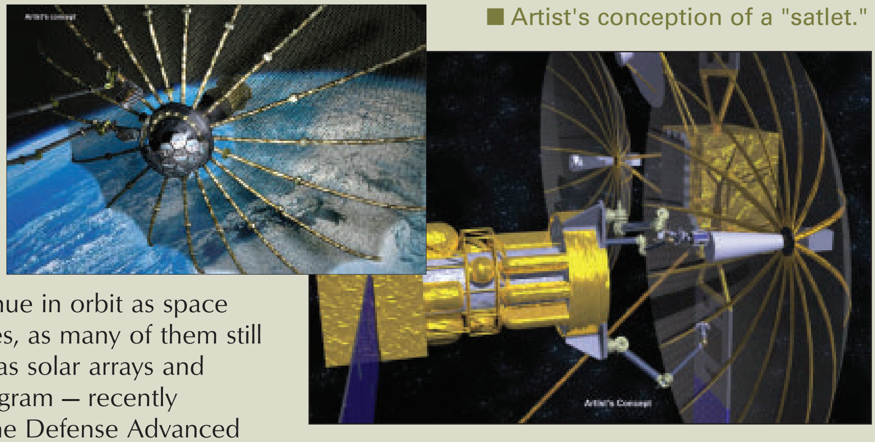
TECH 2012 KNOWLEDGEY

■ BY JEFF ECKERT

EVENTS, ADVANCES, AND NEWS

ADVANCED TECHNOLOGY

ROBOTS TO RECYCLE SPACE JUNK



■ Artist's conception of a "satlet."

It's well known that the US military depends on a slew of communication satellites that circulate in geosynchronous orbit (GEO) about 22,000 miles up. Unfortunately, whenever one fails or otherwise becomes obsolete, we have to send up a replacement and just let the old one continue in orbit as space junk. This is a regrettable waste of resources, as many of them still contain perfectly viable components such as solar arrays and antennas. However, the new "Phoenix" program — recently instigated by some parsimonious folks at the Defense Advanced Research Projects Agency (DARPA, **www.darpa.mil**) — aims to save us some money by salvaging these parts from decommissioned satellites. According to DARPA, "The goal of the Phoenix program is to develop and demonstrate technologies to cooperatively harvest and reuse valuable components from retired, nonworking satellites in GEO and demonstrate the ability to create new space systems at greatly reduced cost. Phoenix seeks to demonstrate around-the-clock, globally persistent communication capability for warfighters more economically by robotically removing and reusing GEO-based space apertures and antennas from decommissioned satellites in the graveyard or disposal orbit." The result will be the collection of small "satlets" resembling nano satellites. The basic units will be sent into orbit via commercial satellite launch and then joined with the used parts upon arrival. Sometime in 2015, DARPA plans to demonstrate the program's viability by actually separating a retired satellite aperture using grappling tools controlled from Earth and reconfiguring it into a "new" free-flying system. To see how it all works, log onto **www.youtube.com/watch?v=uvkhWIlmHEg**. ▲

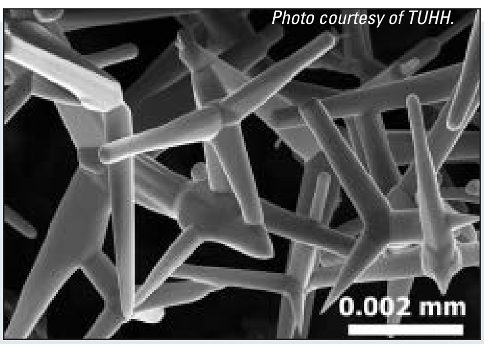


Photo courtesy of TUHH.

■ Tetrapods of zinc oxide form the basis for "Aerographite."

WORLD'S LIGHTEST MATERIAL

In a recent issue of the scientific journal *Advanced Materials*, it was reported that scientists from the University of Kiel (**www.uni-kiel.de**) and Hamburg University of Technology (**www.tu-harburg.de**) have created the "lightest material in the world," consisting of a network of porous carbon tubes that are three-dimensionally interwoven at the nano and micro levels. The material — which has a wide range of potential electrical and mechanical applications — weighs in at only 0.2 milligrams per cubic centimeter, which makes it about 75 times lighter than Styrofoam®. Nevertheless, it is said to be very strong physically. According to the developers, it is jet black, stable, and electrically conductive. It is also ductile and nontransparent. The material — dubbed "Aerographite" — is four times lighter than the previous record holder which was made from a similar nickel-based structure. Interestingly, Aerographite can be compressed 95 percent and pulled back to its original form without damage; it actually becomes stronger when exposed to stress. It also absorbs light rays almost completely, making it the blackest of black materials. As far as electronic applications go, "Aerographite could fit onto the electrodes of Li-Ion batteries. In that case, only a minimal amount of battery electrolyte would be necessary, which then would lead to an important reduction in the battery's weight." This could translate into more compact batteries for electric vehicles. In addition, it could be used in the creation of conductive plastics and for vibration damping in avionics and satellites. The material may also prove useful for filtering out water and air pollutants. ▲

Discuss this article in the Nuts & Volts forums at http://forum.nutsvolts.com.

COMPUTERS AND NETWORKING

ONE MORE OFF THE RIM

■ The new BlackBerry PlayBook with optional ($119.99) keyboard and convertible case.

Late last year, Research in Motion (RIM, **www.rim.com**) entered the tablet market with its BlackBerry PlayBook — a seven inch Wi-Fi tablet running the company's own OS. The original was priced at $499 retail but dropped to $199 a few months later after grabbing a disappointing 3.3 percent market share (compared to 61 percent for Apple's iOS devices, 30 percent for Google's Android, and 4.6 for various Windows products). However, RIM is taking another whack at it with an upgraded 4G/4G LTE version of the device that adds 4G capabilities to the existing W-Fi. Aside from enhanced 4G connectivity, the new PlayBook isn't a whole lot different. You get the same seven inch 1024 x 600 pixel display, HDMI, stereo speakers, and a pair of HD video cameras, but the processing speed has been increased from 1.0 to 1.5 GHz and the **html5test.com** site has rated RIM's OS 2.0 as the best browser for the upcoming HTML5. As of this writing, it is available only in Canada, but according to the company, "Additional variants of the BlackBerry PlayBook tablet supporting various high speed cellular networks are expected to be available in the coming months from carriers in the US, Europe, South Africa, Latin America, and the Caribbean." Quite a few eggs are riding in this basket, as RIM turned in a $518 million loss in the first quarter and is in the process of laying off 5,000 employees — about a third of its worldwide workforce. In fact, the investment advisors at 24/7 Wall Street (**247wallst.com**) have ranked RIM number four in the "Ten Brands that Will Disappear in 2013" list. Whether the new PlayBook and BlackBerry 10 will drag RIM's chestnuts out of the fire remains to be seen. ▲

■ Crucial v4 SSD lineup focuses on older computing devices.

SSDS SUPPORT OLDER MACHINES

While many solid-state drives (SSD) target high-end computers, the new Crucial (**www.crucial.com**) v4 SSDs are designed for customers who still use "mainstream SATA 3 Gb/s" systems, namely, machines built before 2011. Older machines (which actually constitute 80 percent of existing systems) have read/write speed limitations that prevent them from taking full advantage of today's fastest SSDs. Accordingly, the v4 devices are designed to cost less while matching the maximum capabilities of the older computers. The devices offer read speeds up to 230 MB/s and write speeds up to 190 MB/s, providing "faster startup times, faster application downloads, faster transfer data speeds, and increased reliability compared to traditional hard drives." The lineup includes 32, 64, 128, and 256 GB drives priced, respectively, at $49.99, $69.99, $99.99, and $189.99. Crucial drives — which are actually designed and built by the parent company, Micron Technology — come with a three year warranty and are both PC and Mac compatible. ▲

TEXTING FROM YOUR PC

It happens all the time. You put together a thoughtful, fascinating email, perhaps with attached photos or videos, and send it to a buddy. A little while later, you get a terse, cryptic, disappointing, and nearly incomprehensible response with a notice that reads something like, "This email was sent from my mobile device. Please excuse any typos." Unfortunately, the reply-to address is the recipient's email address, not the cell phone, so you can't fire back a stinging rebuke. Or can you? Well, if you know what carrier your buddy uses, you certainly can. To send emails directly from your computer to a cell phone, just use the following address configurations: T-Mobile: **phonenumber@tmomail.net**. Virgin Mobile: **phonenumber@vmobl.com**. Cingular: **phonenumber@cingularme.com**. Sprint: **phonenumber@messaging.sprintpcs.com**. Verizon: **phonenumber@vtext.com**. Nextel: **phonenumber@messaging.nextel.com**. AT&T: **phonenumber@txt.att.net**. AT&T with pix: **phonenumber@mms.att.net**. ▲

CIRCUITS AND DEVICES

BATTERY OFFERS 20+ YEARS OF OPERATION

The NanoTritium battery provides a 20+ year operating life.

Down in Homestead, FL, City Labs, Inc. (**www.citylabs.net**), has been working to revive betavoltaic battery technology which actually originated in the 1970s but didn't take off, largely because (a) lithium batteries — which are much cheaper — came on the scene shortly thereafter, and (b) betavoltaics include a radioactive element that requires a bulkier case. In addition, they don't put out much power, so applications were pretty much limited to things like pacemakers. Today's devices don't suck so much current, so the incredibly long life of these batteries (20+ years) is looking more promising. We're talking about things like sensors on deep water oil rigs, covert surveillance, silicon clocks, SRAM backup, etc. In fact, City Labs has secured a $1 million Air Force grant to develop the technology. The company's first commercial product — called NanoTritium™ — was introduced earlier this year. In addition to the long lifespan, the batteries are resistant to extreme temperatures (-50° to +150°C), as well as extreme vibration and altitude, which partially explains Lockheed Martin's interest in them. The bad news is that the little company is currently capable of building only engineering quantities of them (1,000/year or so), and they cost "a couple thousand" apiece. Prices should drop as manufacturing ramps up. ▲

DEALS ON USED EQUIPMENT

PhotoMachining, Inc. (**www.photomachining.com**), specializes in precision laser micromachining of parts that can't be produced any other way, which is probably of low interest to most *Nuts* readers. However, the company also resells tons of used equipment (snapped up as excess inventory from downsizing corporations) and sells it at a discount. For example, how about a $9,000 LeCroy dual 350 MHz oscilloscope with cart for $1,200? Or, a $2,580 Cole-Palmer temperature calibrator/simulator for $600? And so on. The inventory is updated weekly, so if you're in the market for bargains on such instruments and devices, it's worth a look. ▲

YOU WON'T BE BLOWN AWAY

This month's award for the dumbest electrical gadget goes to the Talme iPhone dock fan which plugs into various iPhone models, as well as the iPod Touch generations 1 through 4. According to the manufacturer, it runs nearly six hours continuously before draining the battery, so it probably moves about enough air to cool off a damp Etruscan pygmy shrew. On the other hand, if you need to buy someone a present that makes you look clever and thoughtful while costing next to nothing, this could be the answer. You can pick one up on **Amazon.com** for only $7.99. For some reason, people who purchase the dock fan also tend to buy a factory-reconditioned Homelite ZR09510 two-cycle blower. Go figure. ▲

The Talme iPhone dock fan.

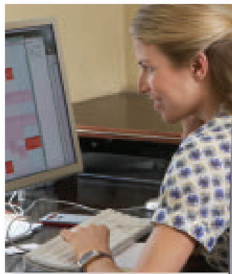
INDUSTRY AND THE PROFESSION

MICROSOFT SNATCHES UP ANOTHER ONE

It has become pretty obvious that those of us who did not spend our college days cloistered in dorm rooms developing social networking software missed a big financial boat. Further evidence is provided by Microsoft's recent purchase of Yammer (**www.yammer.com**) for a nifty $1.2 billion in cold, hard cash. Yammer — in case you've never heard of it — sells cloud-based social networking software for workplace use, "including employee profiles, activity streams, discussion forums, microblogging, wikis, idea generation software, joint document sharing, and editing, as well as tagging, rating, and reviewing of content." The company claims about 200,000 users, which means that Microsoft spent $6,000 per user to acquire the company. **NV**

With APC Back-UPS, your digital life goes on...







even when the power goes off.

Preserve what's most important to you.

Reliable power backup for 24/7 availability

Whether DVRing your favorite show, updating your Facebook status, or playing a live networked game, you depend on your home electronics every day, all day. That's why APC™ by Schneider Electric™ has designed battery backup solutions that protect the constant availability and connectivity you expect … and depend on.

Peace-of-mind protection on two levels

When the power goes out, our popular Back-UPS™ units go to work. They instantly switch your home technologies to emergency power, allowing you to work through brief power outages or safely shut down your systems so you won't lose valuable files — such as digital photos and media libraries. They also feature surge outlets to guard your electronics and data from "dirty" power and damaging power surges — even lightning. So you get two levels of protection in every APC Back-UPS unit!

Energy-saving insurance for what matters most

Our Back-UPS units protect your home office, digital living and home media applications, notebook computers, DVRs, and gaming application. And since we now offer energy-efficient models that reduce electricity costs through unique power-saving outlets, you can realize true energy savings regardless of the applications you're backing up. Throughout your home, the APC Back-UPS is the cost-saving insurance you need to stay up and running and reliably safeguarded from both unpredictable power and wasteful energy drains.

APC power protection products are available at: BEST BUY OfficeMax STAPLES that was easy. Office DEPOT

Keep your electronics up and your energy use down!

ES Series

The ever-popular ES models are priced affordably yet provide enough extended runtime to allow you to work through short and medium power outages. Some power-saving models have been designed to actively reduce energy costs.

The energy-efficient ES 750G

The ES 750G boasts innovative power-saving outlets, which automatically shut off power to unused devices when your electronics are turned off or asleep, eliminating wasteful electricity drains.

- 10 outlets
- 450 watts/750 VA
- 70 minutes maximum runtime
- Telephone/network protection

The best-value ES 550G

The ES 550 uses an ultra-efficient design that consumes less power during normal operation than any other battery backup in its class, saving you money on your electricity bill.

- 8 outlets
- 330 watts/550 VA
- 43 minutes maximum runtime
- Telephone/network protection

Power up to WIN one of seven APC BR700G Battery Back-UPS units (a $130 value)!

Visit www.apc.com/promo Key Code r645v • Call 888-289-APCC x4970 • Fax 401-788-2797

APC™ by Schneider Electric

PICAXE PRIMER

■ BY RON HACKETT

SHARPENING YOUR TOOLS OF CREATIVITY

MEASURING A PICAXE PROCESSOR'S INTERNAL TEMPERATURE

In this month's Primer, we're finally going to take a look at the new M2-class *readinternaltemp* command. However, we also have some unfinished business to discuss. As you may remember, the subroutine that we've been using to convert the DS18B20 data (which is reported in degrees Celsius) to Fahrenheit only works for temperatures that are greater than or equal to 0°F. Since the DS18B20 has a temperature range of -67°F to +257°F, we need to modify our conversion subroutine so that it can handle the entire range. So, let's take care of our old business before we move on!

MENDING OUR WAYS

According to Wikipedia, there are only three countries in the world that still use the Fahrenheit scale as the official way to measure temperature; everyone else has converted to the Celsius scale. Unfortunately, the United States happens to be one of the three holdouts. (The other two are the Cayman Islands and Belize.)

As a result, most readers of this column have to do the extra work of converting from Celsius to Fahrenheit before we can understand our temperature data. Some might say that it serves us right, but we won't get into that here.

In order to understand the task that's facing us, you may want to refer to **Figure 1** which presents four columns of equivalent temperature measurements. We'll soon see why I chose the specific measurements in the **figure**, but before we get to that, let's focus on each of the four columns, in order. The "DS18B20" column contains the data that's returned by the *readtemp* command for the six temperatures presented in the "Celsius" column.

Don't forget, the DS18B20 automatically sets bit 7 of its data byte to 1 whenever the Celsius reading is negative. Of course, bit 7 of a binary number has a decimal value of 128, which explains the relatively large DS18B20 values that correspond to the four negative Celsius temperatures that are listed.

The values in the "Fahrenheit" column are simply the result of converting from Celsius to

DS18B20	Celsius	Fahrenheit	PICAXE
125	+125	+257	+257
0	0	+32	+32
129	-1	+30	+31
146	-18	0	0
147	-19	-2	-2
183	-55	-67	-67

■ FIGURE 1. Selected Equivalent Temperature Measurements.

Fahrenheit, and rounding to the nearest whole degree. If you want, you can double-check my arithmetic by using the standard conversion formula, or any one of the many temperature conversion sites on the Web.

The "PICAXE" column displays the corresponding results that we will obtain when we actually program a PICAXE to do the math for us. The only reason I've included the PICAXE data is to demonstrate the small "errors" that can result from the fact that PICAXE arithmetic is integer based; no decimals or fractions can be stored along the way.

Also, PICAXE BASIC doesn't round the result of a computation to the nearest integer; it just drops the fractional remainder altogether. For example, consider the following line of code:

```
b0 = 9 / 5
```

When the above statement is executed, the *b0* variable will contain 1 (not 2) because the remainder of 4 is simply discarded. Consequently, in a PICAXE program, some Celsius to Fahrenheit conversions will include a small error (e.g., the 31 rather than 30 in the "PICAXE" data in **Figure 1**). An error of 1ºF has never been a problem in any of my projects, but if you need greater accuracy in one of yours, you can always use the *readtemp12* command.

As you can see in **Figure 1**, the data spans the entire range of the DS18B20's measurement capabilities. The values in the first row (C = +125º and F = +257º) each represent the maximum temperature that the DS18B20 is able to measure, and the values in the last row (C = -55º and F = -67º) each represent the minimum temperature that the DS18B20 is able to measure.

Also, I've divided the data into three sections of two rows each. If you examine the data in each of the three sections, you can see how they differ. In the top two-row section, the Celsius and Fahrenheit values are all positive, or 0; in the middle two-row section, the Celsius values are negative, and the Fahrenheit values are positive, or 0; in the bottom two-row section, the Celsius and Fahrenheit values are all negative.

■ FIGURE 2. DS18B20 Data Subset Summary.

DS18B20 Data	Degrees C	Degrees F
0 through 125	>= 0	> 0
129 through 146	< 0	>= 0
147 through 183	< 0	< 0

As you may remember from our earlier Celsius to Fahrenheit conversions, we had to implement our PICAXE computations differently for positive and negative values, so the three data-sections in **Figure 1** provide a clue as to how we need to proceed to implement a conversion subroutine that can handle the entire range of DS18B20 measurements.

In order to clarify our task, **Figure 2** summarizes the important aspects of the data presented earlier in **Figure 1**. It also shows an interesting little anomaly; the DS18B20 data ranges between 0 and 183, but it will never have a value of 126, 127, or 128. However, for our purpose (i.e., converting C to F), the main point is that we need to perform different computations on the three data subsets (0 through 125, 129 through 146, and 147 through 183).

Of course, the programming structure that's most suitable for this task is a *select case* statement, so that's what we'll use in our first program this month. Essentially, our conversion subroutine will look something like this:

```
select case tempC
  case   0 to 125
    …
  case 129 to 146
    …
  case 147 to 183
endselect
```

Before we actually discuss the program in detail, there's another issue that needs to be mentioned. Unless you happen to work in a very well equipped scientific laboratory, there's no way we can actually test our program at every temperature between +257ºF and -67ºF in order to be sure that it functions correctly. Fortunately, there's a simple alternative; we'll have the program generate every possible DS18B20 data byte, and compute the necessary conversion for each one of them. Also, we'll want to see each equivalent Celsius and Fahrenheit value side-by-side, so rather than using our two-digit LED display, we'll just send the data to the terminal window.

With all that in mind, go to the *Nuts & Volts* website, download the three programs we'll be using this month, and print out a copy of each for reference during the remainder of this month's article.

Figure 3 shows a photo of the breadboard setup that I used to test the program. As you can see, it doesn't require any external circuitry at all; the data is sent back to the terminal window via the processor's *serial out* programming line.

When you download the program to your PICAXE and run it, the data will scroll up in the terminal window at a rate that's much too fast

In spite of its limitations, *readinternaltemp* certainly has its uses.

The first thing to note about the *TempRangeTest.bas* program is that it includes comment numbers in brackets along the right edge of the printout (e.g., "[1]"). The following discussion will clarify the 11 program lines that include comment numbers.

[1]: At this point in the program, we have already declared names for the *w0* and *w1* variables, so variables *b0* through *b3* have already been used. Therefore, we'll start our byte variable declarations with variable *b4*.

[2] and [4]: As I mentioned earlier, the DS18B20 data will never have a value of 126, 127, or 128, so we will use two separate *for/next* loops to generate all the possible data values (and none of the impossible values). In order to make the generated data easier to read, the first loop (at [2]) starts at the highest possible temperature, and works down through the count. The second loop (at [4]) may seem to be counting back up, but it's processing negative temperatures (because bit 7 has been set to 1). Larger negative values represent lower temperatures, so we're still counting down through the data.

[3] and [5]: In both *for/next* loops, the first thing we do is assign the current value of *index* to the *tempC* variable. This is necessary because each time through the *for/next* loop our *ConvertC2F* subroutine is called, and the value of *tempC* is changed in the process. If we didn't maintain *index* and *tempC* as two separate variables, the program would behave in an extremely unpredictable and erratic manner. (If you're not convinced of this, try using *tempC* in place of *index*, and then "comment out" statements [3] and [5] and watch what happens!)

[6]: If we didn't include an *end* statement at this point, the program would "fall through" to the *convertC2F* subroutine, which would also produce erratic results.

[7]: In the first *case* clause of the *select case* statement, all the DS18B20 data values that are processed represent positive (or 0) Celsius values, so we can use the standard version of the C to F conversion formula.

[8]: In the second *case* clause, all the DS18B20 data values that are processed represent negative Celsius values. Also, each time the compiler executes the program line tempF = tempC * 9 / 5, the result is less than or equal to 32 (don't forget, the fractional part of the result is dropped), so we subtract the magnitudes as shown in line **[9]** to get the final value. If this explanation is confusing, use the smallest DS18B20 value in this case (129), and do the computations in each step by hand; you should end up with a "PICAXE" value of +31º F (see **Figure 1**).

Then, do all the computations again, this time using the largest DS18B20 value in this *case* clause (146); you should end up with a "PICAXE" value of 0º F. (Again, see **Figure 1**.)

[10]: In the third *case* clause, all the DS18B20 values that are processed again represent negative Celsius values, but this time the computed value of tempC * 9 / 5 is always greater than 32, so we subtract the magnitudes as shown in line [11] to get the final value. Again, if this explanation is confusing, use the smallest DS18B20 value in this case (147), and do the computations in each step by hand; you should end up with a "PICAXE" value of -2º F.

Then, do all the computations again, this time using the largest DS18B20 value in this *case* clause (183); you should end up with a "PICAXE" value of -67º F.

When you understand how the program works (or, you just need a break), we can move on to actually running the program and observing how it functions. I used a PICAXE-08M2 processor for this purpose, but you can use any M2 processor you want — just be sure to edit the program's *#picaxe* directive accordingly.

to read. However, the program only takes about 10 seconds to run, and when it's finished you can use the scroll bar to go back and examine all the data that was generated.

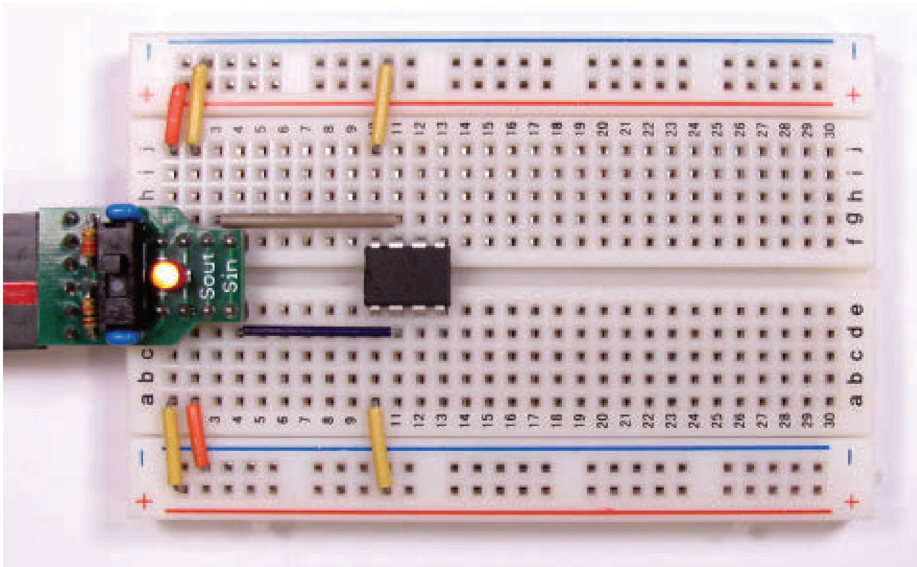
This is also a good time to mention that there is a "Copy Input Buffer" option in the terminal window's Edit menu. You can use this option to copy all the data and then paste it into a word processor or text editor file so that you can save it and/or print it if you want.

Once you have examined the data and confirmed that the program is operating correctly, the *ConvertC2F* subroutine can be used in any project that employs the DS18B20 for temperature measurement. All you need to do is include the necessary variable definitions in your program.

USING THE *READINTERNALTEMP* COMMAND (FINALLY!)

Now that we've taken care of our "old business," we're ready to move on to the new M2-class *readinternaltemp* command. However, before we get into the details, it's important to note a couple of significant limitations of this command. First, it's intended to provide an approximate indication of the PICAXE processor's internal temperature; "It is designed to be used as a cooling failure warning threshold device, not an accurate temperature sensor!" (Refer to the PICAXE Manual, Section 2, *readinternaltemp* documentation.)

Results may vary significantly from one processor to another. In other words, if you have calibrated the command to provide a reasonably accurate temperature measurement with one specific processor and then run the same program on a different M2 processor (even one of the same size), the results may vary significantly. Finally, because the sensing device is embedded in the processor's



■ FIGURE 3. Breadboard Setup for *TempRangeTest.bas* Program.

relatively large plastic case, the internal sensor responds more slowly than the DS18B20 to changes in the ambient temperature.

In spite of the above limitations, I think that the *readinternaltemp* command is accurate enough to be useful in some projects, so let's discuss the details so you can form your own opinion of its usefulness. The complete syntax of the command is *readinternaltemp voltage, offset, variable*. Its three parameters are defined as follows:

Voltage is a constant that specifies the power supply voltage. For a +5V supply, "IT_5V0" should be used; for other supply voltages, see the *readinternaltemp* documentation in Section 2 of the PICAXE Manual.

Offset is a positive or negative correction factor that can be used to calibrate the temperature reading. The *readinternaltemp* documentation states that the *offset* parameter is optional, and that it defaults to 0 if it's not included. However, every time I leave it out, I get a syntax error. If the same thing happens to you, just use a 0 for *offset* until you can arrive at a value that improves the accuracy, which we are about to do.

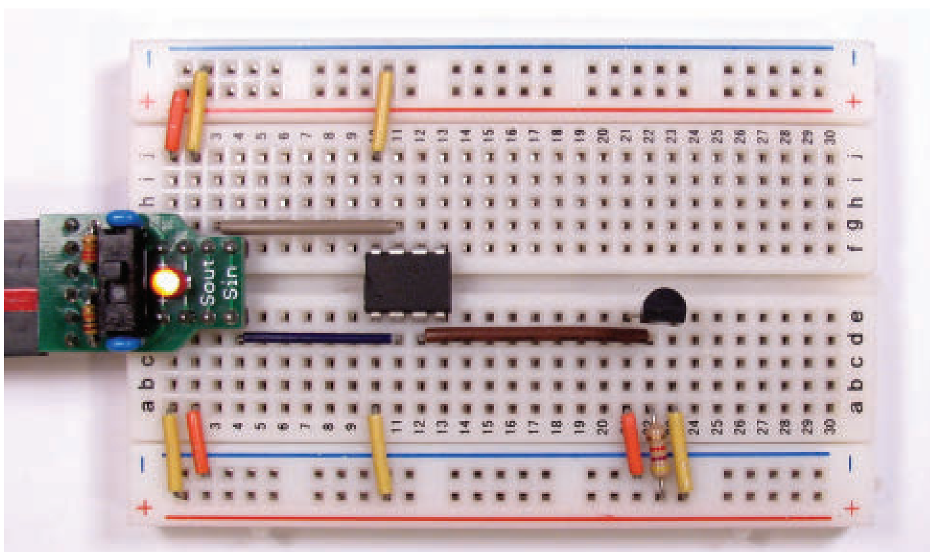
Variable is the variable that you want to use to store the temperature data. The documentation doesn't specify a byte or word variable, but I've been using a word variable just to be safe.

Using the above explanation of the *readinternaltemp* command, here's one example of a complete command: *readinternaltemp IT_5V0, 0, tempC*. In this example, we're using a +5V supply, and *tempC* has been previously declared as a word variable in the program.

The breadboard setup that I used for my initial experiments with the *readinternaltemp* command is shown in **Figure 4**. It's the same setup we just used in **Figure 3** above, with the addition of a DS18B20 that has its data output connected to pin C.4 of the 08M2 processor. The purpose of the DS18B20 is to serve as a reasonably accurate temperature reference for use in calibrating the *readinternaltemp* command. (Don't forget that we need to include a 4.7K pull-up resistor between the DS18B20 data line and +5V in order for the sensor to function correctly.)

We'll use the *TempIntExtCalib.bas* program to calibrate the *readinternaltemp* command. The program is fairly self-explanatory; all we're doing is taking an internal and external temperature reading, and sending both of them to the terminal window so that we can see how the two readings compare. However, I do need to explain how to use the program to calibrate the *readinternaltemp* command.

Essentially, we need to take a "trial-and-error" approach to finding the *offset* value that will produce the same internal and external temperature readings. In other words,



■ FIGURE 4. Breadboard Setup for *TempIntExtCalib.bas* Program.

	offset	tempI	tempE	Adjust
Run #1	0	301	24	-277
Run #2	-277	3	24	+21
Run #3	-256	25	24	-1
Run #4	-257	24	24	0

■ FIGURE 5. Results from the *TempIntExtCalib.bas* Program.

we'll start with an *offset* value of 0, run the program, and use the two temperature readings we obtain to adjust the value of *offset* so that the resulting internal temperature reading becomes closer to the (accurate) external temperature reading. We'll repeat this adjustment process until we get the same internal and external temperature reading.

Figure 5 is a summary of the results that I obtained for my 08M2 processor. The first time I ran the program, I used 0 as the value of the *offset* parameter which resulted in an internal temperature reading of 301ºC. The DS18B20 reported an external temperature reading of 24ºC. We know that the DS18B20 external reading is accurate, so we need to adjust our internal reading to match 24ºC. In other words, we need to lower the *offset* value by 277, so I changed its value from 0 to -277 for the second run of the program.

In Run #2, I obtained the internal and external readings you see in **Figure 5**. This time, the internal temperature was 21º too low, so I added 21 to the *offset* value of -277 which increased it to -256. Using that value for *offset* in Run #3, the internal temperature reading became 1º too large, so I adjusted *offset* down a notch to -257. Using that value for *offset* in Run #4, the internal and external temperatures matched exactly, completing the calibration process.

Download the *TempIntExtCalib.bas* program to your breadboard setup, and implement the above process in order to determine the value of *offset* that works for your processor. I carried out the calibration process with a total of four different processors (three 08M2s and one 20M2), and obtained *offset* values of -257, -239, -254, and -264, respectively, so there's a fair amount of variation among different processors. Of course, that means that any time you change processors, you will need to re-calibrate your internal temperature measurement program.

Min	Internal Celsius	External Celsius	External (Raw)
0	24	24	24
1	20	9	9
2	17	0	0
3	15	-3	131
4	13	-5	133
5	11	-7	135
6	10	-9	137
7	9	-10	138
8	8	-12	140
9	7	-12	140
10	6	-12	140
11	6	-12	140
12	6	-12	140
13	6	-13	141
14	5	-13	141
15	5	-13	141
16	5	-13	141
17	5	-13	141
18	5	-13	141
19	5	-13	141
20	5	-13	141

■ FIGURE 6. Truncated Results from the *TempIntExtCold.bas* Program.

If that were the whole story, the *readinternaltemp* command would be very accurate; unfortunately, that's not the case. A processor that has been calibrated at one specific temperature will be out of calibration at any other temperature, and the greater the temperature change, the greater the inaccuracy.

To demonstrate this phenomenon, I disconnected my breadboard setup from my desktop computer, re-connected it to my laptop, and moved everything into the room where my basement refrigerator/freezer is located. I also wanted to demonstrate the fact that the internal temperature readings take longer than the external readings to adjust to changes in temperature, so I wrote another program that records internal and external temperature readings once a minute for an hour. The resulting program (*TempIntExtCold.bas*) does require some explanation, mainly because I changed the data format somewhat.

My goal was to format the resulting data so that it could be easily copied and pasted directly into an Excel spreadsheet. As before, the numbers preceding the following comments correspond to the numbers along the right edge of the program listing:

[1]: Tab-delimited text is easy to place in an Excel spreadsheet, so we're defining the variable *HT* as 9 because that's the ASCII code for a horizontal tab. We'll use this in the *sertxd* statements of the program.

[2]: Here, we're sending the three column headings, so that extraneous text doesn't appear in each of the program's data transmissions to the terminal window.

[3]: The main *for/next* loop is executed once per minute for one hour. In each iteration of the loop, we read the internal and external temperature, and send the data to the terminal window.

[4]: In the April '12 Primer, we discussed the M2-class *time* variable which is a built-in system variable that automatically updates an internal seconds counter. (Note that we didn't declare *time* as a variable in the program; the *time* variable is built in, so trying to declare it in a program will cause a syntax error.) At the beginning of each iteration of the main *for/next* loop, we initialize *time* to 0. Then, we send the value of the minute counter and the two temperature readings — each separated by a tab character.

[5]: At the end of the main *for/next* loop, the empty *do/loop* simply loops back on itself until *time* has automatically incremented to 60 in the background. As a result, the main *for/next* loop executes about once per minute. As we discussed in the April installment, there is usually a small error in the rate at which the *time* variable increments, but for our current purpose, all that matters is that the two temperatures are recorded at the same time; an interval that's very close to one minute is fine.

When I was sure the *TempIntExtCold.bas* program was running correctly, I carried out the following procedures to collect my temperature data:

First, I placed my laptop next to my basement refrigerator. With the freezer compartment door opened, I started the *TempIntExtCold.bas* download to the 08M2, and then placed the entire breadboard setup (which was still connected to my laptop) in the freezer compartment and shut the door just as the download completed.

Next, I took a well-earned break for about an hour so that the program could run its course. (Actually, I did peek at the data once or twice!)

When the program completed running, I removed the breadboard from the freezer and used the terminal window's "Copy Input Buffer" option to copy all the data.

Finally, I opened a new Excel spreadsheet and pasted the data into it. Each data byte was automatically placed in a separate cell, so all I had to do was save the spreadsheet. (Nice and easy!)

Figure 6 presents the data that I obtained from the first 20 minutes of my little experiment. (The data for minute 20 stayed essentially the same for the remaining 40 minutes, so I truncated the table in **Figure 6**.) The first point that requires explanation is the data in the "External (Raw)" column on the right edge of **Figure 6**. When I originally wrote the *TempIntExtCold.bas* program, I wanted to keep it as simple as possible, so I decided not to include any data conversion at all. Consequently, the DS18B20 data values were all in the "raw" form which is shown in the "External (Raw)" column. (Don't forget, for any DS18B20 temperature reading that's below 0ºC, bit 7 is automatically set to 1 which increases the data value by 128.) However, in order to make the external data more directly comparable to the internal data, I subtracted 128 from all the below-zero values in the "External (Raw)" column, and placed the result (along with a "-" sign) in the "External Celsius" column.

In order to interpret the data of **Figure 6**, just ignore the "Raw" column, and focus on the two "Celsius" columns. If you do that, two significant points become clear:

The internal temperature readings took much longer to adjust to the freezer's temperature that did the external temperature readings. For example, after two minutes, the internal temperature reading had only decreased by 7ºC; in the same two minutes, the external temperature reading had decreased by 24ºC.

After 20 minutes when the internal and external readings both had stabilized, the internal reading was 5ºC (41ºF) and the external reading was -13ºC (9ºF). At that point, my freezer alarm (which is still working fine) indicated the freezer temperature was 10ºF. So, even though I had calibrated the *readinternaltemp* command at a room temperature of 24ºC, it was way off the mark in the much colder environment of my freezer.

The conclusion is obvious and clearly stated in the documentation for the *readinternaltemp* command in Section 2 of the PICAXE Manual: "... this system can never be an accurate sensor and should only be used as an indicator of extreme temperature ..."

However, in spite of its limitations, *readinternaltemp* certainly has its uses. For example, it can be used to implement a warning device for cooling failure (as mentioned in the PICAXE Manual) or as a warning device for below-freezing temperatures in an unoccupied building (as I plan to do for my outdoor shed this winter).

In any application of the *readinternaltemp* command, the most important thing to remember is to calibrate the command at a temperature as close as possible to the critical temperature you want to detect. That way, it will provide an accurate reading at the critical temperature which is the point at which you will want your program to take appropriate action.

That's it for now — see you next time! **NV**

QUESTIONS & ANSWERS

WITH
RUSSELL KINCAID

In this column, Russ answers questions about all aspects of electronics, including computer hardware, software, circuits, electronic theory, troubleshooting, and anything else of interest to the hobbyist. Feel free to participate with your questions, comments, or suggestions. **Send all questions and comments to: Q&A@nutsvolts.com**

EXTRA LOUD CAR TURN SIGNAL

Q My uncle is losing his hearing and has trouble knowing when he has left the car turn signals on too long. He drives an older car (1989 Buick). I located and replaced the turn signal flasher under the dash with an extra loud unit. It is a small round plug-in type. Even with this new louder unit, he still cannot hear the "clicks."

I was hoping that I could build a unit that would attach to his car with a small loudspeaker and maybe even an adjustable volume control so he can hear the clicks. I thought it might help to have a small box with the speaker and some bright LEDs for the left and right signal that I could mount on the dash. Not sure if such a device can be built, but if it could plug in where the flasher plugs in, that would make installation easy.

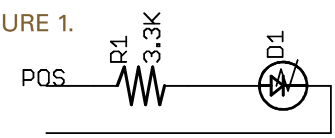
A way to control the speed of the turn signal flashes would be a really nice addition to this device. Probably asking for too many features in one unit, but any ideas you might have would be very welcome.

— Guy Fischetti

A I note that your uncle's car has a sloping windshield and a flat dash where it will be easy to mount an LED which will reflect off the windshield where it will be in the field of view but not obstruct the view. On the theory that the simplest solution is the best one, I offer this suggestion: Solder some #24 hookup wire to the pins of the flasher near the base so you can still plug it in. Run the wires up the side of the dash, and mount the circuit in **Figure 1** on the top of the dash. The LED can be mounted on a project board, Plexiglas, or wood, and attached with screws or doublesided carpet tape.

You will need to measure the voltage at the flasher socket to know which is positive. The flasher doesn't care about polarity, but the LED does.

FIGURE 1.



REGENERATIVE SHORTWAVE RADIO

Q On my 13th birthday, I received two P-Box electronic kits from my father: a regenerative shortwave radio #28-110 and a telephone pickup amplifier #28-116 — now discontinued by RadioShack — which got my interest in electronics.

The regenerative shortwave radio worked well enough that I decided to connect it to the amplifier, entirely eliminating the crystal earphone. Both of these P-Box kits were using germanium transistors (2SB54/2SB56) which are now hard to find and rather expensive when available.

I have three questions — mutually exclusive of each other — about this radio kit:

a) Is germanium transistor 2SB77 an excellent replacement for the transistors (2SB54/2SB56) used in the shortwave radio and amplifiers?
b) Is euro-transistor AC-125 an excellent replacement for these transistors?
c) Can you redesign/modify the basic shortwave radio diagram so that the cheaper and readily available silicon transistors can be used instead of germanium transistors? If, however, the modification/conversion to such silicon transistors would greatly complicate the regenerative shortwave radio, please simply state it's not worth the re-work because it would take away too much from the original design.

— Nate

A The 2SB77 is an equivalent transistor to replace the 2SB54/56. The AC125 is a slightly better transistor (higher Ft).

The original schematic of the radio is shown in **Figure 2**. The Q1 circuit has a good bias arrangement; it will work with silicon transistors (PN2222) with no changes. The audio amp, Q2, Q3, will work better if the following changes are made: Remove R10 and change R9, R4 values to 100K. The DC voltage at the Q2 collector should be between four and seven volts. PN2907 PNP transistors can be used.

TEMPERATURE SWITCH QUESTION

Q I have a question about a temp switch. The circuit I'm sending you was reversed-engineered by me, and I believe it is correct.

The problem I have is getting the output to switch. According to the information I received, R7 is a 150K ohm negative temperature coefficient sensing thermistor.

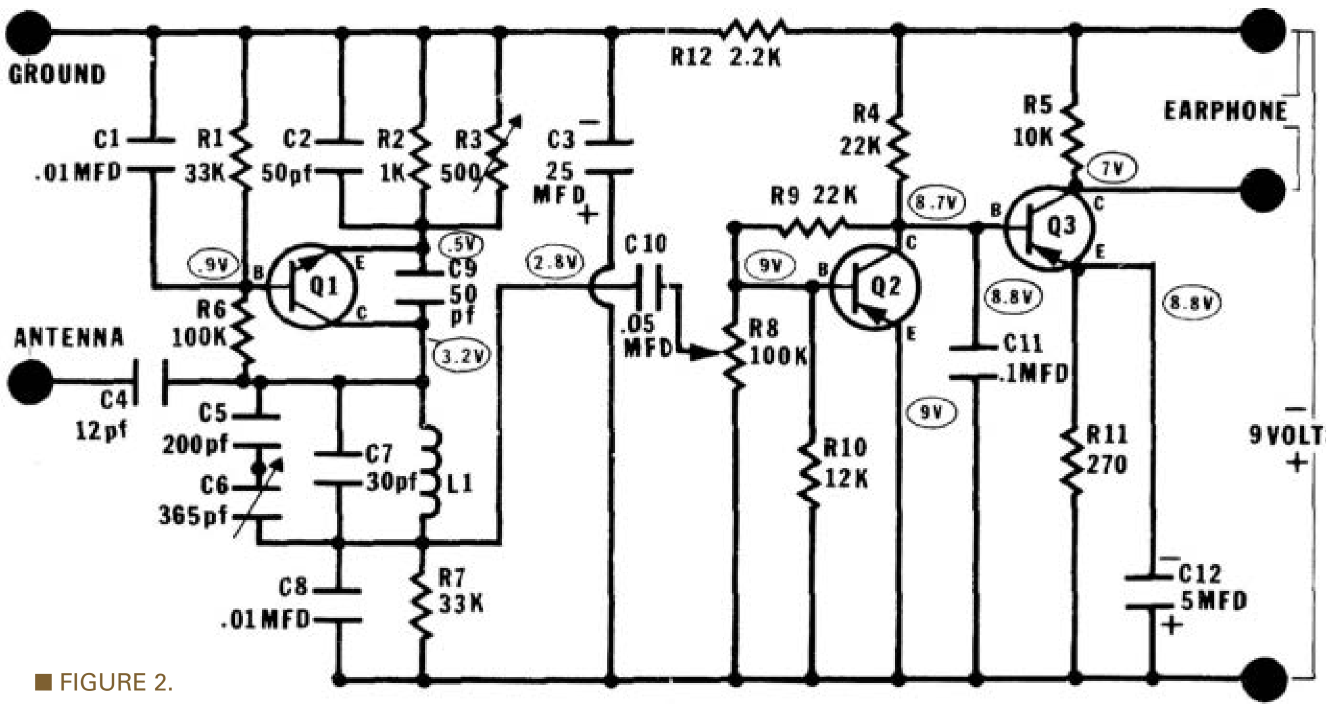
The inverting input isn't even close to the non-inverting input which should be around 10 volts DC. In order for the switch to work, the inverting input has to be higher than the non-inverting input. I believe the op-amp is being used as a comparator. What I had to do was change the R9 value to 160 ohms. After I changed the value, the switch switched correctly, but I don't understand what has changed in the circuit. Any information you could provide would be helpful. By the way, the original op-amp was a U741 but I couldn't get it to work, so I substituted a LM324 and now the simulation works.

I'm new to circuit simulation but I am including the file. By the way, I'm using VisualSpice for simulation.

— Jeff Miller

A I could not read the .ckt file because I do not have VisualSpice, but the circuit diagram that you sent is sufficient. I redrew the schematic so it was easier for me to follow; see **Figure 3**.

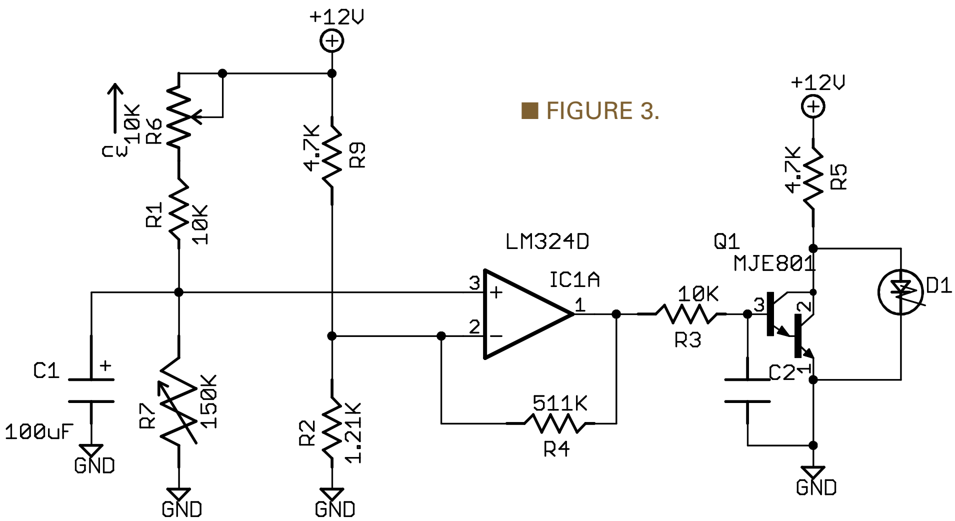
In this circuit, the reference voltage at the negative input is 2.45 volts; the positive input has to be pulled down below that to switch. At normal temperature (where the thermistor is 150K), the voltage at the positive input is 10.5V or higher. The common mode limit of the UA741 is



■ FIGURE 2.

Q1 = Transistor NPN 2N5223
Q2 & Q3 = Transistors PNP 2SB54
R1 & R7 = 33K Resistors
R6 = 100K Resistor
R2 = 1K Resistor
R12 = 2.2K Resistor
R4 & R9 = 22K Resistors
R10 = 12K Resistor
R5 = 10K Resistor
R11 = 270 ohm Resistor

C1 & C8 = 0.01 Disc Ceramic Capacitors
C2 & C9 = 50 pF Disc Ceramic Capacitors
C4 = 12 pF Disc Ceramic Capacitor
C5 = 200 pF Disc Ceramic Capacitor
C7 = 30 pF Disc Ceramic Capacitor
C10 = 0.05 Disc Ceramic Capacitor
C11 = 0.1 µF Mylar Capacitor
C12 = 5 µF Electrolytic Capacitor
C3 = 25 µF Electrolytic Capacitor
R3 = 500 ohm Potentiometer
R8 = 100K Potentiometer
C6 = 365 pF Variable Capacitor



■ FIGURE 3.

9V, so it might get hung up. The thermistor has to get below 5K in order to switch; that is a wide temperature range. What you accomplished by reducing R9 to 160 ohms was to raise the reference voltage so the circuit would switch with a smaller temperature change. With the 10K pot at minimum, the circuit will switch when the thermistor gets to 70K.

A better solution will be to raise the value of R1. If R1 is 270K and R9 is 4.7K, the circuit will switch when the thermistor gets to 70K, and the common mode voltage is

MAILBAG

Dear Russell: Re: Digital Clock, July 2012, pages 23-24. While your schematic for a digital clock will work, it is extremely complex. What you should have suggested would be to use a "clock on a chip" such as the old — but still available — National Semiconductor MM5314N. It is very easy to build a six digit clock with either a 12 or 24 hour display, and it can be optioned with a switch for 50 or 60 Hz operation. It is easily interfaced with transistors or with seven-segment LEDs or Nixie driver chips.

I have one that was built around 1979 and it is still working to this day. It has never needed any repairs of any kind. The datasheet is at **http://html.alldatasheet.com/html-pdf/117107/NSC/MM5314N/54/1/MM5314N.html**.

— Bruce Bubello WA2HWV

Response: Thanks for letting me know about this IC, Bruce, but I did not go looking for such a device because there is no challenge in that.

Dear Russell: Re: Three-Phase Motor Control, July 2012, page 22. The circuit you designed seems very useful, especially for a large three-phase induction motor that could be used as a driver in an electric car, as I envisioned it being applied. It seems that the circuit would act like a stepper motor driver. What is needed to turn this into an electric vehicle controller? I think I understand the timing angles using the 4040, but some more explanation would help. I don't know what use Mr. McGinnis intended for the circuit, so I pose my questions as if the application was for a three-phase AC electric car motor.

I regularly read *N&V* and do some projects once in a while. This one tripped my curiosity. Only some of the major uP, uC articles tantalize me but at the same time scare me. Thank you, Russ, for all your efforts in keeping me continually interested in the things I have the most fun doing. Thanks for the book titles.

— David Jenks

Response: I had not considered this to be an automotive application because 60 Hz three-phase motors would not work over a very wide frequency range (although I know next to nothing about three-phase motors). With an automatic transmission to limit the frequency range, it could work. I got a washing machine motor to try to find out how it works, but still don't know. The schematic appears to be three-phase but the signals are more like a stepper. A motor like that — but 20 to 30 times bigger — could power a car. If I were to build an electric car, I would use a DC motor because it is more easily understood and speed control is easy — just vary the voltage. The starter motor in a five liter V8 is 3 hp and draws 200 amps; boost that to 120 volts, 200 amps and you have enough power for a small car.

The three-phase generator that I designed is just a bunch of time delays to turn the switches on and off at the right time. Some 555 timers could have done the job but not be as accurate.

Thanks for writing.

Dear Russell: Re: Unijunction Transistor, August 2012, page 20. Thought you should know that one of *Nuts & Volts* longest running advertisers has been stocking and selling the 2N2646 for years, at a reasonable price. How 'bout a plug!

— Vic Neumark
ELECTRONIX EXPRESS

Response: Glad to do it, thanks for letting me know.

well within the limits.

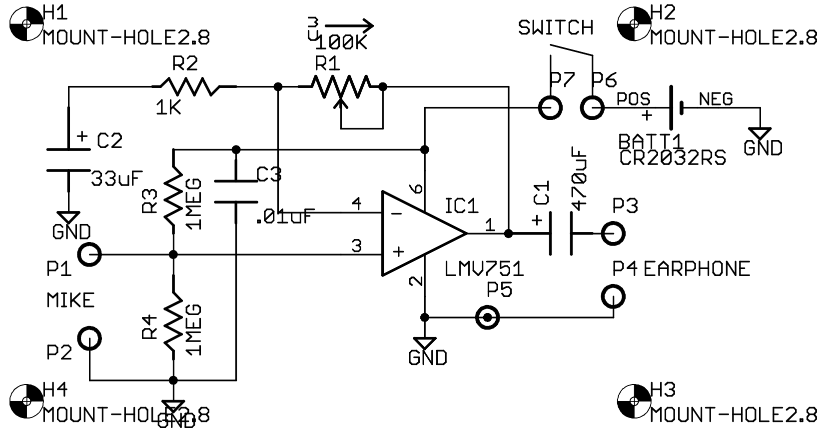
STETHOSCOPE AMPLIFIER

Q I am attempting to build a simple but functional amplified stethoscope for a hearing impaired 11 year old. She has hearing aids in both ears and is virtually deaf without them. The hearing aids prevent standard earpieces from being inserted into her ears. Purchasing a custom commercial unit is cost-prohibitive. I am using a disposable stethoscope, as is found in many hospitals. My idea is to place a small electret mic behind the diaphragm, route the wires through the tubing, and insert a small box with an amplifier at the end of the sensor tube where it "Ys" into the earpiece tubes. A set of over-the-ear headphones would plug into the amplifier. The scope could be powered with a few button cells, and adding an on/off switch and gain control for the amp would complete the system. The amp should have 10-20 db of gain.

Research on stethoscopes indicates that heart and respiratory sounds are typically between 20 and 600 Hz, so those frequencies are all that need amplification — higher frequencies would only contribute to feedback.

Any suggestions you have would be much appreciated.

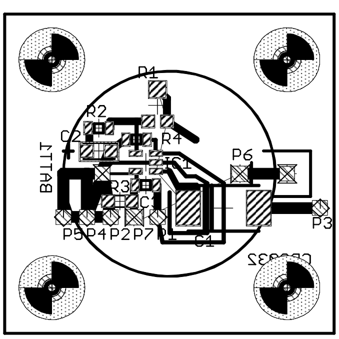
— Joe Pitman

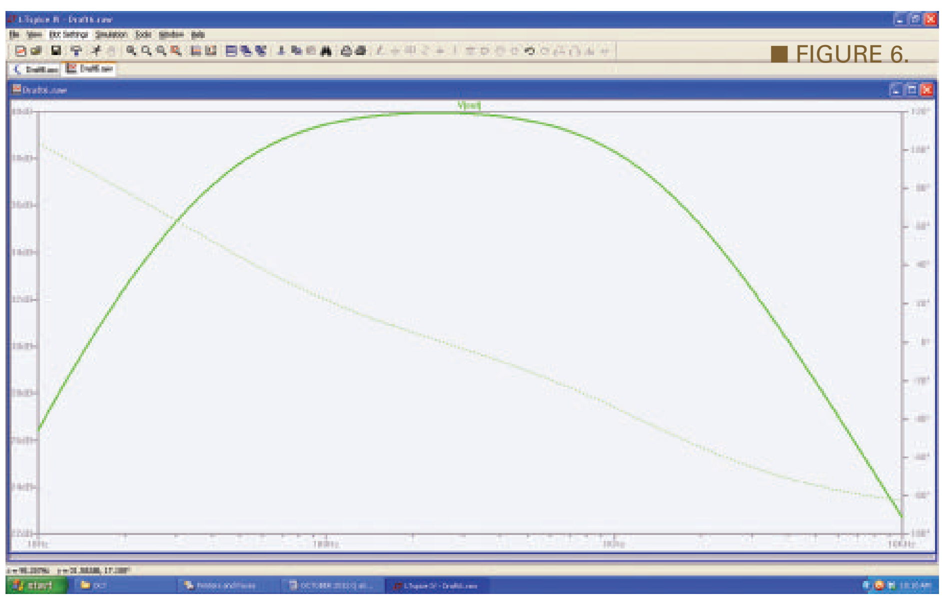


■ FIGURE 4.

A The circuit of **Figure 4** is a single stage non-inverting amplifier which provides high impedance input and low impedance output. The output capacitor is sized to allow the use of eight ohm headphones, but higher impedance will use less power and give better low frequency response. The 100K pot in the feedback allows the gain to be adjusted from 0 dB to 40 dB. The circuit is intended to operate from a single 2032 lithium cell which should last about 100 hours. There is no input capacitor because the electret is a capacitor; the circuit will not work with a dynamic mic because it would change the bias. The layout is in **Figure 5** and the simulated response is in **Figure 6**.

■ FIGURE 5.



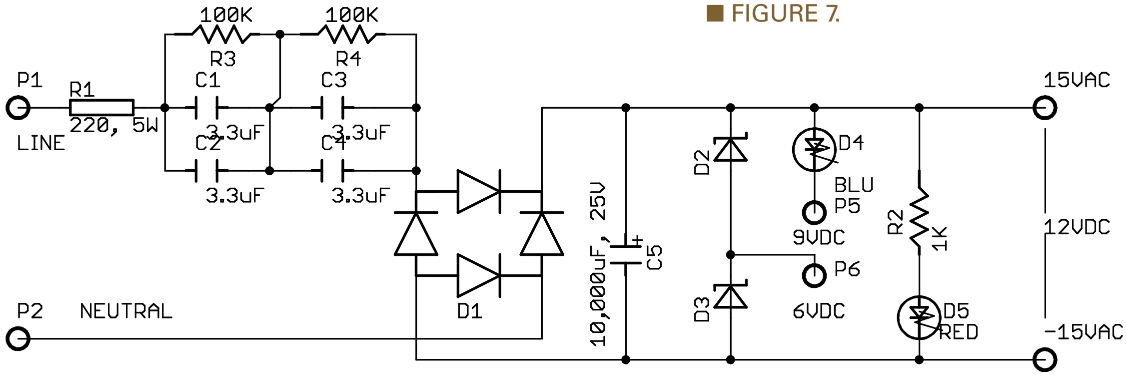


■ FIGURE 6.

TRANSFORMERLESS POWER SUPPLY

Q I am an avid reader of your column and have on rare occasions tried to build a few of the simple circuits in your articles. I am a beginner who has only a very basic understanding of the passive components of electronics. Please enlighten this old

■ FIGURE 7.



man's concept of how you determine what particular components to use and how they work together to achieve the desired effect.

Case in point: Transformerless Power Supply, May 2012, Figure 4 schematic. What if I want all three outputs (6 VDC, 9 VDC, 12 VDC) to have 500 mA each? What would you change in this particular schematic and how did you arrive at these changes?

A I would not like to use this circuit for so much current because if the outputs are not loaded, the zener diodes have to dissipate the whole 13.5 watts that would otherwise go to the loads. However, since it is only an exercise, this is how I would do it. **Figure 7** is the circuit from the May issue; I will use it only for reference.

R1 is needed to limit the inrush current when the circuit is first turned on because C5 is a short circuit until it charges up. The load is 1.5 amps,

so I would set the current limit to two amps. The input voltage is 115 VAC nominal but the peak is 162 volts; therefore, R1 = 162/2 = 81 ohms. The resistor has to be rated to handle the normal 1.5 amps and since P = I^2*R, P = 182 watts. A 200 watt lamp would be an economical way to do that.

The impedance of the capacitors (C1, C2, C3, and C4) has to make up the difference between the input of 162 volts and the load of about 16 volts (15V load plus 1V diode drop). The R1 drop is 1.5A * 81 ohms = 122V; that leaves 160-122-16 = 22 volts for the caps. Ignoring the phase angle and using Xc = E/I, then Xc = 22/1.5 = 15 ohms. Using Xc = 1/(2*PI*f*C), then C = 177 µF. The capacitor has to be non-polar (bi-polar) because it is in an AC circuit. You want the capacitor(s) to have a higher voltage rating for reliability; I would choose 50V and use 100 µF, 47 µF, and 33 µF in parallel. These capacitors have a ±20% tolerance which you should be aware of, but since they are not a large part of the current limit, don't worry about it.

The diode bridge has to handle 1.5 amps on average, and since diodes are normally rated for high inrush current, a 1.5 amp or higher rating diode bridge will be okay. I would use a six amp, 600V bridge for reliability, and also because they are readily available.

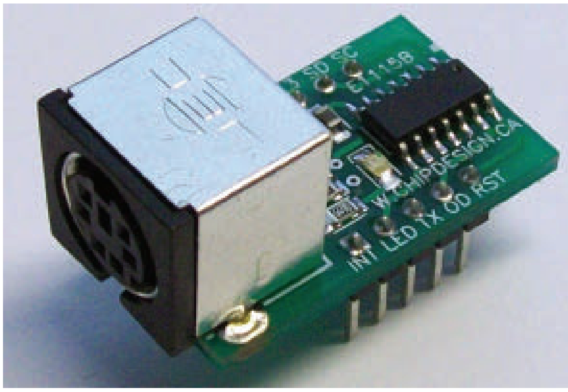
C5 should have a ripple current rating of 1.5 amps because it supplies all the load current between AC peaks. The capacitance value will depend on how much ripple you will tolerate. For 1% ripple (.15V at 15V), I compute the capacitance from: C = I * dT/dE. In this case, dT = 8.3 mS because the rectifier is full wave. C = 83 mF = 83,000 µF — 22,000 µF, 50V is available; four in parallel will do it and the ripple current rating is way higher than needed. The zener diodes, D2 and D3, must dissipate seven watts each. Stud-mount 10 watt units are available. D4 must drop three volts at 1/2 amp; six one amp diodes in series will do it.

There you have it. It is a monstrosity, but it works. I simulated the circuit; the outputs at 400 mA load on all outputs were: 11.9V, 7.7V, and 5.1V. The outputs had a dead battery level at 500 mA. Oops, I see that I used 15V instead of 12V for the output, but you get the idea. Another error I am making is to assume sinusoidal current; the current is in pulses and the I^2R of pulses is greater, so it is wise to over-rate everything. **NV**

NEW PRODUCTS

- HARDWARE
- SOFTWARE
- GADGETS
- TOOLS

PS/2 KEYBOARD TO ASCII CONVERTER MODULE



CHiPdesign announces the availability of the E1115 — a compact PS/2 keyboard to ASCII converter module for embedded microcontroller applications. It offloads the process-intensive PS/2 scancode decoding, while providing a fast single byte output response for both ASCII and non-ASCII keys including multi-media. Requiring only +5V, the easy to use module provides a pin selectable 57.6K/115.2K baud rate, as well as a 100 kHz clocked serial data output. The module also flashes an LED and generates an interrupt pulse.

The converter allows for the quick development of any keyboard-based project. The E1115B is available as a 10-pin DIP style 1.3" x 0.7" module with integrated PS/2 connector, and also as a SOIC-14 chip. An application note on the website describes how to parse the output into commands and data. Pricing starts at $16 for the module and $8 for the chip.

For more information, contact:
CHipdesign
Web: **www.CHiPdesign.ca**

RGB LED MATRIX BACKPACK

Logos Electromechanical announces two new products. The first is an I²C RGB LED matrix backpack. The backpack allows users to drive an 8 x 8 RGB LED matrix over I²C from any microcontroller.

The backpack includes a full frame buffer — write a desired image out once and the backpack takes care of all the matrix operations. Each backpack costs $30 and comes with a 60 mm 8 x 8 RGB LED matrix.

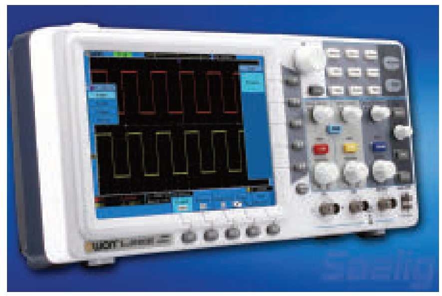
Logos is also introducing their inline current/voltage sensor. This compact sensor (0.5" x 1.675") comes in a variety of bipolar and unipolar current and voltage ratings, ranging up to 30A and 100V. It is designed to connect through ring terminals as large as #10.

The ring terminal holes are 1.125" apart which enables the board to mount to a standard 3/8" pitch barrier terminal strip for easy incorporation into existing systems.

For more information, contact:
Logos Electromechanical
Web: **www.logos-electro.com**

LOW COST TWO-CHANNEL OSCILLOSCOPE

Saelig Company, Inc., announces the SDS5032E: a new, low cost



two-channel oscilloscope which is packed with useful features normally only seen on higher-end DSOs, including external and video-capable triggering, auto-measurements, auto-scaling, a large 8" high resolution full color LCD display, XY mode, auto-set, averaging, math functions, USB output, waveform storage, pass/fail output, and a three year warranty.

FFT functionality is included for frequency spectrum display, in addition to a built-in six-digit frequency meter which can measure frequencies from 2 Hz to 30 MHz.

The autoscale feature can automatically adjust the vertical gain or the horizontal time base, or both together. This is useful for circuit probing. As the probe is moved from point to point on a circuit board, the display auto-adjusts for best trace presentation.

It functions like AutoSet but instead of being a one-time function, it's active until turned off allowing "hands-free" measurements.

The FFT function shows an instant frequency spectrum of the signal under test, while the math function shows the results of the

addition, multiplication, division, and subtraction operations between channel 1 and channel 2, and the FFT operation of channel 1 or channel 2. The SDS5032E can automatically measure and display frequency and peak-peak/rms/mean values, but cursors can also be moved to make individual readings.

A built-in self-calibration facility improves measurement accuracy. Video monitoring is possible with triggering on NTSC/PAL/SECAM line or field waveforms.

SDS5032E features include: manual cursor measurements, up to 19 automatic measurements (including frequency), high speed screen update, storage for up to 15 waveforms and setup parameters, convenient USB serial interface with PC software, and advanced trigger settings including pulse width criteria for detecting rogue signals.

This lightweight (3.3 lb) and compact 13.7" x 6.7" x 3" oscilloscope comes in a slim case and is perfect for an engineer's or student's desk. The USB master and slave connections make printing or storing results easy, as well. With a large 8" 800 x 600 color TFT-LCD, this scope is small enough and light enough to carry anywhere.

Sensitivity range is 5 mV/div to 5V/div, with horizontal display of 4 ns/div~100s/div. An external VGA output port allows the screen information to be displayed on a computer monitor or projected with a video projector. The SDS5032E offers onboard storage and USB output, and is available for $299.

For more information, contact:
Saelig
Web: **www.saelig.com**

TEXT TO SPEECH IC

The SP0-512 RoboVoice text to speech IC from Speech Chips is a pre-programmed microcontroller that accepts English text from a serial connection, converts that text to phoneme codes, then generates audio.

It is ideal for adding a robot voice to embedded designs. Features include:

- Single chip text to speech IC.
- Low power.
- Communicates using a simple serial port (9600 N81).
- 800 rules that convert English text into phoneme codes.
- Built-in 16-bit at 16 kHz DAC. No external filter required.

The RoboVoice is based on the Microchip dsPIC33FJ64GP802 microcontroller. A datasheet is available at **ww1.microchip.com/downloads/en/DeviceDoc/70292F.pdf**. User's should refer to this document for greater detail about the chip itself.

Source code licenses are available. The complete text to speech system fits in about 32K and is written in ANSI C, so it can be ported to almost any embedded processor. Price is $24.99 each.

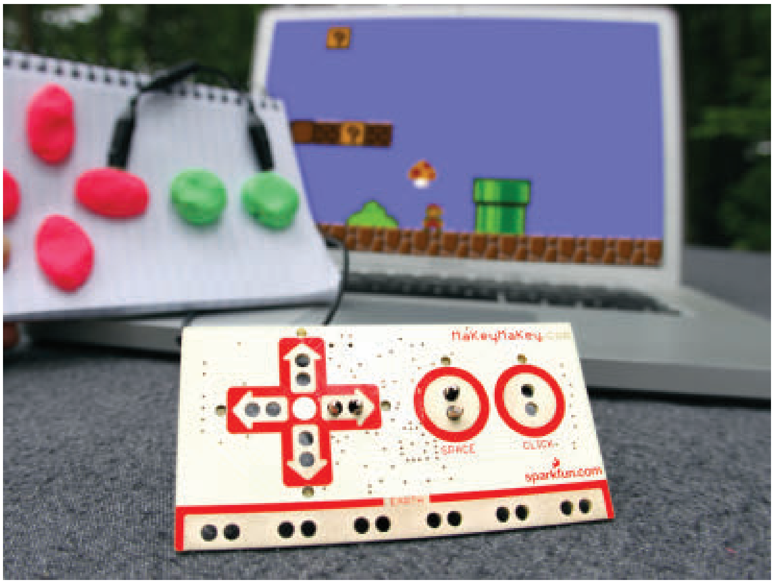
For more information, contact:
Speech Chips
Web: **www.speechchips.com**

MaKey MaKey INVENTION KITS

SparkFun Electronics is now shipping the MaKey MaKey — a new product that invites everyone to be an inventor.

MaKey MaKey was designed by MIT graduate students Jay Silver and Jay Rosenbaum, and is an invention kit that encourages builders to discover creative new ways to interact with computers. It allows users to create inventions using almost any everyday object such as a keyboard key or mouse. For example, you could replace your space key with a banana, use Play-Doh™ to move and click your mouse, or high-five your best friend to advance PowerPoint slides.

"The MaKey MaKey is an amazing kit because it really lowers the point of entry for beginning



electronics enthusiasts," said AnnDrea Boe, SparkFun Director of Marketing Communications. "This kit allows users to start inventing immediately, right out of the box. It's a fantastic way to start exploring DIY electronics."

The MaKey MaKey uses very high resistance switching to detect when any of its 18 inputs are activated, even through materials that aren't very conductive (like leaves, pasta, or people).

The MaKey MaKey communicates with your computer using the USB Human Interface Device (HID) specification which means that it can act just like a USB keyboard or mouse.

In addition, the MaKey MaKey is also an Arduino-compatible platform, so users can easily re-map any of the keys, or further customize it to their needs.

The regular kit is $39.95 and comes with the MaKey MaKey HID board, an alligator clip pack, and a mini-USB cable. The deluxe kit ($49.95) includes everything that the basic kit does, in addition to a second alligator clip pack, a jumper wires pack, and a roll of copper tape for even more possibilities.

For more information, contact:
SparkFun Electronics
Web: **www.sparkfun.com**

BATTERY HOLDERS IN TAPE AND REEL PACKAGING

Battery holders from Linx Connectors are now available in tape and reel packaging, making easy to use automated assembly techniques. Linx has tape and reel in stock for the popular BAT-HLD-001 surface-mount battery holders for CR2032 coin cells. Tape and reel is

Continued on page 56

PRECISION MINIATURE CRYSTAL MODULE

Lemos International Co., Inc., announces the release of the new RFX Ltd high precision miniature SMD GPS disciplined oven controlled crystal oscillator module type GPS1300-10-1000.

This 10.000 MHz oscillator with NMEA data stream (which measures only 40 x 32.5 x 10.5 mm) has evolved from the former successful military design and will help revolutionize the attainment of extreme precision timing signals for use in commercial applications. The accuracy is derived from the loop control discipline of a precision OCXO through the acquisition of satellite GPS data, providing near Caesium accuracy approaching 10-12 one thousandth part per billion.

This product has been designed for use in base stations, wireless networks, system synchronization systems, instrumentation, critical timing and calibration applications, and precision portable reference units. Through the use of control algorithms, the inherently accurate precision OCXO is further regulated towards the onboard satellite Caesium standard. In the event of the GPS signal being temporarily lost, the module provides holdover through the internal OCXO which retains it previously acquired control settings until lock is reinstated.

MULTI-CHANNEL TRANSCEIVER

Lemos International Co., Inc., is also now offering the new SHX1 multi-channel transceiver from Radiometrix which has a usable range of over 5 km, 500 mW RF power, and narrow band performance. Its available on 151-154 MHz MURS band frequencies (unlicensed use under FCC Part 95).

SHX1 is a small VHF transceiver which also features:

- Data rates up to 5 kbps for standard module.
- High performance double superhet PLL synthesizer.
- 1200 baud modem.
- Low power requirements.

Some of the technical specs include:

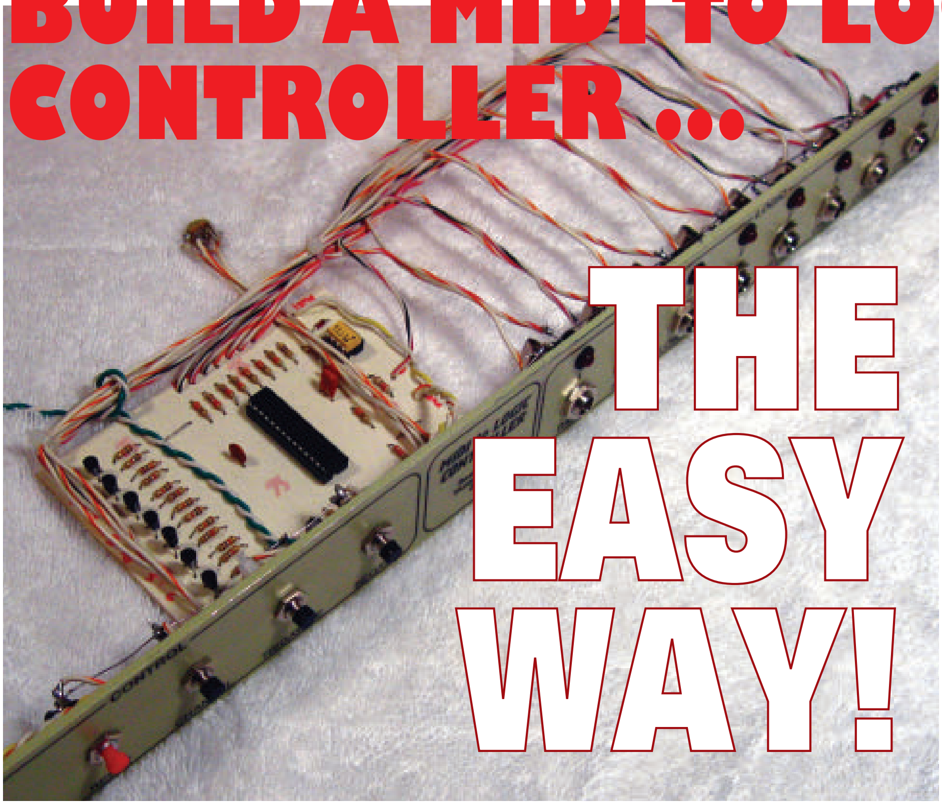
- Part 90 compliant for licensed applications.
- Eight parallel or 255 serial channels.
- Supply range +5V regulated.
- Current consumption: 280 mA transmit, 20 mA receive.
- Size: 67 x 30 x 9 mm.

Applications include remote control, and wireless data and telemetry.

For more information, contact:
Lemos International
Web: **www.lemosint.com**

SHOWCASE

BUILD A MIDI TO LOGIC CONTROLLER ... THE EASY WAY!



By Thomas Henry

If you've ever dreamed of building your own MIDI device but didn't want to get your hands dirty with assembly language, complex programming circuits, or microcontroller architectures which make your head swim, then you've come to the right place! Here's a nifty project which will convert MIDI on and MIDI off signals to logic levels through either eight independent gate or trigger outputs. Musicians will find it useful for such things as firing analog music synthesizers, clocking drum sequencers, and synchronizing recorders. With the addition of relays or opto-isolators, it could also form the heart of light shows, home automation rigs, and the like. Best of all, the whole shebang can be easily implemented and customized since the brain of the unit is the beginner-friendly PICAXE 28X2 microcontroller. As we'll see, no expensive or complicated equipment is required to burn the program into memory, putting the project within easy reach of all.

If you're new to the world of microcontrollers, then here's the deal. With a system like this, you'll be able to write a program in a language closer to what humans use (that's Basic here), but be able to convert it to something the microcontroller likes (that's machine code). Burning simply means storing the machine language equivalent you've just created inside the microcontroller; this puts the chip through its paces.

Versatility is achieved by making either gate or trigger outputs available at the flip of a switch. In gate mode a MIDI on signal brings an appropriate output high, leaving it on until a corresponding MIDI off is detected. On the other hand, in trigger mode a MIDI on signal pulses an output for around 10 milliseconds, with MIDI off being ignored.

As two simple examples, a gate is perfect for controlling a synthesizer envelope generator, while triggers are at home firing analog drum voices or perhaps punching in a recorder on the down-beat. If this sounds interesting, then read on to find out just how simple it all is.

What the PICAXE Provides

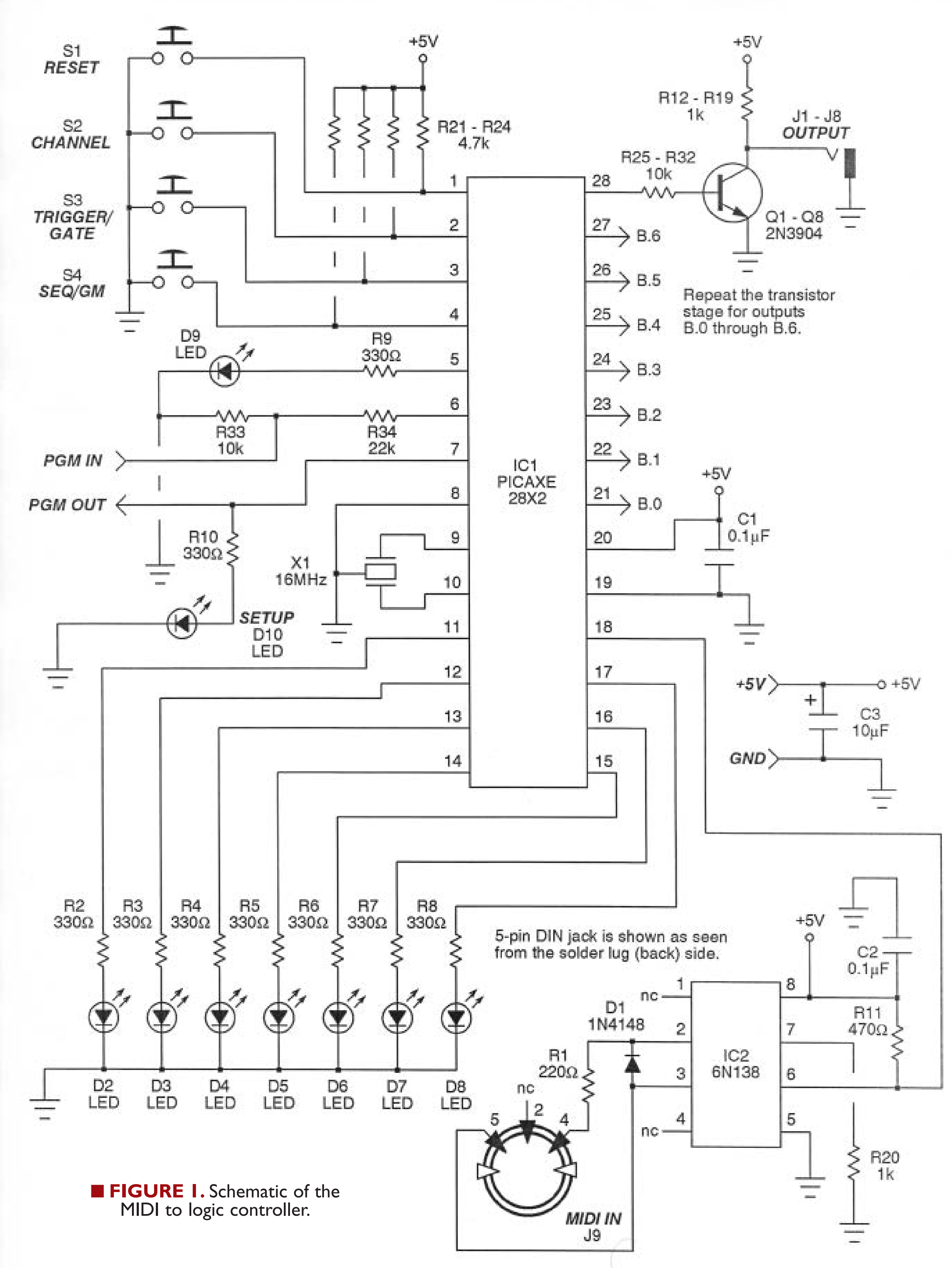
Before getting into the details, let's conduct a general overview of the PICAXE 28X2 chip and see what makes it so useful for this project. First, there are ample input and output ports to handle all of the switches, LEDs, and gates or triggers. In particular, we'll be using four input lines and 17 output lines. A built-in serial interface will easily handle the MIDI signal, which is also serial in nature and runs at 31.25 kHz.

Then, there are the usual things such as resonator inputs, a reset line, and a programming interface.

Inside the chip, you'll also find a huge amount of program space, plenty of variables in RAM, and even some general-purpose EEPROM for holding data after power-down. This truly is an all-in-one unit, and only requires some simple buffers to complete things. If you've peeked ahead at the schematic or even the printed circuit board (PCB) parts placement guide, then you were probably already struck by just how condensed the complete MIDI to logic controller circuit really is. With our quick reconnaissance out of the way, let's dig in more deeply and see what makes the circuit tick.

How It Works

Refer to **Figure 1** which shows the schematic. There are eight main



■ **FIGURE 1.** Schematic of the MIDI to logic controller.

outputs at pins 21 through 28 on the PICAXE 28X2. Since we want to connect these to real world devices, we'll buffer them. Check out pin 28, for example, which drives an inverter configured around transistor Q8. This gives plenty of current for whatever application you have in mind. Note too that the output impedance is 1K which is a typical electronic music standard, as is the +5V level.

The buffers are repeated for a total of eight outputs. These are the lines comprising port B. For greatest versatility, only the software needs to be changed to make these either gate or trigger outputs. Again, when in trigger mode the pulse width will be somewhere in the neighborhood of 10 milliseconds. Remember, the width is irrelevant here; it's the timing of the leading edge that matters.

We'll want a visual indication as to which outputs are being fired. So, let's turn to port C and use pins 11 through 17. The PICAXE is able to directly drive the LEDs via the 330Ω resistors, R2 through R8.

We just saw in the previous paragraph that the outputs in trigger mode blip with a short pulse width. This is really too brief for the human eye to pick up on decently, so we'll let the firmware take care of making sure the corresponding LEDs have an on-time of about 50 milliseconds.

Counting pins, you'll notice that we're one short on the LEDs. That's because pin 18 in port C must be used for the serial MIDI input. No big deal; we'll just impose on pin 5 (which is actually port A.3) to provide the full complement.

The firmware keeps it all straight, so now we have eight outputs and their corresponding eight LEDs under control.

One final LED D10 indicates whether the unit is currently idle, holding in display or setup mode, or actually active and ready to translate the incoming MIDI stream.

Let's take on the switches next. There are four pushbuttons that handle all of the input activity. S1 is a typical reset switch. Normally, reset pin 1 is held high by R24 but briefly pressing the button brings it low, forcing the chip to start from a known condition.

MIDI to Logic Controller User Guide

There are four modes of operation. In Run mode, the unit responds to MIDI data — its primary job. In Display mode, you can confirm the stored defaults. In Set mode, you can revise and store those defaults. Finally, in Test mode the unit will output three different test patterns — useful for setting up your patches in a home studio or on stage.

At power-on reset, the unit comes up in Run mode (LED D10 off) using the defaults currently stored in EEPROM. These may be changed and stored by the procedures listed below.

To turn on Omni or select a specific channel:
1. Simultaneously press both the Reset and Channel buttons.
2. Release the Reset button. The unit enters Set mode.
3. The LEDs begin by performing a quick chaser effect, indicating Omni on. If this is what you desire, then release the Channel button. Otherwise, continue to hold that button and the LEDs begin to flash.
4. Counting the flashes, release the Channel button after the desired channel number has been indicated (1 through 16).
5. The unit now enters Display mode which is described later (LED D10 on).

To toggle Trigger/Gate mode:
1. Simultaneously press both the Reset and Trigger/Gate buttons.
2. Release the Reset button. The unit enters Set mode. The LEDs will show the new mode using a dot-dash code. They will flash three times rapidly to indicate Trigger mode (a dot) or give one long flash (a dash) to indicate Gate mode.
3. The unit now enters Display mode (LED D10 on).

To toggle Sequential/GM mode:
1. Simultaneously press both the Reset and Sequential/GM buttons.
2. Release the Reset button. The unit enters Set mode. The LEDs will show the new mode using the dot-dash code described above. A dot means Sequential mode, while a dash means GM mode.
3. The unit now enters Display mode (LED D10 on).

To confirm your settings:
1. You may confirm your settings in Display mode (LED D10 on).
2. Press Channel. If Omni mode is on, the LEDs will do a quick chaser effect. Otherwise, all LEDs will flash the number of the currently selected channel.
3. Press the Trigger/Gate button. The LEDs will indicate the mode with the dot-dash code described earlier.
4. Press the Sequential/GM button. The LEDs will indicate the mode with the dot-dash code described earlier.

To play a test pattern:
1. While in Display mode, press a pair of buttons, then release both.

Channel + Trigger/Gate	- Test Pattern One	- One at a time
Channel + Seq/GM	- Test Pattern Two	- All outputs at once
Trigger/Gate + Seq/GM	- Test Pattern Three	- Drum pattern

2. The unit will begin playing a test pattern in Trigger mode.
3. Press Reset to quit the test pattern and go to Run mode.
4. Otherwise, press any other button to quit the test pattern and return to Display mode.

To exit Display mode and begin Run mode:
1. Press the Reset button. LED D10 will turn off and the unit is now ready to receive MIDI data.

We'll do some other clever things with this in just a moment.

S2 allows you to set or confirm the MIDI channel. The MIDI standard provides for 16 possible channels, and any one of these may be set as a default which is then stored in EEPROM. Additionally, it is possible to coerce the device to operate in Omni mode, in which case it responds to data from any of the 16 channels. The user instructions in the **sidebar** will make it clear how to set the channel.

Pushbutton S3 is used to toggle between the gate and trigger modes alluded to earlier. Again, the default is saved in EEPROM.

The last switch S4 offers a unique feature of special merit in drum applications. Part of the MIDI protocol allocates some 40 different percussive instruments to note numbers. This aspect is known as General MIDI, or GM for short.

If, in fact, you are playing back a drum part saved as a MIDI file, it is likely the drums (bass, snare, tom-tom, etc.) are designated by the GM note numbers. Well, in GM mode this circuit will automatically translate these and route them to the correct drum outputs.

The alternative is called Sequential mode. In this case, the eight outputs are mapped consecutively to middle C on up to middle G. This makes it easy to play the drums from a piano type keyboard since they are all next to each other on the scale. (In GM mode, the drums are scattered all over the place.)

As a rule, I use Sequential mode for my own compositions, but employ GM mode when playing back someone else's MIDI file. In either event, switch S4 accommodates things nicely.

Nothing has been said yet about how the serial data makes it into the PICAXE. Let's chase that down now. A MIDI cable (from a PC or Mac running sequencer software, for instance) connects to jack J9. This goes to optocoupler IC2, configured in a standard way to convert the current loop input to a digital signal. The inputs are at pins 2 and 3 of the optocoupler, while the output is at pin 6. The conditioned digital signal then feeds pin 18 of the PICAXE. This is port C.7. Recall that we have no choice in this, and must use this pin for serial input.

The PICAXE 28X2 is quite versatile in how it can be clocked. In this application, we need all of the speed we can get, so we'll put it under control of ceramic resonator X1. This is a three-pin device, with the middle pin grounded and the other two going to pins 9 and 10 of IC1.

A resonator rated at 16 MHz is specified here, but note that a phase-locked loop within the PICAXE actually multiplies this by four. Hence, the chip is running at a speedy 64 MHz, which isn't too shabby!

So, what about the programming? As mentioned earlier, that's a snap with the PICAXE series and it is also quite inexpensive to set up. You'll notice pins 6 and 7 are indicated as program lines on the schematic.

R34 is a current-limiting resistor on the input, while R33 is a pull-down to keep the sensitive input line from floating. The free PICAXE manual gives the details of how to program the chip (it's easy), but see the second **sidebar** for a quick look.

To complete our survey of the circuit, observe that pins 8 and 19 of IC1 are grounded, while pin 20 is the +5V line. Decoupling capacitors are used throughout.

Programming the PICAXE 28X2

In the not too distant past, burning a program into a microcontroller was a genuine bearcat for the newcomer. First, there was the business of building a programmer (since commercial units were either priced out of range for the DIYer or simply not available). Then, there was the seemingly infinite learning curve to ascend. Assembly language had to be mastered, and a nontrivial amount of hours spent cracking the hard nut of cross-assemblers. The design cycle was slow and tedious, requiring all sorts of cable switching, yanking parts out of sockets, and re-assembling the code each time a problem was found.

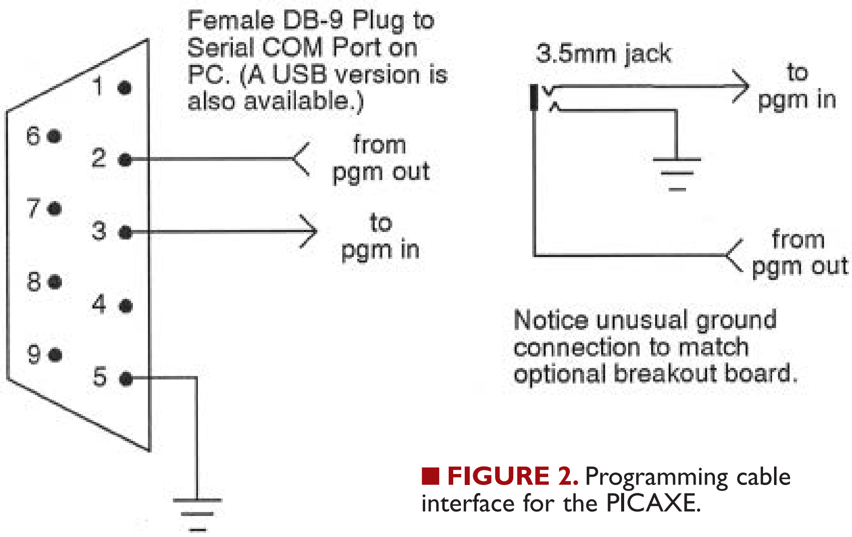
That's all a bad memory now thanks to the PICAXE family of microcontrollers. Creating and burning a custom microcontroller couldn't be simpler, and even better is the fact it can be accomplished on a limited budget. Here's how to get started.

Your first step should be to acquire the PICAXE manual. This is free of charge and can be downloaded from the manufacturer's website at **www.picaxe.com**.

Next up would be to get the PICAXE programming editor and install it on your home computer. Again, this is free of charge even though it is an exceptionally well-written and easy-to-use piece of software. You'll also find it at the PICAXE site.

No complicated burning hardware is required, just a cable! I employ a homemade RS-232 version I tossed together for less than $4. If you're using the official PICAXE breakout board for development on a breadboard like me and want to stay consistent with it, then connection is made by means of a stereo 3.5 mm plug and jack. If that's what you're using, then note the unusual connections indicated in **Figure 2** on the next page. Commercial cables manufactured expressly for the PICAXE are also available and come in either RS-232 or USB flavors. Choose whichever matches your home computer. For example, a couple variations are available at SparkFun Electronics, among other dealers. For details, go to **www.sparkfun.com**.

If you've been following *NV* author Ron Hackett's column here (starting on page 14) over the past several years, then you already know something about programming the PICAXE. If you're new to it all and would like some more background on how to get started, then be sure to look into his initial entry in the December 2007 issue, pp. 17-21.



■ **FIGURE 2.** Programming cable interface for the PICAXE.

An Easy Build

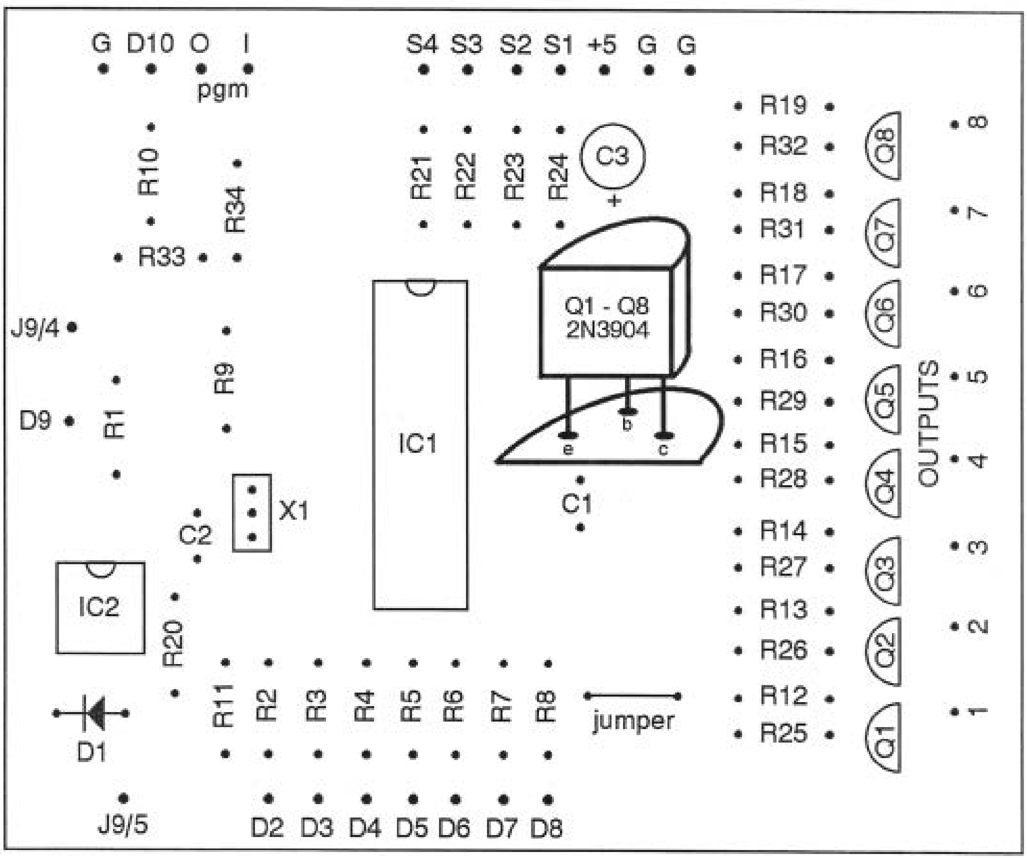
Constructing the MIDI to logic controller couldn't be simpler. After all, the PICAXE 28X2 bears most of the burden. I was able to whip up a PCB design in a short evening and had the thing etched and drilled by the following morning. Since this is a fairly pedestrian layout, we can readily get by with single-sided.

I used Press 'n Peel to transfer the artwork and a small travel iron to bake it on. The PCB artwork is available at the article link, while **Figure 3** gives the parts placement guide. But really, the circuit is so straightforward that it could just as easily be wired by hand on perfboard.

Everything in my studio is rack-mounted, so that's what I went with here. **Figure 4** shows the front panel design. This is a standard 1U panel, weighing in at 1-3/4 inches by 19 inches. The circuit board mounts behind the panel on small angles. **Figure 5** is a snapshot of the completed undertaking.

I wanted to leave the programming jack in place to make it easy to update the firmware later if needed. If you look carefully at **Figure 5**, you'll see it on pigtails hanging off the back of the board. The wiring of **Figure 2** was used.

Since this is an unusual arrangement (employed merely to maintain compatibility



■ **FIGURE 3.** Parts placement guide.

■ **FIGURE 4.** Front rack panel layout.

with my PICAXE breakout board), it is important that the programming jack **not** be panel-mounted unless it is an insulated type.

Firmware, Latency, and Jitter

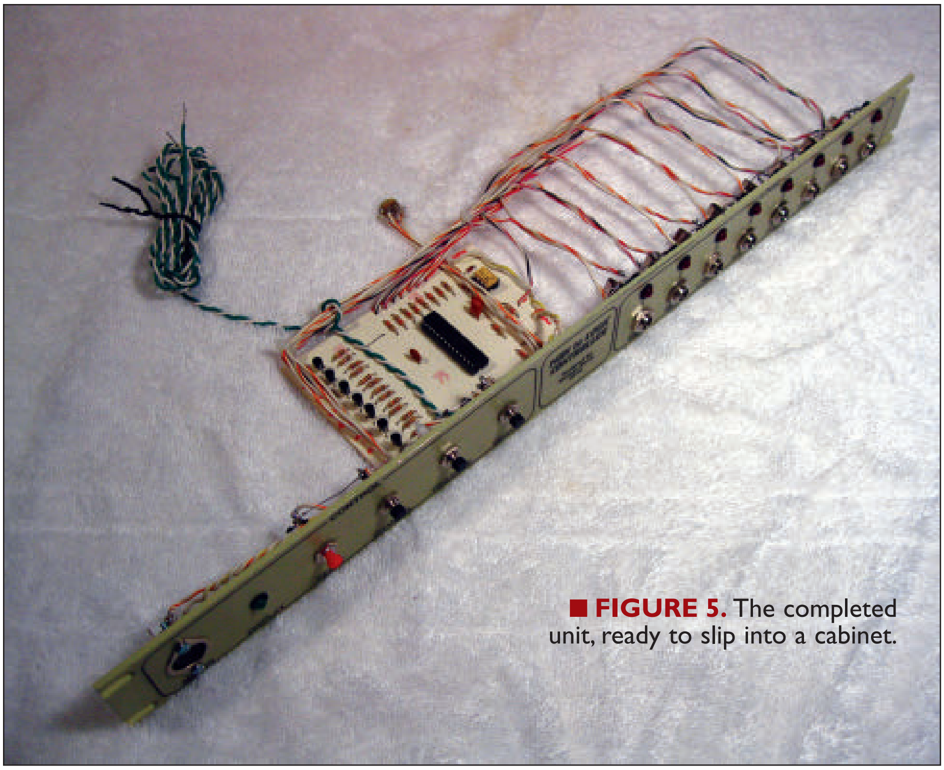
The time has come to discuss the firmware. You'll find the complete source code in Basic at the article link. The code has been extensively documented, so not only should it be a snap to understand but customizing it for your application should be a breeze. Here are some general comments to guide you along.

As you probably know, MIDI (being serial in nature) requires a lot of activity in a very brief period of time. As a consequence, getting the MIDI to logic controller up to speed was a real battle. All of the prototypes essentially worked from the get-go, but early versions suffered from latency and jitter. Latency is a lag time between receiving a command and executing it. Jitter occurs when the latency varies over time.

So, here's the deal. This unit has an incredible amount of work to do in short intervals. The first step was to bump the microcontroller's master clock to 64 MHz. Next up was to optimize the code. Since the PICAXE uses a tokenized Basic language (notoriously slow by any standards), it was necessary to break all the rules of good structured programming and instead just focus on speed.

Also, there is tons of program space available, so I made no attempt to minimize the size of the program — again going instead for fast execution.

My motto was (lifted from an old movie) "waste anything but time." So, as you read the code you'll see just how profligate I was with program space, but it was for a legitimate purpose.



■ **FIGURE 5.** The completed unit, ready to slip into a cabinet.

Actually, gate mode has very little overhead associated with it and never gave any grief. It was trigger mode that really kept my nose to the grindstone. The code provides a clue on how I finally distributed the trigger updates over time to spread out any noticeable jitter error.

One final thing helps. In a real world situation, it is common to use a MIDI filter before any unit (commercial or otherwise) in a setup. This is to ensure that all of that mixed bag of signals coming down the pike be delivered only to the devices really needing them.

Most people — including me — use the wonderful and free program MIDI-Ox in the usual course of things. (You can download it from

PARTS LIST

ITEM	DESCRIPTION
All fixed resistors are 1/4 watt, 5% values.	
R1	220Ω
R2-R10	330Ω
R11	470Ω
R12-R20	1K
R21-R24	4.7K
R25-R33	10K
R34	22K
All capacitors are 16V or better.	
C1, C2	0.1 µF disc
C3	10 µF electrolytic
Other components	
J1-J8	1/4" phone jack, N.O.
J9	Five-pin DIN jack
S1-S4	Momentary SPST switch
Semiconductors	
D1	1N4148 diode
D2-D9	Red LED
D10	Green LED
Q1-Q8	2N3904 NPN transistor
X1	16 MHz ceramic resonator
IC1	PICAXE 28X2 microcontroller
IC2	6N138 optocoupler

Miscellaneous: Printed circuit board, IC sockets, front panel, LED holders, wire, etc.

The PICAXE 28X2 is available from SparkFun Electronics or RobotShop, among others.

www.sparkfun.com
www.robotshop.com

All other parts are commonplace and available from general suppliers.

www.midiox.com). With the filter removing all the chaff not required by the MIDI to logic controller it was possible to greatly simplify the code, hence speeding things up appreciably.

Now it no longer needs to be looking for channel numbers, checking Omni mode, passing over system real time messages, etc.

In short, the final code simply responds to MIDI on and MIDI off messages, leaving the filtering of all else to MIDI-Ox (or something similar). With that in place, very rapid sequences can be played with no objectionable jitter.

When you download the code, you'll see there are two versions. One is the stripped down variant just described. However, if you'd like to try the full-blown rendering with the built-in filtering, that's there, too.

I originally took on this project as a challenge. I had read so many comments on the Web bemoaning the "fact" that the PICAXE just isn't fast enough to do MIDI that I wanted to see for myself.

It took a month of daily experimenting and hair-pulling, and I almost gave up more than once. Truthfully, though, while this circuit pushes the PICAXE to the limit, it really does get the job done and then some.

I hope you enjoy it, and more importantly I hope it helps you to create some new music! **NV**

REMOTE DATA LOGGER & SURVEILLANCE CAM

By Andy Sullivan

Even in today's world of high tech electronic connectivity, remote monitoring can be a challenge. Phone lines and high speed Internet connections don't go everywhere, and low cost wireless solutions have a shorter range. What if you want to know what is going on across town, in the woods, or on the other side of your property?

■ **FIGURE 1.** Finished data logger and camera for remote monitoring using an inexpensive "no contract" cell phone.

My challenge was to keep an eye on our hobby farm. It is located 32 miles outside of town in a remote area of Montana. There is no phone or even power available. I keep a few animals there including pigs, chickens, and turkeys. My family enjoys time at the farm while I refill water and food supplies. Gas is expensive, so it makes sense to space trips out as appropriate while making sure the animals are properly cared for. To provide confidence and peace of mind, I needed a way to look in on the animals without the long drive. Fortunately, there is cell coverage in the area.

My goal was to use a low cost "no contract" cell phone to send me a photo of the farm each day. A picture is worth a thousand words in terms of general farm conditions like: are the animals in the fence, has it rained, do they have water, and the like. In addition to a picture, the text messaging feature of the phone can be used to communicate back information like water level and food inventory. The same techniques could be used to monitor a game stand, transmit back remote weather station data, or even radio back the GPS coordinates of a stolen car.

The Cell Phone

Figure 1 shows the assembled project. I chose to use a Verizon Samsung Gusto 2 "No Annual Contract" phone available from Walmart or Target. The phones are $15, but include $10 of free air time making the hardware cost only $5. Usage costs are low at only 25¢ for a picture message. Text can be included with the picture message, so I can achieve my goal of a photo and text message of data for $5 down and 25¢ per day for the phone service.

I used an Arduino Mega to control the phone. I chose the Mega for the greater number of digital pins available. To have the Mega control all the functions of the phone, I needed 14 pins. An Arduino Uno or other microcontroller would work, depending on other pin use and what functionality you need. Fourteen pins allow the microcontroller to access all keypad buttons including the camera button, and allow sending text messages. A camera-only version I tested used just five pins on an Arduino Uno.

The Arduino is interfaced to the cell phone by simulating button presses. The easiest way I found to do this was to disassemble the phone and solder wires to the buttons on the phone's key pad. There are 26 buttons on the phone that are interfaced to the Arduino. The two volume buttons and the speaker phone button on the sides didn't have a use in this project and are the most difficult to solder, so they were left out.

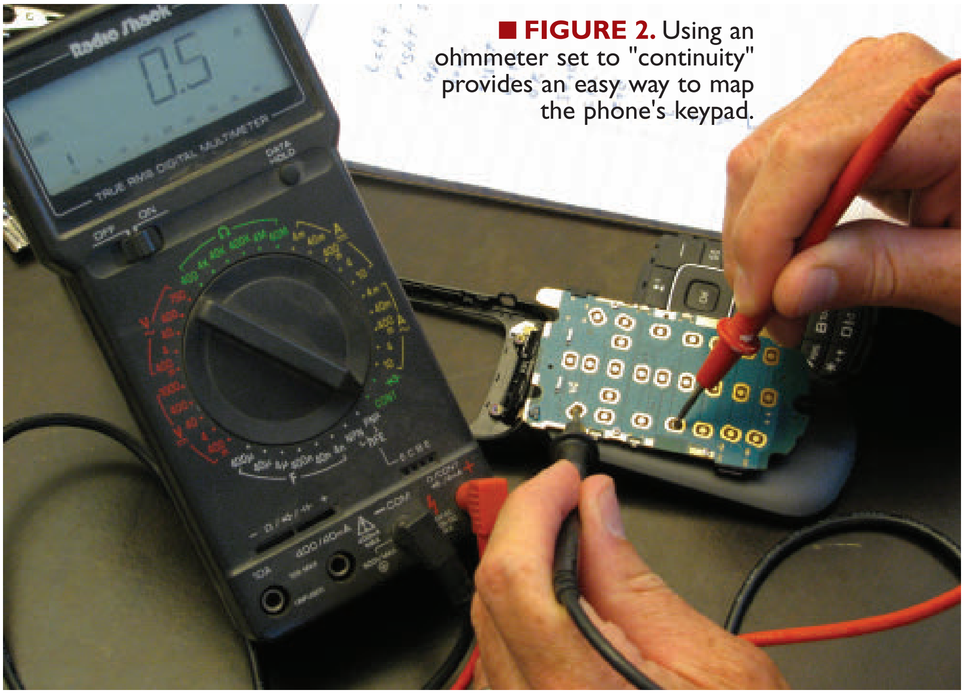
Before taking the phone apart, take a few minutes to set it up and get comfortable with it. Activate the phone and set it up to add minutes over the Internet. Rehearse actions you wish to perform remotely, and take notes on button press sequences.

Disassemble the phone. Look for screws under the battery cover. The Gusto 2 has four tiny screws on the back outside of the battery cover. A different phone didn't have screws, but snapped together tightly; it was difficult to take apart. There may still be a little snap-apart required even if there are screws. A small jewelry screwdriver helps. Pop out the circuit board while looking for screws. The Gusto 2 circuit board just snapped out over a plastic clip. Finally, remove the button membrane and gently peel off the adhesive sheet which holds the button contacts.

Each button on the phone has two copper pads that are contacted by a conductive rubber pad when the button is pressed. That is 52 pads for the 26 buttons, but things are greatly simplified because many of the button pads share a common trace. Said another way, a pad on button "A" might be in continuity with a pad on button "B." Using an ohmmeter set to "continuity" provides an easy way to map the phone's keypad as shown in **Figure 2**.

Figure 3 is the keypad map for the Samsung Gusto 2 cell phone. The button pads are concentric circles on the Gusto 2 with an outer and inner pad. **Figure 3** contains a row for every button on the phone we will interface to. Each button has a single character code that will be used in the Arduino program. Each pad is given a wire number. If the pads are in continuity, they share a wire number. For example, the "1" button center pad happens to be labeled with wire #1. This pad is in continuity with the center pads of buttons 4, 7, *, Send, and "Left Soft."

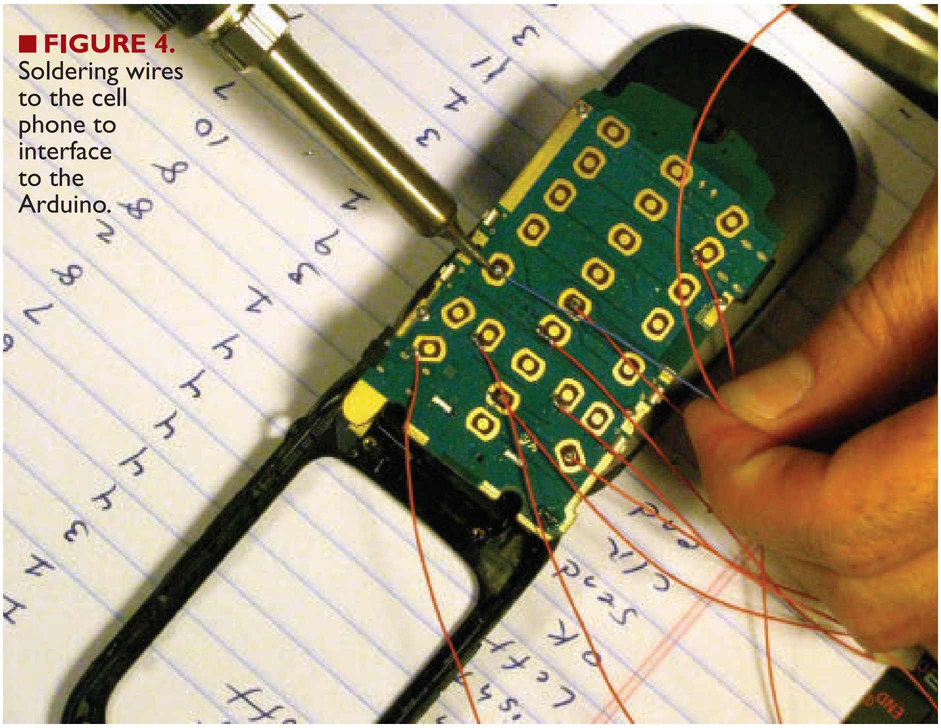
A wire labeled #1 will be



■ **FIGURE 2.** Using an ohmmeter set to "continuity" provides an easy way to map the phone's keypad.

Phone Button	Button Code	Inner Pad Wire #	Outer Pad Wire #
Left Soft	<	1	2
Right Soft	>	3	2
Up	U	4	5
Down	D	4	6
Right	R	4	7
Left	L	4	8
OK	O	4	2
Send	S	1	8
CLR	C	3	8
End	E	9	10
1	1	1	7
2	2	11	7
3	3	3	7
4	4	1	6
5	5	11	6
6	6	3	6
7	7	1	5
8	8	11	5
9	9	3	5
*	*	1	12
0	0	11	12
#	#	3	12
ICE	I	11	2
Messaging	M	4	12
Camera	P	13	14
Voice Mail	V	11	8

■ **FIGURE 3.** Keypad map for the Samsung Gusto 2 cell phone.



■ FIGURE 4. Soldering wires to the cell phone to interface to the Arduino.

soldered to any of these pads.

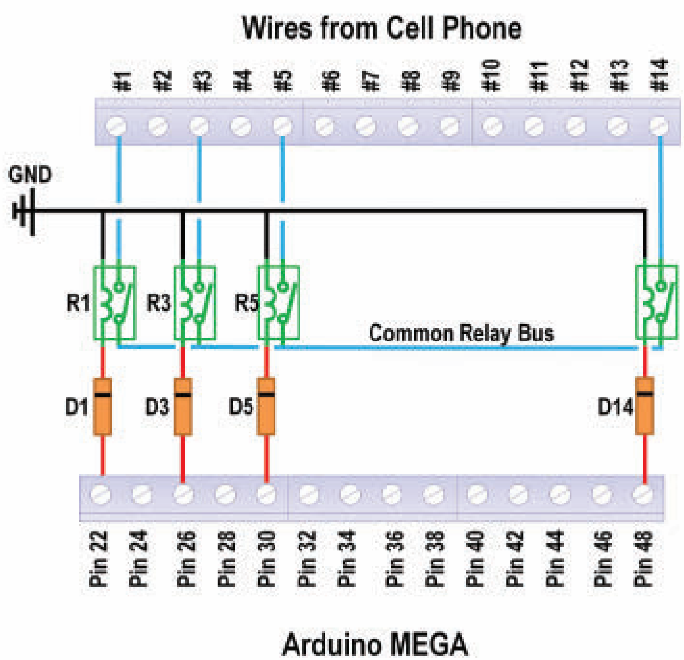
Figure 4 shows soldering the wires to the cell phone pads. Soldering the wires to the cell phone is a little tricky. Use a light gauge wire like 30 gauge wire-wrap wire to prevent excess force on the button pads. A torn pad is trouble especially with the camera button. The camera button is different than the phone's front buttons. It has tiny pads on the back of the phone that are more difficult to solder. An earlier prototype used 22 gauge wire which put too much strain on the connection and broke it. It was difficult to remake the connection with an even smaller pad to work with. Putting a small bit of solder on a pad, then re-melting the wire to the solder and pad seemed to work best.

The Interface Circuit

To simulate a button press, the inner and outer pads of a button must be contacted. This is done with a reed switch. Using a press of button "1" as an example, we see from **Figure 3** that wires #1 and #7 need to be contacted. An earlier prototype used an individual reed relay for each button. A relay for each button was okay for the smaller photo-only prototype that used only five buttons, but wasn't a good solution for sending text messages as most of the 26 buttons might be used.

A more elegant and flexible solution is to use a common electrical bus or header as shown in **Figure 5**. Each of the 14 wires from the cell phone are wired to a reed relay. The other side of each reed relay is tied together on a common bus. To simulate a button press, two reed relays are activated. In the case of cell phone button "1," reed relays #1 and #7 are activated.

Reed relays are connected to even numbered Arduino pins 22 through 48. Even pin numbers were selected so a straight header of continuous pins could be used. A diode was placed on each reed relay on the feed from the Arduino, just in case the small coil in the relay could produce enough kickback voltage to damage the Arduino. The diodes were simply placed in series with the relay coil. Placing the diodes backward across the coil might have been a better solution, but the diodes are good for 100 volts which seems high enough to handle

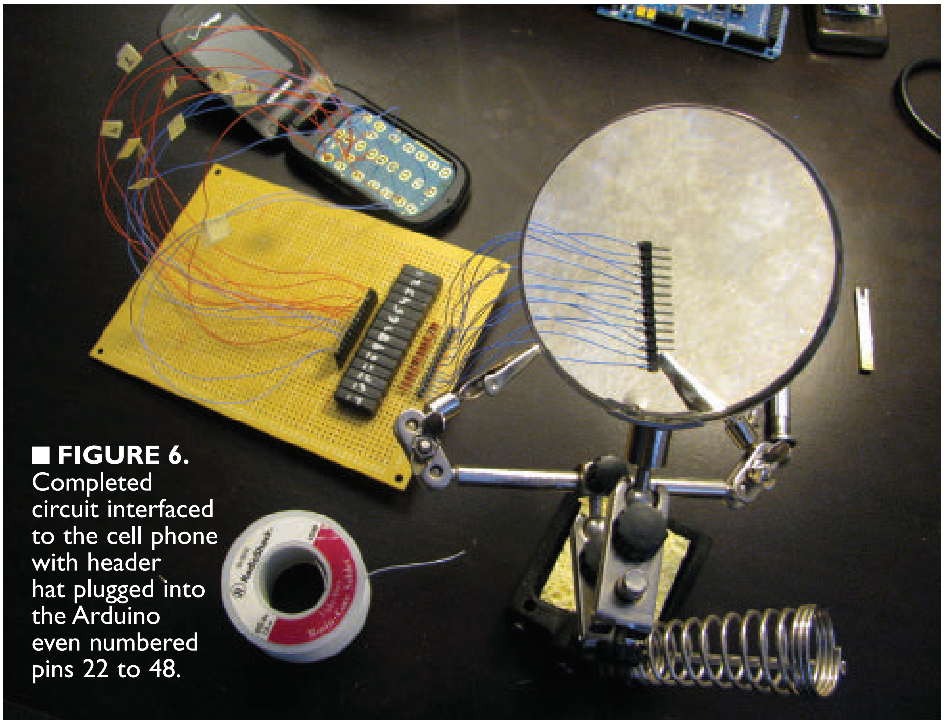


■ FIGURE 5. Circuit diagram of the cell phone interface to the Arduino using reed relays.

kickback from a reed relay.

A transistor is not used to power the reed relays since the Arduino has plenty of power. The reed relays each draw 9 mA of current at 4.3 volts, accounting for the 0.7 volt drop across the diode. The Arduino can source 40 mA per pin and 200 mA total which will accommodate the two relays this project will use at a time.

Figure 6 is a photo of the finished circuit. Strip headers were used to make connecting and separating the phone, Arduino, and custom circuit easy. The headers also help to keep connections secure, prevent wires from popping out, and clean things up a lot. The circuit was assembled point-to-point on a grid style circuit board, or perfboard. **Figure 7** shows the back of the circuit board.



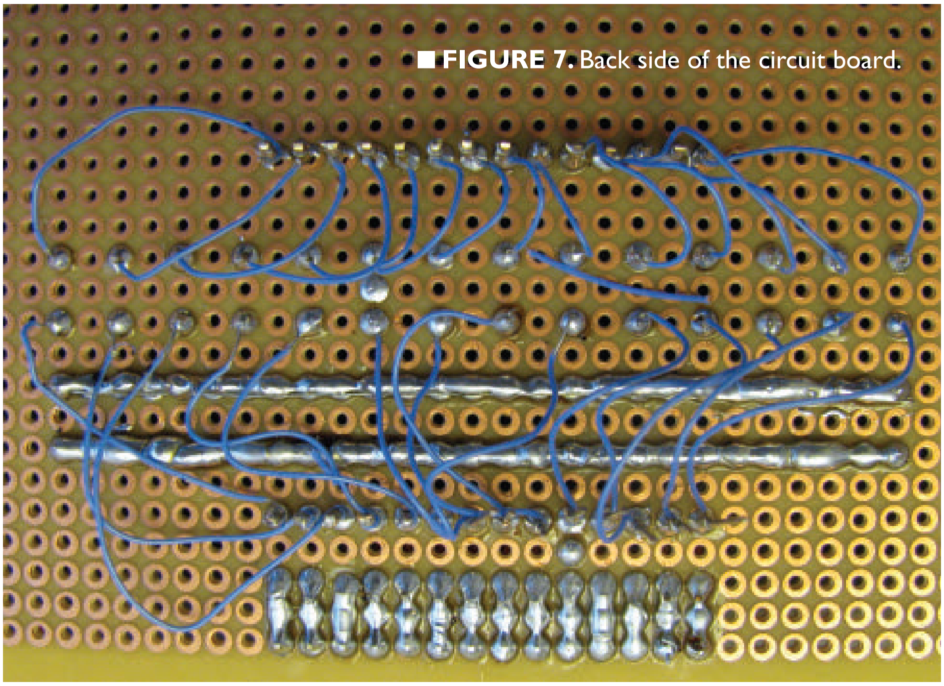
■ **FIGURE 6.** Completed circuit interfaced to the cell phone with header hat plugged into the Arduino even numbered pins 22 to 48.

The Control Program

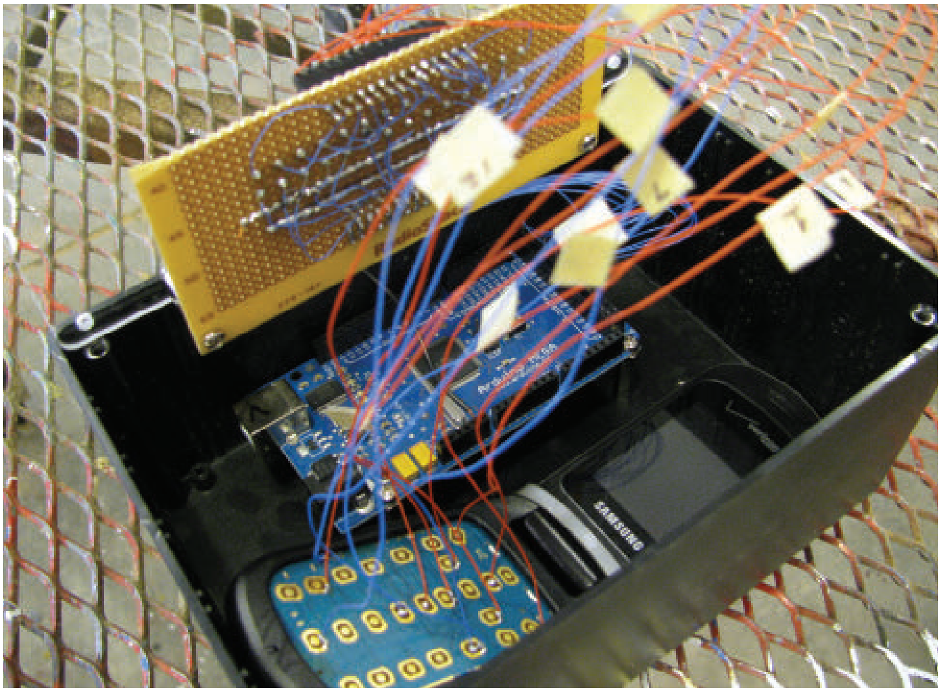
The Arduino program is available at the article link. It is composed of four elements. The main program simply counts milliseconds and calls the picture routine when a defined period has passed. The program is set up to send a picture and text message every 24 hours starting at power-on. Another option would be to initiate a picture or text message when movement or some other event is detected.

A pushbutton routine is used to activate the cell phone buttons. The routine is passed by a one character code representing the button, and the number of milliseconds the button is to be pressed for; 250 milliseconds is a normal push. A three second push of the "end" button turns the phone on or off. Normally, the phone is left off to conserve battery life. The routine works by looking up the button character in an array, then setting the pins for the appropriate relays to be high for the specified time. A section of the main program scans for input from a PC, allowing the user to operate cell phone buttons from their PC via the Arduino. Operating the phone from the computer is a handy troubleshooting tool once the phone's normal buttons are removed.

The process of taking a picture and sending it along with a text message is handled by a picture routine. It calls the pushbutton routine in the same sequence the phone would be used manually. It turns on the phone with a long press of the "End" button. Four short presses of the "End" button clears any messages and makes sure the phone is ready. The "Camera" button is pressed. A couple "Oks" takes the



■ **FIGURE 7.** Back side of the circuit board.



■ **FIGURE 8.** Close-up of the finished project packaged in a case.

photo and prepares to send it. A press of the "4" enters a "G" which brings up my gmail account. "Ok" twice more selects the address.

The picture message is almost sent. Now, the text message is entered using a final routine that accepts a text string as input. The string was created in the main program by piecing together text with the reading from an ultrasonic range finder: "Water level, inches 43.0." The routine works by scanning an array of 47 different characters the phone can send. Complementary arrays contain what button has to be pressed to get the character, and how many times it needs to be pressed. For example, a "c" requires that button "2" be pressed three times quickly. The routine then calls "pushbutton." The button is pressed for 100 milliseconds with a 100 millisecond delay in between presses. Typing the text in is pretty fast and is interesting to watch.

Putting It Together

Figure 8 shows a close-up of the finished project. The device was packaged in a 3 x 5 x 7 inch project box. A small hole was drilled in the back for the camera to see through, and holes were drilled in the bottom to accommodate a barrel jack for power and a USB cord for programming. The box is mounted on a 1 x 1/2 inch stick of wood so it can be easily attached to a fence, tree, or T-post with cable ties.

The Arduino consumes 41 mA in this configuration. That is about one amp-hour of battery draw in a 24 hour period. A six volt lantern battery has about 11 amp-hours of capacity, and would last about 11 days. Cost is $5.88 at Walmart which would add up over time at about $0.50 per day.

A better option is an old car battery. New, they are about $55 but I usually have an old weak one laying around that I use to jump other vehicles or run an electric fence. A smallish car battery has 70 amp-hours of capacity, will run the camera device for a couple months, and can be recharged.

The Arduino Mega and sonar sensor are the majority of the project. Hopefully, the items in the **Parts List** can be recycled from other projects like they were in my case.

What's Next

The device is operating in town well. My next step is to take it to the farm where cell phone reception can be a little spotty. Hopefully, I can pick a good spot where I will make a good connection most of the time. Once that piece is working, I will hook the sonar to one of the water tanks or feed bins. I currently record water and feed weekly. The daily

PARTS LIST

QTY	DESCRIPTION	SUPPLIER	PART #	COST EA.*	TOTAL COST*
1	Samsung Gusto 2 Cell Phone	Walmart		$15.00	$15.00
1	Arduino Mega	SparkFun	DEV-11061	$58.95	$58.95
14	Reed Relays (Spartan SIP Relays)	Mouser	816-9007-05-00	$0.94	$13.16
14	1N914 Diodes	RadioShack	276-1620	$0.06	$0.84
1	Wire-Wrapping Wire (30 AWG)	RadioShack	278-501	$4.99	$4.99
1	Wire-Wrapping Tool	RadioShack	276-1570	$6.49	$6.49
1	Grid Style Printed Circuit Board	RadioShack	276-147	$3.99	$3.99
1	Project Enclosure (7 x 5 x 3 inches)	RadioShack	270-1807	$6.99	$6.99
1	Female Headers	SparkFun	PRT-00115	$1.50	$1.50
1	Break-Away Headers (Straight)	SparkFun	PRT-00116	$1.50	$1.50
1	Ultrasonic Range Finder LV-EZ1	SparkFun	SEn-00639	$25.95	$25.95
					$139.36

*Prices at time of printing. Prices may have changed.

update will give me an indication of a problem much sooner. Ultimately, I could manifold as many as nine level sensors to get a full picture of the farm's status. A motion sensor would also be a nice addition, but my experience with a game camera at the farm shows wind and plant motion creates a lot of false triggers.

Perhaps a beam break sensor would be better, but there are a lot of tumble weeds in the area. Another interesting idea is to place a similar setup with a GPS chip in a floating ball. It would be interesting to monitor progress as the ball floated down the Yellowstone River on its way to the Gulf of Mexico. **NV**

■ BY FRED EADY

ADVANCED TECHNIQUES FOR DESIGN ENGINEERS

BUILDING AN INTELLIGENT DATA RADIO

Those guys and gals that wear pointy hats with the moons and stars on them are plenty popular these days. It seems that everything is going wireless. You and I — being of the embedded persuasion — use wireless technology to monitor and control the gadgets we design and build. Normally, we'll choose an off-the-shelf radio solution and integrate it into our design. However, there is absolutely no reason why we gadget engineers can't don a moon-encrusted star-studded pointy hat.

MENDING OUR WAYS

Before we're done, I am going to show you how to scratch-build a data radio using an RF IC that is melded with a microcontroller. The RF IC — an AXSEM AX5043 — is shown pads down in **Photo 1**. Once we get our arms around the AX5043, we'll tackle its microcontroller enhanced variant — the AX8052F143 — which you can see in **Photo 2**. We can also use the AX5043 with a stand-alone microcontroller such as the AXSEM AX8052F100.

Just about any microcontroller that can spawn an SPI portal can be used to drive an AX5043. However, we can gain a step by using the AX8052 and its development system which is a free tool chain that does everything but solder. Since we're not really ready to solder yet, we'll employ the services of the AX8052 development hardware, which is also known as the DVK-2. We'll use the DVK-2 in conjunction with the development system to create our methodology for the final design goal.

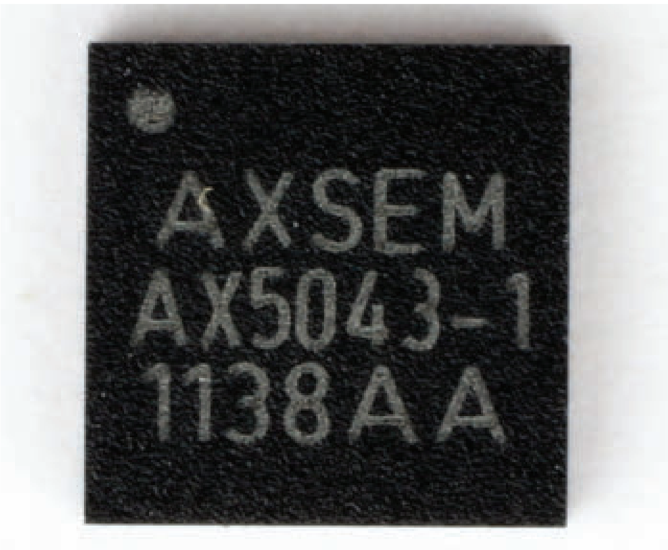
THE AXSEM AX5043

The little scudder smiling in **Photo 1** leads all narrow-band transceivers in performance and low power consumption. While drawing a mere 9.5 mA, the AX5043 provides -126 dBm receive sensitivity at 868 MHz using an FSK-modulated 6.25 kHz channel. Adjacent channel rejection is an amazing 43 dB. Data rates as low as 1,000 bps up to 115.2 kbps can be accommodated.

The AX5043 is designed to be used in battery-powered applications. It can sleep with one eye open on 500 nA of current, and wake up very quickly. Close both eyes and the AX5043 sleeps deeply using a minute 50 nA. Battery operation is made possible by its internal voltage regulator system. A 1.8 to 3.6 volt battery can be attached directly to the VDD_IO pin.

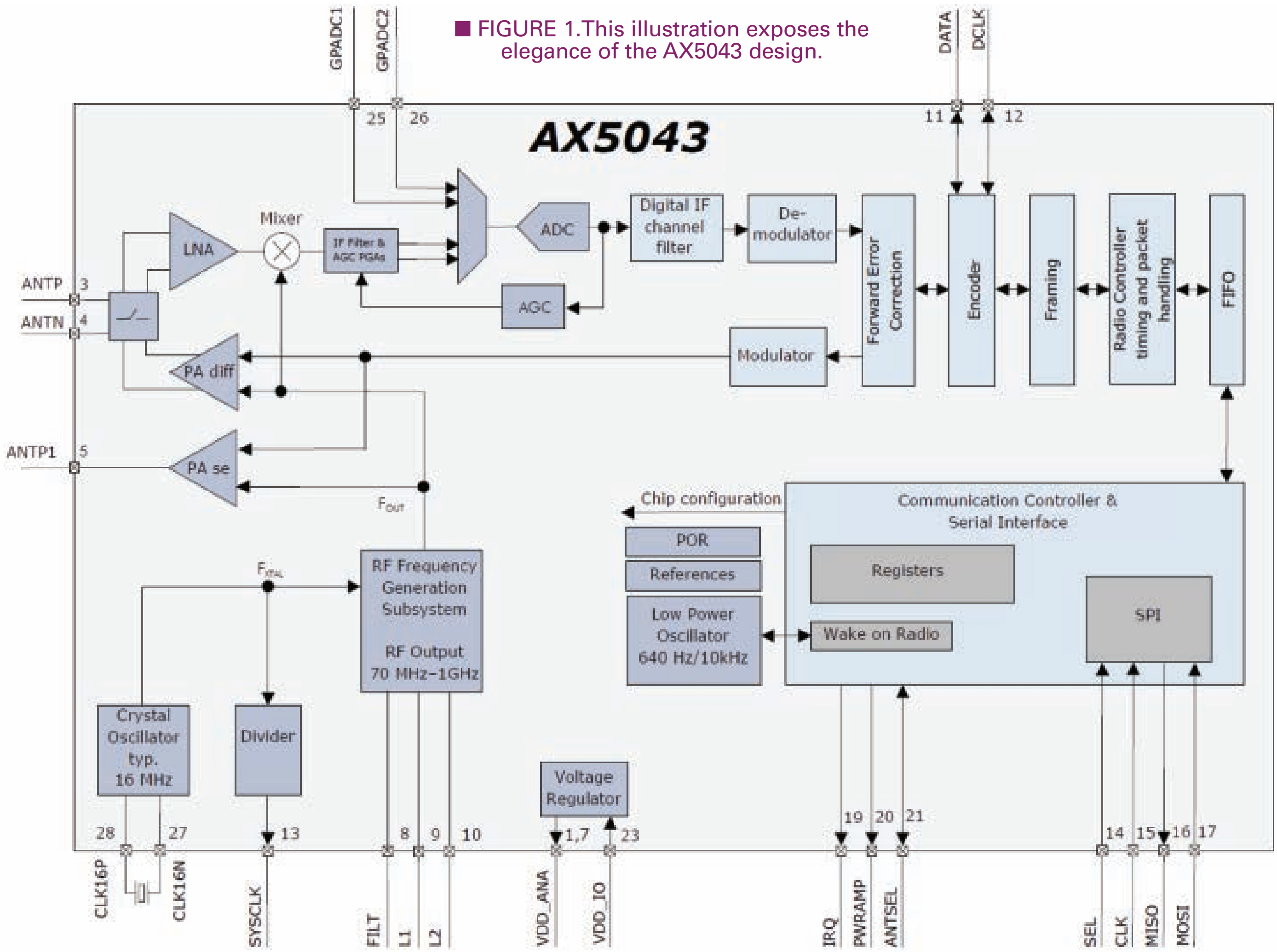
When the AX5043 is powered down, voltages that feed the digital and analog circuitry are switched off. The SPI portal remains operational in power-down mode, but can only be accessed at low speed. When the AX5043 closes both eyes, everything within it is switched off. The only way to wake it back up is to pull its SEL pin logically low. A logical low on the SEL pin resets the AX5043. The attached microcontroller monitors the MISO pin to determine when the AX5043 is fully awake and ready for operation.

The AX5043 is more easily taken in if broken down into smaller pieces. For instance, you can easily pick out where the microcontroller interfaces to it in **Figure 1**. Using the AX5043's SPI portal, a microcontroller can gain access to the internal register bank. The SPI portal is also used to transfer



■ PHOTO 1. The AX5043 is a narrow-band ultra-low power RF transceiver IC. This RF IC can be programmed to operate with a carrier frequency range of 70 to 1,050 MHz.

■ PHOTO 2. Add an 8052-based microcontroller to the AX5043 and you get an AX8052F143 SoC RF microcontroller.



■ FIGURE 1. This illustration exposes the elegance of the AX5043 design.

data to and from the AX5043. Thus, the AX5043 is configured using the SPI portal. From the microcontroller's perspective, the AX5043's SPI portal operates as an SPI slave. Incoming and outgoing data transferred by the communication controller is buffered by the AX5043's FIFO (First In First Out). The other side of the FIFO services the data radio components.

To better understand what happens beyond the AX5043's FIFO, let's define the two modes of operation. When the AX5043 is configured to move data through the SPI portal as frames, it is said to be in frame mode. In frame mode, checksums, preambles, and postambles are automatically generated. Frame mode data flow between the microcontroller and the AX5043 is regulated using interrupts. That explains the presence and purpose of the AX5043's IRQ output pin.

Take another look at **Figure 1** and note that the AX5043's 256 byte FIFO is also a data conduit for the radio controller and framing module. The data contained within the FIFO is organized as chunks. Chunks consist of chunk headers and a payload, and can contain packet data and radio related data such as signal strength and timestamps. Radio related data is inserted into the FIFO under the control of the radio controller.

Outgoing data packets that pass through the framing module are converted into a bit stream that is modulator friendly. Conversely, the framing module packetizes the bit stream that is flowing from the demodulator. The FIFO acts as the communications conduit between the microcontroller and the framing module. The microcontroller can interrogate the FIFO status by reading its status and count registers. To reduce the microcontroller's FIFO management burden, the FIFO can be configured to issue an EMPTY, NOT EMPTY, FULL, or NOT FULL interrupt.

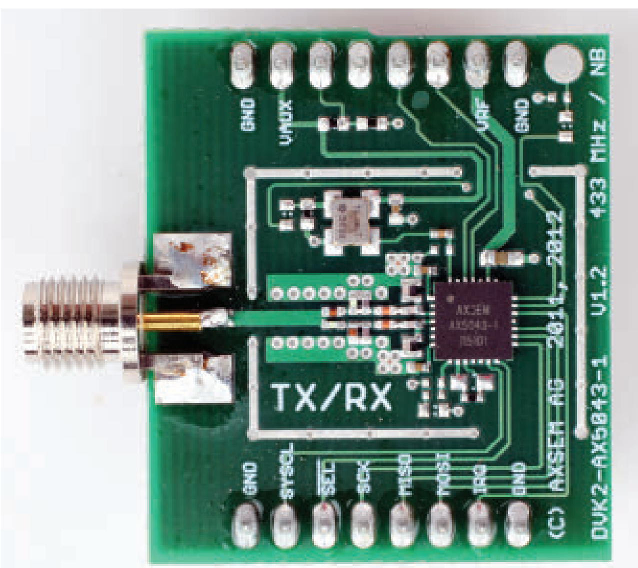
The bit stream from the demodulator and the bit stream to the modulator must pass through the encoder module. The encoder module can be used to invert, encode, or shape the bit stream. Encoding the bit stream entails how the transitions of the signal represent bits. For instance, NRZI (Non Return to Zero Inverted) encoding toggles the signal level to indicate a "1" and does not toggle the signal level to represent a "0." Shaping performed by the encoder is spectral in nature. Spectral shaping involves removing the DC component from the bit stream and guaranteeing the bit stream can be properly demodulated.

Disabling the communication controller forces the AX5043 into wire mode. In wire mode, the data is available directly from the encoder on the AX5043's DATA pin. The associated bit clock is generated by the AX5043 and is available on the DCLK pin. The raw data stream entering and exiting the DATA pin is under the control of the attached microcontroller. Note that the framing module, radio controller, and FIFO are eliminated from the data path. Thus, the AX5043 does not provide checksums, preambles, or postambles while in wire mode. Any modification to the raw data stream must be performed by the attached microcontroller.

In that a bit clock is available, the AX5043 can operate in synchronous wire mode. It can also operate in asynchronous wire mode. When operating in asynchronous wire mode, the AX5043 provides RS-232 start bit recognition when transmitting. That means the AX5043's DATA pin can be tied to the attached microcontroller's UART. When operating in asynchronous mode, the AX5043 inserts or deletes a stop bit to synchronize the RS-232 signal to its internal transmission clock. So, the microcontroller's UART must be configured to send two stop bits and be able to accept one stop bit. The DCLK pin is not used in asynchronous wire mode.

The final frontier as far as digital goes is the digital IF channel filter. This channel filter works with the demodulator to extract the data bit stream from the incoming IF signal. For this process to be successful, all of the ducks must be in a row. That is, all of the modulation schemes and data rates must be configured to match the incoming data bit stream.

What we do in the digital domain of the AX5043 determines what happens in its RF domain. Placing a 16 MHz crystal or TCXO (Thermally Compensated Crystal Oscillator) at pins 27 and 28 is sort of a no-brainer. Configuring the SYSCLK divider is probably pretty straightforward, as well. Download the AX5043 datasheet and take a look at section 6.1. You'll find that



■ PHOTO 3. The 433 MHz DVK-2 add-on board consists of an AX5043, a TCXO, and antenna matching circuitry. The AX5043 DVK-2 add-on board variants include 169 MHz, 433 MHz, and 868 MHz versions.

there is a galaxy of AX5043 register bits that can be twiddled. The question is, which bits do we twiddle and what values should we twiddle them to. Fret not. That pointy hat we want on our heads isn't out of reach yet.

AX-RADIOLAB

AX-RadioLab is a free AX5043 configuration program that also generates C source code for the AX8052F100 and AX8052F143. The code and configuration settings generated by AX-RadioLab are in the form of an AXCode::Blocks project. AX-RadioLab requires the services of the AX8052-IDE software package which is also a free download. The software package includes a copy of AXCode::Blocks, the AXSDB AX8052 debugger, the SDCC 8052 compiler, and the LibMF AX8052 support library. We will exercise all of these AX5043 tool chain members using the DVK-2 development kit.

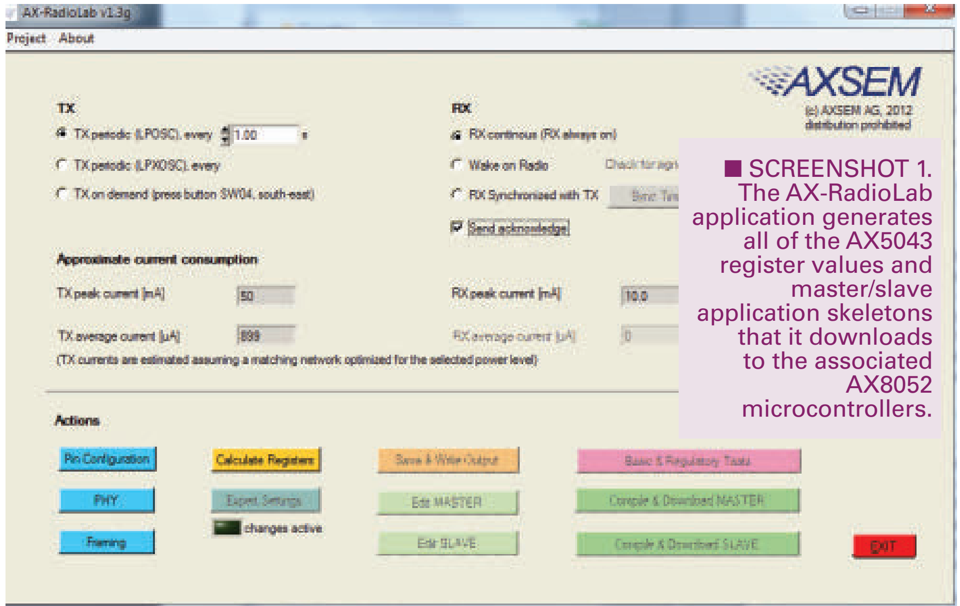
The AX-RadioLab application allows us to build an AX5043 configuration by simply entering our RF design criteria. As you can see in **Screenshot 1**, the AX-RadioLab allows us to configure the GPIO pins, define the PHY, and set up the packet framing parameters. Once all of the configuration data is punched in, AX-RadioLab calculates the register values and builds a set of AXCode::Blocks project files. With the help of the DVK-2's AXDBG hardware, AX-RadioLab can download the C firmware it generated to an AX8052.

Let's put a set of 433 MHz radios on the air. We'll call this AX5043 project *nv-433mhz-project*. Our project will take the shape of the TX and RX selections you see in **Screenshot 1**. The transmitter will send a packet every second, and once received on the other end, an acknowledgement will be returned.

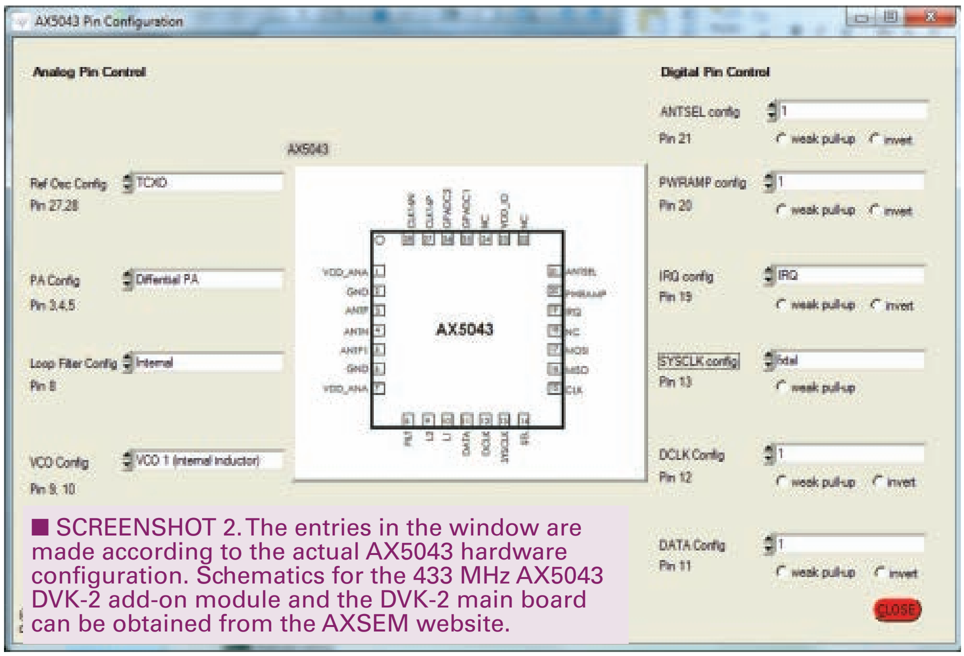
The AX5043's GPIO pins are divided into analog and digital groups. We'll configure the analog and digital pins according to the hardware layout of the 433 MHz DVK-2 add-on module under the lights in **Photo 3**. The schematic diagram for the 433 MHz DVK-2 add-on module can be downloaded from the AXSEM website. Go ahead and get your copy of the schematic because we will be referencing it as we configure the AX5043.

The AX5043 pins are configured in the pin configuration window which is shown in **Screenshot 2**. Let's begin by configuring the analog pins. Referencing the 433 MHz DVK-2 add-on schematic, we see that the AX5043 is being clocked by a TCXO. It can also be clocked using a crystal. However, that's not the case for the 433 MHz DVK-2 add-on module.

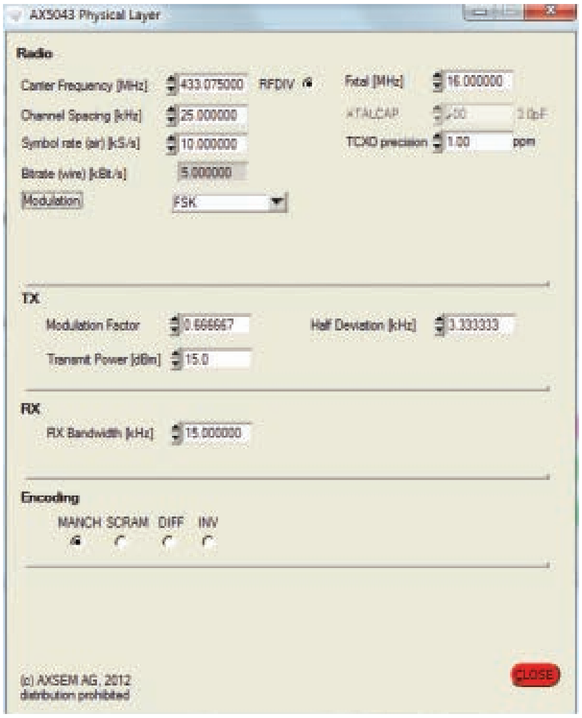
The antenna matching circuitry begins at the ANTP and ANTN differential antenna input/output pins. Note that these input/output pins are



■ SCREENSHOT 1. The AX-RadioLab application generates all of the AX5043 register values and master/slave application skeletons that it downloads to the associated AX8052 microcontrollers.



■ SCREENSHOT 2. The entries in the window are made according to the actual AX5043 hardware configuration. Schematics for the 433 MHz AX5043 DVK-2 add-on module and the DVK-2 main board can be obtained from the AXSEM website.



■ SCREENSHOT 3. This is a very interesting configuration window. Try running the symbol rate up and down. Note that other values follow and show red when you exceed their boundaries.

attached to an internal transmit/receive switch. The transmit/receive switch connects the antenna and antenna matching circuitry to the LNA (Low Noise Amplifier) in receive mode, and switches in the differential PA (Power Amplifier) during transmit. Take another look at **Figure 1** and you'll see that the AX5043 also offers a single ended PA which can be used to drive an external PA.

The AX5043 has an internal loop filter. The internal loop filter can be overridden by an external loop filter inductor. When using the AX5043's internal loop filter, its pins must be shorted. A peek at the 433 MHz DVK-2 add-on module schematic shows the external inductor as a no-connect. Instead, a zero ohm resistor is installed across pins 9 and 10.

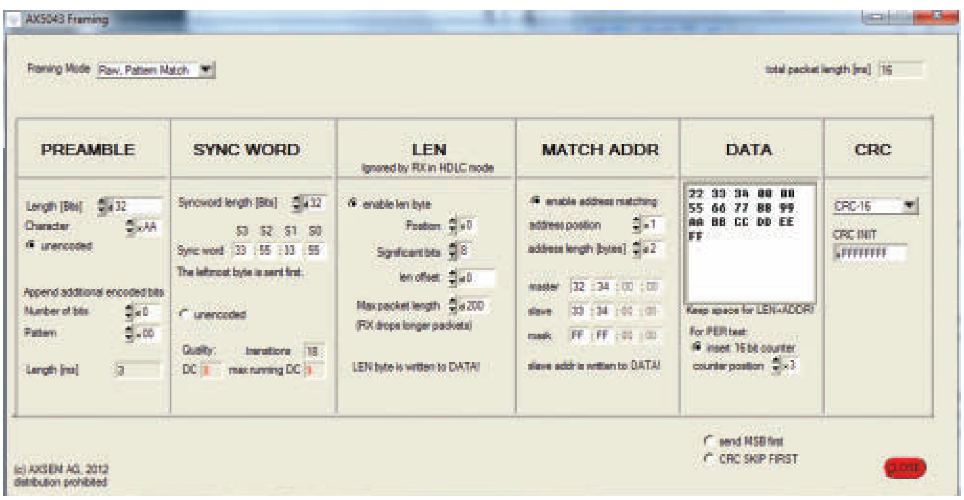
The AX5043 internal loop filter works in conjunction with a charge pump to set the synthesizer loop bandwidth. Its internal VCO (Voltage Controlled Oscillator) uses the charge pump and loop filter's output voltage as a reference to generate the carrier frequency.

We don't have much work to do on the digital pin side. If you consult the DVK-2 add-on module datasheet, you will find that the AX5043's ANTSEL pin is wired to control the TCXO. However, a closer examination of the ANTSEL circuit shows that it is interrupted by an uninstalled component. Instead, the TCXO's control line is tied logically low with a zero ohm resistor. With that, we don't have to do anything with the ANTSEL configuration parameter. The AX5043's PWRAMP pin is also going nowhere.

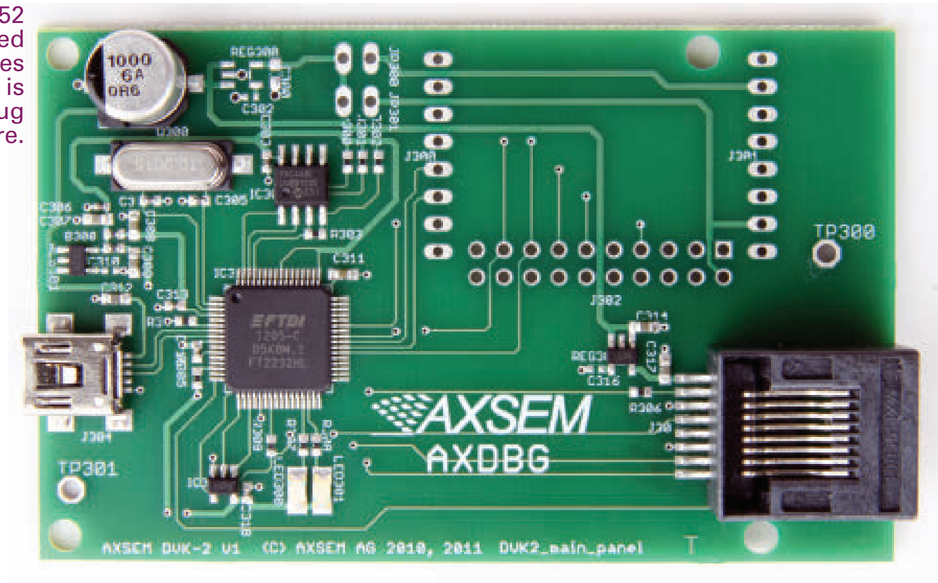
The latest version of AX-RadioLab allows us to choose one of three transmit demo modes. Some of the transmit demos will most likely use the AX5043's IRQ output. The same holds true for the receiver. So, to play it safe, we will enable the IRQ output pin.

The AX8052F100 microcontroller has a multitude of clocking options. According to the DVK-2 main board schematic, the AX5043's SYSCLK line is electrically connected to the AX8052F100's RSYSCLK pin. However, the AX8052F100 can also be clocked in low power crystal oscillator mode with the existing 32.768 kHz crystal. I loaded and ran demos with and without an entry in the SYSCLK slot. All of the demos functioned properly. I also checked the demo program overview located within the AX-RadioLab user manual and there was no mention of the SYSCLK signal being used. Since the AX5043 and AX8052F100 have their respective SYSCLK/RSYSCLK pins electrically connected and the demos run with the AX5043's SYSCLK pin enabled, we will choose to emit 16 MHz via the AX5043's SYSCLK pin. Recall that the DATA and DCLK pins are used in wire mode. We will most likely do something with wire mode in an upcoming Design Cycle. For now, we'll leave them at their default values.

Don't get real excited about the physical layer configuration we've laid out in **Screenshot 3**. The physical layer parameters will make more sense to you as we get further into the AX5043 design. Right now, we want to make sure



■ SCREENSHOT 4. This is my favorite window. Setting up the frames is like playing in the sandbox.



■ PHOTO 4. This piece of AX8052 development system hardware is used to program the AX8052F100 that rules the DVK-2 main board. The AXDBG is also used by AXCode::Blocks to debug the AX8052F100 firmware.

we're within the 433 MHz band. The symbol rate and encoding are wide open. You can also choose to alter the channel spacing. My suggestion is to experiment with the settings by driving up the symbol rate until other parameters turn red. The red parameters are warning you that you have stepped outside of the settings that would guarantee reliable communications.

Screenshot 4 is the sandbox portion of AX-RadioLab. This is where we build the packet structure. You can choose one of three packet protocols which are: HDLC, pattern match, and PN9 compatibility. As you can see, we are using pattern match. Again, I urge you to experiment with the framing settings.

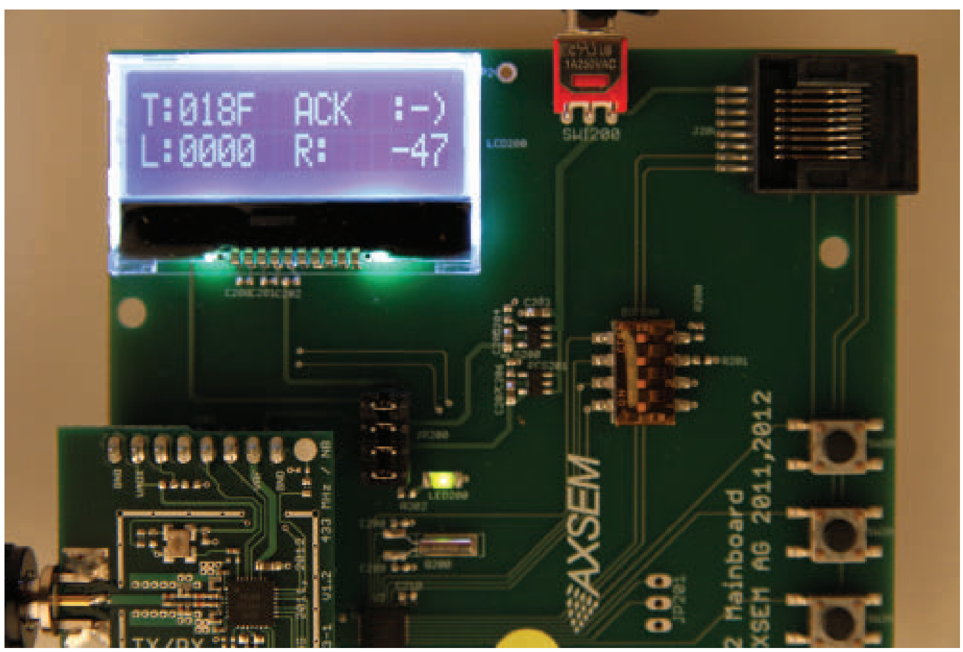
We are ready to have AX-RadioLab calculate the register values. Once it's done with the calculations, the expert settings button is exposed. Expert settings allow us to override the register values that AX-RadioLab calculated. We're still trying to get our pointy hats, so we'll not tweak any values just yet.

The Save & Write Output button is also exposed following the register calculation process. Clicking on this button puts the project files into the project directory we specified when opening AX-RadioLab. When the project files are in place, the Edit MASTER and Edit SLAVE buttons are exposed. Choosing either edit button will automatically open AXCode::Blocks and populate it with the MASTER or SLAVE project files. Those that already own pointy hats can modify the demo applications if they desire. However, we'll pass right now.

Three additional buttons are exposed with the pair of edit buttons. The Basic & Regulatory Tests button opens a window that allows us to set transmit patterns and do BRE (Bit Rate Error) tests. The remaining two buttons compile and download the MASTER and SLAVE demo applications and AX5043 configuration data.

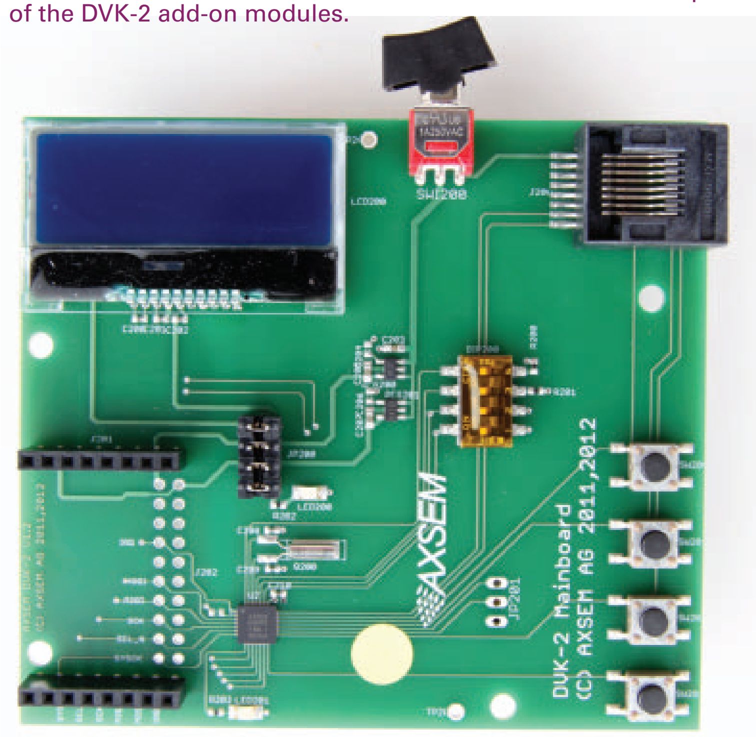
PROGRAMMING AND TESTING

AX-RadioLab uses the services of the AXDBG hardware showing its teeth in **Photo 4** to program the DVK-2 main boards' AX8052F100. From the looks of **Photo 5**, we've all



■ PHOTO 5. The smiley face on the display indicates that the receiver is acknowledging the transmitted data. I want quarter moons and stars on my pointy hat!

■ PHOTO 6. The DVK-2 main board is universal and can accept all of the DVK-2 add-on modules.



got pointy hats coming. The beauty of the AX8052 development system is that if we failed to get the radios to communicate, it's very easy to make a register change using AX-RadioLab and reprogram the AX8052F100s.

The smiley face on the transmitter LCD tells us that the receiver has acknowledged the last transmitted packet. So far, 399 packets have been transmitted with no packet loss. The last acknowledgement RSSI is -47.

I'll leave you with a more detailed shot of the DVK-2 main board in **Photo 6**. Meanwhile, straighten up that pointy hat because you're going to need it again. Oh yeah, you can now include the AX5043 in your Design Cycle. **NV**

Lemos International Co., Inc.
AX5043
AX8052F100
AX8052F143
AX8052
433 MHz DVK-2
Add-on Module
AX8052 Development System
AXCode::Blocks
AX8052-IDE
Software Package
AX-RadioLab
www.lemosint.com

Continued from page 30

also available for through-hole mount CR2032 holders, as well as surface- and through-hole mount holders fitting CR1216, CR1225, CR2016, and CR2025 coin cells.

These surface- and through-hole mount battery holders from Linx are for cost-sensitive, space-constrained applications involving computers, remote controls, and other compact electronics. Their nickel-plated phosphor bronze construction provides good conductivity and strength. Two metal arms secure the battery to the contact patch.

For more information, contact:
Linx Technologies
Web: **www.linxtechnologies.com**

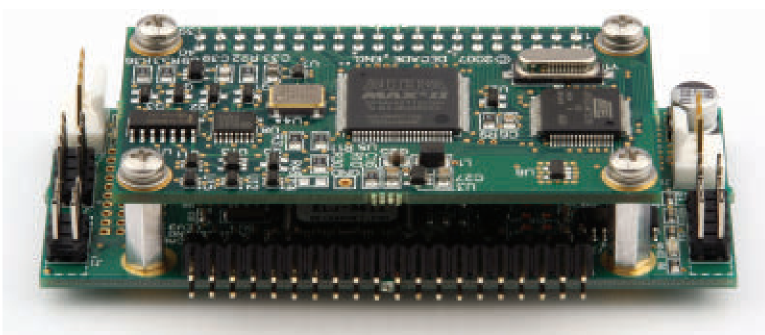
SERIAL SRAM PORTFOLIO EXPANDED

Microchip Technology, Inc., has expanded its serial SRAM portfolio with four new devices that feature the industry's largest densities and speeds. They are also the industry's first with 5V operation, which remains prevalent in automotive and industrial applications. These 512 Kb and 1 Mb SPI devices maintain the portfolio's low power consumption and small eight-pin packages. Speeds of up to 80 Mbps are achieved via the quad-SPI, or SQI™ protocol, providing the zero write-cycle times with near instantaneous data movement needed for offloading graphics, data buffering, data logging, displays, math, audio, video, and other data-intensive functions.

Two additional family members — the 23LCV512 and 23LCV1024 — offer cost-effective options for non-volatile, unlimited-endurance RAM via battery backup. These serial NVSRAM devices feature high speed operation without the high pin counts of parallel NVSRAM and comparable power consumption to

FRAM. This is beneficial for applications such as meters, black boxes, and other data recorders which require unlimited endurance or instantaneous writes along with non-volatile storage.

Microchip is creating a PICtail™ daughterboard which expects to be available in February 2013 for use



RAPID DEPLOYMENT PLATFORM FOR GRAPHICS DISPLAYS

BUD — now available from Decade Engineering — is both an Atmel mega-AVR microcontroller development board and a rapid deployment platform for Decade's popular BOB-4 video character generator module. With BOB-4 installed, BUD easily generates complex text and graphics displays on analog TV monitors. BUD delivers the ATmega640 processing power and peripherals to completely implement many BOB-4 applications. Real time measurement data can be superimposed on video from a camera or any NTSC/PAL video source. Demonstration software allows new users to immediately exercise the combined functionality of BUD and BOB-4. AVR-GCC source code is available for the GPS data display, TV-Typewriter, and real time clock example apps. BUD sells for $89.95 each (single). Production quantity discounts are available.

For more information, contact:
Decade Engineering
Web: **www.decadenet.com**

with its Explorer series of modular PIC microcontroller development boards, as well as the XLP 16-bit development board. This daughterboard will demonstrate the features of both the volatile and non-volatile devices in Microchip's new serial SRAM family, enabling designers to quickly evaluate them.

For more information, contact:
Microchip
Web: **www.microchip.com**

ROBOTIC CRAWLER KIT

RoboBrothers, Inc., is now offering the RoboCrawler kit. It is a simplified robotic crawler which offers an affordable option to hobbyists interested in multi-legged walkers.

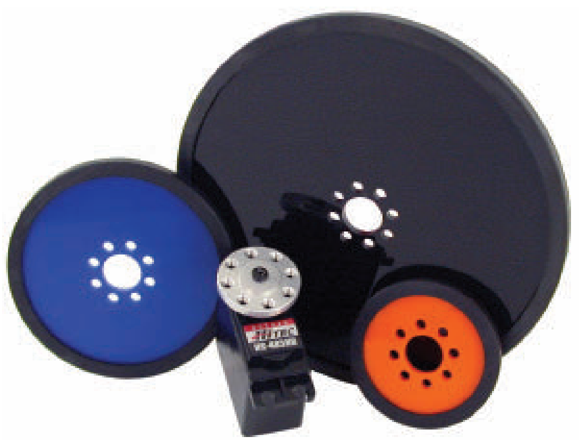
Features include:

- Eight servos for eight degrees of freedom for the legs.
- Each motion routine can have up to 30 sequences, and each sequence can have up to 15 poses.
- Sequence and pose can be reused for other motions to save Flash memories.
- One motion routine can have up to 450 pose transitions.
- RS-232 serial connection to PC for motion programming and execution.
- IR handheld remote to execute user-created program motions.
- Powered by five AA batteries.
- Spare ports available for installation of sensors for autonomous operation.

For more information, contact:
RoboBrothers, Inc.
Web: **www.robobrothers.com**

SERVO HUB AND PRECISION DISK WHEELS

ServoCity introduces their new .770" aluminum servo hubs. These new hubs are machined from



7075 T6 aluminum and have eight 6-32 tapped orbital holes that allow users to attach any of ServoCity's components that have the .770" pattern. They mate up perfectly with ServoCity's line of gears, sprockets, pulleys, and wheels which utilize the same .770" hub pattern. The aluminum servo hubs have broached splines to allow for a solid connection to any standard size Hitec or Futaba spline. The 1/2" diameter protrusion in the center of the hubs ensures that the attachment is perfectly centered.

In order to complement the new aluminum servo hubs, ServoCity has updated its entire line of precision disk wheels to incorporate the new .770" hub pattern. The precision disk wheels are offered in a wide variety of colors and diameters. The wheels provide excellent traction due to the rubber ring which surrounds the wheel. Each wheel is capable of holding up to 15 lbs without flexing.

0.770" GEARS AND SPROCKETS

ServoCity has also updated their line of plastic gears and sprockets. The gears and sprockets now utilize the .770" hub pattern with 6-32 holes and a 1/2" center bore which is used throughout their entire line of products. The gears are constructed of acetyl and are

offered in 1/8", as well as 3/16" face width, 32 and 48 pitch, and a vast range of sizes in order to fit almost any application. Like the hub gears, the new hub sprockets are constructed of acetyl for superior strength and durability. The sprockets are .100" thick and accept standard .250" (1/4") chain.

For more information, contact:
ServoCity
www.servocity.com

Model 324
WORLD'S MOST VERSATILE
CIRCUIT
BOARD HOLDERS
Our line of Circuit Board Holders add versatility & precision to your DIY electronics project. Solder, assemble & organize with ease.
MONTHLY CONTEST
Visit us on Facebook® to post a photo of your creative PanaVise project for a chance to win a PanaVise prize package.
Model 201
PANAVISE®
Innovative Holding Solutions
VISIT US ON
7540 Colbert Drive • Reno • Nevada 89511
1 (800) 759-7535 | www.PanaVise.com

Web-based Control and Monitoring
Get your project online... in minutes!
> No Programming Required
> No Port Forwarding or Dynamic DNS
> No Network Configuration Necessary
> No Need to Host a Web Server
IO-204 Ethernet Module
WHAT WILL YOU DO?
Real-time, secure control and monitoring...
> Servo and Relay Control
> X10 Home Automation
> RS-232 Serial Communication
> Temperature and Analog Sensors
> Digital Inputs and Outputs
all from a web browser.
> Event-based Alerts, Email and SMS
> Embed Controls in your Web Site
> Encrypted, Secure Access
> Data-logging Service
> Open API for Web Applications
http://www.iobridge.com
Garage Door
Outside
Outside Light
Inside
Driveway Cam
Solar Panel Output
(example dashboard)
10% Off
Discount Code: NVDiscount
ioBridge
iobridge.com/store
sales@iobridge.com
1-508-630-2165

Brandon built his own Mars Rover out of LEGO bricks.
His mom gave him a subscription to SERVO so he could take the next step.
Get access to the digital edition for free with your print subscription!
12 issues for $24.95
Call 1-877-525-2539
or go online at
www.servomagazine.com

■ BY LOUIS FRENZEL W5LEF

NEAR FIELD COMMUNICATIONS

Short Range, Long Results

Most of today's wireless communications are short range, meaning they work over a very limited distance.

- **Cellular**
- **Wi-Fi**
- **Bluetooth**
- **ZigBee**
- **Remote Keyless Entry**
- **Garage Door Openers**

and others are all short-range technologies.

What do we mean by "short?" For cellular, short may be a mile or so. For Wi-Fi, short means less than 300 feet. Bluetooth and ZigBee are even shorter with a maximum range of about 30 feet.

Now, there is a newer wireless technology that is the king of "short." It's called near field communications (NFC). "NFC short" means 4 or 5 cm or a couple of inches.

What good is that?

SO, WHAT GOOD IS IT?

As it turns out, NFC is amazingly useful despite its normal short range which is typically less than 20 cm. A number of growing applications like electronic payments replacing credit cards and access to facilities like a key are emerging. Here is a quick look at this wireless technology and some of its hot applications.

NFC BASICS

Wireless devices create radio waves that are emitted from an antenna. The resulting radio signal is made up of two parts: the near field and the far field. Both fields consist of a magnetic field at a right angle to an electric field.

The far field is the "real" radio wave because it is the signal that travels through the air to the receiving antenna.

These electric and magnetic fields behave as Scottish physicist James Maxwell predicted by his famous equations back in the 19th century. These two waves regenerate one another as they pass through space, until they eventually induce a voltage in the receiving antenna.

The power of the far field strength falls off at a rate proportional to the square of the range (r) between the transmitter and receiver, or $1/r^2$. The far field begins approximately several wavelengths from the antenna.

The near field also consists of electric and magnetic fields, but they are more independent. In fact, the magnetic field is more dominant and actually is the most useful component of the signal. The near field extends up to about one wavelength from the antenna.

The best way to think about the near field is a magnetic field generated by the primary of an air core transformer. The receiving antenna then is the secondary of that transformer. The primary produces the magnetic field and induces a voltage into the secondary winding over a short distance. The strength of that magnetic field really decreases

WAVELENGTH

Wavelength is the physical distance between peaks of the magnetic or electric fields. It is a function of the speed of light (radio waves) and the frequency of the signal. Wavelength is represented by the Greek lower case lambda (λ). The basic formula for wavelength in meters is:

$$\lambda = 300/f\text{MHz}$$

For example, the distance between the power peaks of a 400 MHz signal is:

$$\lambda = 300/400 = 0.75 \text{ meter}$$

A meter is 3.28 feet, so the wavelength is:

2.46 feet

NFC operates at 13.56 MHz with a wavelength of:

$$\lambda = 300/13.56 = 22.12 \text{ meters or } 72.6 \text{ feet}$$

with range (r) by a factor of $1/r^6$. The two antennas have to be very close together if any communications are to take place.

HOW NFC WORKS

Most short-range wireless technologies use ultra-high frequencies (UHF) greater than 300 MHz, or microwaves (> 1 GHz). NFC uses 13.56 MHz. It is one of the industrial-scientific-medical (ISM) frequencies designated by the Federal Communications Commission (FCC) for low power, unlicensed applications. Most other countries recognize that frequency for unlicensed use. Power levels are very low, typically less than one watt. The modulation is amplitude shift keying (ASK).

NFC radios (often called NFC controllers) consist of a transmitter and a receiver, as well as an embedded microcontroller that implements the sophisticated protocols of the technology. The radio operates half duplex, meaning in any communications between two NFC devices, only one device transmits at a time. The basic approach is for the receiver to listen first, and transmit only if there is no other signal on the frequency.

The protocols are set up so that one of the communicating devices acts as an initiator that polls the channel for others that may be listening. These other devices are called targets and are the other half of the communications. The target responds to the initiator's invitation and a link is established at which point the two devices exchange data based on the application.

There are three basic modes of operation: Read/Write, Peer-to-Peer, and Smartcard emulation. In the Read/Write mode, one device may store information in the memory of the other. It is used for the process of programming one device by the other. The Peer-to-Peer mode is just a data transfer mode that lets the two devices communicate. The Smartcard emulation mode means that the target device acts like a typical smart credit card — the kind with the built-in chip that can be read by some kind of point of sale (POS) retail terminal.

■ FIGURE 1. This is a smart card and smartphone payment terminal. Also shown is the austriamicrosystems AS3911 NFC IC that puts most of the retail terminal on a single chip. This chip can handle the new VHBR of 6.8 Mb/s.

The Read/Write and Peer-to-Peer modes are commonly active communications modes where both communicating devices have their own power supply, battery, or are AC operated. In the Smartcard emulation mode, one of the devices is a passive device like a Radio Frequency Identification (RFID) tag. A passive device does not have its own power source. Instead, the initiator's signal induces a voltage into the passive target's antenna that is then rectified and filtered into a DC voltage that powers up the transceiver. The target device can then transmit a signal back to the initiator by load-modulating the antenna. This is a form of ASK that the initiator can receive. The signal levels are all very low but the close range makes it work.

The data rates used in NFC are very low compared to the rates of other short-range wireless devices. The most common rates are 106, 212, and 424 kb/s. A very high bit rate (VHBR) has been proposed for the standard providing a rate of 6.8 Mb/s.

There are a variety of international standards defining the protocol and specification of NFC devices. For more details on the standards, modes of operation, and applications, go to the NFC forum website at **www.nfc-forum.org**. This non-profit organization promotes NFC, maintains standards, and provides device testing and certification services to make sure that all devices are interoperable.

NFC has been the proverbial solution looking for a problem.

WHAT DO YOU DO WITH NFC?

NFC does not seem all that useful given its short range. As it turns out, however, NFC is finding many applications. The largest and most promising is mobile payments or what is being called e-commerce. The idea here is to build NFC into all smartphones and program them to act as credit card replacements. Then when you go to buy something, all you do is wave your phone in front of an NFC reader terminal and the transactions take place instantaneously (see **Figure 1**).

Such a payment system requires more than the radio technology to make it work. Retailers must collaborate with credit card companies, banks, and even the cellular network providers to make a

■ FIGURE 2. Square's card reader dongle that turns a smartphone or tablet into a credit card terminal.

■ FIGURE 3. Just tap your smartphone on the reader to pay for your train fare. *Courtesy NXP Semiconductor.*

transaction happen. It is a complex process and one that has been slow to develop. All the major players are trying to get a piece of the financial action that such a system will generate. Currently, there are several major systems in place or in the trial stage. You can do mobile payments right now, but the number of phones containing NFC is limited as are the number of retailers that have NFC readers.

The first major effort is the Google Wallet. A good number of Google Android smartphones already have NFC. Furthermore, Google has set up a payment system in cooperation with MasterCard and Citi-Group to handle the payment transaction for those retailers who have NFC readers. Some retailers now offer the service, but there has not been a massive movement to this payment system.

Another effort to promote the e-payment system is Isis, which is a payment venture by the cellular carriers AT&T, T-Mobile, and Verizon. Tests are now going on in Texas and Utah to fine-tune the process.

The most recent venture that may really help this approach become more widely adopted is the one put together by Wal-Mart, Target, 7-Eleven, Best Buy, and a dozen others retailers. Called the Merchant Customer Exchange (MCX), it could provide the critical mass leading to more widespread use.

There are many other smaller efforts by individual companies (e.g., Starbucks), but none have major traction yet. The whole mobile payment movement is still in its early stage of development. As of this year, only about 15% of smartphones have NFC. That will grow over time, and if Apple adds it to its new iPhone5, the impact could be great. Still only about 50% of the US population has a smartphone. That too is expected to grow, making NFC payment a distinct possibility for many more.

E-payments may not be universal. Most customers are not likely to give up their wallets for a smartphone-only payment method. Many are still concerned over security issues. While NFC is inherently secure because its short range makes the signals hard to intercept and encryption techniques are routinely used in the transaction, many fear stolen credit card numbers, false charges, and stolen identity.

The financial projections for NFC payments are astronomical. While many will adopt this method, others will stay with the still convenient credit card system. Clever new card readers like Square's new dongle (see **Figure 2**) make it easier than ever for retailers to adopt credit card payments. The device plugs into most smartphones and tablets using a special app program that allows the cell phone to complete the transaction over the cellular network and Internet. This device has been a major hit with small businesses and individuals. Square only charges a 2.75% fee — lower than traditional credit card fees. Pay Pal also offers a similar device call the Pay Pal Here and a related transaction service.

Another major application is to use the NFC empowered smartphone as a key to unlock access to facilities. Employees may use it to get into their secure place of work. Or, the key may provide access to other facilities like parking lots. Transactions requiring payment are also a target application. Fares on buses and trains, for example, could be paid by smartphone (see **Figure 3**).

Another emerging application is the use of passive NFC tags to provide information to smartphone users. The tags — called stickers — are thin plastic or paper backings holding an NFC passive or RFID memory chip. A smartphone acting as an NFC reader can interrogate the sticker to get advertisements, maps and directions, promotional information, and other data. The stickers could appear anywhere. Examples are posters, buildings, walls, or any place it might be helpful to provide information. These tags have a memory big enough to hold URLs, credit card data, advertisements, maps, business card information, or interesting graphics.

One big potential use of NFC is pairing other wireless devices. As you probably know, some short-range radios need to exchange some initial set-up information or passwords in order to establish communications. Bluetooth and Wi-Fi are examples. One positive proposal is that this pairing be accomplished by including an NFC chip in each device like a laptop or tablet so that when the two devices to be paired get near one another, they automatically exchange the necessary data so that communications are established without human interaction. This one we can all get behind.

FINAL THOUGHTS

The potential for NFC is great. It was originally developed in 2002 by Sony and Philips Semiconductor (now NXP). These companies — along with Nokia — started the NFC Forum in 2004 and the standards have been around since 2006. So, NFC is no new technology. It has been the proverbial solution looking for a problem. Many applications have been established and are in use but they are not universal. NFC is a work in progress and it remains to be seen how widely adopted it becomes. Like most other short-range wireless technologies, it will no doubt find its niche. **NV**

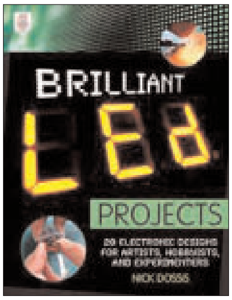
For complete product details, visit our webstore!!

The Nuts & Volts WEBSTORE

GREAT FOR DIYers!

NEW!

Brilliant LED Projects
by Nick Dossis

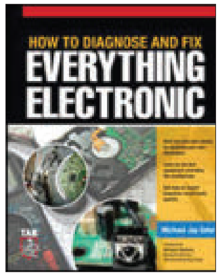


Brilliant LED Projects presents 20 hands-on, step-by-step projects for you to make using inexpensive, commonly available components. Projects range from simple, functional devices — such as a "green" LED flashlight and a flashing rear bike light — to more complex designs, including color-changing disco lights and persistence-of-vision (POV) gadgets — all featuring easy-to-follow instructions, highlighted with detailed illustrations.

$24.95

How to Diagnose and Fix Everything Electronic
by Michael Jay Geier

Master the Art of Electronics Repair!



In this hands-on guide, a lifelong electronics repair guru shares his tested techniques and invaluable insights. *How to Diagnose and Fix Everything Electronic* shows you how to repair and extend the life of all kinds of solid-state devices, from modern digital gadgetry to cherished analog products of yesteryear. **About the Author:** Michael Jay Geier began operating a neighborhood electronics repair service at age eight that was profiled in *The Miami News*.

$24.95

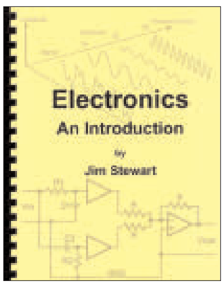
Programming Arduino Getting Started with Sketches
by Simon Monk

Program Arduino with ease!

Using clear, easy-to-follow examples, *Programming Arduino: Getting Started with Sketches* reveals the software side of Arduino and explains how to write well-crafted sketches using the modified C language of Arduino. No prior programming experience is required! The downloadable sample programs featured in the book can be used as-is or modified to suit your purposes.

$14.95

Electronics An Introduction
by Jim Stewart

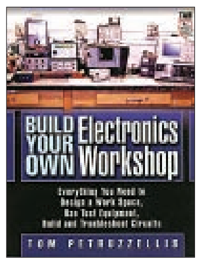


This book is designed as an in-depth introduction to important concepts in electronics. While electronics can be highly mathematical, this text is not about calculations. It is about how electronic equipment is able to extract, process, and present information held in electrical signals. If you are in — or studying to be in — a profession that requires the use of electronic equipment, then this book will provide the insight necessary to use such equipment effectively.

$33.95*

Build Your Own Electronics Workshop
by Thomas Petruzzellis

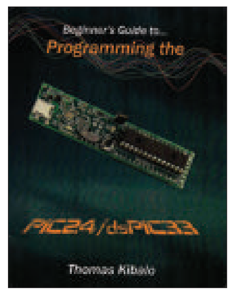
BUILD YOUR OWN DREAM ELECTRONICS LAB!



This value-packed resource provides everything needed to put together a fully functioning home electronics workshop! From finding space to stocking it with components to putting the shop into action — building, testing, and troubleshooting systems. This great book has it all! And the best part is, it shows you how to build many pieces of equipment yourself and save money, big time!

Reg Price $29.95
Sale Price $26.95

Beginner's Guide to ... Programming the PIC24/dsPIC33
by Thomas Kibalo

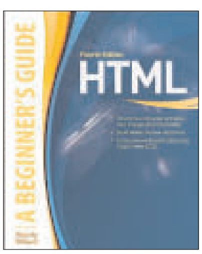


Kibalo takes you step by step through the fundamentals of programming the PIC24H which can equally be applied to the dsPIC33. His clear explanation of the inner workings make learning the PIC24H/dsPIC33 16-bit architecture easy. His code examples demonstrate how to perform the functions most applications require. The hardware is shown in a simple breadboard setup so even a beginner can build it, along with very few extra components needed.

$39.95*

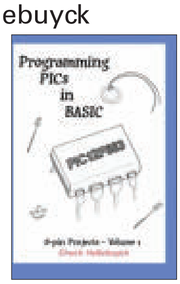
HTML: A Beginner's Guide
by Wendy Willard

Create highly functional, impressive websites in no time. Fully updated and revised, *HTML: A Beginner's Guide*, Fourth Edition explains how to structure a page, place images, format text, create links, add color, work with multimedia, and use forms. You'll also go beyond the basics and learn how to save your own web graphics, use Cascading Style Sheets (CSS), create dynamic web content with basic JavaScript, and upload your site to the web. By the end of the book, you'll be able to build custom websites using the latest HTML techniques.



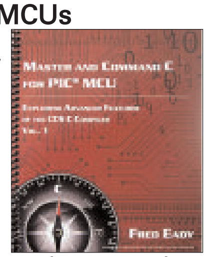
$29.95

Programming PICs in Basic
by Chuck Hellebuyck



If you wanted to learn how to program microcontrollers, then you've found the right book! Microchip PIC microcontrollers are being designed into electronics throughout the world and none is more popular than the eight-pin version. Now the home hobbyist can create projects with these little microcontrollers using a low cost development tool called the CHIPAXE system and the Basic software language. Chuck Hellebuyck introduces how to use this development setup to build useful projects with an eight-pin PIC12F683 microcontroller. **$14.95**

Master and Command C for PIC MCUs
by Fred Eady



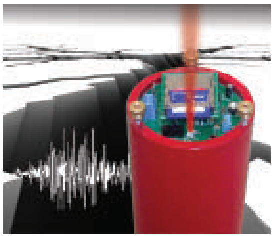
Master and Command C for PIC MCU, Volume 1 aims to help readers get the most out of the Custom Computer Services C compiler for PIC microcontrollers. The author describes some basic compiler operations that will help programmers particularly those new to the craft create solid code that lends itself to easy debugging and testing. As Eady notes in his preface, a single built-in CCS compiler call (output_bit) can serve as a basic aid to let programmers know about the "health" of their PIC code.

$14.95

CALL 1-800-783-4624 today!
Order online @ www.nutsvolts.com

PROJECTS

Seismograph Kit

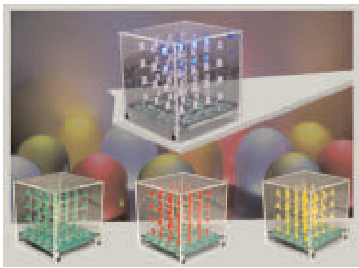


As seen in the May 2012 issue. Now you can record your own shaking, rattling, and rolling.

The Poor Man's Seismograph is a great project /device to record any movement in an area where you normally shouldn't have any. The kit includes everything needed to build the seismograph. All you need is your PC, SD card, and to download the free software to view the seismic event graph.

Subscriber's Price **$79.95**
Non-Subscriber's Price **$84.95**

3D LED Cube Kit



This kit shows you how to build a really cool 3D cube with a 4 x 4 x 4 monochromatic LED matrix which has a total of 64 LEDs. The preprogrammed microcontroller that includes 29 patterns that will automatically play with a runtime of approximately 6-1/2 minutes.
Colors available: Green, Red, Yellow & Blue

Subscriber's Price **$57.95**
Non-Subscriber's Price **$59.95**

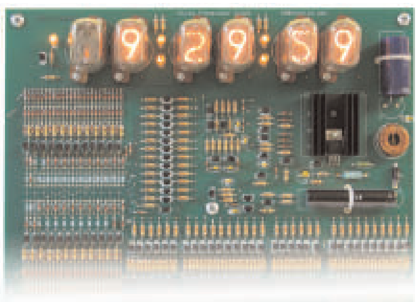
Battery Marvel Kit

As seen in the November 2011 issue. Battery Marvel helps protect cars, trucks, motorcycles, boats, and any other 12V vehicles from sudden battery failure. This easy-to-build kit features a single LED that glows green, yellow, or red, indicating battery health at a glance. An extra-loud piezo driver alerts you to any problems.

For more info, please visit our website.

Subscriber's Price **$18.95**
Non-Subscriber's Price **$19.95**

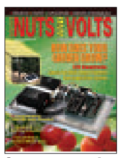
Neon Transistor Clock Kit



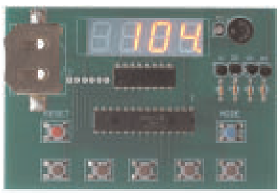
Add HIGH VOLTAGE to your clock!
This is a Nixie Tube display version of the Transistor Clock. It uses only discrete components — no integrated circuits.
For more info, see the April 2012 issue.

Subscriber's Price **$245.95**
Non-Subscriber's Price **$249.95**

Sorting Counter Kit



As seen in the July 2011 issue.

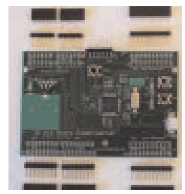


Sorting counters have many uses — keeping score, counting parts, counting people — it is just a handy gadget to have around. This is a very simple project for those who want to learn to solder or are interested in using microprocessors and how they function. No special tools are needed, just a small tip soldering iron. It has no box as it stands alone, therefore there is no drilling.

Subscriber's Price **$33.95**
Non-Subscriber's Price **$39.95**

32-Bit Micro Experimenter Board

The 32-Bit Micro Experimenter is the fastest way to learn 32-bit microcontrollers.

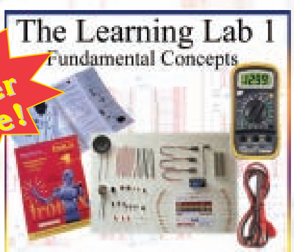


The kit includes onboard 46 programmable I/O and USB, free software, carefully documented step-by-step experiments for USB, embedded web server, graphics and audio, wireless, RTOS, and file I/O. User pushbuttons, LEDs, and 32 kHz clock crystal. Can be used in solderless breadboard environment or stand-alone.

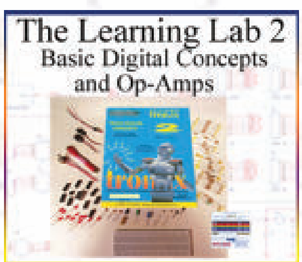
Subscriber's Price **$89.95**
Non-Subscriber's Price **$93.95**

FOR BEGINNER GEEKS!

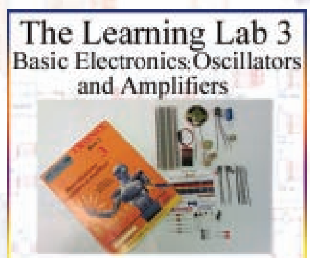
New Lower Price!



$59.95



$49.95



$39.95

The labs in this series — from GSS Tech Ed — show simple and interesting experiments and lessons, all done on a solderless circuit board.

As you do each experiment, you learn how basic components work in a circuit, and continue to build your arsenal of knowledge with each successive experiment.

For more info and a promotional video, please visit our webstore.

ELECTRONET

For the ElectroNet online, go to www.nutsvolts.com click Electro-Net

BY JOE PARDUE

SMILEY'S WORKSHOP #51

FRITZING WITH THE ARDUINO — Part 3

Last month, we continued learning about Fritzing — a novice-friendly electronics hardware design package that we are using to design a real time clock (RTC) shield for an Arduino. We saw how to use Fritzing to take the breadboard design from Part 1 and turn it into a schematic drawing that is crucial for understanding and documenting a design. We also saw how to take that schematic and create a printed circuit board (PCB) design that we sent off to BatchPCB. Well, after three weeks the PCB arrived and it works! Let's take a look at it and then go a bit further with Fritzing to start learning how to make parts.

THE PCB FINALLY ARRIVES!

Figures 1, **2**, and **3** show the Fritzing breadboard, the schematic, and the PCB views, respectively, that were used to generate the Gerber files that I sent to BatchPCB. I purchased one PCB for $26.27 and received it three weeks later (really not bad at all). Plus, they sent me an extra board for free — a two for one hidden sale that you might (or might not) get!

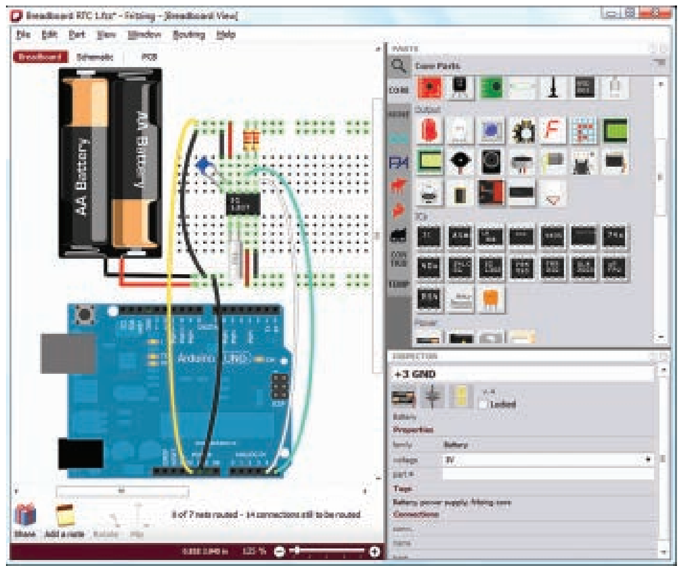


FIGURE 1. A real time clock on a breadboard.

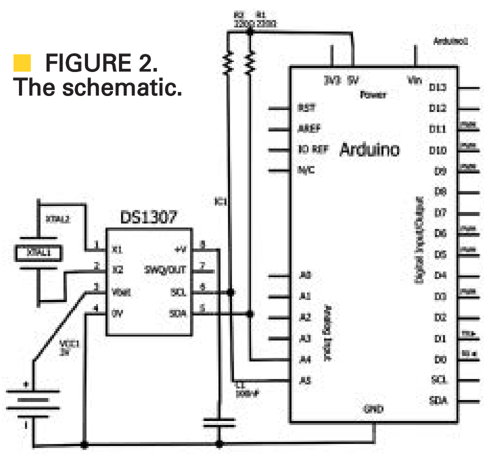


FIGURE 2. The schematic.

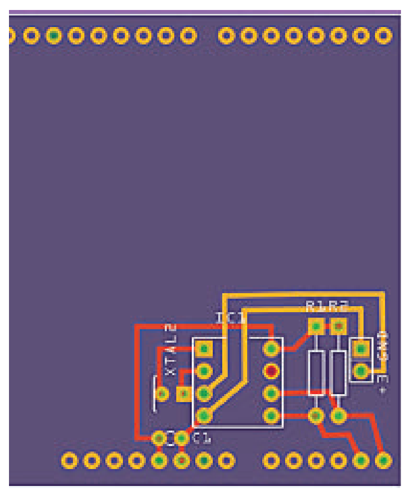


FIGURE 3. The PCB.

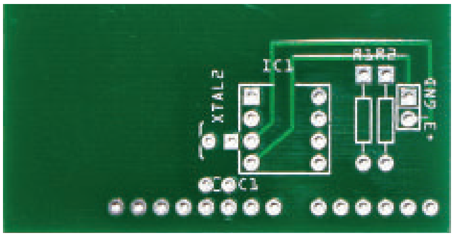


FIGURE 4. Fritzing RTC PCB, top side.

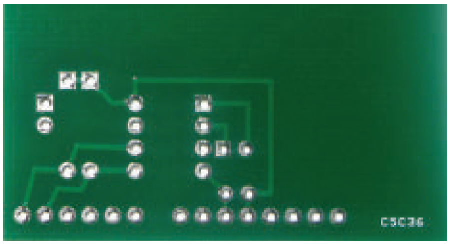


FIGURE 5. Fritzing RTC PCB, bottom side.

The PCB is shown front and back in **Figures 4** and **5**.

The RTC part of the design is fairly small and tucked away in the bottom left corner to be near the Arduino pins needed to provide power and the I²C connections required by the DS1307 chip. That seems to leave a lot of wasted board space, and indeed when I designed (not using Fritzing) a nearly identical circuit, I made the board much smaller. Remember, though, that the purpose of all this is to introduce Fritzing and how to use it to design an Arduino shield. To be a compliant shield, you need the PCB to fit on an Arduino — thus, the rows of holes on the top and bottom to fit over the Arduino connectors.

GETTING BETTER ACQUAINTED WITH FRITZING

Fritzing is a too-cool tool. We've had a decent introductory 'quick start.' Let's drill deeper and see what else we can do. Please note that Fritzing is evolving and works on several platforms, so your version may not be exactly what is shown here, but it should be close enough that you can figure out the (hopefully) slight differences.

Figure 6 shows the necessary parts, which are listed in **Table 1**. **Figure 7** shows the board with the parts soldered on and plugged into an Arduino for testing. Plug it into the Arduino as shown in **Figure 8**. Yup, it worked using the DS1307 RTC software discussed in the July 2012 Smiley's Workshop.

So, you noticed all that empty space. And we mentioned why we can't make it smaller. We would have to design another part that would represent only those pins on the lower part of an Arduino shield, and then actually only four of those pins. To do that, we would need to design our own part. We'll get to that momentarily.

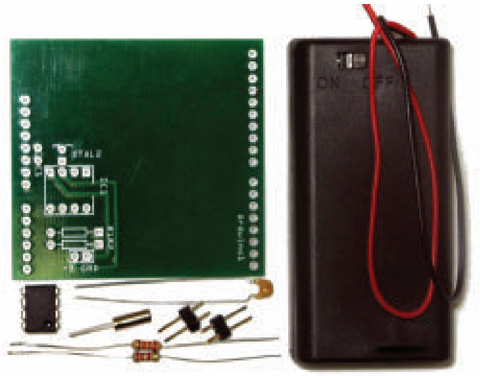


FIGURE 6. RTC PCB and parts.

#	Name	Distributor Part #
1	DS1307	Mouser 700-DS1307
1	32.768 watch crystal	Mouser 815-AB38T-32.768KHZ
1	Cap 0.1 µF	Mouser 21RZ310-RC
2	Resistor 2.2K ohm	Mouser 660-MF1/4DCT52R2201F
1	Battery box - 2 AA	Jameco 216120
2	Two-pin male header	Jameco 108338

Table 1. RTC bill of materials.

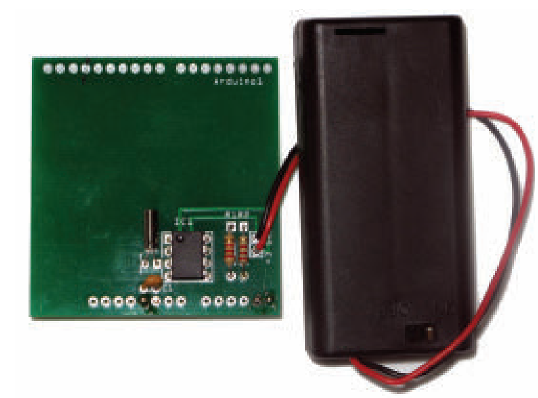


FIGURE 7. The built RTC PCB.

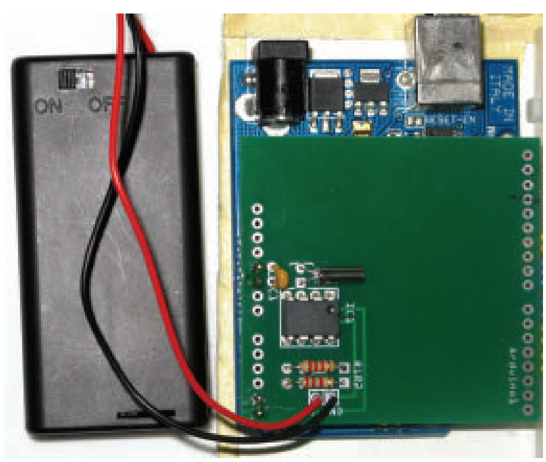


FIGURE 8. RTC PCB tested on an Arduino.

Window Views

Open the Core parts bin, and drag and drop the generic IC onto the breadboard as shown in **Figure 9**. In this view, we see the Parts and Inspector views on the right of the Fritzing window. These items are docked to the right, but can be dragged out and undocked as shown in **Figure 10**. There are other windows that we can view by opening the Windows menu item as shown in **Figure 11**.

Having all the windows open and docked on the right makes things a bit crowded, so undocking them while using them makes most sense as shown in **Figure 12**.

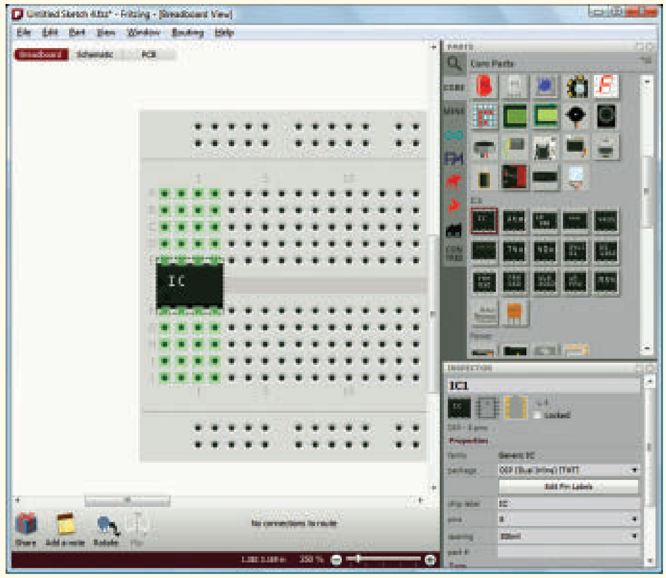


FIGURE 9. Select a generic IC.

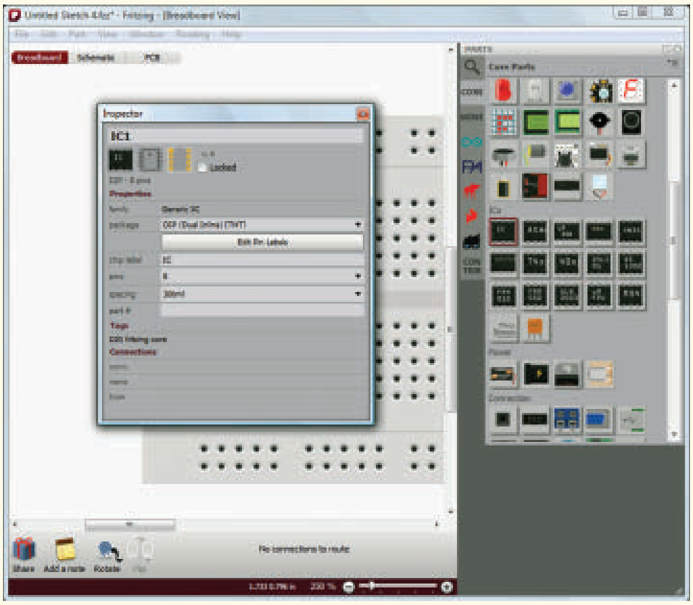


FIGURE 10. Inspector undocked.

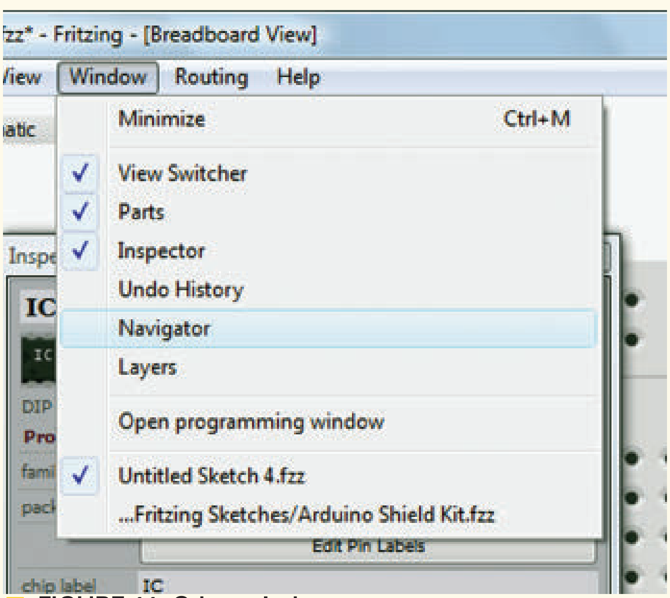


FIGURE 11. Other windows.

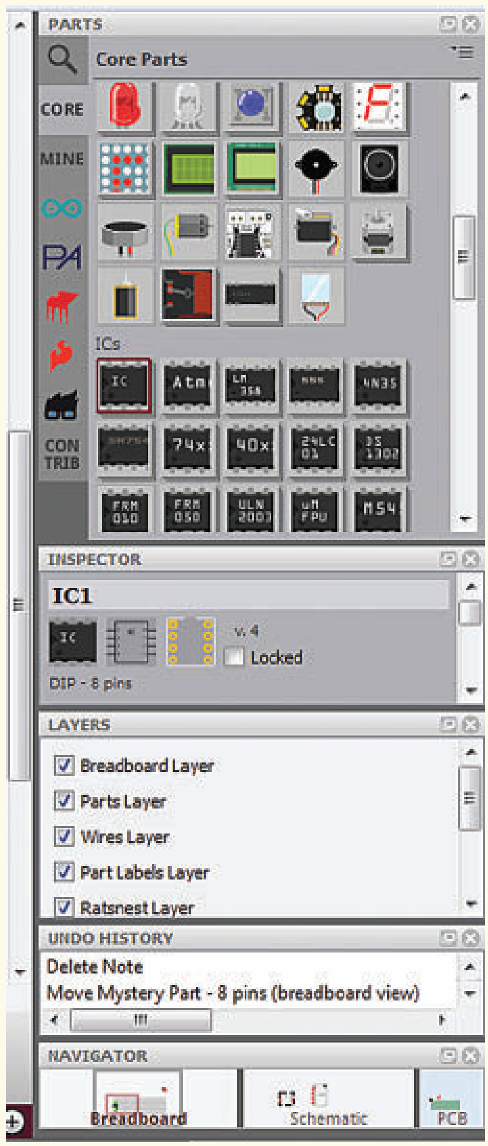


FIGURE 12. All windows open.

Setting Preferences

The Preferences dialog (**Figures 13** and **14**) has four tabs for setting preferences for Fritzing: General, Breadboard View, Schematic View, and PCB View. In the General tab, we see that we can select from a bunch of languages, but we'll leave it on English. Also, we can change Colors for highlighting, plus we can change the mouse wheel behavior and whether we autosave or not, and if so, how often. Let's leave these at their default values.

In the Breadboard View tab shown in **Figure 15**, we see that we can set the background color. Clicking on that gives us **Figure 16**, where we click on Custom Color that gives us the dialog box shown in **Figure 17**. We then select the Val for 245 which gives us a much lighter gray background that provides better contrast with the breadboard parts (in my opinion).

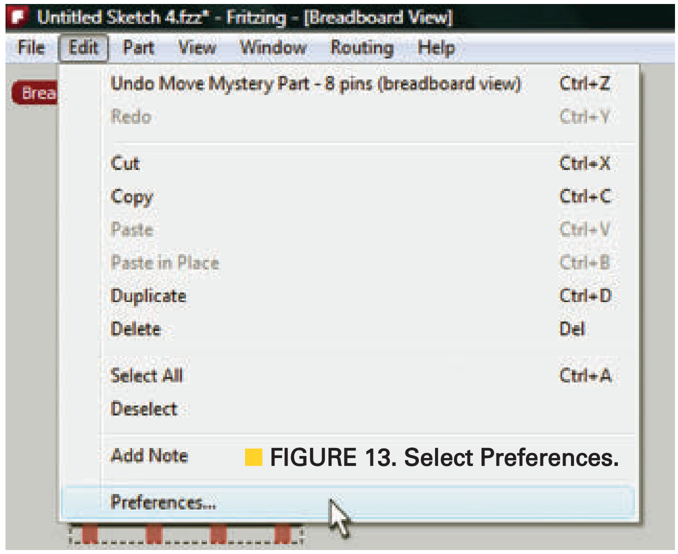


FIGURE 13. Select Preferences.

You can open the other tabs and change the colors for the Schematic or PCB View as you prefer. I lightened up the PCB background — again for better contrast — but you might like it the way it is.

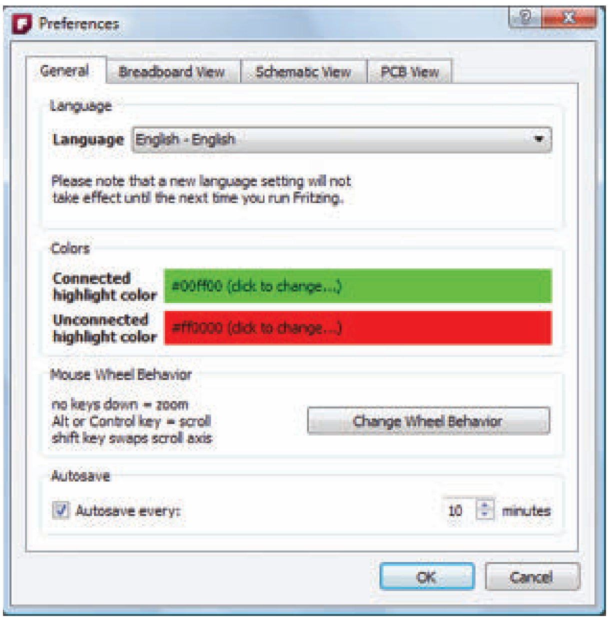


FIGURE 14. Preferences dialog box.

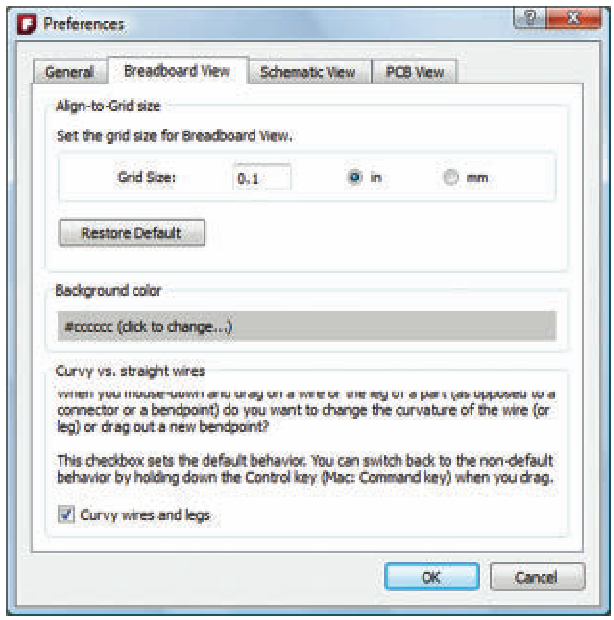


FIGURE 15. Preferences Breadboard View.

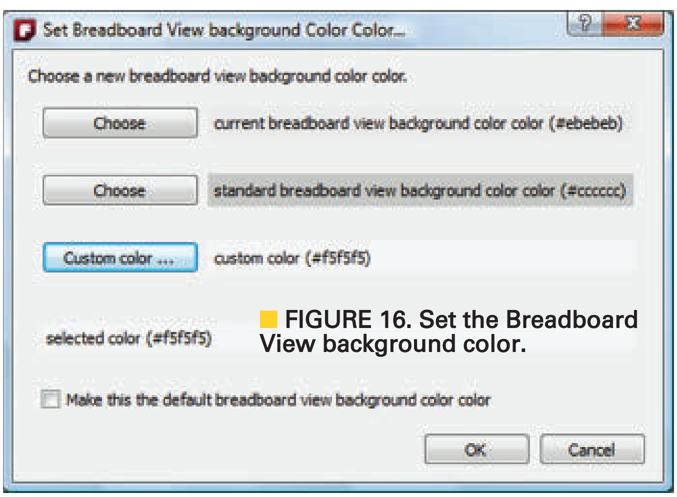


FIGURE 16. Set the Breadboard View background color.

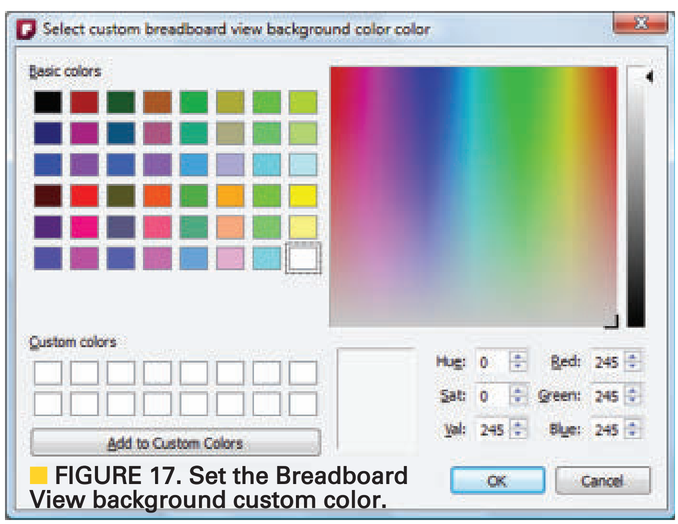


FIGURE 17. Set the Breadboard View background custom color.

Add a Note

You can add a note to your project by clicking the icon shown in **Figure 18**. You can resize the note, move it around, and input multiple lines as shown in **Figure 19**.

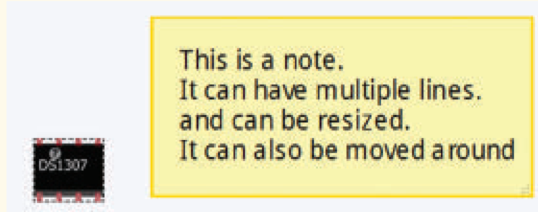


FIGURE 19. An actual note.

Part Floating Menu

Move your cursor over a part and click the right button to open a floating menu of options for things you can do with that part. **Figure 20** shows the example of 'Rotate 45º Clockwise.' Play with these features for a while to get familiar with them.

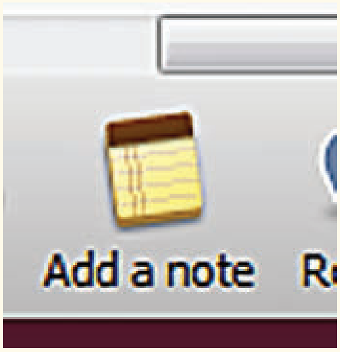


FIGURE 18. Add a note.

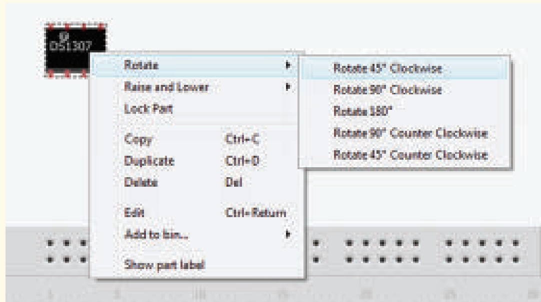


FIGURE 20. Part Floating menu.

CREATING PARTS IN FRITZING

Using the Generic IC to Create a 74HC595 Shift Register

Many of the core parts in Fritzing are generic and are quite easy to modify or create new parts with. We'll look at how to turn the generic IC into a 74HC595 serial-to-parallel shift register. This part is available in the Fritzing core, but we'll build this part as an exercise to get better acquainted with Fritzing's features. Drag and drop a generic IC from the Core bin of the Parts dialog box onto the Breadboard View, then undock the Inspector dialog and set it next to the part as shown in **Figure 21**.

In the Part Inspector, we change the top input box from IC2 to 595, and the chip label to '595. Take a moment to open the package drop-down menu and you'll see a long list of possible packages — mostly SMD that we won't be using. Change the pins from eight to 16, giving us the new Breadboard View part shown in **Figure 22**. Click the 'Edit Pin Labels' button to open the dialog shown in **Figure 23**. Now, open a datasheet for the 74HC595 (actually any 74xx595 part will do) as shown in **Figure 24** and then change the pin labels to match **Figure 25**.

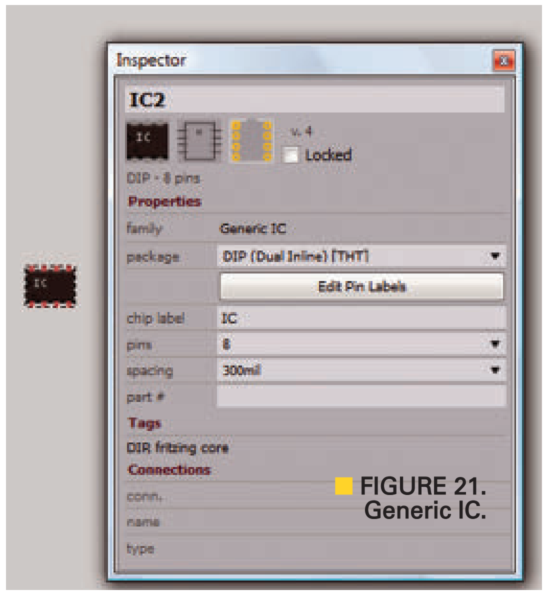


FIGURE 21. Generic IC.

Now, we have a new part for the '595 serial-to-parallel converter. **Figure 26** shows the new '595 part in the three views (Breadboard, Schematic, and PCB). (By the way, if you're wondering why I selected the '595, you might want to take a look at the December 2010 Workshop.)

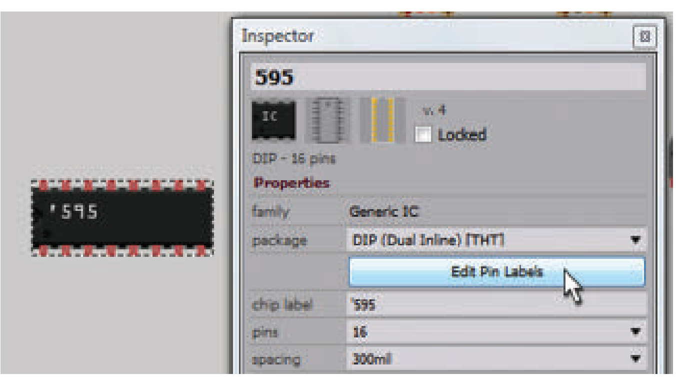


FIGURE 22. The 595 Edit Pin Labels.

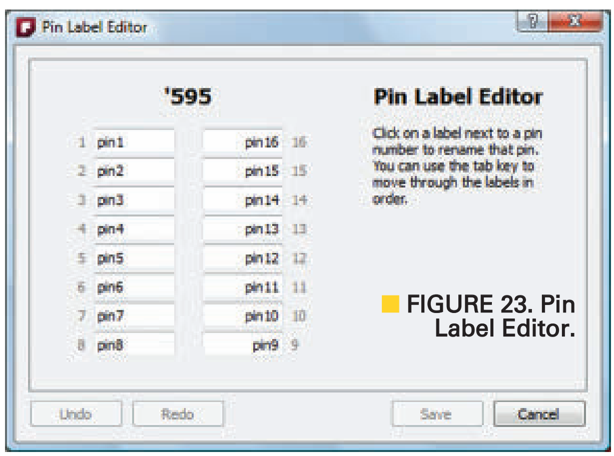


FIGURE 23. Pin Label Editor.

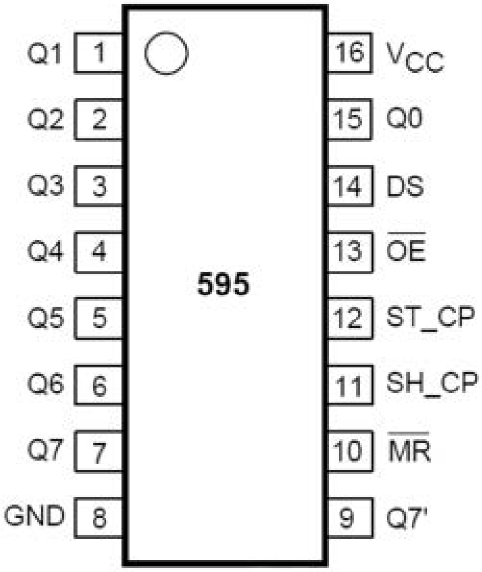


FIGURE 24. The '595 pins.

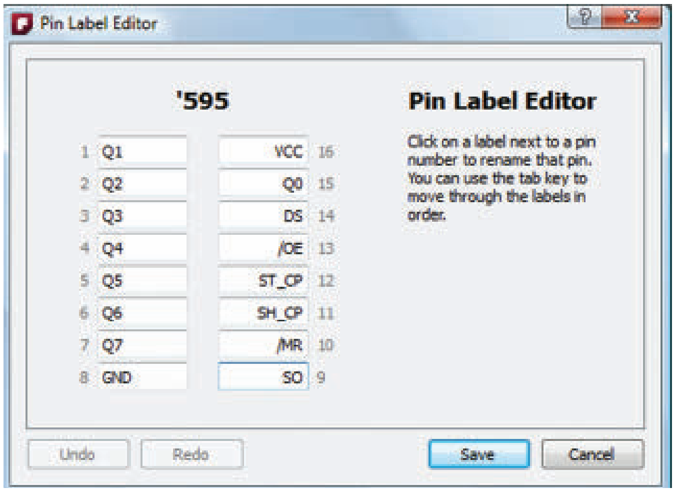


FIGURE 25. Pin labels for the '595.

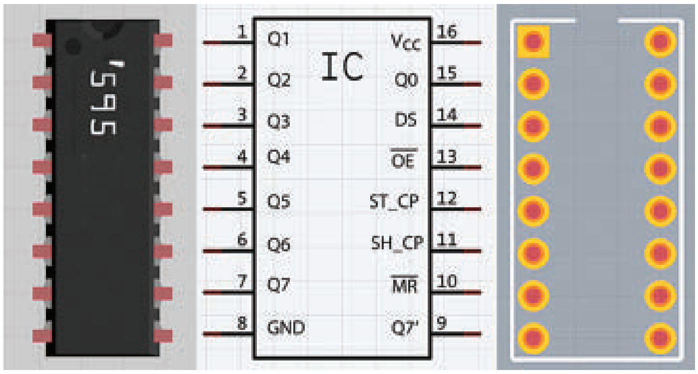


FIGURE 26. The '595 part for the breadboard schematic and PCB.

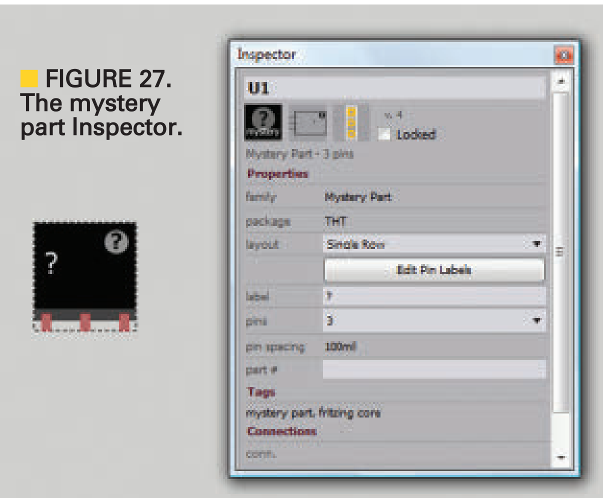


FIGURE 27. The mystery part Inspector.

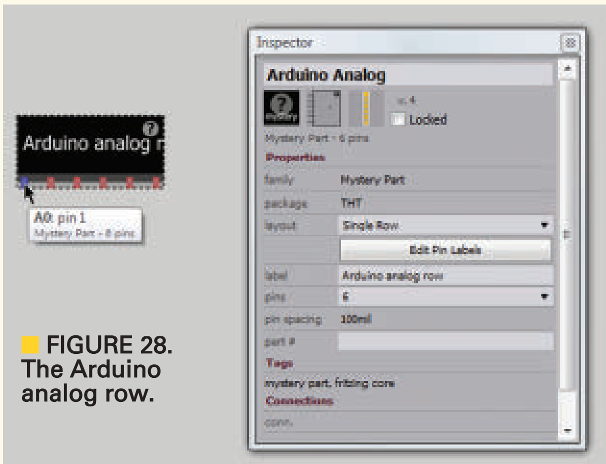


FIGURE 28. The Arduino analog row.

Using the Mystery Part to Create a Partial Arduino Shield

Fritzing has a mystery part that is somewhat similar to the generic IC. We will use it to create a new part that represents the bottom row of an Arduino shield so that we can repeat the RTC circuit design, but for a much smaller PCB (the RTC doesn't need the top row of the Arduino, so why include it?). So, let's use these parts to redo the RTC PCB from last month. Open that file and save a copy — just in case.

Select the mystery part from the Core Parts bin and you'll see the part as shown in **Figure 27**. We will use this to create a part for the Arduino analog pins. Set the 'pins' to six, change the name from U1 to Arduino Analog, and then use the 'Edit Pin Labels' to change the pin names to A0 to A7. Double-check that you've labeled them correctly as shown in **Figure 28**.

That was easy, so let's follow the same procedure and create an Arduino Power Row part that we will use to make a new PCB with only the Arduino bottom row.

Figures 29, **30**, and **31** show how we used these two parts to replace the Arduino in our design (you might want to refer to last month's Workshop for the original process). One thing to notice is that the breadboard image doesn't relate to the real world since we don't actually have physical parts that are the Arduino Power and Analog Row — we are still using the Arduino in our real world breadboard, but we are substituting these 'virtual' parts so we can get our smaller PCB.

So there you have it. Our PCB is less than half the original size and will be proportionally cheaper to fabricate.

Using the Design Rule Check

In the PCB View, open the Routing menu and select (DRC). Uh oh! **Figure 32** shows that we've broken some rules. So, we move things around a little and repeat the DRC; we get **Figure 33**. It is now ready to go.

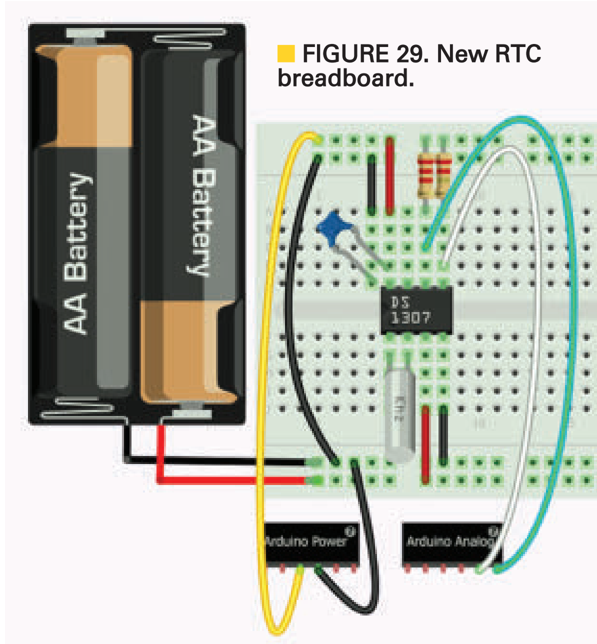


FIGURE 29. New RTC breadboard.

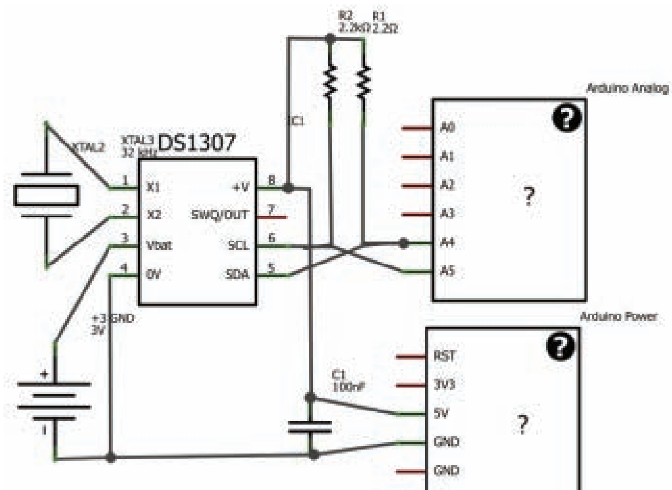


FIGURE 30. New RTC schematic.

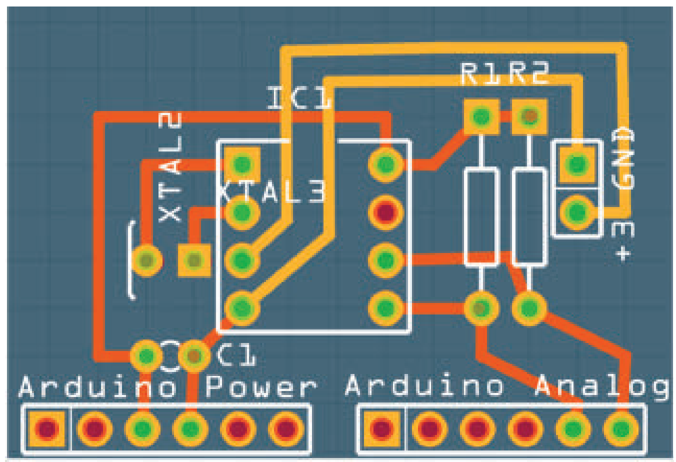


FIGURE 31. Smaller RTC PCB.

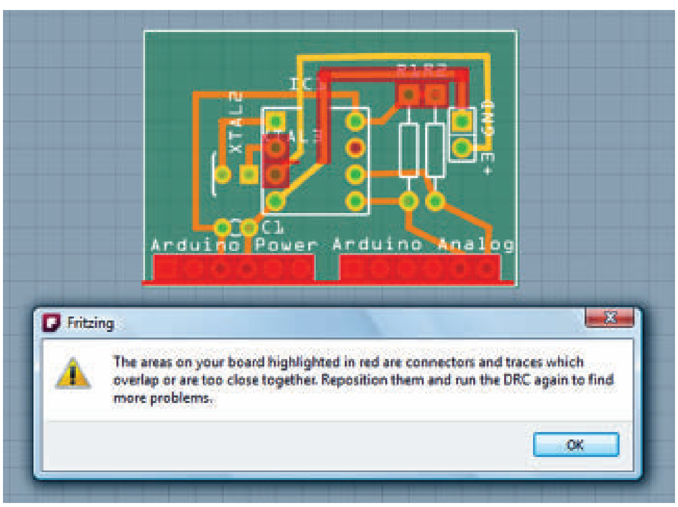


FIGURE 32. Design Rule Check Bad window.

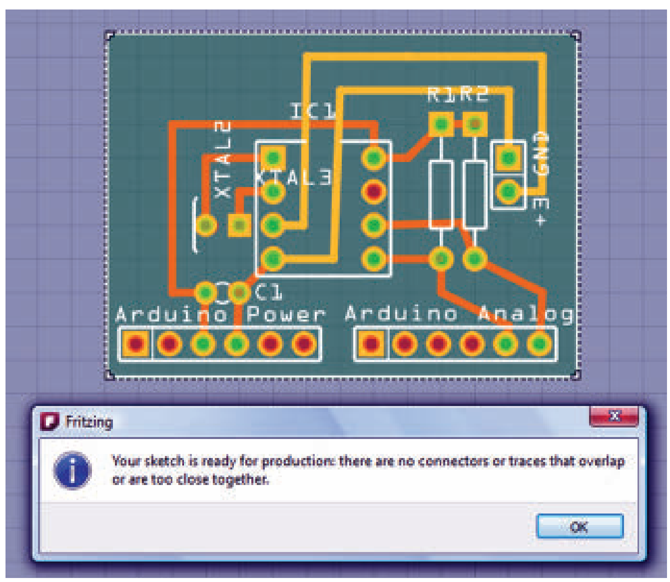


FIGURE 33. Design Rule Check Good window.

MODIFYING AN EXISTING PART

Using Inkscape to Modify the Part

I chose Inkscape — an open source SVG graphics editor — to modify this part in the .svg format used by Fritzing because it is not just free, but a darn good program. I also have the expensive Adobe Illustrator program that draws .svg parts, but I find it less easy to use than Inkscape, so why bother?

I am going to have to assume that you either know how to use vector graphics drawing programs or are willing to learn since it would take many Workshops to explain how to do this.

Let me just say that I keep a browser open and Google almost every action I take when drawing because the procedures are arcane (in both Inkscape and Illustrator) and far from intuitive. I will give a few hints along the way, though, for issues that might take you too long to figure out on your own.

We will only use Inkscape superficially to modify a part this month, but next month we will use it to create a part from scratch. So, if you think you're going to really get into making parts with Fritzing, I suggest you take some time to learn Inkscape. The effort will come in really handy next month.

Another way to create a part is to modify an existing part. This is a lot harder though, and requires messing with the .svg file, so be forewarned. Let's fix that seven-segment LED part, which just annoyed me from the get-go. I wondered why, and then I realized that it was displaying the letter F (**Figure 34**). Being grade conscious, I wanted an A.

You can find this part's drawing in the Fritzing directory under **\parts\svg\core\breadboard\7-segment_13.svg**. Before messing with it, make a copy and paste it in the directory so you can restore it if you flub something up. Drag the part into Inkscape. Make sure you set Inkscape up as before with the 0.1" spacing and the page resized to fit the part.

Select the object and ungroup it. I do this by selecting the part and then holding down the shift and ctrl key, then pounding on the 'g' key until all the sub-object borders appear as shown in **Figure 35**.

Next, use the arrow key to highlight each of the gray segments on the left and then select the 'Pick Colors from Image' tool on the left. Click a red segment which changes the color of the selected segment to that same shade of red as shown in **Figure 36**.

Repeat for the lower right segment and you've turned the F into an A. Another problem for me was that my particular real world seven-segment LED isn't white, but a light gray.

So, I selected the white background box and clicked a light gray color. I then clicked the Object menu group item which gives me the image shown in **Figure 37**.

Now, open the grid, set it to 0.1", and disable the snap. You'll notice that the pins are not on 0.1" centers. Change the size of the part so that the pins fit 0.1" centers, and the upper and lower pins are 0.6" apart.

Finally — and this is critical — before saving this part, resize the page to fit the part and then — even more critical — save the part as a plain .svg, not an Inkscape .svg (I forget this nearly every time). When you open the 13 mm seven-segment LED in Fritzing, you will see (as shown in **Figure 37**) that you are now getting an A. Good for you!

Next month, we will go a bit deeper into making parts with Fritzing. Some of this will be easy, some hard, and some darn near impossible — but hopefully my hints will help keep you from bloodying yourself on some of the same walls I smacked into. **NV**

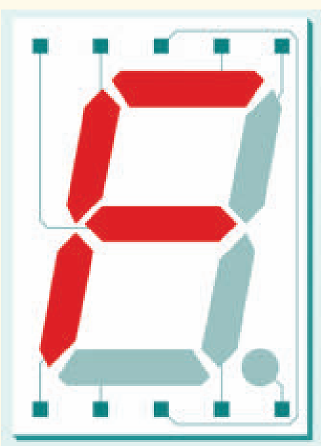


FIGURE 34. Seven-segment F.

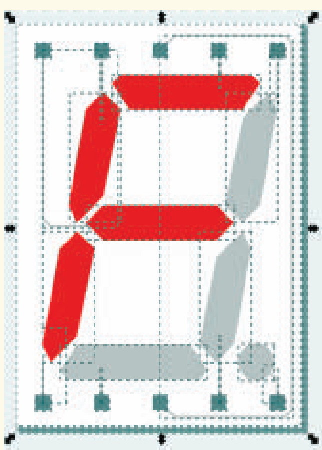


FIGURE 35. Seven-segment object borders.

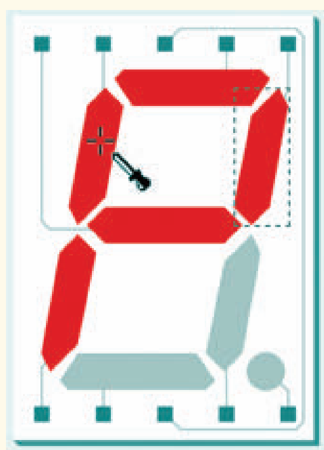


FIGURE 36. Pick colors from image.

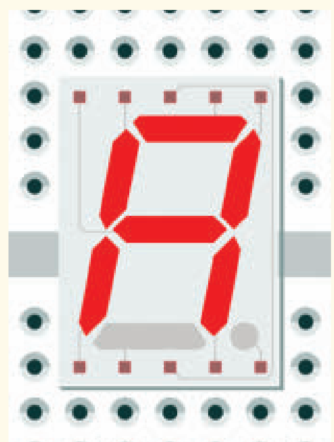


FIGURE 37. Seven-segment A.

CLASSIFIEDS

KITS/PLANS

Electronic Kits

- Fun and Educational
- Discount White and Blue LEDs
- Quantity Discounts

www.bakatronics.com

LIGHTING

HARDWARE WANTED

DEC EQUIPMENT WANTED!!!

Digital Equipment Corp. and compatibles.
Buy - Sell - Trade

CALL KEYWAYS 937-847-2300
or email buyer@keyways.com

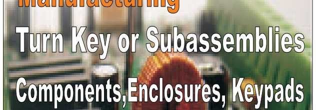
ENGINEERING/ DESIGN SVCS

DEVELOTECH, INC.

IP, PCB & Circuit Design, Firmware, Smart Sensors, Test, Low Quantity Builds

NV@DEVELOTECH.NET
Orlando, FL 407.756.6823

FLEXREV China Direct Manufacturing



$0.007 per soldering point
2 layer PCBs $0.14/in2
Small jobs welcome

FLEXREV.COM

ROBOTICS

THE NEXT GENERATION OF MAXSONAR

The HRLV- MaxSonar Sensors

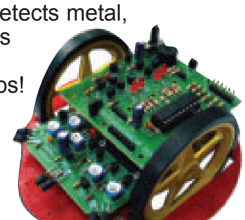
- Amazing One-Millimeter Resolution
- Simultaneous Multiple Sensor Operation
- Superior Noise Rejection
- Target Size Compensation for Accuracy
- Temperature Compensation ($4.95)
- Outputs now include TTL Serial

$34.95 (MSRP)

www.MaxBotix.com

QKITS.COM

AVR Treasure Finder Robot kit detects metal, stops and beeps!



$46.95 USD

1-888-GO 4 KITS

VACUUM TUBES

WANTED

To buy **COMPLETE** *vacuum tube collections, **ANY** quantity. Will buy new, new w/o boxes, and used. We buy all types and will not cherry pick your collection. Will travel to inspect and pick up large hoards.

Paul, Sound Ideas, 3215 NW 13th Street, Gainesville, FL 32609.

pwb@soundideasstereo.com

*Please list tubes in the note line of your email

352-378-0192 ext. 300
FAX: 352-371-1791
(9 am-6 pm EST M-F, 10 am-5 pm Sat)

Subscribe today!
www.nutsvolts.com

COMPONENTS

RF PARTS™ CO.

Tubes, Transistors, Power Components
Email: rfp@rfparts.com • Web: www.rfparts.com
800-737-2787 Fax 888-744-1943

uM-FPU64
64-bit Floating Point Coprocessor

64-bit and 32-bit IEEE 754 compatible
SPI or I2C interface, 3.3V Supply, 5V Tolerant I/O
Extensive floating point support, FFT, matrix operations
GPS input, local device support, Flash functions, IDE

DIP-28, SOIC-28,TQFP-44

Robotics
Navigation
Sensor Modules
Embedded Systems

Additional products...
uM-FPU V3.1 32-bit FPU
uM-PWM1 Servo Coprocessor

www.micromegacorp.com

SURPLUS

SURPLUS ELECTRONIC PARTS & ACCESSORIES

Over 18,000 Items in Stock.

Cables, Connectors, Displays, Fans, Hardware, LEDs, Motors, Potentiometers, Relays, Semiconductors, Service Manuals, Speakers, Switches, Test Equipment, Tools, VCR Parts

Surplus Material Components
SMC ELECTRONICS
www.smcelectronics.com

No Minimum Order.
Credit Cards and PAYPAL Accepted.
Flat $4.95 per order USA Shipping.

MISC FOR SALE

BUSINESS CLOSED!

LIQUIDATING $250,000 SPEAKERS AND PARTS. $200,000 ELECTRONIC WORLD STORE. $25,000 TONER CARTRIDGE BUSINESS. $100,000 INDUSTRIAL EQUIPMENT.
ISE-WORLDWIDE LIQUIDATOR FOR OVER 32 YEARS.
www.iseliquidator.com

ENERGY SOLUTIONS

Save Money On Your Electric Bill
www.sunny-solutions-shop.com
800-315-0309

Looking for robot stuff?
www.servo magazine.com

TESLA PRODUCTS

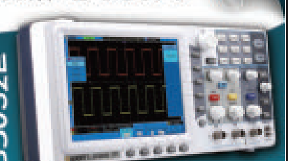
TESLA'S SECRET LAB

HOME OF THE "MIRACLE" TESLA PURPLE ENERGY PLATES!
Plus Tesla Books, Patents, DVDs & Conspiracies!

www.TeslaSecretLab.com
Tesla's Secret Lab • Box 753-N
New Brunswick, NJ 08903
732-602-3407
Free Bizarre Bazaar Newsletter

OSCILLOSCOPES

LOW-COST, VERSATILE 30MHZ SCOPE
NEW



SDS5032E
ONLY $299
WWW.SAELIG.COM

READER-TO-READER TECHFORUM

All New TechForum Online At www.nutsvolts.com/tech-forum

>>> QUESTIONS

Interface to a Cell Phone

I am using two perimeter mics via a small mixer to monitor outside buildings. I now want to connect this to my old Nokia 1200 to do off-site monitoring. Any suggestions?

#10121 Riaan
Newcastle, South Africa

SD/MMC Ribbon Cable to USB

My Brother Lowrance GPS fish finder model #LCX 112C has been discontinued by Lowrance.

The dual SD/MMC reader interface (PCB #0170920-001 rev.2, mfg by Yang-An Electronics) has been corroded by salt water.

I'm looking for a replacement PCB or a way to connect a miniature 10-pin ribbon cable to USB connector as a replacement to read information from the SD Card.

#10122 James Nolan
St John's, NL, CANADA

Headphones and Speakers

I have a pair of 7.1 surround sound headphones. I would like to buy or make a unit to switch all eight speaker outputs to all eight headphone inputs, but I am worried about overloading/blowing up the headphones. I assume some kind of attenuator will be required. Any Ideas?

#10123 Elwyn Horton
Gillingham, Kent, UK

Vintage Parts Hunter

Does anyone have or happen to know where I can obtain a pair of 455 kHz IF transformer cans, and a Local oscillator coil for my homebrew AA five tube super heterodyne AM receiver?

I am using the 12BE6, 12BA6, 12AV6, 50C5, and my friendly 35W4 to build my set. Are there any "old new stock parts" establishments still around these days that may have the vintage parts I require?

I know I can still buy yesteryear radio sets that have what I need. However, I would be more likely to repair them as they are, than to scrap any of them for the parts.

I have my trusty 365 pF dual gang air core variable tuning capacitor with the reduced capacity second rotor set. I hope to have no difficulty obtaining all the rest of the parts, as well.

Can any of you high tech experts direct me to where I can get the above named parts I am lacking? Thanks for any assistance.

#10124 Howard Daniel Rollins III
Vintage Radio Constructor
& Diehard Antique Radio Enthusiast
Kinder, LA

Android Platform

Are there any companies that are developing an Android device that is designed to be wall or panel mounted? I like the operating system and would like to integrate an Android tablet into some automation, but haven't found anything online. My employer would also like to explore some possibilities for industrial applications and a panel mount would work best for them, too.

#10125 John Harris
Delaware, OH

Remagnetizing Coils

I did not use my turntable on my sound system for some years, but the other day, I had the urge to rediscover some of my old vinyl discs.

To my dismay, there was no sound when I put on a disc. Troubleshooting led to the phono cartridge, an AKG P8ES.

I went online to see if any of these are still available, and at the same time came across some audio blogs describing the exact same problem being that these particular cartridges are moving magnet types and appear to be prone to losing their magnetism over a period of time. This does not occur with moving coil cartridges. Would anyone know of a way to remagnetize the coils in these types of cartridges?

#10126 David Asselin
St Lambert, Quebec, Canada

>>> ANSWERS

[#8123 - August 2012]
Audio Sound Spectrum

Is it possible to take the sound heard in the frequency range of 0 to 40 kHz and space it evenly into the 10 Hz to 20 kHz range? I want to hear what a dog hears but in the human range. I know this will make people's voices sound funny, but how would this be done and is there such a device? Is there even a mic for that upper range of up to 40 kHz?

One easy approach is the digital divide circuits used in bat detectors. For example, check **http://pw1.netcom.com/~t-rex/BatDetector.html**. Basically, it just divides the input frequency by some power of two (2, 4, 8, 19, ...). In this case, divide 0 to 40

Check out the *ALL NEW* Tech Forum at **www.nutsvolts.com/tech-forum** where you'll find additional material to this column even before it's seen in print. You can participate, comment on answers, even get emailed when a new answer or comment is posted. *Check It Out!*

Send all questions and answers by email to **forum@nutsvolts.com**
*Check at **www.nutsvolts.com** for tips and info on submitting to the forum.*

kHz by 4 and get 0 to 10 kHz. The digital divide circuit is pretty simple but you will lose all amplitude information. Given how simple the circuit is that might be acceptable.

A suitable microphone will be a bit of a problem. Many mics will give a signal at frequency up into the 40 kHz range, but the sensitivity will drop off a fair amount. Piezo transducers are often used for bat calls and they'll give some signal up to — perhaps as high as — 100 kHz. However, piezos have resonant frequencies and they'll be super sensitive near that frequency. A typical 1" diameter thin piezo resonates at about 3 kHz.

There are ultrasonic transducers designed for 40 kHz that work well as a mic at that frequency, but their sensitivity at lower frequencies drops off.

Jim Sluka
Greenwood, IN

[#8124 - August 2012]
Connecting an Old Printer to a Current PC

I have an IMP-24 printer with a Centronics interface. I would like to print to it from a Windows 7 or XP computer using some form of Basic. I have VB and Visual Studio 2010 Ultimate. However, there seems to be no drivers for this old printer. Is there a way to address LPT1 from Visual Basic or Visual C so I can send the data to the printer directly without going through the operating system?

Here is one method that I think will work for you, but I don't know how it will handle control characters. It's fairly quick to set up, though.

On XP:
- Click Start.
- Point to Settings, then click on Printers and Faxes.
- Double-click on Add Printer.
- Click on Next.
- Click on Local Printer.
- Select Local Printer Attached to this Computer.
- Click on Next.
- Let it search; it will come up with "The Wizard was unable ..."
- Click on Next.
- Select the LPT port you want to use.
- Click on Next.
- From the left column, select Generic.
- Then from the right column, select Generic/Text Only.
- Click on Next.
- If you desire, change the printer name.
- Select whether or not you want this to be the default printer.
- Click on Next.
- Select whether or not you want to share this printer on your network.
- Select No for printing a test page.

It will now show you your selections. If it looks okay, select Finish.

It will show that it's copying some files and then the window will disappear indicating that it is complete.

To cause Windows to NOT spool the printer output:
- Right-click on the printer you just created.
- Click on the Advanced tab.
- Select Print directly to printer.
- Click Apply, click OK.

(The following code is from a VB6 app that I wrote that was using an HP printer with an HP driver.)

In VB, there is a Printer object.
'It needs some setup code:

```
'Printer defaults
  With Printer
    ColorMode =
      vbPRCMMonochrome
    .Copies = 1
    .Orientation = vbPRORPortrait
    .Zoom = 100
    .ScaleMode = vbTwips
    .FontBold = False
    .FontItalic = False
    .FontStrikethru = False
    .FontTransparent = True
    .FontUnderline = False
    .FontName = "Courier New"
    'not proportional letters
    .Orientation = vbPRORPortrait
    .ScaleMode = vbInches
    .FontItalic = False
    .FontBold = False

'You may not need the following:
    .FontSize =

  If .PaperSize <> 1 And .PaperSize
  <> 9 Then
    .PaperSize = 1
    End If

    'sets top, left of page
    Printer.CurrentY = 0
    Printer.CurrentX = 0

  'end of May Not Need

' Here's where you actually print
' the text"
Printer.Print strStringOfText

' note: If you want to print more on
' the same line,
' add a trailing semicolon for all but
' the last print

Printer.Print strStringOfText;

Printer.NewPage    'ie: formfeed

'If your printer is spooling, the following will cause the document to print.
'It will also signal End of Document so
you can start the next document
Printer.EndDoc
```

Mark Peterson
Plymouth, MN

[#9122 - September 2012]
How Long Does My Furnace Run?

I need to see how long my furnace blower motor runs and track the overall time in a digital display. I would need a reset button to reset the *time when needed. I have a switch that activates when power is sent to the motor line.*

#1 How complex do you want it to be? A microP with a digital display and many hours work, or a home brew digital circuit (many many hours), or an old PC with a serial port and a couple hours for programming (just use the switch to ground the RTS pin and have a QBasic periodically check if RTS is high or low), or hack a $5 digital kitchen timer to do the job?

I would go with the $5 digital kitchen timer (or digital stopwatch). Rip the case open and explore how the Start/Stop button works. It may be pretty easy to interface that button with the switch that is operated when the motor is turned on.

Jim Sluka
Greenwood, IN

#2 I think I found exactly what you need; see Multifunction Counter-Timer kit K8035. I found it at **apogee kits.com**, under the timing kits and modules index on the left side of the web page.

Pete Lunt
Raleigh, NC

#3 It depends a lot on what you want to spend. If you do a search on "hour meter," you will find lots of suitable products. In general, hour meters are not resettable.

www.ingramproducts.com/Hour_Meters_Counters-Rectangular_Hour_Meter_12_to_48VDC.html is an hour meter that's non-resettable, cheap, and easy to install. The datasheet says it will operate on AC power and there are only two connections. You could just connect between C and Y in your furnace if you want to monitor cooling; C and G if you want to monitor the fan; and C and W if you want to monitor heat. There will be 24 VAC present between these terminals in the modes above. Between C and G, there is 24V all the time. You would also have to decide if you wanted to ignore the time the fan was on by itself. That would require creating an OR condition between the C and Y, and C and W contacts using two relays by putting the normally open contacts in parallel.

www.kep.com/catalog/ii/pages/products/HB26-hour-run-meter.html is an electro-mechanical timer that also meets the specs with reset.

Some furnaces such as some Carrier models don't use the traditional terminals and the motors may be three-phase internally and would be harder to interface. A heat pump would be slightly different. Some furnaces have a humidifier or 120 VAC electronic air cleaner voltages available.

Some timers come with non-replaceable batteries with a 5+ year life. A "contact closure" is a common way to cause timing. A "contact closure" is easy to get. Just connect a 24 VAC relay across the terminals above. Some of the smaller rectangular meters would have a "standard" cut-out.

www.laurels.com/magna_clock.htm would meet your spec, but it has way too many features. With electronics, the case, power supply, and "real estate" (size of the PCB) make up most of the cost, so a throw-away, battery powered meter with no reset is the simplest.

The key here is knowing what to search for with "hour meter" and some interfacing possibilities. It looks like 120 VAC and 24 VAC are options for power and timing, and a contact closure is an option for timing with one or two additional relays.

DC meters might work on low voltage AC too. Check the datasheet.

Ron Dozier
Wilmington, DE

[#9123 - September 2012]
3D Location System

I am trying to build a close-range 3D location system that works within about a 10 ft square area.

I need X, Y, and Z coordinates from one fixed point; then want to record the points into a computer.

I prefer a laser based system that records its rotation in degrees, and for both Z and X axis, and the length of the laser to the object for a calculated Y coordinate.

#1 Can't help with most of your questions except for one "length of the laser to the object for a calculated Y coordinate" is a pretty tricky thing to do — even with a computer — at such short distances. The speed of light is about 300 million meters per second (roughly one billion feet/sec). If you are thinking of blinking the laser then measuring how long it takes the beam to bounce off an object 10 feet away and return to a sensor, then you'll need to measure time intervals of a fair bit less than one billionth of a second; one GHz is a billionth of a second.

To get an accuracy of say one foot out of 10, you would need to measure time at 10 GHz; to get 1", you would be up near 100 GHz. That is possible but not for cheap. Even a $1,000 o'scope wouldn't be able to measure a time interval that short.

I believe vision systems like what you want (e.g., XBox Kinect) use multiple lasers and a video system. With two laser beams, you get two dots; if the two beams are parallel, the distance between the dots is proportional to the distance from the laser to the object. A PC does the distance calculation based on the video image.

I think your best bet would be to start with an XBox Kinect system and mount the transmitter/receiver on a rotating platform. The Kinect will do three dimensions all by itself. The rotating platform will let you look in all directions. I believe there are software platforms for communicating with a Kinect system.

Jim Sluka
Greenwood, IN

#2 Since you didn't mention precision, price and ease of construction must be more important to you. Therefore, I don't think a laser based system is necessary or appropriate.

For approximately a three foot cube working space, you could use a webcam parallel to one axis and another webcam perpendicular to that axis. Using an object of known size and location, you could calibrate that photogrammetry system in software.

If you feel you need a laser system, a lens in front of a Microsoft Kinect system could shrink its working space, though that might not be necessary in your application.

Both of these systems assume that there is an unobstructed view of the target (or targets), and that the object is reflective. Other approaches are necessary for transparent objects.

Craig Cantello
Schenectady, NY

[#9125 - September 2012]
A/C Mod

I have a central A/C unit and the inside fan blower has a control module referred to as an "r" mod. It has a few components on it and is designed to kick-start the fan motor. I can't afford the $1,200 to get it replaced. Can I just replace the bad components on the module? It looks like two small capacitors and an electromagnet. It's a Carrier Tech 200 SS.

#1 If the 'device' has two wires going to it and three wires out to the motor, then it is likely a motor starter. It can be a thermal device with a relay; that is most common.

Many compressors use them and they do fail. Most any A/C parts counter clerk should be able to match it to a direct or universal replacement for about $20.

Len Powell
Finksburg, MD

#2 Of course, you can replace the bad components, but how do you know the module is bad? Perhaps the motor is bad.

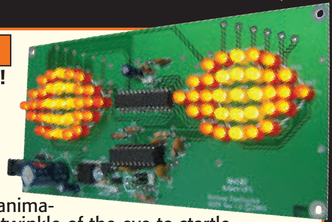
A motor starter usually involves a large capacitor, but I guess an inductor could be used. I suspect that the fan has to be kick-started because the compressor will overheat and shut down if the fan is not running.

Russ Kincaid
Milford, NH

RAMSEY

Spooktacular Halloween Treats!

Blinky-Eyes Animated Display



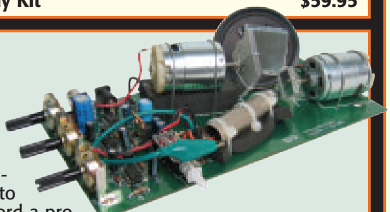
- ✔ **Animated display of 66 super bright LED's!**
- ✔ **Microcontroller controlled!**
- ✔ **Changes brightness automatically!**
- ✔ **Animated with constant motion!**

The ultimate animated LED kit that will dazzle you and delight your friends! Uses a microcontroller to randomly select from many different animations such as a long pause before a wink, or a twinkle of the eye to startle passers-by!

Four modes to satisfy any enthusiast's desires: 1. Off for long random periods, then blinks or winks. Designed to scare! 2. On for long periods before performing an animation, perfect for costumes and displays! 3. Animates all the time for constant motion, perfect display attention-getter. 4. Random fire! When placed in a pumpkin will light it up like you wouldn't believe! As if this weren't enough, the BE66 can also control a small hobby motor to shake bushes at random intervals or signal an external player to make a scary sound! Also has a CDS cell to sense light. In one mode, the display will dim as it gets dark for battery operation, and in the other mode it will turn off when it's too bright, so it plays only in the dark!

BE66	**Blinky-Eyes Animated Display Kit**	**$59.95**

Laser Light Show



- ✔ **Audio input modulates pattern!**
- ✔ **Adjustable pattern & size!**
- ✔ **Projects neat motorized patterns!**
- ✔ **Uses safe plastic mirrors!**

You've probably seen a laser show at concerts or on TV. They're pretty impressive to say the least! Knowing that you can't afford a professional laser display we challenged our engineers to design one that's neat and easy to build, yet inexpensive.

Well, the result is the LLS1 Laser Light Show! This thing is sweet and perfect for your haunted house or halloween parties! It utilizes two small motors and a small standard laser pointer as the basics. Then, we gave it variable pattern and speed controls to customize the pattern!

Not enough, you say? How about a line level audio input to modulate the pattern with your CD's, music, or spooky sound effects? You bet! Everything is included, even the small laser pointer. Runs on 6-12 VDC or our standard AC adapter.

LLS1	**Laser Light Show Kit**	**$49.95**
AC125	**110VAC Power Supply**	**$9.95**

Tri-Field Meter & "Ghost Detector"

- ✔ **See electric, magnetic, and RF fields!**
- ✔ **Watch the magnetic fields of the earth!**
- ✔ **Sense different magnetic poles**
- ✔ **Detect RF transmitter fields**
- ✔ **Graphical LED display allows you to "see"the invisible fields**
- ✔ **Great learning tool for EMF, RF, and manetic field theory.**

Call it a Tri-Field Meter, an Electrical, Magnetic, and RF Detector, a Ghost Detector, or a Tricorder, but walking around with this on Halloween will seriously impress even the most doubting of Thomases out there!

The TFM3C has three separate field sensors that are user selectable to provide a really cool readout on two highly graphical LED bargraphs! Utilizing the latest technology, including Hall Effect sensors, you can walk around your house and actually "SEE" these fields around you! You will be amazed at what you see. How sensitive is it? Well, you can see the magnetic field of the earth... THAT'S sensitive!

The technical applications are endless. Use it to detect radiation from monitors and TV's, electrical discharges from appliances, RF emissions from unknown or hidden transmitters and RF sources, and a whole lot more! If you're wondering whether your wireless project or even your cell phone is working, you can easily check for RF! A 3-position switch in the center allows you to select electric, magnetic, or RF fields. A front panel "zero adjust" allows you to set the sensors and displays to a known clean "starting point."

If the TFM3C looks familiar, it's probably because you saw it in use on the CBS show Ghost Whisperer! It was used throughout one episode (#78, 02-27-2009) to detect the presence of ghosts!

The concept is simple, it is believed (by the believers!) that ghosts give off an electric field that can be detected with the appropriate equipment. In the electric mode, the TFM3C's displays will wander away from zero even though there isn't a clear reason for it (not scientifically explainable, aka paranormal!). This would mean something has begun to give off an electric field. What it was in the Ghost Whisperer was a friendly ghost. What it will be in your house... who knows! Makes a GREAT leaning project besides! Requires 4 AA batteries.

TFM3C	**Tri-Field Meter Kit With Case**	**$74.95**

Automatic Animated Ghost

- ✔ **Automatically greets your visitors!**
- ✔ **Responds to sudden noises!**
- ✔ **Built-in microphone!**
- ✔ **Adjustable sensitivity**

Who says ghosts are make believe? Once your friends come upon this one they'll think differently! The unique circuit board design includes two ominous blinking eyes that change with various conditions, including sudden changes in ambient noise. A highly sensitive built-in microphone picks up anything from noises to talking and makes the ghost dance with its built-in motor, make eerie sounds with the built-in speaker, and randomly blink. A white cloth and a hanger are included as shown to make it look like the real thing. Runs on 2 AAA batteries (Not included).

MK166	**Automatic Animated Ghost Kit**	**$21.95**

Halloween Pumpkin

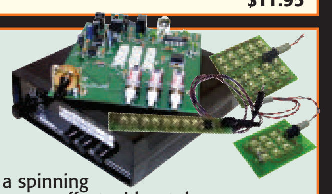
- ✔ **25 bright LED's!**
- ✔ **Random flash simulates flickering candle!**
- ✔ **Super bright LED illuminates entire pumpkin!**
- ✔ **Simple & safe 9V battery operation**

The perfect "starter" kit with a terrific Halloween theme! You won't be scraping the seeds and guts out of this pumpkin! Six transistor circuit provides a neat random flash pattern that looks just like a flickering candle. Then a super bright LED illuminates the entire pumpkin with a spooky glow!

The pumpkin face is the actual PC board, and assembly is easy through-hole soldering of all components and LED's. Your pumpkin is powered by a standard 9V battery (not included) which snaps to the back of the pumpkin. An on/off switch is also included. Create a new kind of pumpkin this year, and learn about LED's and electronics at the same time!

MK145	**Electronic Halloween Pumpkin Kit**	**$11.95**

High Power LED Strobe Light



- ✔ **Everlasting LEDs won't burn out!**
- ✔ **Variable flash rate and audio trigger!**
- ✔ **Bass and treble trigger modes**
- ✔ **Safe low voltage construction, no fragile high voltage Xenon tube to break!**

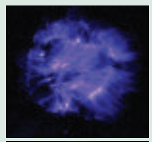
Everyone has seen Xenon stroboscopes "stop" a spinning quarter or flash to music. Now you can get the same effect without the expense and hassle of high voltage Xenon tubes! Great for Halloween displays and parties!

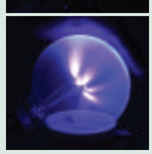
A plug-in 3x3 array of super bright Telux™ LEDs creates a brilliant, sharp flash similar to a Xenon flash tube. The LEDs can also be mounted directly on the main PC board or on a remote display board. Optional display boards also available. Variable flash rate from 1-220 FPS plus an audio flash trigger with selectable bass or treble triggering. Easy to connect with standard stereo RCA audio connectors. External trigger can be looped to additional units for simultaneous flash displays from the same source! Runs on safe 12-15VDC so a great kit for the kids to build!

LEDS1C	**High Power LED Strobe Light Kit with Case**	**$49.95**
LEDS8	**Display Board, Inline Array of 8 LEDs**	**$29.95**
LEDS20	**Display Board, 5x4 Array of 20 LEDs**	**$19.95**
AC125	**110VAC Power Supply**	**9.95**

Spark Generating HV Plasma Generator

- ✔ **Generate 2" sparks to a handheld screwdriver!**
- ✔ **Light fluorescent tubes without wires!**
- ✔ **Build your own Plasma Balls!**
- ✔ **Generates up to 25kV @ 20 kHz from a solid state circuit!**





This popular kit was conceived by one of our engineers who likes to play with things that can generate large, loud sparks, and other frightening devices! And at Halloween there's no better effect than high voltage sparks flying through the air! The PG13 Plasma Generator creates a very impressive 25,000 volts at 20 kHz, to provide a stunning display of high voltage! It will draw a cool looking 2" spark to a hand held screwdriver, or light fluorescent tubes without any connection!

It produces stunning lighting displays, drawing big sparks, to perform lots of high voltage experiments. In the picture, we took a regular clear "Decora" style light bulb and connected it to the PG13 - WOW! A storm of sparks, light tracers and plasma filled the bulb. Holding your hand on the bulb doesn't hurt a bit and you can control the discharge! It can also be used for powering other experiments; let your imagination be your guide! Operates on 16VAC/VDC for maximum output. Can also be run from 5-16VAC/VDC to reduce the output voltage.

PG13	**Plasma Generator Kit**	**$64.95**
PS21	**110VAC Input, 16VAC Output, Power Supply**	**$19.95**

Music Light Controller
Plug in your standard 110VAC incancescent lights and control up to 3 channels of lights! Lights flicker to different tone ranges of your audio making a great display! Each channel can handle 300W. Includes matching case. Runs on 110VAC.
ML1C Music Light Controller Kit $59.95
Practice Guitar Amp & DI
Practice your guitar without driving your family or neighbors nuts! Works with any electric, acoustic-electric, or bass guitar. Plug your MP3 player into the aux input and practice to your favorite music! Drives standard headphones and also works as a great DI!
PGA1 Personal Practice Guitar Amp Kit $64.95
Passive Aircraft Monitor
The hit of the decade! Our patented receiver hears the entire aircraft band without any tuning! Passive design has no LO, therefore can be used on board aircraft! Perfect for airshows, hears the active traffic as it happens! Available kit or factory assembled.
ABM1 Passive Aircraft Rcvr Kit $89.95
LED Blinky
Our #1 Mini-Kit for over 35 years! Alternately flashes two jumbo red LED's. Great for signs, name badges, model railroading, and more. Used throughout the world as the first learning kit for students young and old! Great solder practice kit. Runs on 3-15 VDC.
BL1 LED Blinky Kit $7.95
LED SMT Blinky
The subminiature high-tech version of the BL1! Utilizes all SMT components to drive a pair of high intensity SMT LED's. Runs on 2 button cells (included). Great attention grabber for signs, hats, or other applications.
BL2 LED SMT Blinky Kit $17.95
Electronic Siren
Exactly duplicates the upward and downward wail of a police siren. Switch closure produces upward wail, releasing it makes it return downward. Produces a loud 5W output, and will drive any speaker! Horn speakers sound the best! Runs on 6-12VDC.
SM3 Electronic Siren Kit $7.95
LED Sound-To-Light Board
What a neat little kit! It starts with an on-board ultra sensitive mic that captures ambient, spoken, and music audio. It then converts it to a neat 6-LED display. Great for hearing impaired alarms! Adjustable sensitivity. Runs on a 9V battery.
CK457 LED Light To Sound Display Kit $15.95
Electronic Watch Dog
A barking dog on a PC board! And you don't have to feed it! Generates 2 different selectable barking dog sounds. Plus a built-in mic senses noise and can be set to bark when it hears it! Adjustable sensitivity! Unlike the Saint, eats 2-8VAC or 9-12VDC, it's not fussy!
K2655 Electronic Watch Dog Kit $39.95
USB DMX Lighting Interface
Control DMX fixtures with your PC via USB! Controls up to 512 DMX channels each with 256 different levels! Uses standard XLR cables. Multiple fixtures can be simply daisy chained. Includes Light Player software for easy control. Runs on USB or 9V power.
K8062 USB DMX Interface Controller Kit $67.95
20 Watt Mini Audio Amp
Delivers a super clean 20W output from one SMT package! Ultra efficient class D design produces no heat. PCB can be snapped into a small circle for special applications. Runs on 18VDC for rated output, or down to 10VDC for reduced output.
UAM2 20W Subminiature Amp Kit $49.95
Touch Switch
Touch on, touch off, or momentary touch hold, it's your choice with this little kit! Uses CMOS technology. Actually includes TWO totally separate touch circuits on the board! Drives any low voltage load up to 100mA. Runs on 6-12 VDC.
TS1 Touch Switch Kit $9.95
Dual White LED Strobe
This small dual LED strobe is a great attention grabber and a whole lot more! Adjustable flash rate from 1-60 FPS makes it an ideal "pocket strobe". Also makes a great handy tool to "stop" rotating machinery. Runs on a standard 9V battery, not included.
MK147 Dual White LED Strobe Kit $10.95
Mad Blaster Warble Alarm
If you need to simply get attention, the "Mad Blaster" is the answer, producing a LOUD ear shattering raucous racket! Super for car and home alarms as well. Drives any speaker. Runs on 9-12VDC.
MB1 Mad Blaster Warble Alarm Kit $9.95
Xenon Tube Strobe Light
Create amazing effects with an authentic Xenon tube strobe light! Creates a super bright white FLASH with a variable speed of 2 to 20 flashes second. Just connect 110VAC and and you have a complete strobe light!
K5300 Xenon Tube Strobe Light Kit $19.95
Stereo Ear Super Amplifier
Ultra high gain amp boosts audio 50 times and it does it in stereo with its dual directional stereo microphones! Just plug in your standard earphone or headset and point towards the source. Incredible gain and perfect stereo separation!
MK136 Stereo Ear Amp Kit $9.95
Water Sensor Alarm
This little $8 kit can really "bail you out"! Simply mount the alarm where you want to detect water level problems (sump pump!). When the water touches the contacts the alarm goes off! Sensor can even be remotely located. Runs on a standard 9V battery.
MK108 Water Sensor Alarm Kit $6.95
Air Blasting Ion Generator
Generates negative ions along with a hefty blast of fresh air, all without any noise! The steady state DC voltage generates 7.5kV DC negative at 400uA, and that's LOTS of ions! Includes 7 wind tubes for max air! Runs on 12-15VDC.
IG7 Ion Generator Kit $64.95
LED Windmill Display
This "Windmill" displays 31 dazzling bright red LEDs in a rotating animated pattern that is sure to grab attention! The rotational speed is adjustable with a built-in pot. Runs on 9-12VDC.
CK176 LED Windmill Display Kit $24.95
SMT Soldering Lab
Learn all about SMT parts and how to solder them! Surface mount parts are tiny and require a special skill to solder. This lab and course covers it all, and you end up with a great "Decision Maker" kit when done!
SM200K SMT Soldering Lab Kit $22.95
Soldering Lab
The perfect beginner's project specifically designed to teach you the fundamentals of soldering and PC boards. You will not only learn soldering, but how to troubleshoot soldering problems and how to fix them! Final project runs on 9V battery.
SP1A Soldering Lab Kit $9.95
Soldering Parts Lab
Not only will you learn soldering and desoldering but you will also learn component identification and color codes! Includes 163 solder points, and you will learn how to solder all types of components. The comprehensive manual even includes exams!
SP3B Soldering Parts Lab Kit $9.95
IC AM/FM Radio Lab
Learn all about AM/FM radio theory, IC theory, and end up with a high quality radio! Extensive step-by-step instructions guide you through theory, parts descriptions, and the hows and whys of IC design. Runs on a standard 9V battery.
AMFM108K AM/FM IC Radio Lab Kit $34.95
Non-Conductive Tweezers
We've discussed tweezers and magnifiers while working with SMT components. Working with highly sensitive components, ESD safe tweezers can be a life saver! This set of 4 non-conductive tweezers are perfect for any static sensitive devices, and are priced right!
VTTWSET2 Non-Conductive Tweezers Set $3.95
3-In-1 Multifunction Lab
The handiest item for your bench! Includes a RoHS compliant temp controlled soldering station, digital multimeter, and a regulated lab power supply! All in one small unit for your bench! It can't be beat!
LAB1U 3-In1 Multifunction Solder Lab $134.95
GET THE NUTS & VOLTS DISCOUNT!
Mention or enter the coupon code NVRMZ12 and receive 10% off your order!
Get the latest 2012 Ramsey Hobby Catalog today! Filled with some of the neatest goodies around! Order yours today... Or download the PDF at www.ramseykits.com/catalog!
ramsey
amazon.com
www.ramseykits.com
800-446-2295
ramsey
RAMSEY ELECTRONICS®
590 Fishers Station Drive
Victor, NY 14564
(800) 446-2295
(585) 924-4560
Where Electronics Is Always Fun!
✦ Build It! ✦ Achieve It!
✦ Learn It! ✦ Enjoy It!
Prices, availability, and specifications are subject to change. Not responsible for typos, stupids, printer's bleed, or really weird Halloween tricks! Visit www.ramseykits.com for the latest pricing, specials, terms and conditions. Thanks Robin... for reminding me of this ad deadline and that Halloween is approaching! Therefore today's color theme is Pantone 021C Orange! Copyright 2012 Ramsey Electronics®...so there!


Detention Punishment:
Write "I will not play video games in class"
300 Times
10 X = 0
20 Print "I will not play video games in class"
30 X = X + 1
40 IF X < 300 GOTO 20
50 END
President Bush